

**UNIVERSIDAD CATÓLICA DE SANTA MARÍA**

**FACULTAD DE CIENCIAS E INGENIERÍAS FÍSICAS Y FORMALES**

**ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS**



**“MECANISMO ÁGIL PARA LA GESTIÓN Y DESARROLLO DE  
PROYECTOS DE SOFTWARE, APLICADO EN INSTITUCIONES DE  
SEGUROS Y DE TELECOMUNICACIONES”**

Tesis presentada por la Bachiller:

Flores Roca, Karen

Para optar el Título Profesional de:

**INGENIERO DE SISTEMAS**

**Arequipa – Perú**

**2016**

## PRESENTACIÓN

Sra. Directora de la Escuela Profesional de Ingeniería de Sistemas.

Sres. Miembros del Jurado.

De conformidad con las disposiciones del Reglamento de Grados y Títulos de la Escuela Profesional de Ingeniería de Sistemas, ponemos a vuestra consideración el presente trabajo de investigación titulado:

“Mecanismo Ágil para la Gestión y Desarrollo de Proyectos de Software, aplicado en instituciones de Seguros y de Telecomunicaciones”; el mismo que de ser aprobado, me permitirá optar el Título Profesional de Ingeniería de Sistemas.

Bach. Karen Flores Roca.

# DEDICATORIA

Con todo mi amor y agradecimiento a Dios, quien guía mis pasos cada día, aquel que me dio la oportunidad de ser esposa y madre.

Para mi amado esposo, Víctor Manuel González Valenzuela por su amor y cuidado y para mi tesoro, mi hijo, Sebastián Mateo González Flores, mi fuerza motivadora.

Con amor y cariño a mis padres y a mi querida hermana; Jhon Remy Flores Carnero, Ruth Roca Talavera de Flores y Karla Flores Roca, quienes siempre me apoyaron, confiaron y esforzaron a ser una mujer de bien.

Con cariño para mi querido abuelo, Luis Alonso Flores Molina, quien lleno mi infancia de sonrisas, lindas experiencias y graciosas historias.

Para mis abuelas, Vilma Carnero y Rosa Talavera, por enseñarme a ser valiente, esforzada y a emprender nuevos retos, las llevo siempre en mi corazón.

Con afecto a mis tíos Víctor González y Sonia Valenzuela, gracias por su cariño y sus consejos.

# RESUMEN

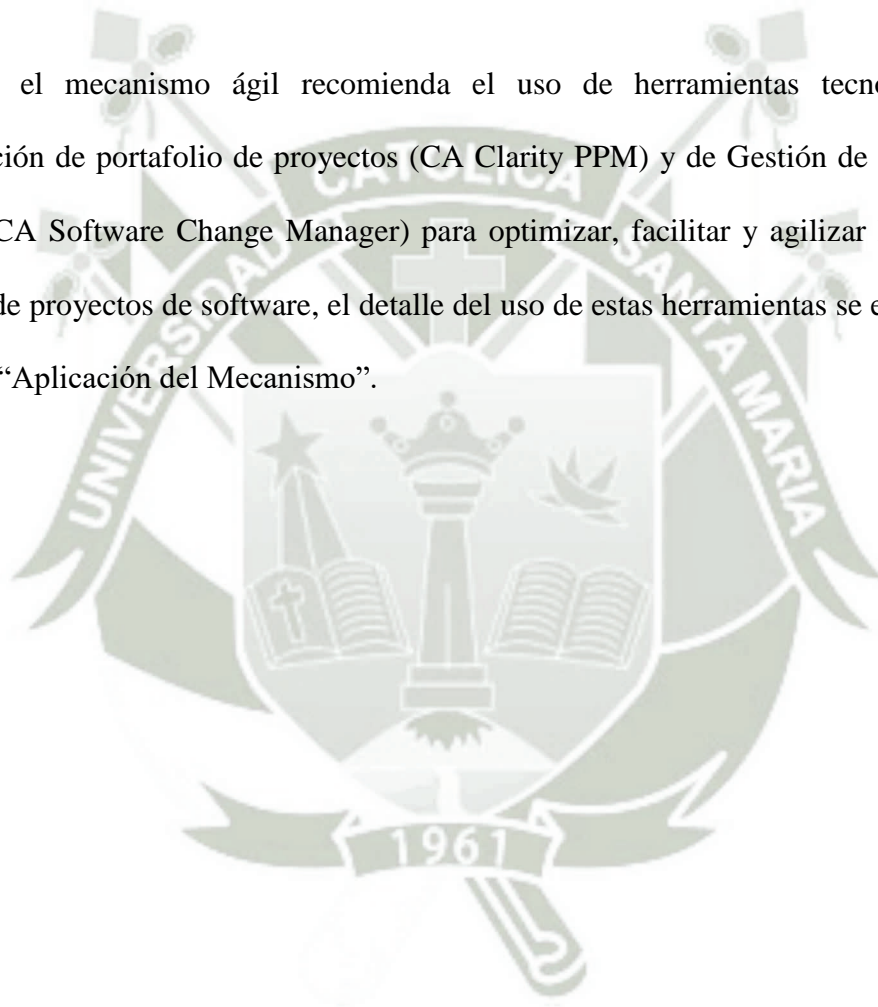
La alta competitividad de los conglomerados empresariales, hacen que se requiera una rápida respuesta ante el requerimiento de tareas o proyectos de software de tipo estratégicos, tácticos u operacionales, ello dependiendo de la necesidad de la empresa, para ello es necesario que la empresa o institución cuente con una cultura de gestión de proyectos de software ágil, optima, ordenada y práctica que imparta responsabilidades a cada uno de los participantes en el desarrollo y gestión de proyectos de software desde su inicio hasta su cierre, de manera que los cambios que puedan surgir durante el ciclo de vida del proyecto o su producto puedan ser gestionados rápidamente sin afectar a otros proyectos pudiendo disponer del producto o servicio en el mejor de los casos, libre de errores, incidencias, incongruencias o incumplimiento de los requerimientos del usuario.

De esta manera el mecanismo ágil que se elaboró es una guía práctica para desarrollar y gestionar proyectos de software de complejidad baja, mediana y alta, desde su inicio hasta su cierre, dando a conocer las etapas y fases por las que pasa el proyecto, como también los roles, actividades y artefactos (documentos) de cada participante en el desarrollo y gestión de proyectos de software.

El uso de artefactos (documentos) apoyarán el desarrollo y facilitarán la gestión, seguimiento y control durante el ciclo de vida del proyecto, evitando la generación de artefactos que no sean necesarios o que no apliquen según la complejidad de cada proyecto.

El mecanismo ágil propuesto, se elaboró con la recopilación de lecciones aprendidas y experiencias laborales en desarrollo y gestión de proyectos. Por lo que se incluyen casos de estudio para analizar la problemática de proyectos en una etapa y fase específica, rescatando lo que se debió de haber hecho para evitar re – planificaciones, re – trabajo, solicitudes de cambio u otro tipo de inconvenientes que pueden surgir durante el desarrollo y gestión del proyecto de software.

Finalmente el mecanismo ágil recomienda el uso de herramientas tecnológicas de administración de portafolio de proyectos (CA Clarity PPM) y de Gestión de Cambios de Software (CA Software Change Manager) para optimizar, facilitar y agilizar la gestión y desarrollo de proyectos de software, el detalle del uso de estas herramientas se expone en el Capítulo 4 “Aplicación del Mecanismo”.



## Índice

<b>CAPITULO 1: PLANTEAMIENTO DEL PROBLEMA</b>	<b>13</b>
<b>1.1. Planteamiento del problema</b>	<b>13</b>
<b>1.2. Objetivos</b>	<b>15</b>
a) General	15
b) Específicos	16
<b>1.3. Preguntas</b>	<b>16</b>
<b>1.4. Tipo y Nivel de Investigación</b>	<b>16</b>
1.4.1. Tipo de Investigación	16
1.4.1. Nivel Investigación	16
<b>1.5. Solución propuesta</b>	<b>17</b>
<b>1.6. Justificación</b>	<b>17</b>
<b>1.7. Descripción de la solución</b>	<b>17</b>
1.7.1. Caso de estudio	18
1.7.1.1. Explicación de proyecto	18
1.7.1.2. Incidencias	18
1.7.1.3. Consideraciones	18
<b>1.8. Alcances</b>	<b>23</b>
<b>1.9. Palabras Clave</b>	<b>23</b>
<b>CAPITULO 2: MARCO TEORICO</b>	<b>23</b>
<b>2.1. Estado del arte</b>	<b>24</b>
<b>2.2. Marco de gestión de proyectos según PMBOK</b>	<b>25</b>
2.2.1. Proyecto	25
2.2.1.1. Gestión de proyectos	26
2.2.1.2. Involucrados del Proyecto (Stakeholders)	26
2.2.1.3. Portafolio de proyectos	29
2.2.1.4. PMO (Oficina de Proyectos)	29
2.2.2. Entender el ciclo de vida del proyecto	30
2.2.2.1. Inicio de un proyecto	31
2.2.2.2. Planificación del proyecto	34
2.2.2.3. Ejecución del Proyecto	36
2.2.2.4. Seguimiento y control del proyecto	38
2.2.2.5. Cierre del proyecto	41
<b>2.3. Marco de gestión de proyectos ágil</b>	<b>45</b>
2.3.1. SCRUM	45
2.3.2. Historia ágil y una visión general	46
2.3.3. Objetivos de la agilidad	49
2.3.4. Comparación principios ágiles y tradicionales	51

2.3.5.	Enfoque ágil _____	56
2.3.5.1.	Enfoque ágil para Proyectos _____	56
2.3.5.2.	Un equipo ágil trabaja como uno _____	56
2.3.5.3.	Equipos Ágiles trabajan en iteraciones cortas _____	57
2.3.5.4.	Equipos ágiles entregan partes funcionales en cada iteración _____	57
2.3.5.5.	Equipos ágiles en las prioridades del negocio _____	58
2.3.6.	Conceptos en proyectos ágiles _____	60
2.3.6.1.	Historia _____	60
2.3.6.2.	“Sprint” / Iteración _____	60
2.3.6.3.	Backlog _____	60
2.3.6.4.	Reunión de stand-up _____	60
2.3.7.	Planeación, monitoreo y adaptación _____	61
2.3.7.1.	Planificación adaptativa _____	61
2.3.7.2.	Iteración y planificación de liberación _____	62
2.3.7.3.	Diario de planificación _____	63
2.3.7.4.	Seguimiento del proyecto ágil _____	63
2.3.7.4.1.	Radiadores de Información _____	63
2.3.7.4.2.	Autoevaluación _____	64
2.3.8.	Gestión de la calidad del proyecto _____	64
2.3.8.1.	Introducción _____	64
2.3.8.2.	Costo de la calidad _____	65
2.3.8.3.	Test Driven Development (TDD) _____	66
2.3.8.4.	Integración continua _____	66
2.3.8.5.	Entrega incremental _____	67
2.3.9.	Gestión de las comunicaciones del proyecto _____	67
2.3.9.1.	Gerencia de comunicaciones _____	67
2.3.10.	Métricas ágiles y estimaciones _____	69
2.3.10.1.	Puntos de historia _____	69
2.3.10.2.	Tiempo ideal _____	69
2.3.10.3.	Tiempo ideal y Desarrollo de Software _____	70
2.3.11.	Gestión de costos del proyecto ágil _____	71
2.3.11.1.	Gestión de costos del proyecto _____	71
2.3.12.	Gestión de riesgos del proyecto ágil _____	72
2.3.12.1.	Plan de gestión de riesgos _____	72
2.3.12.2.	Identificar los riesgos _____	73
2.3.12.3.	Plan de respuestas del riesgo _____	74
2.3.12.4.	Vigilancia y control de riesgos _____	74
2.3.12.5.	Puntos de verificación en proyectos ágiles _____	74
2.3.12.6.	Errores comunes de gestión de riesgos _____	75
2.3.13.	OpenUp _____	75
2.3.13.1.	Definición _____	75
2.3.13.2.	Principios básicos _____	75
2.3.13.2.1.	Colaborar para alinear los intereses y un entendimiento compartido _____	76
2.3.13.2.2.	Balancear las prioridades para maximizar el valor de los stakeholder _____	76
2.3.13.2.3.	Enfocarse en articular la arquitectura _____	76
2.3.13.2.4.	Evolucionar para obtener continuamente retroalimentación y progreso _____	77

2.3.13.3.	Fases _____	77
2.3.13.3.1.	Fase de Inicio _____	77
2.3.13.3.2.	Fase de Elaboración _____	78
2.3.13.3.3.	Fase de Construcción _____	79
2.3.13.3.4.	Fase de Transición _____	80
2.3.13.4.	Iteración _____	81
2.3.13.5.	Beneficio de las iteraciones _____	81
2.3.13.6.	Caso de uso _____	81
2.3.13.7.	Arquitectura de software _____	83
2.3.13.8.	Roles _____	84
2.3.14.	Proceso Unificado de Rational (RUP) _____	84
2.3.14.1.	Características esenciales _____	84
2.3.14.1.1.	Dirigido por Casos de Uso _____	84
2.3.14.1.2.	Proceso centrado en la arquitectura _____	84
2.3.14.1.3.	Proceso iterativo e incremental _____	86
<b>CAPITULO 3: MECANISMO PROPUESTO</b> _____		<b>88</b>
<b>3.1.</b>	<b>Resumen de mecanismo</b> _____	<b>88</b>
<b>3.2.</b>	<b>Solución propuesta</b> _____	<b>88</b>
<b>3.3.</b>	<b>Generalidades</b> _____	<b>88</b>
3.3.1.	Etapas y fases _____	89
3.3.1.1.	Roles _____	89
3.3.1.2.	Actividades y artefactos _____	91
<b>3.4.</b>	<b>Mecanismo ágil de gestión y desarrollo de proyectos de software</b> _____	<b>104</b>
3.4.1.	Proceso de iniciación _____	104
3.4.1.1.	Asignar Líder del proyecto _____	105
3.4.1.2.	Elaborar Ficha de Inicio del Proyecto _____	106
3.4.1.3.	Registrar Proyecto en el PPM _____	106
3.4.2.	Proceso de planificación _____	107
3.4.3.	Reunión de Kick – Off _____	107
3.4.3.1.	Acta de Reunión _____	108
3.4.3.2.	Elaborar Checklist del Proyecto _____	108
3.4.3.3.	Fase Análisis _____	109
3.4.3.3.1.	Elaborar Documento de Especificación Funcional _____	110
3.4.3.3.2.	Línea base de alcance _____	110
3.4.3.4.	Fase de Diseño _____	111
3.4.3.4.1.	Elaborar Documento de especificación técnica _____	112
3.4.3.4.2.	Elaborar Plan de Pruebas Unitarias _____	113
3.4.3.4.3.	Elaborar Plan de Pruebas Integrales _____	114
3.4.3.4.4.	Elaborar Plan de Pruebas de Usuario _____	114
3.4.3.5.	Elaborar Plan de Gestión de Comunicaciones _____	114
3.4.3.6.	Elaborar Plan de gestión de riesgos _____	114
3.4.3.7.	Plan de gestión de cambios _____	115
3.4.3.7.1.	Nivel de Impacto _____	115
3.4.3.7.2.	Roles de la gestión y aprobación de cambios _____	116

3.4.3.7.3.	Plan de contingencia ante solicitudes de cambio urgentes _____	117
3.4.3.7.4.	Documento de solicitud de cambio _____	117
3.4.3.8.	Planeamiento de proyecto _____	117
3.4.3.8.1.	Comité técnico _____	118
3.4.3.8.2.	Gestión de aprobaciones _____	119
3.4.3.8.3.	Priorización y estimación de desarrollo _____	120
3.4.3.8.4.	Registrar documentos en el sistema PPM _____	120
3.4.3.8.5.	Elaborar matriz de riesgos _____	120
3.4.3.8.6.	Línea Base del Tiempo _____	120
3.4.4.	Proceso de ejecución _____	121
3.4.4.1.	Fase de Desarrollo _____	121
3.4.4.1.1.	Caso de estudio _____	122
3.4.4.1.1.1.	Explicación de proyecto _____	122
3.4.4.1.1.2.	Incidencias en fase de desarrollo _____	122
3.4.4.1.1.3.	Recomendaciones y/o lecciones aprendidas _____	123
3.4.4.1.2.	Elaborar Evidencias de Pruebas Unitarias _____	125
3.4.4.1.3.	Elaborar Evidencias de Pruebas Integrales o Funcionales _____	125
3.4.4.1.4.	Publicar los documentos del proyecto en el PPM _____	125
3.4.4.1.5.	Publicar los objetos del desarrollo en el SCM _____	126
3.4.4.1.6.	Elaborar Manual de usuario _____	127
3.4.4.2.	Transición _____	128
3.4.4.2.1.	Caso de estudio _____	128
3.4.4.2.1.1.	Explicación de proyecto _____	128
3.4.4.2.1.2.	Incidencias en fase de Certificación (QA) _____	128
3.4.4.2.1.3.	Recomendaciones y/o lecciones aprendidas _____	128
3.4.4.2.2.	Certificación del proyecto _____	129
3.4.4.2.3.	Producción y estabilización del proyecto _____	130
3.4.5.	Seguimiento y Control _____	130
3.4.5.1.	Dirigir y Gestionar la Ejecución del Proyecto _____	130
3.4.5.2.	Ejecutar seguimiento y control del Proyecto _____	131
3.4.5.3.	Gestionar y Monitorear los Riesgos del Proyecto _____	131
3.4.5.4.	Ingresar y Actualizar Entregables del Proyecto _____	133
3.4.5.5.	Ingresar y actualizar bitácora del proyecto _____	133
3.4.5.6.	Atención de solicitud de cambios _____	134
3.4.5.7.	Registrar lecciones aprendidas _____	134
3.4.6.	Cierre _____	135
3.4.6.1.	Realizar cierre del proyecto _____	135
3.4.6.2.	Registrar cierre de Proyecto en el PPM _____	135
	<b>CAPITULO 4: APLICACIÓN DE MECANISMO _____</b>	<b>137</b>
	<b>4.1. Proyecto de Automatización del Tablero de Control _____</b>	<b>137</b>
4.1.1.	Introducción _____	137
4.1.2.	Proceso de iniciación _____	137
4.1.2.1.	Asignar Líder del proyecto _____	137
4.1.2.2.	Registrar Proyecto en el CA Clarity PPM _____	137
4.1.3.	Proceso de planificación _____	137

4.1.3.1.	Reunión de Kick – Off _____	137
4.1.3.2.	Etapa Análisis _____	138
4.1.3.2.1.	Elaborar Documento de Especificación Funcional _____	138
4.1.3.3.	Etapa de Diseño _____	138
4.1.3.3.1.	Elaborar Documento de especificación técnica _____	138
4.1.3.4.	Elaborar Plan de gestión de riesgos _____	138
4.1.4.	Proceso de ejecución _____	139
4.1.4.1.	Etapa de Desarrollo _____	139
4.1.4.1.1.	Elaborar Evidencias de Pruebas Integrales _____	139
4.1.4.1.2.	Publicar los documentos del proyecto en el PPM _____	139
4.1.4.1.3.	Publicar los objetos del desarrollo en el SCM _____	140
4.1.5.	Seguimiento y Control _____	<b>¡Error! Marcador no definido.</b>
4.1.5.1.	Dirigir y Gestionar la Ejecución del Proyecto _____	140
4.1.5.2.	Ejecutar seguimiento y control del Proyecto _____	141
4.1.5.3.	Gestionar y Monitorear los Riesgos del Proyecto _____	141
4.1.5.4.	Ingresar y actualizar bitácora del proyecto _____	141
<b>CAPITULO 5: EVALUACIÓN DE MECANISMO _____</b>		<b>142</b>
<b>5.1.</b>	<b>Método de evaluación _____</b>	<b>142</b>
<b>5.2.</b>	<b>Resultados de evaluación _____</b>	<b>143</b>
<b>5.3.</b>	<b>Información de Expertos _____</b>	<b>147</b>
<b>CONCLUSIONES _____</b>		<b>148</b>
<b>BENEFICIOS _____</b>		<b>148</b>
<b>RECOMENDACIONES _____</b>		<b>149</b>
<b>REFERENCIAS _____</b>		<b>151</b>
<b>ANEXO A _____</b>		<b>151</b>
FICHA DE INICIO _____		153
	FI_NombreProyecto_FichaInicio_CódigoProyecto_versión_Fecha.docx _____	153
KICK OFF _____		157
	KO_NombreProyecto_Kick Off_CódigoProyecto_versión_Fecha.ppt _____	157
ACTA DE REUNIÓN _____		162
	AR_NombreProyecto_Acta_Reunion_CódigoProyeccto_versión_Fecha.docx _____	162
CHECK LIST _____		164
	CL_NombreProyecto_CheckListProyecto_CódigoProyecto_versión_Fecha.xls _____	164
ESPECIFICACIÓN FUNCIONAL _____		172
	EF_NombreProyecto_EspecificaciónFuncional_CódigoProyecto_versión_Fecha.docx _____	172
ESPECIFICACIÓN TÉCNICA _____		176
	ET_NombreProyecto_EspecificaciónTécnica_CódigoProyecto_versión_Fecha.docx _____	176
PLAN DE COMUNICACIONES _____		179
	PC_NombreProyecto_PlanComunicaciones_CódigoProyecto_versión_Fecha.xlsx _____	179
MATRIZ DE RIESGOS _____		180
	MR_NombreProyecto_MatrizRiesgos_CódigoProyecto_versión_Fecha.xls _____	180
PRUEBAS FUNCIONALES _____		181
	PI_NombreProyecto_PruebasIntegrales_CódigoProyecto_versión_Fecha.docx _____	181

PLAN DE GESTIÓN DE CAMBIOS _____	182
GC_NombreProyecto_GestiónCambios_CódigoProyecto_versión_Fecha.docx _____	182
SOLICITUD DE CAMBIOS _____	184
SC_NombreProyecto_SolicitudCambio_CódigoProyecto_nroSRTI_versión_Fecha.docx _____	184
MANUAL DE INSTALACIÓN _____	<b>¡Error! Marcador no definido.</b>
MI_NombreProyecto_ManualInstalación_CódigoProyecto_versión_Fecha.docx _____	<b>¡Error! Marcador no definido.</b>
MANUAL DE OPERACIÓN _____	<b>¡Error! Marcador no definido.</b>
MI_NombreProyecto_ManualOperación_CódigoProyecto_versión_Fecha.docx _____	<b>¡Error! Marcador no definido.</b>
MANUAL DE USUARIO _____	186
MU_NombreProyecto_ManualUsuario_CódigoProyecto_versión_Fecha.docx _____	186
INFORME DE SALUD DEL PROYECTO _____	188
IS_NombreProyecto_InformeSaludProyecto_CódigoProyecto_versión_Fecha.pptx _____	188
INFORME DE MONITOREO DE RIESGOS _____	192
IR_NombreProyecto_InformeRiesgos_CódigoProyecto_versión_Fecha.docx _____	192
FICHA DE CIERRE DEL PROYECTO _____	194
FC_NombreProyecto_FichaCierre_CódigoProyecto_versión_Fecha.docx _____	194
<b>ANEXO B _____</b>	<b>196</b>
KICK OFF _____	196
ESPECIFICACIÓN FUNCIONAL _____	198
ESPECIFICACIÓN TÉCNICA _____	203
MATRIZ DE RIESGOS _____	208
PRUEBAS INTEGRALES _____	211
INFORME DE SALUD _____	217

## Lista de Figuras

Figura 1. Descripción de problemas comunes en la gestión y desarrollo de proyectos de software. ____	14
Figura 2. Flujo de procesos, Proyecto de Automatización de Suspensiones. ____	<b>¡Error! Marcador no definido.</b>
Figura 3. Fechas del Proyecto (Herramienta Ca. Clarity). _____	<b>¡Error! Marcador no definido.</b>
Tabla 4. Actividades y áreas responsables. _____	20
Figura 5. Diferentes estados en el ciclo de vida de un proyecto. Cada uno de estos estados representan un grupo de procesos _____	31
Figura 6. Ilustración de las relaciones entre los principales bloques de construcción del panorama general de iniciación de un proyecto _____	34
Figura 7. Una vista de alto nivel de las interacciones y el flujo de datos entre los diferentes componentes de la planificación de proyectos. _____	36
Figura 8. Grupo del proceso de ejecución _____	38
Figura 9. Escenario del grupo de proceso de monitoreo y control _____	40
Tabla 10. Etapas del ciclo de vida de un proyecto: los grupos de procesos del proyecto. _____	45
Figura 11. Radiador de la Información. _____	64
Figura 12. Evolución de la arquitectura durante las fases de RUP. _____	85
Figura 13. Iteración RUP _____	86
Tabla 14. Actividades y roles del mecanismo propuesto _____	92
Figura 15. Etapas y fases del mecanismo ágil de gestión y desarrollo de proyectos de software. ____	104
Figura 16. Subprocesos del proceso de iniciación _____	107
Figura 17. Subprocesos de la fase de Análisis _____	111
Figura 18. Subprocesos en la fase de Diseño. _____	113
Cuadro 19. Roles para la Gestión y Aprobación de Cambios _____	116
Figura 20. Flujo de subprocesos para la planificación del proyecto _____	118
Figura 21. Subprocesos en la fase de Desarrollo _____	122
Figura 22. Fechas de proyecto _____	<b>¡Error! Marcador no definido.</b>
Figura 23. Acta de reunión. _____	124
Figura 24. Cronograma del proyecto (Estructura de trabajo desglosada). _____	131
Figura 25. Riesgos del proyecto. _____	133
Figura 26. Documentación del proyecto. _____	133
Figura 28. Solicitud de cambio del proyecto. _____	134
Figura 31. Subprocesos de cierre _____	136

## CAPITULO 1: PLANTEAMIENTO DEL PROBLEMA

### 1.1. Planteamiento del problema

Actualmente, los negocios operan en un entorno global que cambia rápidamente. Tienen que responder a nuevas oportunidades y mercados, condiciones económicas cambiantes y la aparición de productos y servicios competidores. El software es parte de casi todas las operaciones de negocio, por lo que es fundamental que el software nuevo se desarrolle rápidamente para aprovechar nuevas oportunidades y responder a la presión competitiva. Por lo tanto, actualmente el desarrollo y entrega rápidos son a menudo los requerimientos más críticos de los sistemas de software. De hecho, muchas compañías están dispuestas a una pérdida en la calidad del software en favor de una entrega rápida.

Debido a que estas compañías operan en un entorno cambiante, a menudo es prácticamente imposible obtener un conjunto completo de requerimientos de software estables. Los requerimientos que se proponen cambian inevitablemente, porque a los clientes les resulta imposible predecir cómo afectará un sistema a la manera de trabajar, cómo interactuará con otros sistemas y qué operaciones de los usuarios se deben automatizar. Es posible que los requerimientos reales sólo queden claros cuando se haya entregado el sistema y los usuarios hayan adquirido experiencia. Cuando los requerimientos cambian o cuando se descubren problemas con ellos, el diseño o implementación del sistema se tiene que volver a realizar o probar. Como consecuencia, normalmente se prolonga en el tiempo un proceso en cascada convencional o basado en la especificación y el software definitivo se entrega al cliente mucho tiempo después de que fuera inicialmente especificado.

En un entorno de negocios que se mueve con rapidez, esto puede causar verdaderos problemas. Para cuando esté disponible el software, la razón original de su adquisición puede haber cambiado tan radicalmente que el software sea en realidad inútil. Por lo tanto, en particular para los sistemas de negocio, los procesos de desarrollo que se basan en el desarrollo y entrega rápidos de software son esenciales.

La Figura 1 ilustra los problemas más comunes vivenciados en empresas de seguros, tecnologías de información y de telecomunicaciones agrupado por los indicadores de productividad, calidad y flexibilidad:

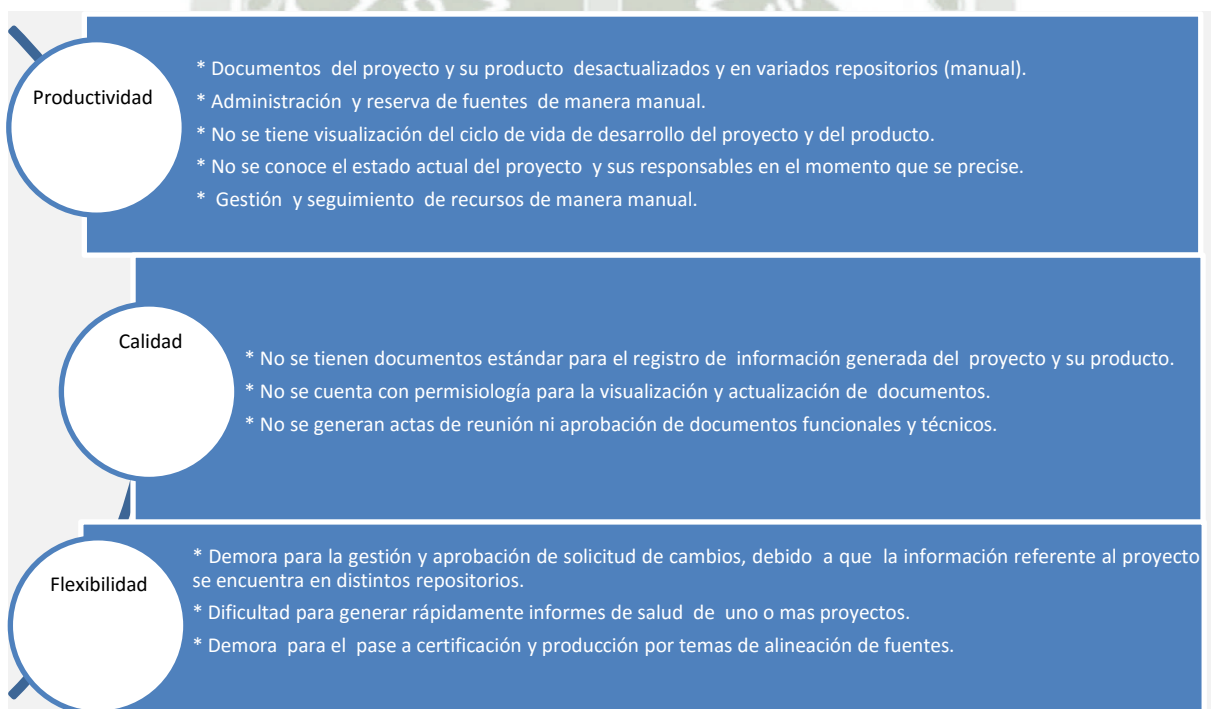


Figura 1. Descripción de problemas comunes en la gestión y desarrollo de proyectos de software.

Fuente propia.

Es por ello que el presente trabajo de investigación, provee un mecanismo ágil, que sirva de guía práctica para la gestión y el desarrollo óptimo, ordenado, ágil y práctico del ciclo de vida de uno o más proyectos y sus productos ya sea que tengan una complejidad, baja, media o alta.

El mecanismo ágil pondrá a disposición del lector, procedimientos e información sobre las etapas y fases por las que pasa el proyecto a desarrollar y de las responsabilidades de los actores que participan en la gestión y desarrollo del mismo, dando recomendaciones y/o lecciones aprendidas, mediante casos de estudio reales a fin de minimizar las re – planificaciones, re – trabajo, solicitudes de cambio, demoras o el rechazo del proyecto por parte del usuario, también sugiere el uso de formatos y documentos que facilitarán la gestión y seguimiento durante el ciclo de vida del proyecto, evitando la generación de documentos que no serán necesarios o que no aplican según la complejidad del proyecto.

Finalmente el mecanismo ágil recomienda el uso de herramientas tecnológicas de administración de portafolio de proyectos (CA Clarity PPM) y de gestión de cambios de software (CA Software Change Manager) para optimizar, facilitar y agilizar la gestión y desarrollo de proyectos de software.

## **1.2. Objetivos**

### **a) General**

Entregar una guía de trabajo (mecanismo ágil) para la gestión de proyectos y desarrollo de software utilizando la sinergia de las metodologías PMBOK y SCRUM.

## **b) Específicos**

1. Disminuir los riesgos del proyecto.
2. Optimizar la gestión del proyecto y su producto.
3. Flexibilidad para realizar cambios de software, pases a certificación y producción.
4. Formalización de documentos para la gestión y desarrollo de proyectos de software.

### **1.3.Preguntas**

El trabajo de investigación nos plantea las siguientes preguntas:

- ¿El mecanismo propuesto incrementa la productividad de los recursos durante la gestión y desarrollo de proyectos de software?
- ¿El mecanismo propuesto mejora la calidad del proyecto y su producto?
- ¿El mecanismo propuesto permite la flexibilidad para la gestión de cambios durante el proyecto?
- ¿El mecanismo propuesto optimiza la gestión, planificación, ejecución y control del proyecto?

### **1.4. Tipo y Nivel de Investigación**

#### **1.4.1. Tipo de Investigación**

Tecnológica

#### **1.4.1. Nivel Investigación**

Experimental

### **1.5. Solución propuesta**

MECANISMO ÁGIL PARA LA GESTIÓN Y DESARROLLO DE PROYECTOS DE SOFTWARE, APLICADO EN INSTITUCIONES DE SEGUROS Y DE TELECOMUNICACIONES.

### **1.6. Justificación**

Actualmente se vienen utilizando métodos de desarrollo y gestión de proyectos de software clásicos y ágiles de manera aislada, es decir, no se tiene un mecanismo concreto que combine las bondades de los métodos clásicos y ágiles para gestionar y desarrollar proyectos de software. Por lo que el presente trabajo de investigación, provee un mecanismo ágil que combina las mejores prácticas de ambos enfoques, sirviendo de guía práctica para la gestión y el desarrollo óptimo, ordenado, ágil y práctico del ciclo de vida de uno o más proyectos de software y sus productos ya sea que tengan una complejidad, baja, media o alta.

### **1.7. Descripción de la solución**

El mecanismo ágil pondrá a disposición del lector, procedimientos e información sobre las etapas y fases por las que pasa el proyecto a desarrollar y de las responsabilidades de los actores que participan en la gestión y desarrollo del mismo, dando recomendaciones y/o lecciones aprendidas, mediante casos de estudio reales a fin de minimizar las re – planificaciones, re – trabajo, solicitudes de cambio, demoras o el rechazo del proyecto por parte del usuario, también sugiere el uso de formatos y documentos que facilitarán la gestión y seguimiento durante el ciclo de vida del proyecto, evitando la generación de documentos que no serán necesarios o que no aplican según la complejidad del proyecto.

Finalmente el mecanismo recomienda el uso de herramientas tecnológicas de administración de portafolio de proyectos (CA Clarity PPM) y de gestión de cambios de software (CA Software Change Manager) para optimizar, facilitar y agilizar la gestión y desarrollo de proyectos de software.

### **1.7.1. Caso de estudio**

#### **1.7.1.1. Explicación de proyecto**

Se solicita desarrollar el Proyecto de automatización de Transacciones de manera automática.

#### **1.7.1.2. Consideraciones**

Es importante en una empresa mediana o grande, donde se gestionan y desarrollan en simultáneo números considerables de Proyectos de software de complejidad baja, media o alta, contar con una cultura de gestión y desarrollo de proyectos ágil, ordenada, práctica y flexible, de no contar con un mecanismo de gestión y desarrollo de proyectos de software, la gestión se tornará inmanejable, ya que es necesario establecer responsabilidades, etapas y fases de trabajo cada una ellas con sus documentos correspondientes para administrarlas de la manera más óptima posible. De esta manera podremos gestionar, desarrollar y controlar óptimamente los proyectos en sus distintos estados, como son, inicio, planificación, ejecución y cierre.

Respecto al proyecto del caso de estudio, es un proyecto que deberá de ser puesto en “stand by”, hasta que se terminen de actualizar las plataformas, pues el proyecto depende de ellas para continuar con su ejecución, estando el proyecto en estado de “stand by”, los recursos asignados al proyecto serán liberados y re asignados a otros proyectos, hasta que el estado del proyecto retorne a ejecución.

El riesgo de poner el proyecto en “stand by”, es, que cuando se retome la ejecución del proyecto, se deberá de evaluar el tiempo en que los recursos re – asignados se liberarán de sus proyectos, en algunos casos pueden ser días y en otros semanas o meses, esto generará retrasos, el líder del proyecto deberá de coordinar la liberación de sus recursos y evaluar el tiempo en que dispondrá de ellos o si es necesario asignar nuevos recursos al proyecto, considerando la curva de aprendizaje de cada uno de ellos, en muchos casos resulta conveniente continuar con los recursos que iniciaron el desarrollo debido al conocimiento (“Know – How”) que tienen del proyecto, esto depende de cuan complejo o técnico sea el proyecto, de ser un proyecto sencillo donde la curva de aprendizaje es mínima, es recomendable asignar nuevos recursos y evitar demoras por liberación de los recursos iniciales.

En la gran mayoría de proyectos, el desarrollo de software, involucra crear objetos de bases de datos (procedimientos almacenados, tablas, funciones, etc.), objetos de interfaz (servicios web), objetos de código fuente en distintos lenguajes de programación y para diferentes aplicativos de negocio. El mecanismo ágil recomienda la división del desarrollo de software en grupos o áreas, por ejemplo, para el proyecto anteriormente mencionado, el analista funcional analizará sus requerimientos de desarrollo de software, identificando las áreas responsables de dichos desarrollos, la Tabla 4 ilustra un ejemplo de las actividades a realizar y las áreas encargadas para llevar a cabo el desarrollo del proyecto:

Actividades a realizar	Área o Equipo encargado
Elaboración de Propuesta de solución (Especificación Funcional y Técnica)	Líder de proyecto (Analista Funcional)

Elaboración de documento de Arquitectura conceptual	Arquitecto de Solución
Elaboración de documento de Arquitectura de datos	Arquitecto de Datos
Elaboración de documento de Arquitectura de servicios	Arquitecto de Servicios e Integración
Elaboración de documento de Arquitectura de plataforma	Arquitecto de Plataforma
Elaboración de documento de Arquitectura de aplicación	Arquitecto de Aplicación
Elaboración de documento de Arquitectura de Producto	Arquitecto de Producto
Actualización y creación de procedimientos almacenados	Equipo de desarrollo según el aplicativo impactado
Creación de tablas, triggers e índices	Equipo de desarrollo según el aplicativo impactado
Creación de servicios web	Área de Arquitectura e integración

Tabla 4. Actividades y áreas responsables.  
Fuente propia.

Esta manera de trabajo, permitirá tener objetos de software o módulos específicos y menos complejos que sean prácticos de probar, verificar o actualizar, generando responsabilidad por cada uno ellos en las áreas encargadas de desarrollarlas.

De manera distinta sucede, cuando le encargamos a una o más personas de un mismo equipo de trabajo el desarrollo de todos los objetos y recursos de software del proyecto. Al subdividir por áreas o equipos, cada una de ellas tiene conocimiento de sus propias normas y estándares que deben de cumplir para sus desarrollos, por lo que al momento de hacer el pase a certificación se minimizan las devoluciones de paquetes de software por incumplimiento de normas y estándares de programación o calidad.

Al contar con desarrollos por áreas especializadas, es posible, generar reuniones con los encargados de cada área antes de empezar la ejecución del proyecto, para explicar la planificación del proyecto y recibir recomendaciones técnicas o funcionales a tomar en cuenta en el desarrollo, esto gracias al “know how” que tiene cada área y sus miembros, a este tipo de reuniones les denominaremos “comités técnicos” y serán la última oportunidad para adecuar la planificación (especificación funcionales y técnica) del proyecto sin generar solicitudes de cambio, luego de que el proyecto cambie del estado de planificación a ejecución, cualquier cambio, deberá de ser gestionada con la respectiva solicitud de cambio.

Como beneficio adicional, esta manera de trabajo le facilita la gestión y seguimiento del proyecto al líder de proyecto (analista funcional), ya que conoce el estado actual de cada desarrollo y el responsable de entregarla.

Cabe resaltar que en las actividades mencionadas anteriormente se incluyen documentos de arquitectura, es beneficioso contar con la participación de un Arquitecto de Solución en el proyecto ya que confirmará el uso de Bases de datos, tablas, “Store Procedures”, campos, servicios web que deban y puedan re-utilizarse o aquellos que deberán de crearse.

En caso considere necesario y así lo demande el Arquitecto de Solución, podrá solicitar la asignación de un arquitecto especialista de datos, servicios e integración, aplicación, plataforma y/o producto.

Para explicar brevemente las ventajas de incluir los roles de arquitectos en nuestro mecanismo vamos a considerar que en nuestro proyecto es necesario modificar un “Store Procedure” (SP) para agregar un campo de entrada adicional, este “Store Procedure” es agregado en la especificación funcional (EF) por parte del Analista Funcional, el Arquitecto de Solución al recibir el avance de la EF valida el uso del SP, verifica que funciona correctamente en producción y verifica el impacto que pueda tener su modificación, sin embargo no cuenta con la suficiente información como para confirmar que su modificación no impactará en otras aplicaciones que lo consuman, por lo que solicita al Jefe de Proyectos la asignación de un Arquitecto de Datos para que realice una auditoría al SP e identifique las aplicaciones que lo consumen.

En caso se tengan aplicaciones impactadas por la modificación del SP estas deberán de ser incluidas como parte del desarrollo del proyecto, debiendo de asignar un Analista programador (AP) por aplicativo impactado.

Los entregables del Arquitecto de solución será: Documento de Arquitectura conceptual (se aprueba junto con la especificación funcional y se adjunta a la especificación técnica), Documento de Arquitectura Lógica (Documento input para los AP, para la elaboración de la especificación de diseño de aplicativo o servicio) y Documento de Arquitectura Física (Documento usado por los Analistas de soporte de aplicaciones para la instalación y configuración del desarrollo en el ambiente de Producción).

Estas y muchas otras consideraciones y recomendaciones se exponen a lo largo del presente trabajo de investigación.

## 1.8. Alcances

Proyectos de Software de complejidad baja, media y alta.

## 1.9. Palabras Clave

- PMBOK
- SCRUM
- OpenUp
- RUP
- Historias
- Especificación Funcional
- Especificación Técnica
- Acta de Reunión
- Kick Off
- Informe de Salud
- Informe de Riesgos
- Comité técnico

## CAPITULO 2: MARCO TEORICO

El marco teórico del presente trabajo de investigación contempla los conceptos, métodos y procesos relacionados a la gestión de proyectos (PMBOK, SCRUM) y desarrollo de software (OpenUp, RUP); los cuales combinados con la experiencia de la investigadora dan como resultado el mecanismo propuesto, el que es: “Mecanismo ágil para la gestión y desarrollo de proyectos de software, aplicado en Instituciones de Seguros y de Telecomunicaciones”.

## **2.1. Estado del arte**

### **2.1.1. Propuesta de un modelo de asociación en el gerenciamiento de riesgos de proyectos informáticos basado en el estándar PMBOK y CMMI-SW.**

#### **2.1.1.1. Problema principal**

Los proyectos de software para aplicaciones comerciales son claramente difíciles de administrar y una gran cantidad de ellos terminan en fracaso. En un proyecto de software, éste se puede traducir en una mala calidad del producto, incumplimiento de planes u objetivos y hasta el fracaso del proyecto. La gestión de riesgos en proyectos de software pretende identificar, estudiar y eliminar las fuentes de riesgo antes de que comiencen a amenazar el éxito o la finalización exitosa de un proyecto de desarrollo de software. Se define el riesgo como la posibilidad que un evento adverso, desgracia o contratiempo pueda manifestarse produciendo una pérdida. El riesgo es una posibilidad futura, por lo tanto una gestión adecuada puede determinar la ocurrencia o no ocurrencia de éstos. La Identificación de Riesgos en proyectos de software consiste en la determinación de elementos de riesgos potenciales mediante la utilización de algún método consistente y estructurado; este es, probablemente, el paso más importante entre todos aquellos que componen las actividades de Administración de Riesgos, ya que sin la correcta determinación de los mismos, no es posible desarrollar e implementar anticipadamente respuestas apropiadas a los problemas que puedan surgir en el proyecto. El resultado de la identificación de riesgos es una lista conteniendo los riesgos que se han identificados y su categoría correspondiente. Existen varios modelos de Administración de Riesgos, pero el más aceptado consta de cinco pasos (Identificación, Análisis, Planificación, Seguimiento y Control) los que comparten como actividades comunes las de documentación y comunicación.

### **2.1.1.2. Objetivos específicos**

1. Describir los procesos involucrados en la identificación, análisis, planeación, presupuesto y plan de respuesta de los riesgos del proyecto informático.
2. Identificar los problemas que se generan en la implementación del modelo de madurez.
3. Detallar normas para llevar a cabo el análisis cuantitativo y cualitativo de los riesgos en los proyectos informáticos.
4. Proponer un esquema para la planeación de riesgos a partir de la planificación estratégica de riesgos, la monitorización y control y la mitigación de los mismos.
5. Proponer un subescenario para el presupuesto de riesgos, a partir del esquema de planeación de riesgos.
6. Proponer un escenario para el plan de respuesta de riesgos.

## **2.2. Marco de gestión de proyectos según PMBOK <sup>1</sup>**

Esta sección incluye referencias al PMBOK rescatando aquellos puntos considerados necesarios.

<sup>1</sup> PMI. (2013). Project Management Body Of Knowledge (PMBOK GUIDE). (Quinta Edición).

### **2.2.1. Proyecto**

Un proyecto es un esfuerzo temporal emprendido para crear un producto o servicio único. Es temporal, pues cada proyecto tiene un inicio y un final definido, y único ya que el producto o servicio es diferente en alguna forma que lo distingue de todos los demás productos o servicios.

Los proyectos se implementan frecuentemente como un medio para llevar a cabo el plan estratégico de la empresa. En la mayoría de las empresas, los proyectos son un medio para

responder a aquellas demandas que no pueden ser atendidas dentro de sus límites operacionales normales.

### **2.2.1.1. Gestión de proyectos**

La metodología de gestión de proyectos, debe de involucrar los siguientes puntos:

- Alcance, necesario para asegurar que el proyecto incluya todo el trabajo requerido y sólo lo requerido para completar exitosamente el proyecto.
- Plazos, necesarios para asegurar la culminación a tiempo del proyecto.
- Recursos humanos, necesarios para hacer más efectivo el uso de las personas involucradas con el proyecto.
- Comunicaciones, necesarias para garantizar que la información del proyecto sea generada, colectada, distribuida, almacenada y finalmente dispuesta en forma oportuna y apropiada.
- Riesgos, necesarios para conocer a qué oportunidades o riesgos está expuesto el proyecto.
- Calidad, necesaria para asegurar que el proyecto satisfaga las necesidades para las cuales fue emprendido.

### **2.2.1.2. Involucrados del Proyecto (Stakeholders)**

Los involucrados del proyecto son individuos y organizaciones que están activamente involucrados con el proyecto o cuyos intereses pueden estar afectados positiva o negativamente por los resultados de la ejecución del proyecto y sus resultados. Una vez identificado los involucrados del proyecto, el jefe de proyectos debe determinar los requerimientos de estos

involucrados, y luego gestionar e influenciar esos requerimientos para asegurar el éxito del proyecto.

Los involucrados de un proyecto varían de acuerdo al proyecto y pueden adoptar uno o más roles dentro del mismo. A continuación se describen los roles que en la mayoría de los casos forman parte de un proyecto, pudiendo definirse roles adicionales para alguno de ellos:

- **Responsable del Proceso (RP):** pertenece a las áreas usuarias, es un Jefe, Gerente o Vicepresidente. Dueño del proceso que es impactado directamente por la funcionalidad modificada o implementada.
- **Usuario Líder (LU):** pertenece a las áreas usuarias, es un Jefe, Gerente o Vicepresidente. Es el responsable de la definición, verificación y aprobación (por parte del lado de los usuarios) de todos los requerimientos solicitados, asegurando que se cumplan y que cumplan con la satisfacción de los usuarios.
- **Usuario (US):** pertenece a las áreas usuarias. Es quien usa el sistema implementado y apoya en la etapa de las pruebas, el usuario también puede participar en la etapa de definición de requerimientos.
- **Jefe de PMO (JPMO):** Es el encargado de gestionar la cartera de proyectos, ya sean internos de la Compañía o contratados por externos.
- **Analista Funcional:** Es el responsable del análisis funcional del requerimiento o proyecto de software a desarrollar, da soporte y absuelve consultas a los analistas programadores, gestiona y da seguimiento al proyecto desde el inicio al cierre del proyecto.
- **Analista de Soporte de Aplicaciones (SOAP):** Corresponde al cargo Analista de Soporte de aplicaciones, es el encargado de dar soporte técnico a los sistemas en el ambiente de

producción. Apoya en la realización de los pases a certificación y Producción, además de brindar el soporte que sea necesario a los programadores.

- Jefe de Proyecto (JP): Es el facilitador y encargado de la gestión y seguimiento del proyecto, maneja una cartera de proyectos asignados por el Jefe de PMO.
- DBA: Corresponde al cargo Administrador de Base de Datos.
- Analista de Sistemas (AS): corresponde a los cargos Analista Programador, Analista Programador Senior o Analista de Sistemas.
- Jefe de QA (JQA): Jefe de QA es el encargado de garantizar la calidad y certificación de los sistemas implementados, abarcando la documentación y el cumplimiento de los requerimientos funcionales de los sistemas.
- Analista QA (AQA): corresponde a los cargos Analista de Calidad. Es quien verifica el cumplimiento del plan de pruebas y cronograma de pruebas. Verifica los entregables y documentación. Realiza y verifica las pruebas en el ambiente de certificación.
- Supervisor QA (SQA): rol que supervisa el cumplimiento del cronograma de pruebas y comunica el avance del proceso de pruebas a los roles AQA y JQA.
- Programador (APROG): corresponde a los cargos Analista Programador o Analista Programador Senior. Es quien se encarga de programar el sistema a través de lenguajes de programación y bajo una arquitectura establecida para el proyecto.
- Analista de Seguridad de la Información (OSI): corresponde al cargo de Coordinador de Seguridad de Información. Es quien se encarga del cumplimiento de los requerimientos de Seguridad, con la finalidad de resguardar la confidencialidad, integridad y disponibilidad de la información, alineado a la normatividad vigente.

- Oficial de Cumplimiento Normativo (OCN): es quien se encarga de asegurar el cumplimiento de la normativa legal vigente que aplica a la empresa.
- Proveedor (PROV): corresponde a la entidad que presta algún servicio específico durante la realización de los proyectos.

### **2.2.1.3. Portafolio de proyectos**

El portafolio de proyectos es un conjunto de proyectos que se gestionan de manera centralizada. La gestión de portafolio incluye identificar, establecer prioridades, autorizar, dirigir y controlar proyectos. La gestión del portafolio se centra en asegurar que los proyectos se revisen a fin de establecer prioridades para la asignación de recursos, y en que la gestión del portafolio sea consistente con las estrategias de la organización y esté alineada con ellas.

### **2.2.1.4. PMO (Oficina de Proyectos)**

La PMO es una entidad que tiene varias responsabilidades asignadas con relación a la dirección centralizada y coordinada de aquellos proyectos que se encuentran bajo su jurisdicción.

Una función fundamental de la PMO es brindar apoyo a los jefes de proyecto de diferentes formas, entre ellas:

- Gestionar recursos compartidos por todos los proyectos dirigidos por la PMO;
- Identificar y desarrollar una metodología, mejores prácticas y normas para la dirección de proyectos;
- Instruir, orientar, capacitar y supervisar;

- Vigilar el cumplimiento de las políticas de normas, procedimientos y plantillas de la dirección de proyectos mediante auditorías del proyecto;
- Desarrollar y gestionar políticas, procedimientos, plantillas y otra documentación compartida del proyecto, y coordinar la comunicación entre proyectos.

Una de las diferencias entre el rol de los jefes de proyecto y PMO es que el jefe de proyecto gestiona las restricciones (alcance, cronograma, costo y calidad, entre otras) de los proyectos individuales, mientras que la PMO gestiona las metodologías, normas, oportunidad/riesgo global e interdependencias entre los diferentes proyectos que conforman el portafolio.

### 2.2.2. Entender el ciclo de vida del proyecto

Un proyecto tiene un comienzo y un fin. El intervalo de tiempo desde el principio del proyecto hasta el final del proyecto se denomina ciclo de vida del proyecto. Si un proyecto tiene fases múltiples, todas las fases se completan durante su ciclo de vida con el fin de completar el proyecto. Independientemente de si el proyecto tiene varias fases o sólo una fase, durante este ciclo de vida, se inicia el proyecto, organizado y preparado es llevado a cabo y se cierra. En la terminología estándar, un proyecto se inicia, planifica, ejecuta, supervisa y controla y se cierra.

Desde su inicio hasta la autorización para su cierre, un proyecto pasa por un ciclo de vida que incluye la definición de los objetivos del proyecto, la planificación del trabajo, la realización de la obra, la supervisión y el control del progreso y el cierre del proyecto después de recibir aceptación del producto. La Figura 5 muestra las etapas del ciclo de vida del proyecto:

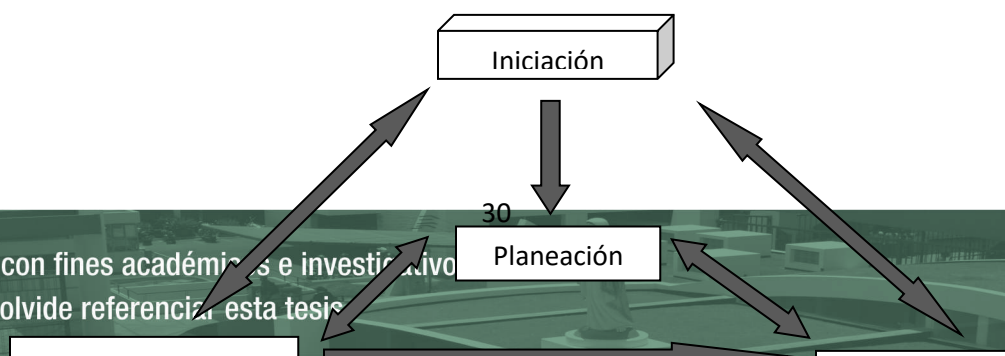


Figura 5. Diferentes estados en el ciclo de vida de un proyecto. Cada uno de estos estados representan un grupo de procesos.

Fuente: CAPM – Certified Associate in Project Management Study Guide for the CAPM Exam, Paul Sanghera.

### 2.2.2.1. Inicio de un proyecto

Esta fase define y autoriza el proyecto. El director del proyecto debe su nombre, el proyecto se pone en marcha oficialmente a través de un documento firmado, llamado acta de constitución del proyecto (Project charter), el cual contiene elementos tales como la finalidad del proyecto, una descripción del producto de alto nivel, un resumen del cronograma de hitos, y un caso de negocio para el proyecto. Otro resultado de esta etapa es un documento que se llama el registro de stakeholders, que identifica a los interesados en el proyecto e información importante acerca de ellos. Los procesos utilizados para llevar a cabo este escenario se denominan procesos de iniciación.

Por lo tanto, iniciar un proyecto significa definir el proyecto, obtener la aprobación para comenzar, identificación y análisis de las partes interesadas del proyecto. Durante esta etapa, en el alcance inicial del proyecto se define, los recursos iniciales se determinan y asignan, un gerente de proyecto con el nivel de autoridad adecuado es asignado, y los participantes del proyecto son identificados. La definición del proyecto incluye lo siguiente:

- Desarrollar los objetivos del proyecto y describir cómo se relacionan con los objetivos de negocio de la organización y su estrategia.
- Especificar los entregables del proyecto, tales como productos, servicios o resultados que satisfagan los objetivos. En algunos casos, un objetivo y un entregable puede ser el mismo.
- Con base en los objetivos y resultados, definir el alcance inicial del proyecto, explicando lo que se hará, trazando las fronteras, es decir, indicando lo que no se incluirá.
- Con base en el alcance inicial, estimar la duración del proyecto y los recursos necesarios. Las estimaciones más precisas se harán durante la planificación.
- Definir los criterios de éxito, que pueden ser mejorados durante la planificación.
- Autorizar el proyecto. Mientras que diferentes organizaciones pueden tener un proceso diferente para aprobar los proyectos, la forma estándar de hacerlo es aprobar el documento que contiene la definición del proyecto, tales como el acta de constitución del proyecto.

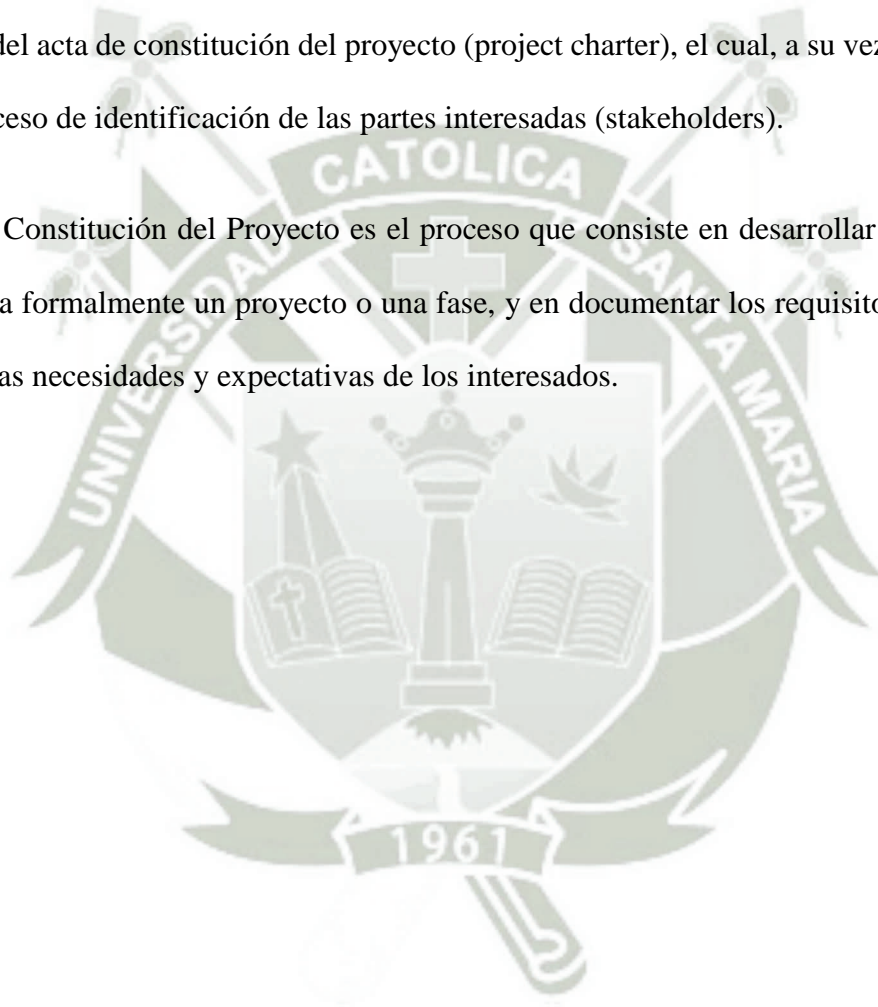
Con base en la definición del proyecto, se podrán identificar las partes interesadas del proyecto.

Nota:

Es una buena práctica involucrar a los clientes y otras partes interesadas importantes en la fase de inicio del proyecto. Se les dará un sentimiento de propiedad compartida que contribuirá en gran medida al éxito del proyecto, influyendo positivamente en factores tales como la aceptación de los entregables y la satisfacción de las partes interesadas.

La Figura 6 presenta el panorama general de la iniciación de un proyecto ilustrando las relaciones entre los elementos más importantes de esta etapa. La necesidad de la empresa surge a partir de la estrategia de negocio de la organización, y en base a las necesidades del negocio, se hace la declaración de trabajo y el modelo de negocio. El modelo de negocio y la declaración de trabajo son los puntos de partida para desarrollar el alcance del proyecto inicial y por lo tanto determinar los recursos iniciales. Toda esta información se convierte en la entrada para el desarrollo del acta de constitución del proyecto (project charter), el cual, a su vez, es un insumo para el proceso de identificación de las partes interesadas (stakeholders).

El Acta de Constitución del Proyecto es el proceso que consiste en desarrollar un documento que autoriza formalmente un proyecto o una fase, y en documentar los requisitos iniciales que satisfacen las necesidades y expectativas de los interesados.



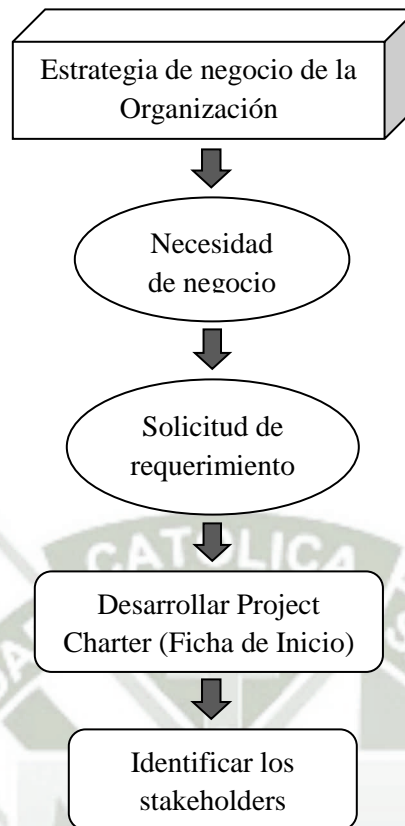


Figura 6. Ilustración de las relaciones entre los principales bloques de construcción del panorama general de iniciación de un proyecto.

Fuente: CAPM – Certified Associate in Project Management Study Guide for the CAPM Exam, Paul Sanghera.

### 2.2.2.2. Planificación del proyecto

En esta etapa, el gerente del proyecto, junto con el equipo de gestión del proyecto, establece el alcance total del esfuerzo, define y refina los objetivos del proyecto, y desarrolla la línea de acción requerida para alcanzar dichos objetivos. Los procesos de planificación desarrollan el plan de gestión del proyecto y los documentos proyecto que se utilizarán para llevarlo a cabo. Este proceso incluye los siguientes planes: plan de gestión de cambios, plan de configuración,

plan de gestión del proyecto, plan de gestión de la calidad, plan de recursos humanos, plan de gestión de comunicaciones.

Los cambios importantes que ocurren a lo largo del ciclo de vida del proyecto generan la necesidad de reconsiderar uno o más de los procesos de planificación y, posiblemente, algunos de los procesos de iniciación. Esta incorporación progresiva de detalles al plan de gestión del proyecto recibe generalmente el nombre de “planificación gradual”, para indicar que la planificación y la documentación son procesos repetitivos y continuos.

Es importante planificar el proyecto, ya que no todos los proyectos necesitan todos los procesos de planificación, ni todos ellos necesitan el mismo grado. Por lo tanto, el contenido del plan de gestión del proyecto, el documento final elaborado a partir de la planificación del proyecto, dependerá de un proyecto específico. A medida que el proyecto pasa por diferentes etapas, el plan de gestión del proyecto puede ser actualizado y revisado por el proceso de control de cambios.

La Figura 7 presenta el panorama general de la planificación del proyecto. Dependiendo de la complejidad del proyecto, el plan de gestión del proyecto puede ser un resumen o una colección de los planes subsidiarios y componentes, que pueden incluir los siguientes:

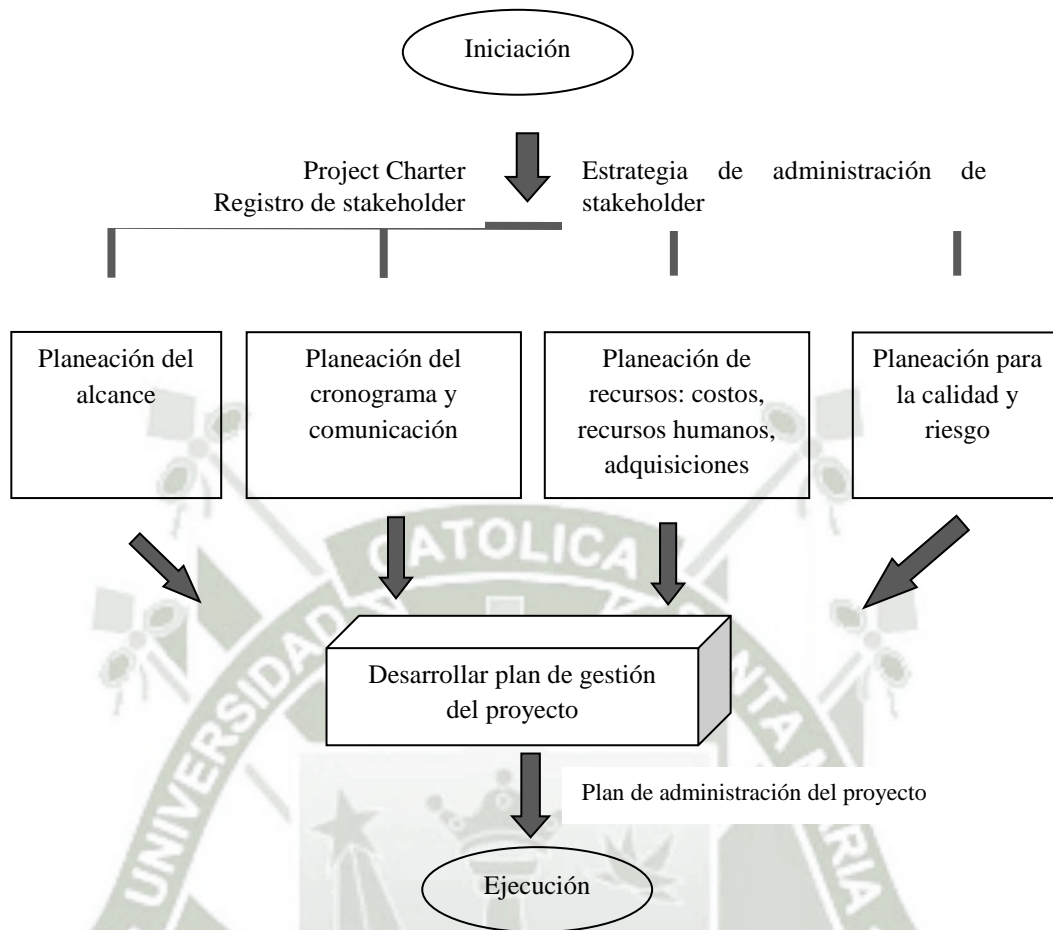


Figura 7. Una vista de alto nivel de las interacciones y el flujo de datos entre los diferentes componentes de la planificación de proyectos.

Fuente: CAPM – Certified Associate in Project Management Study Guide for the CAPM Exam, Paul Sanghera.

En pocas palabras, la planificación del proyecto consiste en determinar exactamente lo que se hará y cómo se hará.

### 2.2.2.3. Ejecución del Proyecto

En esta etapa, el gerente del proyecto, realiza la ejecución del plan de gestión del proyecto y el equipo de proyecto realiza el trabajo programado en la etapa de planificación. Se coordinan todas las actividades que se realizarán para lograr los objetivos del proyecto y cumplir con los

requisitos del proyecto. Por supuesto, el principal resultado de este proyecto son los entregables del proyecto.

Los cambios aprobados, recomendaciones y reparaciones de defectos, también se aplican en esta etapa. Pero, ¿de dónde vienen estos cambios y recomendaciones? Surgen del seguimiento y control del proyecto. Los interesados también pueden sugerir cambios, que deben pasar por un proceso de aprobación antes de su ejecución. La ejecución del proyecto se lleva a cabo mediante procesos que pertenecen a un grupo llamado el grupo de procesos de ejecución. Así pues, el proyecto de trabajo definido en el plan de gestión del proyecto se ejecuta mediante el uso de los procesos de ejecución, como se muestra en la Figura 8. Como se dijo anteriormente, corresponde al equipo del proyecto determinar cuál de estos procesos es relevante para el proyecto en cuestión. Los procesos de este grupo se utilizan para lograr un objetivo triple:

- Coordinar las personas y los recursos utilizados para realizar las actividades del proyecto.
- Integrar y administrar las actividades del proyecto que se están realizando.
- Velar por el cumplimiento del alcance del proyecto y de cambios aprobados.

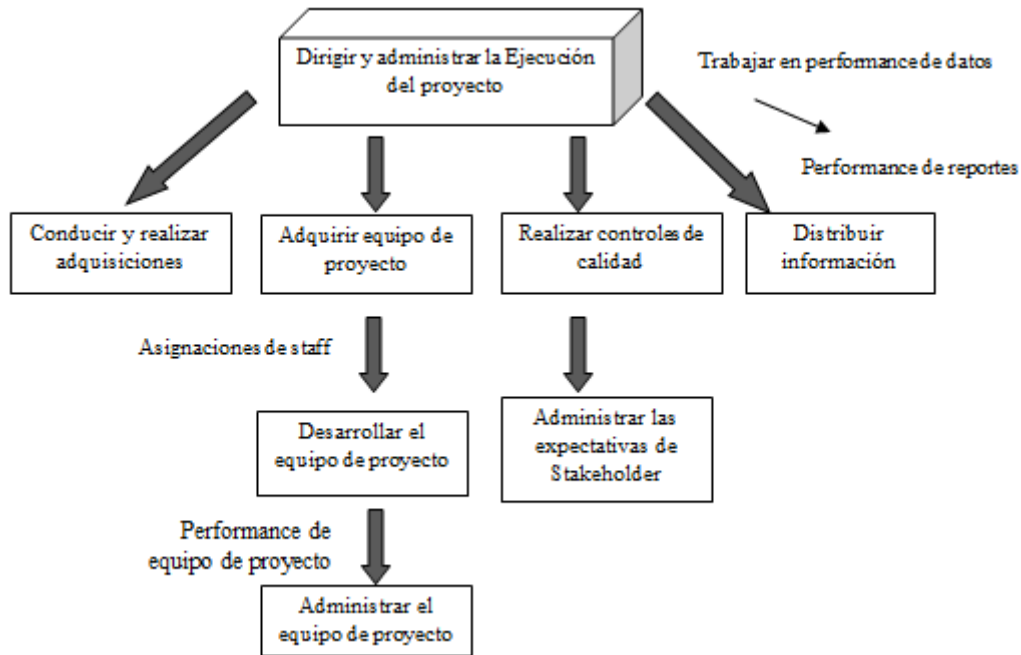


Figura 8. Grupo del proceso de ejecución.

Fuente: CAPM – Certified Associate in Project Management Study Guide for the CAPM Exam, Paul Sanghera.

#### 2.2.2.4. Seguimiento y control del proyecto

El monitoreo y control del proyecto es a través de su ciclo de vida, incluida la fase de ejecución. El proceso de seguimiento y control incluye la defensa del proyecto contra la corrupción del alcance (cambios no aprobados a los alcances del proyecto), el seguimiento del progreso del proyecto y el rendimiento para identificar cambios del plan, y recomendar acciones preventivas y correctivas para que el proyecto esté en línea con las expectativas previstas. Las solicitudes de cambios, tales como cambios en el alcance del proyecto, también se incluyen en esta etapa, ya que pueden venir del gerente o de cualquier parte interesada. Los cambios deben pasar por un proceso de aprobación, y sólo los cambios aprobados son implementados.

Los procesos utilizados en este escenario se denominan el grupo de procesos de seguimiento y control. La vigilancia y el grupo de procesos de control incluye los procesos de monitoreo y control de proyectos de rendimiento, cambios y riesgos.

El monitoreo incluye los resultados del proyecto de medición, recopilación de información sobre el desempeño de los proyectos y la evaluación de la información de rendimiento para ver las tendencias. El monitoreo continuo ayuda a que el equipo de gestión del proyecto identifique las áreas que necesitan ser controlados estrechamente por ejemplo, la adopción de medidas correctivas o preventivas.

Nota:

El seguimiento y control no se inicia sólo después de la ejecución del proyecto, por el contrario, el proyecto debe ser supervisado y controlado todo el camino desde el inicio hasta el cierre.

La Figura 9 muestra todos los procesos que se encuentran en el grupo de procesos de control, excepto el proceso de dirigir y gestionar la ejecución del proyecto, que pertenece al grupo de ejecución.

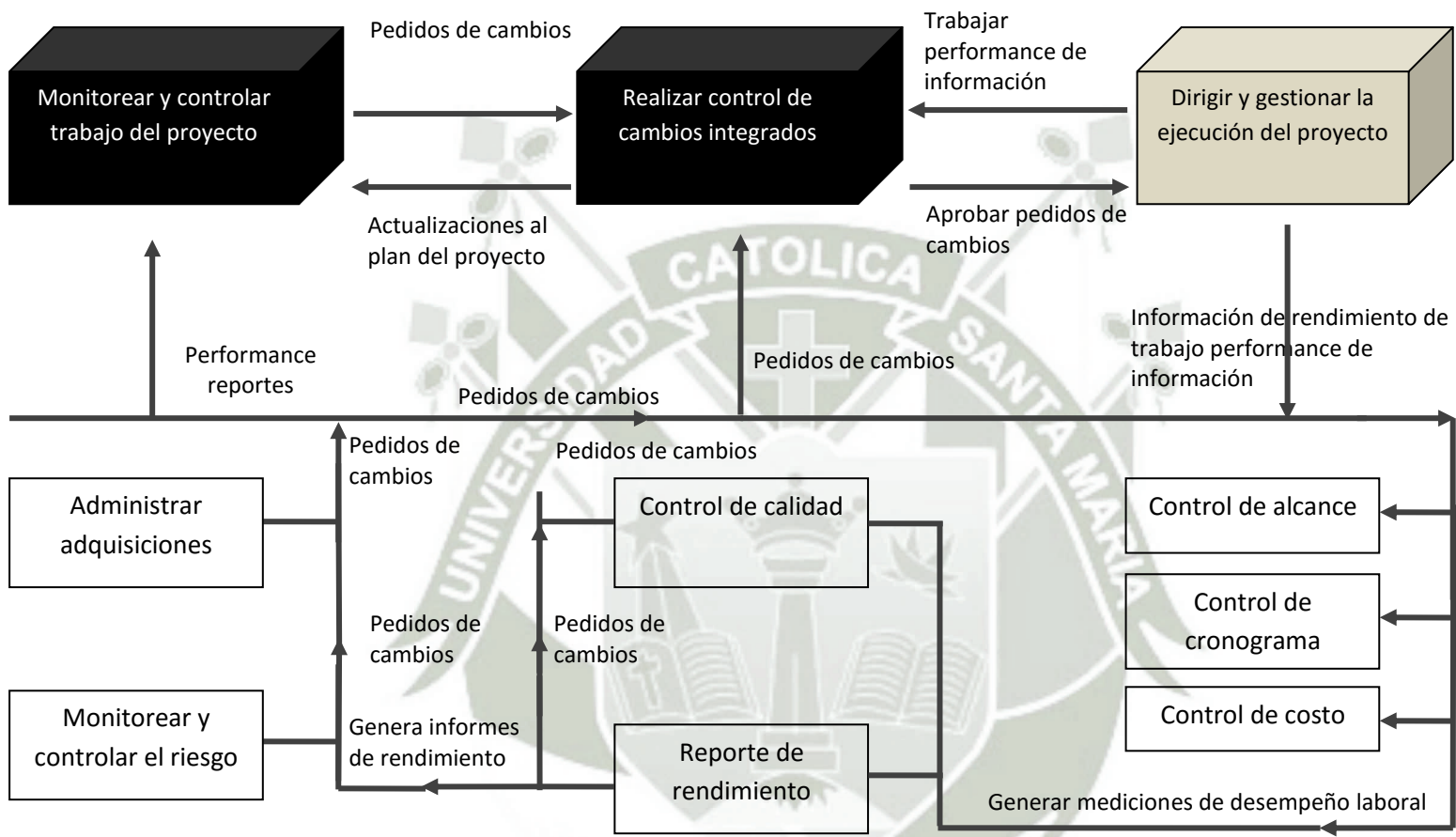


Figura 9. Escenario del grupo de proceso de monitoreo y control.  
 Fuente: CAPM – Certified Associate in Project Management Study Guide for the CAPM Exam, Paul Sanghera.

### 2.2.2.5. Cierre del proyecto

El Grupo del Proceso del Cierre está compuesto por aquellos procesos realizados para finalizar todas las actividades a través de todos los grupos de procesos de la dirección de proyectos, a fin de completar formalmente el proyecto, una fase del mismo u otras obligaciones contractuales.

Este grupo de procesos, una vez completado, verifica que los procesos definidos se hayan completado dentro de todos los grupos de procesos a fin de cerrar el proyecto o una fase del mismo, según corresponda, y establece formalmente que el proyecto o fase del mismo ha finalizado. En el cierre del proyecto o fase, puede ocurrir lo siguiente:

- Obtener la aceptación del cliente o del patrocinador,
- Realizar una revisión tras el cierre del proyecto o la finalización de una fase,
- Registrar los impactos de la adaptación a un proceso,
- Documentar las lecciones aprendidas,
- Aplicar actualizaciones apropiadas a los activos de los procesos de la organización,
- Archivar todos los documentos relevantes del proyecto en el sistema de información para la dirección de proyectos para ser utilizados como datos históricos y cerrar las adquisiciones.

Nota:

Lo que denominamos fases del proyecto aquí no son en realidad las fases del proyecto. Una fase de proyecto se enmarca dentro del proyecto conjunto en el que ciertos hitos o resultados del proyecto se han completado. Todas estas etapas, técnicamente

denominados grupos de procesos, se pueden aplicar a cualquiera de las fases de un proyecto que se divide en varias fases.

Así, el cierre del proyecto se refiere a un conjunto de tareas que se requieren para terminar formalmente el proyecto. Hay dos tipos de proyectos que se necesita para cerrar formalmente:

- Proyectos completados.

Un proyecto que ha cumplido los criterios de terminación entra en esta categoría.

- Proyectos terminados.

Un proyecto que fue cancelado antes de su finalización entra en esta categoría.

Un proyecto puede ser terminado en varias etapas, por diversas razones. Los siguientes son algunos ejemplos:

- El plan de gestión del proyecto no ha sido aprobado por la razón que sea.
- El proyecto se ha venido ejecutando, pero se han quedado sin recursos, y los recursos no están disponibles.
- El proyecto ha sido cancelado debido a que no se llega a nada.
- El proyecto ha sido pospuesto indefinidamente porque no hay un mercado lo suficientemente grande para el producto.

Nota:

Los procesos del grupo de cierre pueden utilizarse para cerrar un proyecto, así como para cerrar una fase de un proyecto.

El grupo de procesos de cierre incluye los siguientes procesos:

- Cerrar el proyecto o fase.
- Cerrar adquisiciones del proyecto.

El Cierre del proyecto incluye las siguientes actividades:

- Actividades para verificar que todas las entregas se han previsto y aceptado.
- Actividades para confirmar que todos los requisitos del proyecto, incluidos los requisitos de los interesados, se han cumplido.
- Actividades para verificar que los criterios de finalización o de salida se han cumplido.
- Actividades para asegurar que el producto del proyecto se transfiere a la persona o el grupo correcto.
- Actividades para revisar el proyecto, obtener las lecciones aprendidas y archivar los registros del proyecto.

Se necesita obtener un cierre final, como signo de aceptación del compromiso, cierre de contrato, o recibos, tanto para la parte interna y la parte de la contratación del proyecto y de los dos proveedores internos y externos y clientes. Debe realizar esta tarea mediante el uso de las prácticas habituales de contabilidad y siguiendo los procedimientos pertinentes organizativas y legales.

Panorama general de cierre del proyecto en pocas palabras:

1. Verifique el alcance de los resultados del proyecto desarrollado.

2. Aceptar los entregables obtenidos a través del proceso de cierre de adquisiciones.
3. Obtener los entregables del Paso 1 al Paso 2 y lograr que sean aceptadas por el cliente o patrocinador para cerrar realmente el proyecto.
4. Archivar documentos del proyecto.

La Tabla 10 presenta un resumen de la vida del proyecto. La etapa de iniciación autoriza un proyecto nombrando al director del proyecto, la etapa de planificación define además los objetivos del proyecto y los planes de trabajo para lograr estos objetivos, la etapa de ejecución ejecuta la obra, la supervisión y el control de los monitores de escenario, los avances del proyecto y control para mantenerse en línea con el plan, y la fase de cierre formalmente cierra el proyecto mediante la obtención de la aceptación del producto. Cada una de estas etapas se realiza mediante el uso de un grupo de procesos. De esta manera, estas etapas se llaman grupos de procesos.

<b>Grupo de proceso</b>	<b>Correspondiente nombre para la etapa de proyecto</b>	<b>Meta importante</b>	<b>Salida importante</b>
Inicio	Empezar el proyecto	Autorizar el proyecto	Acta de constitución del proyecto
Planeación	Organizar y preparar	Plan y cronograma del trabajo para realizar el proyecto	Plan de administración del proyecto
Ejecución	Ejecución de la obra	Realizar el trabajo del proyecto	Entregas de proyecto: producto, servicio o resultado

Cierre	Cierre del proyecto	Cierre del proyecto formalmente	Aceptación del producto, cierre del contrato y archivamiento.
Monitoreo y control	Abarca todo el ciclo de vida del proyecto	Monitorear el progreso del proyecto para identificar las varianzas con el plan y corregirlas	Informes de rendimiento del proyecto

Tabla 10. Etapas del ciclo de vida de un proyecto: los grupos de procesos del proyecto.

Guía del PMBOK quinta edición 2013, PMI

### 2.3. Marco de gestión de proyectos ágil <sup>2</sup>

Esta sección contiene referencias al “SCRUM Manager” se han rescatado definiciones importantes para comprender la metodología de SCRUM.

#### 2.3.1. SCRUM

Scrum es un proceso en el que se aplican de manera regular un conjunto de mejores prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto. Estas prácticas se apoyan unas a otras y su selección tiene origen en un estudio de la manera de trabajar de equipos altamente productivos.

En Scrum se realizan entregas parciales y regulares del producto final, priorizadas por el beneficio que aportan al receptor del proyecto.

<sup>2</sup> Juan Palacio, Claudia Ruata (2011). Manifiesto ágil. *Scrum Manager Gestión de Proyectos*, 20 - 21. España: Safe Creative.

Por ello, Scrum está especialmente indicado para proyectos en entornos complejos, donde se necesita obtener resultados pronto, donde los requisitos son cambiantes o poco definidos, donde la innovación, la competitividad, la flexibilidad y la productividad son fundamentales.

### 2.3.2. Historia ágil y una visión general

‘La agilidad es dinámica, con contenido específico, ajustable al cambio de manera dinámico y orientado al crecimiento’. Steven Goldman.

Las metodologías ágiles no son nuevas, las personas han estado desarrollando varios tipos de proyectos ágiles por mucho tiempo; sin embargo, hoy en día, adoptar más metodologías ágiles parece haber ganado mucho impulso por varias razones:

- Algunas de las metodologías y herramientas para implementar proyectos ágiles como Scrum, se han vuelto más entendidas y aceptadas.
- Muchas compañías han demostrado éxito implementando proyectos ágiles como prácticas actuales y se añadió conocimiento base de que trabajo hacer o que trabajo no hacer en diferentes entornos de trabajos.
- Las ventajas y desventajas asociadas con el balance para el control suficiente de un proyecto y aseguramiento de la agilidad son más comprendidas, y hay muchas maneras de lograr el balance de esos objetivos con una apropiada mezcla de las metodologías tradicionales y ágiles.

El origen de todo el movimiento ágil de hoy en día se remonta al manifiesto ágil, el cual fue desarrollado en febrero del 2001, en Wasatch – Utah. El manifiesto es muy simple y está comprendido por los siguientes enunciados a valorar:

- A los individuos y su interacción, por encima de los procesos y las herramientas.
- El software que funciona, por encima de la documentación exhaustiva.
- La colaboración con el cliente, por encima de la negociación contractual.
- La respuesta al cambio, por encima del seguimiento de un plan.

Valoramos más a los individuos y su interacción que a los procesos y las herramientas.

Este es posiblemente el principio más importante del manifiesto.

Por supuesto que los procesos ayudan al trabajo. Son una guía de operación. Las herramientas mejoran la eficiencia, pero en trabajos que requieren conocimiento tácito, sin personas con conocimiento técnico y actitud adecuada, no producen resultados.

Las empresas suelen predicar muy alto que sus empleados son lo más importante, pero la realidad es que en los años 90 la teoría de producción basada en procesos, la reingeniería de procesos ha dado a éstos más relevancia de la que pueden tener en tareas que deben gran parte de su valor al conocimiento y al talento de las personas que las realizan.

Los procesos deben ser una ayuda y un soporte para guiar el trabajo. Deben adaptarse a la organización, a los equipos y a las personas; y no al revés. La defensa a ultranza de los procesos lleva a postular que con ellos se pueden conseguir resultados extraordinarios con personas mediocres, y lo cierto es que este principio es peligroso cuando los trabajos necesitan creatividad e innovación.

Valoramos más el software que funciona que la documentación exhaustiva.

Poder ver anticipadamente cómo se comportan las funcionalidades que se esperan sobre prototipos o sobre partes ya elaboradas del sistema final ofrece un "feedback" muy estimulante y enriquecedor que genera ideas y posibilidades imposibles de concebir en

un primer momento, y difícilmente se podrían incluir al redactar un documento de requisitos detallados antes de comenzar el proyecto.

El manifiesto no afirma que no hagan falta. Los documentos son soporte de documentación, permiten la transferencia del conocimiento, registran información histórica, y en muchas cuestiones legales o normativas son obligatorios, pero se resalta que son menos importantes que los productos que funcionan. Menos trascendentales para aportar valor al producto.

Los documentos no pueden sustituir, ni pueden ofrecer la riqueza y generación de valor que se logra con la comunicación directa entre las personas y a través de la interacción con los prototipos. Por eso, siempre que sea posible debe preferirse reducir al mínimo indispensable el uso de documentación, que genera trabajo que no aporta un valor directo al producto.

Si la organización y los equipos se comunican a través de documentos, además de perder la riqueza que da la interacción con el producto, se acaba derivando a emplear a los documentos como barricadas entre departamentos o entre personas.

Valoramos más la colaboración con el cliente que la negociación contractual.

Las prácticas ágiles están especialmente indicadas para productos difíciles de definir con detalle al principio del proyecto, o que si se definieran así tendrían al final menos valor que si se van enriqueciendo con retroinformación continua durante el desarrollo. También para los casos en los que los requisitos van a ser muy inestables por la velocidad del entorno de negocio del cliente.

Para el desarrollo ágil el valor del resultado no es consecuencia de haber controlado una ejecución conforme a procesos, sino de haber sido implementado directamente sobre el producto.

Un contrato no aporta valor al producto. Es una formalidad que establece líneas divisorias de responsabilidades, que fija los referentes para posibles disputas contractuales entre cliente y proveedor.

En el desarrollo ágil el cliente es un miembro más del equipo, que se integra y colabora en el grupo de trabajo. Los modelos de contrato por obra no encajan.

#### Valoramos más la respuesta al cambio que el seguimiento de un plan

Para un modelo de desarrollo que surge de entornos inestables, que tienen como factor inherente el cambio y la evolución rápida y continua, resulta mucho más valioso la capacidad de respuesta que el de seguimiento y aseguramiento de planes pre-establecidos. Los principales valores de la gestión ágil son la anticipación y la adaptación; diferentes a los de la gestión de proyectos ortodoxa: planificación y control para evitar desviaciones sobre el plan.

### **2.3.3. Objetivos de la agilidad <sup>3</sup>**

#### 1.-Valor

La gestión ágil se necesita en los mercados rápidos.

Su objetivo es dar el mayor valor posible al producto, cuando éste se basa en:

- Innovación. La permanencia de estas empresas depende de su capacidad de

<sup>3</sup> Juan Palacio, Claudia Ruata (2011). Gestión de Proyectos ágil". *Scrum Manager Gestión de Proyectos*, 41-43. España: Safe Creative.

innovación continua. Del lanzamiento continuo de novedades, que compiten con los productos de otras empresas que también están en continua innovación.

- Flexibilidad. El producto no sólo es valioso por su valor en el momento de su lanzamiento, sino también por su capacidad de adaptación y evolución a través de actualizaciones y ampliaciones.

## 2.- Reducción del tiempo de salida al mercado

En la década de los 90, el tiempo medio de salida al mercado de los nuevos productos en EE.UU. se redujo de 35,5 a 11 meses (Wujec & Muscat, 2002).

Este tiempo es un factor competitivo clave en determinados sectores.

Las estrategias de la gestión ágil para producir resultados en menos tiempo que la gestión tradicional son:

- Solapamiento de las fases de desarrollo.
- Entrega temprana de las primeras partes del producto, que corresponden con las de mayor urgencia para el cliente, de forma que puede lanzar la primera versión en el menor tiempo posible.

## 3.-Agilidad

Capacidad para producir partes completas del producto en periodos breves de tiempo.

## 4.-Flexibilidad

Capacidad para adaptar la forma y el curso del desarrollo a las características del proyecto, y a la evolución de los requisitos.

5.- Resultados fiables

La fiabilidad está en la entrega temprana y continua de resultados (fracciones del producto), no en el cumplimiento de planes.

**2.3.4. Comparación principios ágiles y tradicionales <sup>4</sup>**

<b>Principio del Manifiesto ágil</b>	<b>Relación con la práctica de gerencia de proyectos tradicionales</b>
<p>Nuestra principal prioridad es satisfacer al cliente a través de la entrega temprana y continua de software de valor</p>	<p>No hay contradicciones con el principio; sin embargo, la práctica tradicional en muchos casos enfatiza control, predicción y minuciosidad de testeado en las entregas tempranas.</p>
<p>Son bienvenidos los requisitos cambiantes, incluso si llegan tarde al desarrollo. Los procesos ágiles se dobligan al cambio como ventaja competitiva para el cliente</p>	<p>Controles de cambio son vistas como elementos esenciales para la estabilidad de un proyecto, pero un proceso de cambio no tiene que ser necesariamente rígido.</p> <p>La gestión ágil necesita enriquecer y dar valor a la visión del producto.</p>

	<p>La nueva información durante el desarrollo no son “modificaciones de requisitos”, sino fuente de análisis y valor para el producto. Los requisitos que surgen al probar partes ya desarrolladas o lo que la competencia lanzó al mercado, no son modificaciones que amenazan el plan, sino requisitos con información que aumenta el valor del producto.</p>
<p>Entregar con frecuencia software que funcione, en periodos de un par de semanas hasta un par de meses, con preferencia en los periodos breves</p>	<p>Lograr este punto con metodologías de desarrollo tradicional es complicado, y esta es la parte donde realmente la agilidad vale la pena; por supuesto, la ganancia en esta área necesita ser controlada para no causar impactos potenciales en otras áreas.</p> <p>La entrega temprana y constante de valor: de partes funcionales cerradas que resultan valiosas, bien porque ya puede salir al mercado, o bien porque generan información para los requisitos siguientes con mayor valor para el producto.</p>
<p>Las personas del negocio y los desarrolladores deben trabajar juntos</p>	<p>Este es un rol que ha crecido significativamente para la gente del</p>

<p>de forma cotidiana a través del proyecto</p>	<p>negocio que deben ir más allá de la definición típica de los requerimientos y fases de pruebas de las metodologías de desarrollo tradicional. Esto implica una relación de trabajo cercano y colaborativo entre las personas de negocio y desarrollo y un compromiso más fuerte del lado de negocio para participar activamente en el proceso de desarrollo.</p> <p>Mucha gente de negocio no están preparadas para aceptar ese tipo de compromiso que se necesita, y en muchos casos, ese nivel de participación no se toma en cuenta. Como resultado, la metodología del proyecto necesita encajar con las necesidades de participación y necesidades de la gente de negocio que provee esa participación.</p>
<p>Construcción de proyectos en torno a individuos motivados, dándoles la oportunidad y el respaldo que necesitan y procurándoles confianza para que realicen la tarea</p>	<p>Este principio particular implica un nivel alto de delegación y confianza que normalmente no se encuentra en desarrollos tradicionales. Un nivel alto de delegación involucra algunas pérdidas de control, pero puede ser una guía aceptable</p>

	<p>en el entorno adecuado con las personas correctas en el equipo del proyecto.</p> <p>Esto implica que las personas deben estar bien preparadas y capacitadas para asumir niveles altos de responsabilidad. Muchas personas usan estos conceptos para justificar anarquía y caos sin liderazgo.</p> <p>Conociendo lo que se está trabajando y lo que no y deliberadamente decidir cuando y donde tomar acciones correctivas (y como debería ser) es esencial para cualquier proyecto.</p>
<p>La forma más eficiente y efectiva de comunicar información de ida y vuelta dentro de un equipo de desarrollo es mediante la conversación cara a cara</p>	<p>Se debe tener la habilidad de colocar miembros en los equipos que permitan la comunicación cara a cara y también debe estar limitado al tamaño del equipo. En muchas situaciones, herramientas como compartir escritorios (con o sin video) y portales de colaboración pueden aliviar la necesidad de una comunicación directa en equipos distribuidos especialmente cuando se usan las herramientas de desarrollo para la web.</p>

<p>La simplicidad como arte de maximizar la cantidad de trabajo que se hace, es esencial</p>	<p>En la metodología tradicional, las construcciones elaboradas, densas y complejas hacen incómoda la evolución. El desarrollo ágil se basa en la construcción iterativa. En la entrega continua de pequeños módulos de valor. Los desarrollos se basan en la modularidad sobre “piezas” funcionales simples.</p>
--	---

Adoptar muchos de estos principios ágiles envuelve ventajas o desventajas con las prácticas de gerencia de proyectos aceptadas, se deben de tomar atención a:

- Control y predicción
- Gerencia del riesgo del proyecto
- Entrega de un aceptable nivel de calidad

En muchas ocasiones, esto se percibe como un propósito todo o nada y la decisión está entre:

- Muy rígido el control y con mucha documentación, y
- No usar metodología y documentación

En algunas organizaciones las decisiones son extremas, por ejemplo mucho control y en otras casi nada de control. La lección clave que se debe aprender es:

4 Charles G. COBB. (2011). Agile History and Overview. *Making Sense of Agile Project Management: Balancing Control and Agility*, 40-43. New Jersey, Estados Unidos: Jhon Wiley & Sons.

- Tener un plan y enfoque bien realizado que equilibre la agilidad y control, si se hace correctamente, no se necesita sacrificarse para tener control y lograr agilidad, pero se necesita habilidad para lograr ambas agilidad y control sin sacrificar ninguna para lograr la otra.
- Tomar las decisiones correctas sobre cuál es el correcto balance de agilidad y control, es algo que debe hacerse junto con el lado del negocio y el lado de desarrollo de la organización basada en un entendimiento mutuo de ventajas y desventajas y el impacto potencial. Tiene cierta importancia la madurez de la organización para hacer estas decisiones efectivas.

### 2.3.5. Enfoque ágil <sup>5</sup>

#### 2.3.5.1. Enfoque ágil para Proyectos

Con una comprensión de los principales valores ágiles, podemos convertir nuestra atención a lo que un equipo ágil se ve en la práctica. Se incluye:

- Trabajo en equipo
- Trabajo en iteraciones cortas
- Entregar partes funcionales cerradas cada iteración
- Centrarse en las prioridades del negocio
- Revisar y adaptar

#### 2.3.5.2. Un equipo ágil trabaja como uno

Para el éxito de un proyecto es que todos los participantes en el proyecto ven a sí mismos

<sup>5</sup> Mike Cohn. (2005). An Agile Approach. *Agile Estimating and Planning*, 21-25. Massachusetts, Estados Unidos: Pearson Education.

como un equipo destinado a un objetivo común. No hay lugar para que cada uno trabaje aisladamente, los analistas no pueden simplemente dejar los requisitos para los diseñadores.

Los diseñadores y arquitectos no pueden simplemente dejar los diseños sueltos para los programadores, los programadores sueltan el código medio probado a los validadores. Un equipo ágil exitoso debe tener una mentalidad de todos estamos en esto.

Mientras que un equipo ágil debe trabajar juntos, hay una serie de funciones específicas en el equipo. Vale la pena identificar y clarificar los roles que se desempeñan para una estimación y planificación.

#### **2.3.5.3. Equipos Ágiles trabajan en iteraciones cortas**

En un proyecto ágil no hay delimitación de fases, hay requisitos iniciales, fase seguida de análisis, seguido de diseño arquitectónico y así sucesivamente, una vez el proyecto ha comenzado, todo el trabajo (requerimientos, diseño, codificación, pruebas, y así sucesivamente) ocurre simultáneamente dentro de cada iteración.

#### **2.3.5.4. Equipos ágiles entregan partes funcionales en cada iteración**

Es importante que cada iteración contenga los requisitos correctos, codificados, probados y de calidad listo para entregar la parte funcional de software.

Una sola iteración normalmente no proporciona suficiente tiempo para completar una nueva funcionalidad para satisfacer los deseos del usuario o cliente, se introduce el

concepto más amplio de lanzamiento. Una versión comprende una o más iteraciones que construyen mutuamente un conjunto de funcionalidades relacionadas.

Mientras que son más comunes las iteraciones de una a cuatro semanas, un lanzamiento es típicamente dos a seis meses. Por ejemplo, en un sistema de gestión de inversiones, una versión puede incluir todas las funciones relacionadas con la compra y venta de fondos mutuos y fondos de mercado de dinero. Esto puede tomar seis iteraciones de dos semanas para completar (aproximadamente tres meses). Un segundo lanzamiento puede agregar acciones y bonos comerciales y tomar cuatro iteraciones adicionales de dos semanas. Pueden ocurrir a intervalos variables. Un primer lanzamiento puede tomar seis meses para desarrollarse, puede ser seguido de otro lanzamiento de tres meses más tarde, y así sucesivamente.

#### **2.3.5.5. Equipos ágiles en las prioridades del negocio**

Los equipos ágiles demuestran un compromiso con las prioridades del negocio de dos maneras. En primer lugar, ofrecen características en el orden especificado por el propietario del producto, que se espera priorice y combine funciones en una versión que optimiza el retorno de la inversión de la organización en el proyecto.

Para lograr esto, se crea en función de las capacidades del equipo y una lista de prioridades deseados en un plan de lanzamiento. Se prioriza coordinando con el propietario del producto, las características deben ser escritas con el fin de minimizar las dependencias técnicas entre ellos.

Es difícil para el propietario de un producto priorizar funciones en un plan de liberación si la selección de una función requiere el desarrollo previo de otros. Es poco probable que

un equipo alcance una meta sin dependencias, sin embargo, a menudo es factible disminuir las dependencias.

La segunda manera es completar y entregar características valoradas por el usuario en lugar de completar tareas aisladas. Algo recomendable es trabajar con las historias de usuarios que son una técnica ligera de expresar los requisitos de software.

Una historia de usuario es una breve descripción de la funcionalidad vista por un usuario o cliente del sistema. Las historias de usuario son de forma libre y no existe ninguna sintaxis obligatoria. Sin embargo, puede ser útil pensar en una historia de colocación generalmente de la forma: “ < tipo de usuario >, quiero <capacidad> para <valor para el negocio >”.

Las historias de usuario son livianas porque el trabajo para recopilar y documentar no se realiza por adelantado. En lugar de escribir una especificación de requerimientos larga, los equipos ágiles han encontrado una mejor manera de especificar los requerimientos con el enfoque “JUST-IN-TIME”. Típicamente esto comienza con una breve descripción de una historia de usuario que está escrito a mano en una tarjeta. La tarjeta de la historia es sólo el comienzo, sin embargo, la historia de cada usuario va acompañada de tantas conversaciones entre los desarrolladores y el propietario del producto según sea necesario. Estas conversaciones ocurren tan a menudo como sea necesario. La documentación escrita puede seguir existiendo cuando se utiliza un enfoque de requerimientos basados en la historia. Sin embargo, el enfoque puede ser cambiado dramáticamente de escrito a la comunicación verbal.

### 2.3.6. Conceptos en proyectos ágiles <sup>6</sup>

Los conceptos ágiles comunes son:

#### 2.3.6.1. Historia

Una historia representa la unidad más pequeña medible de trabajo desde el punto de vista externo. Cada historia está asociada a alguna unidad de valor del negocio.

#### 2.3.6.2. “Sprint” / Iteración

Representa la unidad más pequeña que se entrega (parte del producto), está terminada y probada. Durante este período, todos los miembros del equipo están trabajando por un período fijo de tiempo, normalmente 1-4 semanas. Al trabajar en horario fijo con lanzamientos regulares, los equipos alimentan de valor a la organización.

#### 2.3.6.3. Backlog

Representa el alcance total del proyecto conocido hasta la fecha. A medida que los elementos del backlog se perfeccionaron a lo largo del tiempo. El Backlog evolucionará de forma continua mientras el producto esté en el mercado, para darle valor de forma continua, y mantenerlo útil y competitivo.

#### 2.3.6.4. Reunión de stand-up <sup>7</sup>

Son reuniones diarias de corta duración que se llevan a cabo para conocer el estado del

<sup>6</sup> Juan Palacio, Claudia Ruata (2011). Elementos de Scrum. *Scrum Manager Gestión de Proyectos*, 60, 61, 74, 75, 76,85. España: Safe Creative.

<sup>7</sup> S.Chandramouli, Saikat Dutt. (2012). Monitoring and Controlling (Iterative cycle review,daily stand-up). *PMI Agile Certified Practitioner*. India: Dorling Kindersley.

proyecto, impedimentos o bloqueantes del sprint o el proyecto en general, etc.

### 2.3.7. Planeación, monitoreo y adaptación

#### 2.3.7.1. Planificación adaptativa <sup>8</sup>

Con la planificación adaptativa se espera que el director del proyecto defina un plan detallado para la iteración inmediata siguiente y no solo un plan de alto nivel. Esta es siempre una situación mucho más cómoda y realista. Esto también se conoce como un equilibrio en la planificación haciéndola más clara y resolviendo complejidades.

Sólo los hitos clave se resaltan en las etapas iniciales.

La planificación innovadora o adaptativa se fundamenta principalmente en:

- La creencia de que el valor principal de la planeación no descansa en los planes que se producen, sino en el proceso de producirlos. El proceso es el producto más importante; por tanto, la importancia de la planeación para los ejecutivos reside en su participación en el proceso, no en el consumo de ese producto. A partir de esto se deduce que la planeación efectiva no puede hacerse para una organización, sino por esa organización.
- Gran parte de la necesidad actual de planeación obedece a la falta de administración y controles efectivos. El principal objetivo no es corregir las deficiencias producidas por acciones hechas anteriormente, sino que se dirige a crear el futuro deseado.
- Nuestro conocimiento del futuro se puede clasificar en tres tipos: certidumbre, incertidumbre e ignorancia; cada uno de ellos requiere un tipo distinto de

<sup>8</sup> S.Chandramouli, Saikat Dutt. (2012). What is adaptive Planning. *PMI Agile Certified Practitioner*. India: Dorling Kindersley.

- planeación; compromiso para aquellos aspectos del futuro de los que no tenemos una seguridad virtual; contingencia, para los aspectos sobre los cuales no podemos estar relativamente ciertos, pero podemos asegurarnos razonablemente de cuáles son sus posibilidades. En estos casos se necesita una planificación contingente; es decir, preparar un plan para cada posibilidad para así poder aprovechar rápidamente las oportunidades que se presenten cuando se decida el futuro. Por último, para aquellos aspectos que no se puedan prever hay que recurrir a la sensibilidad, a una planeación reactiva, encaminada al diseño de una organización y de un sistema de administración que pueda detectar fácilmente las desviaciones de la ruta asignada y reaccionar ante ellas de forma efectiva. Se trata de hacer flexible la organización y de instalar una forma de reaccionar.

#### **2.3.7.2. Iteración y planificación de liberación <sup>9</sup>**

El marco de planificación adaptativa ocurre en fases porque el objetivo se hace más claro sólo cuando llegamos hasta el nivel de detalle. Hay tres grandes fases en la planificación, se menciona a continuación:

- Planificación de liberación (Planificar versiones que se entregarán)
- Planificación de la iteración (Sprint, seleccionar y preparar las actividades que se realizarán, identificar y comunicar cuantas actividades se realizarán durante el sprint).

<sup>9</sup> S.Chandramouli, Saikat Dutt. (2012). Iteration and Release Planning. *PMI Agile Certified Practitioner*. India: Dorling Kindersley.

- Planificación diaria (Informes de trabajo, para toma de decisiones, ¿Qué he hecho?, ¿Qué voy hacer?, ¿Qué impedimentos tengo?).

### 2.3.7.3. Diario de planificación <sup>10</sup>

Durante la reunión diaria de stand-up (reuniones Scrum), el equipo ofrece actualización sobre el estado de los trabajos del día anterior, esto sucede por una duración máxima de 15 minutos.

Las reuniones diarias no deben de tomar mucho tiempo. Los miembros del equipo recibirán un entendimiento común de cómo está progresando el sprint. Recuerde que no están reportando el estado al jefe de proyecto, sino a sí mismos.

Las reuniones diarias principalmente ocurren al comienzo del día. El líder del equipo facilita el Scrum diario (Sincronización de actividades), asisten todos los miembros del equipo. Es responsabilidad del líder del equipo tomar nota de los impedimentos de los miembros del equipo y obtener aquellos con la ayuda de la administración.

### 2.3.7.4. Seguimiento del proyecto ágil <sup>11</sup>

#### 2.3.7.4.1. Radiadores de Información

Un radiador de información es una pantalla de información del equipo situado en un lugar donde la gente puede ver cómo están trabajando, se entiende de un vistazo y es fácil de mantener al día, se pueden usar tarjetas, gráficos, tablas, cuadros, flujos de procesos, etc. que son plasmadas en la pared o pizarras.

<sup>10</sup> S.Chandramouli, Saikat Dutt. (2012). Daily Planning. *PMI Agile Certified Practitioner*. India: Dorling Kindersley.

<sup>11</sup> S.Chandramouli, Saikat Dutt. (2012). Agile Project Tracking. *PMI Agile Certified Practitioner*. India: Dorling Kindersley.

#### 2.3.7.4.2. Autoevaluación

La importancia de los radiadores de la información radica en el hecho de que ayudan al equipo a generar auto-corrección para evaluar el progreso de su equipo y tomar medidas correctivas a través de la discusión en equipo. Las tablas y gráficos que se mencionen en la reunión de equipo permiten una mejor comprensión de los procesos del proyecto.

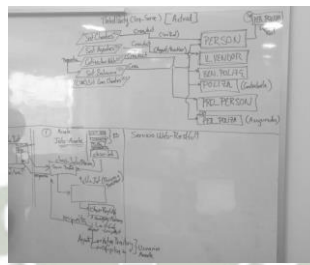


Figura 11. Radiador de la Información.

#### 2.3.8. Gestión de la calidad del proyecto <sup>12</sup>

##### 2.3.8.1. Introducción

La calidad se define como el grado en que un conjunto de características inherentes cumplen ciertos requisitos. El grado es una categoría de medición de elementos o características de un producto. La baja calidad es un problema de bajo grado. El costo total de todas las actividades relacionadas con la calidad en todo el ciclo de vida del producto se llama coste de la calidad.

La gestión de la calidad incluye los procesos y actividades de la organización ejecutante que determinan las políticas de calidad, objetivos y responsabilidades para que el proyecto satisfaga las necesidades para las cuales fue emprendido.

<sup>12</sup> S.Chandramouli, Saikat Dutt. (2012). *Product Quality. PMI Agile Certified Practitioner*. India: Dorling Kindersley.

La gestión de calidad moderna como TQM (gestión de calidad total) complementa la gestión de proyectos y también reconoce la importancia de las siguientes características.

- La satisfacción del cliente.
- Obtener beneficios para todos los miembros de la empresa.
- Mejoras en las condiciones de trabajo y en la formación del personal.

### 2.3.8.2. Costo de la calidad

El COQ se refiere a los costes totales de todos los esfuerzos relacionados con la calidad en todo el ciclo de vida del producto, como se muestra a continuación:

#### Costos de conformidad:

1. Prevención de costos
  - Planeación, encuestas
  - Capacitación e inducción
  - Revisiones de requerimientos y diseños
  - Revisión de códigos
  - Actividades para promover buen SW
2. Costos de evaluación:
  - Inspecciones, revisiones, testing
  - Testeo en procesos y calibración

#### Costos de no conformidad:

3. Costos de fracasos internos
  - Retrabajo

- Reparación, tiempo de inactividad
  - Evaluaciones defectuosas
  - Medidas correctivas
4. Costos de fracasos externos
- Rechazos, retorno, quejas
  - Inspecciones de cliente

### 2.3.8.3. Test Driven Development (TDD)

El desarrollo guiado por pruebas (TDD) es una práctica de programación que involucra otras dos prácticas: Escribir las pruebas primero (Test First Development) y Refactorización (Refactoring). Para escribir las pruebas generalmente se utilizan las pruebas unitarias. En primer lugar, se escribe una prueba y se verifica que las pruebas fallan. A continuación, se implementa el código que hace que la prueba pase satisfactoriamente y seguidamente se refactoriza el código escrito. El propósito del desarrollo guiado por pruebas es lograr un código limpio que funcione. La idea es que los requisitos sean traducidos a pruebas, de este modo, cuando las pruebas pasen se garantizará el software cumple con los requisitos que se han establecido.

### 2.3.8.4. Integración continúa

La integración continua es un modelo informático propuesto inicialmente por Martin Fowler que consiste en hacer integraciones automáticas de un proyecto lo más a menudo posible para así poder detectar fallos cuanto antes. Entendemos por integración la compilación y ejecución de test de todo un proyecto.

En la gestión de proyectos ágil, integrar y construir sucede muchas veces al día, cada vez que se completa una tarea. Permite a los desarrolladores crear un pequeño incremento de

las entregas en cada ciclo de iteración. La integración continua automatizada es posible con sofisticadas herramientas de construcción y herramientas de configuración. No sólo reduce los problemas relacionados con integraciones, también se asegura de que las pruebas se pueden hacer con frecuencia.

#### **2.3.8.5. Entrega incremental**

La entrega se divide en incrementos (ciclo iterativo) con la entrega de cada parte de la funcionalidad requerida y los más altos requisitos de prioridad se incluyen en incrementos iniciales.

La gestión de proyectos tradicional trata de dar más prestaciones provisionales antes de lanzar el producto final. Pero la gestión de proyectos ágiles trata de dar un producto incremental de sí mismo en todas las etapas del proyecto, en lugar de dar una entrega provisional en forma de documentos. Las siguientes son las ventajas de la entrega del producto incremental.

- Justo a tiempo (JIT)
- Entregar lo más rápido posible
- Los primeros incrementos actúan como prototipo para requisitos posteriores
- Construir integridad en el sistema y el equipo

#### **2.3.9. Gestión de las comunicaciones del proyecto**<sup>13</sup>

##### **2.3.9.1. Gerencia de comunicaciones**

En general, más del 90% del tiempo de un gerente de proyecto tradicional se destina sólo

<sup>13</sup> S.Chandramouli, Saikat Dutt. (2012). *Communications. PMI Agile Certified Practitioner*. India: Dorling Kindersley.

para la comunicación, lo que indica la importancia de las habilidades de comunicación. Suponiendo que un gerente de proyecto pasa 100 horas en un proyecto, más de 90 horas de su tiempo se utiliza sólo para la comunicación.

El marco ágil da gran importancia a la comunicación, tanto interna como externa. La retroalimentación inmediata y rápida por parte del cliente es el principio fundamental de cualquier proyecto Ágil.

Tradicionalmente, los jefes de proyecto crean un plan de gestión de comunicaciones, preparan informes basados en esto, discuten con las partes interesadas (stakeholders) acerca de sus necesidades, identifican sus necesidades de comunicación y con frecuencia las discuten en las reuniones del equipo para evitar problemas relacionados con las comunicaciones.

#### **2.3.9.2. Barreras para las comunicaciones**

1. Falta de canales de comunicación claros
2. Distancia física o temporal
3. Dificultades con el lenguaje técnico
4. Distracción de factores ambientales
5. Actitudes perjudiciales

#### **2.3.9.3. Plan de comunicaciones**

Es un proceso de determinación de las necesidades de información de los interesados del proyecto y la definición de un enfoque de comunicación en función de quién necesita qué información, cuándo lo necesitan, y cómo se les dará a ellos y por quién.

La comunicación efectiva significa que la información se proporciona en el formato adecuado y en el momento adecuado y con el impacto adecuado sobre el remitente y el receptor, mientras que la información eficiente significa proporcionar sólo la información necesaria. La estructura de la organización juega un gran impacto en los requisitos de comunicaciones del proyecto.

Para el director del proyecto y el líder de equipo no sólo es necesario planificar la comunicación de información eficiente, también necesita asegurarse de que la comunicación en general es eficaz. Los resultados del plan de comunicación deben revisarse regularmente y según sea necesario para garantizar la aplicabilidad continuada.

### **2.3.10. Métricas ágiles y estimaciones <sup>14</sup>**

#### **2.3.10.1. Puntos de historia**

Como se dijo, los puntos de la historia son una unidad de medida para expresar el tamaño total de una historia de usuario, función u otro trabajo. Cuando estimamos con puntos la historia le asignamos un valor de punto a cada elemento.

#### **2.3.10.2. Tiempo ideal**

Un problema familiar que la gente encuentra en la estimación del tiempo es el momento ideal y el tiempo real. Momento ideal es el número de días u horas que una tarea tomará si se concentra completamente en él y trabajar en él sin ningún tipo de interrupciones. Pero esto no suele ser así ya que el equipo va a experimentar distracción debido a una o varias de las siguientes acciones:

<sup>14</sup> S.Chandramouli, Saikat Dutt. (2012). Agile, Metrics and Estimations. *PMI Agile Certified Practitioner*.

India: Dorling Kindersley.

- Reuniones
- E-mails
- Soporte a aplicaciones
- Las llamadas telefónicas
- Capacitaciones
- Ser reasignado a otro proyecto
- Cambio de tareas

A medida que estas distracciones son sucesos inciertos, lo que representa para ellos en la estimación del tiempo no va a producir una estimación eficiente.

La mejor manera es combinar la velocidad del equipo en proyectos anteriores con la estimación de tiempo ideal para llegar a una estimación realista.

### **2.3.10.3. Tiempo ideal y Desarrollo de Software**

En un proyecto de software, el momento ideal difiere del tiempo transcurrido a causa de los tiempos de espera, pases incompletos, y los errores, debido a la sobrecarga física que experimentan los desarrolladores y actores del proyecto.

Además, en el estudio de por qué el tiempo ideal no hace el mismo tiempo que el transcurrido, considere que los administradores son capaces de trabajar un promedio de sólo cinco minutos entre las interrupciones (Hobbs 1987). Incluso si el programador típico se interrumpe sólo una tercera parte con una frecuencia de quince minutos ya es una interrupción.

Los problemas pueden surgir cuando un gerente le pide a un miembro del equipo la inevitable pregunta: "¿Cuánto tiempo va a tomar"? El miembro del equipo responde "cinco días", por lo que el gerente cuenta cinco días en su calendario y marca el día con una gran X roja. El miembro del equipo, sin embargo, en realidad quería decir: "Cinco días si eso es todo por hacer, pero sí surgen muchas otras cosas por hacer probablemente sean dos semanas".

En un proyecto de software, la multitarea también ensancha la brecha entre el tiempo ideal y el tiempo transcurrido. "Un desarrollador de software que se cuenta para realizar múltiples tareas pierde mucha eficiencia".

### **2.3.11. Gestión de costos del proyecto ágil**

#### **2.3.11.1. Gestión de costos del proyecto**

Se centra principalmente en el costo de los recursos necesarios para completar las actividades definidas. También considera los requisitos de los interesados del proyecto. Los diversos interesados medirán los costos del proyecto de diferentes maneras y en tiempos diferentes. Por ejemplo, el costo de adquisición de un artículo puede medirse cuando se toma la decisión o se hace el compromiso de adquirir el artículo en cuestión, cuando se realiza su pedido o se hace entrega del mismo, o cuando se incurre en el costo real o éste se registra a los fines de la contabilidad del proyecto. Dentro del ciclo iterativo (sprint) la estimación de costos y presupuestos de costos están estrechamente vinculados.

La capacidad de los actores para influir en el coste es mayor en las primeras etapas del proyecto. La gestión de costes son los procesos involucrados en la planificación, estimación, presupuesto y control de costes para asegurar que el proyecto se complete dentro del presupuesto aprobado.

- **Estimar los Costos.** Es el proceso que consiste en desarrollar una aproximación de los recursos financieros necesarios para completar las actividades del proyecto.
- **Determinar el Presupuesto.** Es el proceso que consiste en sumar los costos estimados de actividades individuales o paquetes de trabajo para establecer una línea base de costo autorizada.
- **Controlar los Costos.** Es el proceso que consiste en monitorear la situación del proyecto para actualizar el presupuesto del mismo y gestionar cambios a la línea base de costo.

El proyecto está precedido por un esfuerzo de planificación del equipo de dirección del proyecto. Este esfuerzo de planificación es parte del proceso desarrollar el plan para la dirección del Proyecto, lo cual produce un plan de gestión de costos que determina el formato y establece los criterios necesarios para planificar, estructurar, estimar, presupuestar y controlar los costos del proyecto.

### 2.3.12. Gestión de riesgos del proyecto ágil <sup>15</sup>

#### 2.3.12.1. Plan de gestión de riesgos

El proceso del plan de gestión de riesgos decide la forma de abordar y planificar las actividades de gestión de riesgos para un proyecto. La planificación de la gestión de riesgos cuidadosa y explícita mejora la probabilidad de éxito de otros procesos de gestión de riesgos. El objetivo principal de la planificación de riesgo ágil es permitir que el equipo cumpla con sus compromisos a largo plazo. Este contiene el proceso de cómo vamos a

<sup>15</sup> S.Chandramouli, Saikat Dutt. (2012). Agile Project Risk Management. *PMI Agile Certified Practitioner*. India: Dorling Kindersley.

identificar los riesgos y en qué nivel, ¿cómo vamos a analizar los riesgos?, ¿cómo vamos a monitorear y controlar los riesgos?, resultando en un plan de gestión de riesgos.

El objetivo principal de la planificación ágil del riesgo es permitir al equipo cumplir sus compromisos a largo plazo. El propósito de la mitigación del riesgo es reducir el impacto del riesgo.

### **2.3.12.2. Identificar los riesgos**

Este proceso determina qué riesgos pueden afectar al proyecto y documenta las características de cada riesgo.

El proyecto ágil puede ser cancelado incluso si nos encontramos con riesgos y problemas durante la fase de planificación. La identificación de los riesgos se realiza continuamente a lo largo del proyecto.

Las reuniones diarias stand-up identifican los riesgos continuos y también ayudan a identificar los problemas y riesgos del proyecto ágil. Todo el personal del proyecto debe ser alentado a identificar los riesgos, incluidos los clientes.

Tenemos que considerar el riesgo como una característica importante del producto y la necesidad de identificar y evaluar los riesgos de los proyectos en base diaria, no aisladamente, sino en función de proximidad y colaboración. Necesitamos proporcionar visibilidad sobre los riesgos del proyecto a cada participante del proyecto y actuar sobre ellos para eliminarlos o minimizar su impacto rápidamente.

### **2.3.12.3. Plan de respuestas del riesgo**

El proceso de desarrollar opciones y acciones para mejorar las oportunidades y reducir las amenazas a los objetivos del proyecto se refiere a la respuesta a los riesgos. El propósito de la mitigación de riesgos en el proyecto ágil es reducir el impacto del riesgo.

El plan de respuestas del riesgo es el proceso de determinación de las acciones con diversas posibilidades para mejorar las oportunidades y reducir las amenazas a los objetivos del proyecto.

La planificación de respuesta a los riesgos se debe basar en la gravedad del riesgo, el costo efectivo para superar el reto y oportunidad a tener éxito.

### **2.3.12.4. Vigilancia y control de riesgos**

El proceso de evaluación de la eficacia de riesgos en todo el proyecto se llama control de los riesgos.

Es el proceso de hacer seguimiento de los riesgos e identificar nuevos riesgos, asegurando la ejecución de los planes de riesgo, evaluación de la efectividad del plan y el cambio del plan, si es necesario.

### **2.3.12.5. Puntos de verificación en proyectos ágiles**

La principal ventaja ágil sobre otros enfoques es que el cliente está involucrado en el proceso de toma de decisiones no sólo en todas las etapas de la iteración y reuniones diarias stand-up, etc., sino también en los puestos de control. El cliente acepta la entrega incremental al final del ciclo iterativo, que se denomina un punto de control, el proceso de comprobación es similar a la comprobación del proceso de alcance. En el puesto de

control del equipo y del cliente se mira lo que se hizo, lo que salió mal, lo que salió bien y la planificación de la funcionalidad futura que se construirá en el próximo ciclo iterativo.

#### **2.3.12.6. Errores comunes de gestión de riesgos**

- Completar la identificación de riesgos sin conocer los detalles del proyecto.
- La identificación del riesgo no es una actividad puntual sino que es continua.
- Copiar ejemplos de plantillas proporcionadas o pasadas.
- Los riesgos se agrupan en general y no se especifican al proyecto.
- Revisar los riesgos a intervalos regulares.
- Sin la participación de equipo para el proceso de identificación.
- No identificar los propietarios del riesgo.
- No definir claramente los roles y responsabilidades para cada propietario del riesgo.

#### **2.3.13. OpenUp**

##### **2.3.13.1. Definición**

OpenUP (Proceso Unificado Abierto) es un método de desarrollo de software de fuente abierta compuesto por una serie de complementos (plugins) que en conjunto conforman el proceso.

##### **2.3.13.2. Principios básicos**

OpenUP se basa en cuatro principios básicos que se apoyan mutuamente.

### **2.3.13.2.1. Colaborar para alinear los intereses y un entendimiento compartido**

El software es creado por personas con diferentes intereses y habilidades quienes trabajan juntos para crear software eficientemente.

Por consiguiente, es necesario desarrollar prácticas que fomenten un ambiente de equipo saludable, lo que permitirá la colaboración efectiva, encuadrando los intereses de los participantes del proyecto (equipo de desarrollo, aseguramiento de la calidad, stakeholders del producto, clientes), desarrollando un entendimiento compartido del proyecto.

### **2.3.13.2.2. Balancear las prioridades para maximizar el valor de los stakeholder**

Los participantes del proyecto y los stakeholders deben colaborar para desarrollar una solución que maximice el beneficio de los stakeholders y cumpla con las restricciones planteadas en el proyecto. Lograr un balance es un proceso dinámico porque si los stakeholders y los participantes del proyecto aprenden más acerca del sistema, será beneficioso para el buen desarrollo y termino del proyecto.

### **2.3.13.2.3. Enfocarse en articular la arquitectura**

Sin un fundamento arquitectónico, un sistema evolucionará en una forma ineficiente y casual, presentando dificultades para evolucionar, reutilizarse o integrarse sin una reconstrucción sustancial. Esto también dificulta organizar el equipo o comunicar las ideas sin el enfoque técnico común que la arquitectura proporciona.

Por consiguiente, se debe usar la arquitectura como un punto focal para que los desarrolladores alineen sus intereses e ideas, articulando las decisiones técnicas esenciales a través de una arquitectura creciente.

#### **2.3.13.2.4. Evolucionar para obtener continuamente retroalimentación y progreso**

Usualmente no es posible conocer todas las necesidades de los “Stakeholders”, ser consiente de todos los riesgos, comprender todas las tecnologías del proyecto, o saber cómo trabajar con sus colegas. Aún si fuese posible conocer todas estas cosas, es probable que cambien durante la vida del proyecto.

Por consiguiente, divida el proyecto en mini proyectos, iteraciones encajadas en tiempo para demostrar valor incremental y obtener retroalimentación temprana y continua.

#### Consideraciones clave

Los proyectos podrían tener una o más iteraciones en la fase de inicio. Entre otras razones para tener múltiples iteraciones en la fase de inicio, se tiene a las siguientes:

1. El proyecto es grande y es difícil definir su alcance.
2. Sistemas sin precedentes.
3. Stakeholders con necesidades encontradas y relaciones complejas.
4. Mayores riesgos técnicos demandan la creación de un prototipo o prueba de conceptos.

#### **2.3.13.3. Fases**

##### **2.3.13.3.1. Fase de Inicio**

La primera de las cuatro fases en el ciclo de vida del proyecto es acerca de entender el alcance, los objetivos del proyecto y obtener suficiente información para confirmar que el proyecto es viable.

El propósito en esta fase es lograr concurrencia entre todos los involucrados (stakeholders) sobre los objetivos del ciclo de vida para el proyecto

Hay cuatro objetivos para la fase de Inicio que clarifican el alcance, los objetivos del proyecto y la viabilidad de la solución proyectada:

3. Entender qué construir, determinar la visión, el alcance del sistema y sus límites.  
Identifique quién está interesado en este sistema y por qué.
4. Identificar la funcionalidad clave del sistema, decidir qué requerimientos son los más críticos.
5. Determinar al menos una posible solución, identifique al menos una arquitectura candidata y su viabilidad.
6. Entender el costo, el cronograma y los riesgos asociados al proyecto.

#### **2.3.13.3.2. Fase de Elaboración**

El propósito de esta fase es establecer la línea base de la arquitectura del sistema y proporcionar una base estable para el esfuerzo de desarrollo.

Hay objetivos para la fase de elaboración que ayudan a direccionar los riesgos asociados con los requisitos, la arquitectura, los costos y el cronograma:

1. Obtenga un entendimiento más detallado de los requisitos, se deberá de tener un buen entendimiento de los requisitos que le permitan crear un plan más detallado y obtener ganancia de los stakeholders.
2. Establecer la línea base para la arquitectura, diseñar, implementar y probar un esqueleto estructural del sistema. Aunque la funcionalidad no sea completa aún,

muchas de las interfaces entre los bloques de construcción son implementadas y probadas. Esto se refiere a una arquitectura ejecutable.

3. Mitigar los riesgos esenciales y producir un cronograma exacto y costos estimados.

### **2.3.13.3.3. Fase de Construcción**

La construcción se enfoca en el diseño, implementación y prueba de las funcionalidades para desarrollar un sistema completo.

El propósito de esta fase es completar el desarrollo del sistema basado en la arquitectura.

Los siguientes objetivos de la fase de construcción ayudarán a tener un desarrollo completo con costo eficiente, una versión operativa del sistema que pueda ser entregada a la comunidad de usuarios:

1. Desarrollar iterativamente un producto que esté listo para hacer transición a su comunidad de usuarios. Describir los requisitos restantes, completar la implementación y probar el software. Liberar la primera versión operativa del software (beta) y determinar si los usuarios están listos para que la aplicación sea desplegada.
2. Minimizar el costo de desarrollo y alcance, optimizar los recursos y promover el paralelismo de desarrollo entre desarrolladores o equipos de desarrolladores, por ejemplo, asignar componentes que puedan ser desarrollados independientemente una del otro.

#### Consideraciones Clave

Típicamente, la fase de construcción tiene más iteraciones (dos o cuatro) que las otras fases, dependiendo del tipo de proyecto:

1. Proyecto Simple: Una iteración para construir el producto (liberación beta).
2. Proyecto mediano: Una iteración para exponer un sistema parcial y una para madurar este a un beta testing.
3. Proyecto grande: Tres o más iteraciones, dado el tamaño del proyecto (número de requisitos a implementar para una liberación beta).

#### **2.3.13.3.4. Fase de Transición**

Cuarta y última fase en el ciclo de vida del proyecto.

El propósito en esta fase es asegurarse que el software está listo para entregarse a los usuarios.

Los siguientes objetivos de la fase de transición ayudarán a afinar la funcionalidad, el desempeño y la calidad total de la versión beta del producto:

- La prueba beta valida que las expectativas del usuario sean satisfechas, esto típicamente requiere algunas actividades de afinamiento, tales como depuración de errores y mejora del desempeño y la usabilidad.
- Lograr que los stakeholders concuerden en que la implementación ha terminado, esto puede implicar varios niveles de pruebas de aceptación del producto, incluyendo pruebas formales, informales y pruebas beta.
- Mejorar el desempeño en futuros proyectos a través de lecciones aprendidas, documentar las lecciones aprendidas y mejorar el ambiente de los procesos y las herramientas para el proyecto.

#### **2.3.13.4. Iteración**

Una iteración es un conjunto de periodos de tiempo dentro de un proyecto en los cuales se produce una versión del producto ejecutable, estable, junto con cualquier otra documentación de soporte, instalación de scripts o similar, necesarios para usar esta liberación.

#### **2.3.13.5. Beneficio de las iteraciones**

Las aplicaciones de software de hoy son bastante complejas para definir los requisitos secuencialmente, idear una arquitectura y diseño, hacer una implementación, llevar a cabo una prueba y obtener lo correcto. Si se usan aproximaciones reiterativas o de cascada, sus requisitos iniciales, arquitectura, diseño y código serán sub - óptimos. Con el desarrollo en cascada no se obtiene retroalimentación significativa sobre mejoras que pueden ser hechas en el proyecto hasta que es bastante tarde lo que lo hace costoso. En contraste, dividir el proyecto en una serie de iteraciones encuadradas en tiempo le permite entregar funcionalidades que pueden ser evaluadas por los stakeholders al final de cada iteración. Esta aproximación provee ciclos de retroalimentación rápida y oportuna permitiendo hacer evaluaciones dirigidas y hacer mejoras a un bajo costo mientras el presupuesto y el tiempo todavía lo permiten y antes de que el proyecto haya ido tan lejos que se requieran mayores reconstrucciones.

#### **2.3.13.6. Caso de uso**

Un caso de uso describe lo que el sistema debe hacer para dar valor a los stakeholders.

Una instancia de caso de uso se define como una secuencia de acciones desempeñadas por el sistema que produce un resultado observable de valor para un actor en particular.

Hay varias palabras clave en esta definición:

- *Instancia de caso de uso*, la secuencia referida en la definición es realmente un flujo específico de eventos a través del sistema, o una instancia. Muchos flujos de eventos son posibles, y muchos podrían ser muy similares. Para hacer un caso de uso entendible, se deberá de agrupar flujos similares de eventos dentro de un único caso de uso; identificar y describir un caso de uso realmente significa identificar y describir un grupo de flujos de eventos relacionados.
- *Acciones*, una acción es un procedimiento computacional o algorítmico, se invoca cuando el actor provee una señal al sistema o cuando el sistema obtiene un evento. Una acción puede implicar transmisión de señales a cada uno de los actores involucrados u otros actores. Una acción es atómica, lo cual significa que es desempeñada toda o definitivamente nada de ella.
- *Desempeñada por el sistema*, esto significa que el sistema proporciona el caso de uso. Un actor se comunica con una instancia de caso de uso del sistema.
- *Un resultado observable de valor*, se puede poner un valor sobre un caso de uso desempeñado satisfactoriamente. Un caso de uso debe asegurarse de que un actor pueda desempeñar una tarea que tiene un valor identificable. Esto es muy importante para determinar el nivel correcto de granularidad para un caso de uso.
- *Un actor particular*, los actores son claves para hallar el caso de uso correcto, especialmente porque el actor ayuda a evitar casos de uso que son demasiado largos. Como un ejemplo, considere una herramienta de modelado visual. Hay realmente dos actores para esta aplicación: un desarrollador, alguien quien desarrolla el sistema usando la herramienta como soporte; y un administrador del sistema, alguien que

administra la herramienta. Cada uno de estos actores tiene su propia demanda sobre el sistema y además requerirá su propio conjunto de casos de uso.

La funcionalidad de un sistema está definida por diferentes casos de uso, cada uno de los cuales representa una meta para un actor en particular. La descripción de un caso de uso define qué sucede en el sistema cuando el caso de uso es llevado a cabo.

### **2.3.13.7. Arquitectura de software**

La arquitectura de software representa la estructura o estructuras del sistema que consisten en componentes de software, las propiedades externas visibles de esos componentes y las relaciones entre ellos.

En OpenUP la arquitectura de un sistema de software (en un punto dado) es la organización o estructura de los componentes más significativos del sistema interactuando, a través de interfaces, con componentes sucesivamente más pequeños.

La arquitectura se refiere únicamente a algunos aspectos específicos:

- La estructura del modelo, los patrones de organización, por ejemplo, capas.
- Los elementos esenciales, los casos de uso críticos, clases principales, mecanismos comunes, y así sucesivamente.
- Unos cuantos escenarios clave que muestran el control de los flujos principales de todo el sistema.
- Los servicios, para captar la modularidad, características opcionales, aspectos de línea de productos.

### 2.3.13.8. Roles

Entre los roles usados en OpenUp tenemos:

- Analista
- Arquitecto
- Desarrollador
- Project Manager
- Stakeholder
- Analista de Calidad

### 2.3.14. Proceso Unificado de Rational (RUP) <sup>16</sup>

Es una plataforma de proceso de desarrollo de software configurable que entrega mejores prácticas comprobadas y una arquitectura configurable.

#### 2.3.14.1. Características esenciales

##### 2.3.14.1.1. Dirigido por Casos de Uso

En RUP los casos de uso no son sólo una herramienta para especificar los requisitos del sistema. También guían su diseño, implementación y prueba. Los casos de uso constituyen un elemento integrador (Requisitos, Análisis y diseño, Implementación, Pruebas) y una guía de trabajo.

##### 2.3.14.1.2. Proceso centrado en la arquitectura

La arquitectura de un sistema es la organización o estructura de sus partes más relevantes,

<sup>16</sup> Jacobson Sten. (2002). The Rational Objectory Process - A UML - based Software Engineering Process.

lo que permite tener una visión común entre todos los involucrados (desarrolladores y usuarios) y una perspectiva clara del sistema completo, necesaria para controlar el desarrollo.

En el caso de RUP además de utilizar los casos de uso para guiar el proceso se presta especial atención al establecimiento temprano de una buena arquitectura que no se vea fuertemente impactada ante cambios posteriores durante la construcción y el mantenimiento. En la Figura 12 se ilustra la evolución de la arquitectura durante las fases de RUP. Se tiene una arquitectura más robusta en las fases finales del proyecto. En las fases iniciales lo que se hace es ir consolidando la arquitectura por medio de las líneas base y se va modificando dependiendo de las necesidades del proyecto.

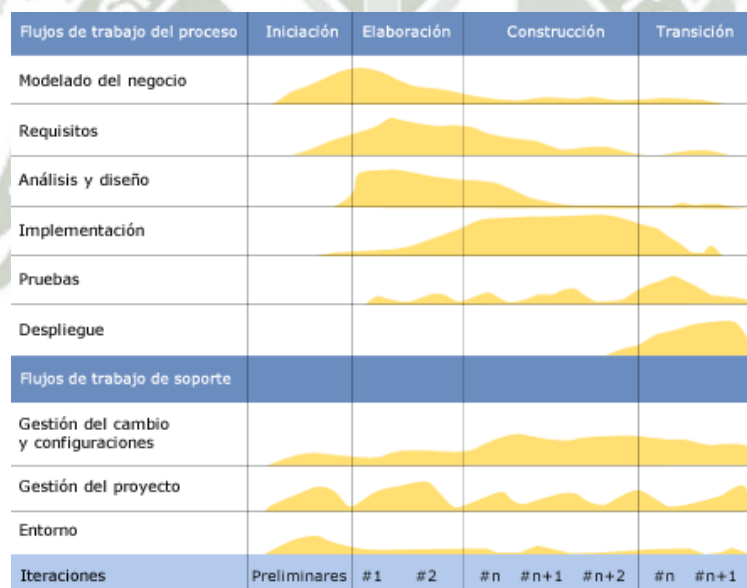


Figura 12. Evolución de la arquitectura durante las fases de RUP.

The Rational Objectory Process - A UML-based Software Engineering Process,  
Jacobson, Sten, 2002.

### 2.3.14.1.3. Proceso iterativo e incremental

El equilibrio correcto entre los casos de uso y la arquitectura es algo muy parecido al equilibrio de la forma y la función en el desarrollo del producto, lo cual se consigue con el tiempo. Para esto, la estrategia que se propone en RUP es tener un proceso iterativo e incremental en donde el trabajo se divide en partes más pequeñas o mini proyectos, permitiendo que el equilibrio entre casos de uso y arquitectura se vaya logrando durante cada mini proyecto, así durante todo el proceso de desarrollo.

Cada mini proyecto se puede ver como una iteración (un recorrido más o menos completo a lo largo de todos los flujos de trabajo fundamentales) del cual se obtiene un incremento que produce un crecimiento en el producto.

Una iteración puede realizarse por medio de una cascada como se muestra en la Figura 13, se pasa por los flujos fundamentales (Requisitos, Análisis, Diseño, Implementación y Pruebas), también existe una planificación de la iteración, un análisis de la iteración y algunas actividades específicas de la iteración. Al finalizar se realiza una integración de los resultados con lo obtenido de las iteraciones anteriores.

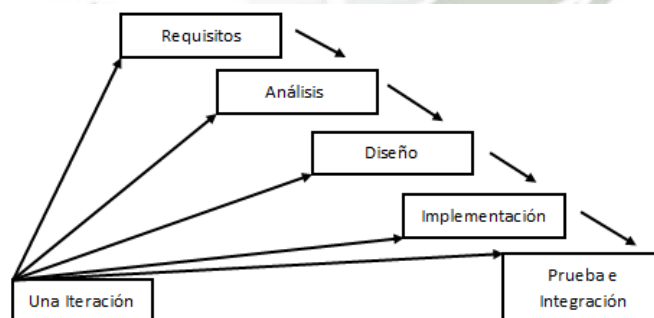


Figura 13. Iteración RUP

The Rational Objectory Process - A UML-based Software Engineering Process,  
Jacobson, Sten, 2002.

El proceso iterativo e incremental consta de una secuencia de iteraciones. Cada iteración aborda una parte de la funcionalidad total, pasando por todos los flujos de trabajo relevantes y refinando la arquitectura. Cada iteración se analiza cuando termina, se puede determinar si han aparecido nuevos requisitos o si han cambiado los existentes, afectando a las iteraciones siguientes. Durante la planificación de los detalles de la siguiente iteración, el equipo también examina cómo afectarán los riesgos que aún quedan al trabajo en curso. Toda la retroalimentación de la iteración pasada permite reajustar los objetivos para las siguientes iteraciones.

RUP divide el proceso en cuatro fases, en las primeras iteraciones las fases de inicio y elaboración se enfocan hacia la comprensión del problema y la tecnología, la delimitación del ámbito del proyecto, la eliminación de los riesgos críticos, y al establecimiento de una línea base de la arquitectura.

Durante la fase de inicio las iteraciones ponen mayor énfasis en actividades de modelado del negocio y de requisitos.

En la fase de elaboración, las iteraciones se orientan al desarrollo de la línea base de la arquitectura, abarcan más los flujos de trabajo de requerimientos, modelo de negocios (refinamiento), análisis y diseño.

En la fase de construcción, se lleva a cabo la construcción del producto por medio de una serie de iteraciones.

Finalmente en la fase de transición se garantiza que se tiene un producto preparado para su entrega a la comunidad de usuarios.

## CAPITULO 3: MECANISMO PROPUESTO

### 3.1. Resumen de mecanismo

El siguiente diagrama resume el mecanismo ágil propuesto en el presente trabajo de investigación.



Diagrama. Resumen de mecanismo ágil propuesto.  
Fuente propia.

### 3.2. Solución propuesta

Mecanismo ágil para la gestión y desarrollo de proyectos de software, aplicado en instituciones de seguros y de telecomunicaciones.

### 3.3. Generalidades

Para el desarrollo del mecanismo ágil se han utilizado las mejores prácticas de la guía del Project Management Body of knowledge (PMBOK), SCRUM, OpenUp y RUP, como también información propia generada gracias a la experiencia laboral en empresas de prestigio.

El mecanismo ágil contiene etapas y fases, dentro de ellas se tienen tres conceptos básicos los que permiten el funcionamiento y buen desempeño del mecanismo ágil, los que son: roles, actividades y artefactos.

### 3.3.1. Etapas y fases

El uso de etapas y fases en el mecanismo ágil de desarrollo y gestión de proyectos como se indican en la Figura 3.2 permite estructurar el ciclo de vida del proyecto de manera ordenada y clara, permitiéndonos seguir un flujo de actividades, procesos y responsabilidades, que son de conocimiento de todos los involucrados y responsables del proyecto, asegurando el cumplimiento de hitos y demás procesos que se llevan a cabo durante el ciclo de vida del proyecto desde su inicio hasta su puesta en producción.

#### 3.3.1.1. Roles

- Usuario solicitante.- Es el usuario que solicita la elaboración de una tarea o proyecto de software a la Gerencia de Tecnologías de Información.
- Analista Funcional.- Es el encargado de realizar el análisis funcional del proyecto en base al requerimiento solicitado por el usuario, da soporte, seguimiento y control durante la ejecución y cierre del proyecto.
- Líder Analista Programador.- El Líder AP es el encargado de dirigir y controlar al equipo de analistas programadores, asigna y libera recursos de las tareas o proyectos que el Project Manager le asignó.
- Líder de SOAP.- Es el líder de equipo de los Analistas de Soporte de Aplicaciones, los que trabajan en el ambiente de producción.
- Encargado de Arquitectura.- Es el responsable del equipo de arquitectos de software.
- Líder de Calidad.- Es el encargado del equipo de calidad, debiendo validar el cumplimiento de los estándares y normas en el desarrollo del proyecto y del correcto

funcionamiento del mismo según los documentos aprobados por la Gerencia y Jefaturas de tecnologías de información.

- Analista de Arquitectura.- Responsable de validar que el desarrollo del proyecto cumpla con los estándares y normas de arquitectura, cumplen labores de análisis y diseño referentes a la arquitectura del proyecto, pueden especializarse en arquitectos de solución, datos, servicios, aplicaciones, plataforma y producto según cada institución.
- Líder QA.- Responsable de asignación de recursos y tiempos de prueba para cada proyecto dependiendo de la prioridad que tengan asignados, es el encargado de asignar los Analistas de QA que llevarán a cabo las pruebas y certificación de cada proyecto.
- Analista de QA.- Responsable de validar que el desarrollo del proyecto cumpla con los estándares y normas de calidad y del desarrollo de las pruebas QA se encarga de solicitar la conformidad de las pruebas QA con el Usuario y Analista Funcional.
- DBA.- Encargado de la administración de bases de datos.
- Analista Programador.- Encargado del diseño y desarrollo de uno o más módulos del proyecto.
- Analista SOAP.- Encargado del soporte de aplicaciones, da soporte en caso de errores o incidencias, da ayuda para el pase de proyectos a los ambientes de producción o para generar pruebas post producción.
- Encargado de fuentes.- Encargado del suministro y mantenimiento de fuentes.
- Jefe de Proyecto.- Es el encargado de controlar el ciclo de vida de los proyectos asignados a su portafolio y del cumplimiento del cronograma de los mismos, como también de elaborar el Informe de Estado de cada uno de sus proyectos asignados por el Project Manager o Gerente de Proyecto según la cultura organizacional de cada institución, también se encarga de revisar y validar los documentos elaborados para la ejecución del proyecto (Especificación Funcional y Técnica), debiendo de estar correctamente detallados y claros. A su vez deberán de solicitar los recursos necesarios para el inicio del

proyecto como: Analistas Funcionales, Analistas Programadores, Arquitectos, esta solicitud se deriva a la PMO o a la Oficina de Gestión de Recursos según sea el caso.

- Project Manager.- Es el encargado del seguimiento y control de los proyectos asignados a su portafolio, puede tomar el rol de Gerente de Proyecto, mayormente es el Jefe de Área en Tecnologías de Información.
- Gerente de TI.- Encargado de la administración presente y futura de las tecnologías de la información de la empresa.
- Jefe de Seguridad de la Información.- Encargado del cumplimiento y alineación de los estándares y normas de seguridad de información de la empresa a los proyectos de software.

### 3.3.1.2. Actividades y artefactos

El mecanismo propuesto propone el uso de los siguientes roles, actividades y artefactos los cuales aseguran y facilitan la gestión, desarrollo, control y cierre del proyecto, la Tabla 14 lista los roles, actividades y artefactos según la etapa y fase en la que se encuentra el proyecto.

Etapa	Proceso	Plantilla / Actividad	Frecuencia	Rol de Responsable	Participantes	Aplica a Proyecto	Observaciones
<b>INICIO</b>							
Inicio	Asignar Líder de Proyecto	Aplicativo PPM Clarity	Uno por Proyecto	Jefe de Proyecto	Gerente de proyecto Jefe de Proyecto	Si	
Inicio	Reunión con el usuario para realizar el levantamiento de información sobre las necesidades del usuario.	Documento de acta de reunión	Según lo requerido	Jefe de Proyecto	Jefe de Proyecto	Si	El AF y/o el AQ participarán de esta actividad cuando haya sido solicitado por el JdP a la Oficina de gestión de recursos
Inicio	Reunión con las áreas involucradas al proyecto.	Documento de acta de reunión	Según lo requerido	Jefe de Proyecto	Jefe de Proyecto	Si	El AF y/o el AQ participarán de esta actividad cuando haya sido solicitado por el JdP a la Oficina de gestión de recursos
Inicio	Elaborar Ficha de Inicio de Proyecto	Documento de Ficha Inicio Proyecto	Uno por Proyecto	Jefe de Proyecto	Jefe de Proyecto Analista Funcional	Si	
Inicio	Aprobar Ficha de inicio de proyecto	Documento de Ficha Inicio Proyecto	Uno por Proyecto	Gerente de proyecto	Gerente de proyecto Jefe de Proyecto	Si	
<b>PLANIFICACIÓN</b>							
Planificación	Reunión de Kick – Off	Presentación Kick - Off	Según lo requerido	Jefe de Proyecto	Gerente de proyecto Jefe de Proyecto Analista Funcional Arquitecto	Si/No	Reunión para el lanzamiento del proyecto
Planificación	Acta de Reunión Kick – Off	Documento de acta de reunión	Según lo requerido	Líder de Proyecto (Analista Funcional)	Jefe de Proyecto Analista Funcional	Si/No	Registrar acuerdos, observaciones y/o sugerencias de reunión

Planificación	Elaborar CheckList (Entregables del Proyecto)	CheckList	Uno por Proyecto	Líder de Proyecto (Analista Funcional)	Jefe de Proyecto Analista Funcional	Si/No	
Análisis	Reunión con el usuario para realizar el levantamiento de información sobre el proyecto.	Documento de acta de reunión	Según lo requerido	Jefe de Proyecto	JdP Arquitecto AF Usuario Soap Líder desarrollo Líder calidad	Si	
Análisis	Reunión con las áreas de TI para evaluar el impacto del proyecto	Documento de Arquitectura Conceptual Documento de Especificación Funcional	Uno por Proyecto	Analista Funcional	JdP Arquitecto AF Usuario Soap Líder desarrollo Líder calidad Seguridad de la Información	Si	
Análisis	Elaborar Documento de Especificación Funcional	Documento de Especificación Funcional	Uno por Proyecto	Analista Funcional	Jefe de Proyecto Arquitecto Analista Funcional	Si/No	
Análisis	Elaboración de la arquitectura conceptual	Documento de Arquitectura Conceptual	Uno por Proyecto	Arquitecto	Jefe de Proyecto Arquitecto Analista Funcional	Si	En caso de ser necesario el Arquitecto solicitará la asignación del Arquitecto de Solución, Arquitecto de Datos, Arquitecto de Servicios o Arquitecto de Aplicaciones

Análisis	Validar Especificación Funcional con el Usuario, Seguridad y Gerente	Documento de Especificación Funcional	Según lo requerido	Jefe de Proyecto	JdP Arquitecto Usuario AF Soap Líder desarrollo Líder calidad Seguridad GdP	Si	El documento de Especificación Funcional podrá incluir como anexo el Documento de Arquitectura Conceptual según el criterio del AF
Análisis	Convocar a comité técnico	Documento de Especificación Funcional	Uno por Proyecto	Jefe de Proyecto	JdP Arquitecto AF Soap Líder desarrollo Líder calidad Seguridad GdP	Si	El Comité Técnico se convoca con 48 hrs de anticipación, en la convocatoria se debe adjuntar la última versión de la EF de manera que los participantes puedan conocer el proyecto que se expondrá y aporten sugerencias, observaciones o mejoras.
Análisis	Sustentación de la EF en el Comité Técnico	Documento de Especificación Funcional	Uno por Proyecto	AF AQ	JdP Arquitecto AF Soap Líder desarrollo Líder calidad Seguridad GdP	Si	El AF debe de sustentar los RF planteados en la EF
Análisis	Elaborar acta de reunión del comité técnico	Documento de Acta de Reunión	Según lo requerido	Jefe de Proyecto	JdP Arquitecto	Si	El Jefe de proyecto deberá de elaborar el Acta de reunión del comité técnico y el

					AF Soap Líder desarrollo Líder calidad Seguridad GdP		Analista Funcional actualizará la EF de haber adecuaciones en los RF.
Análisis	Enviar por mail la EF, acta de reunión y el acta de asistencia a todos los participantes del comité técnico	Documento de Especificación Funcional	Según lo requerido	Jefe de Proyecto	Jefe de Proyecto	Si	
Análisis	Elaborar documento de Arquitectura Lógica	Documento de Arquitectura Lógica	Uno por Proyecto	Arquitecto	Jefe de Proyecto Arquitecto Analista Funcional	Si/No	Luego de contar con la EF cerrada, el Arquitecto asignado deberá de elaborar o completar el documento de Arquitectura lógica que servirá como input para la estimación de tiempos de desarrollo, debe de estar lista 2 o 3 días luego de ejecutado el comité técnico
Análisis	Solicitar las aprobaciones de la PS a las siguientes personas: - Usuario Solicitante - Seguridad de la Información - Gerente de Proyecto	Documento de Especificación Funcional	Según lo requerido	Jefe de Proyecto	Jefe de Proyecto Analista Funcional	Si	Se solicitará la aprobación formal de la EF y de la Matriz de Riesgos (de aplicar cambios mayores o complejos)

Análisis	Solicitar estimación de esfuerzo inicial en QA (vía email)	Documento de Especificación Funcional	Según lo requerido	Jefe de Proyecto	Jefe de Proyecto	Si	Se deberá de enviar un correo a QA solicitando estimación de tiempos de pruebas en QA, estos tiempos serán el input para elaborar el cronograma del proyecto
Análisis	Solicitar asignación de analistas programadores	Documento de Especificación Funcional	Según lo requerido	Jefe de Proyecto	Jefe de Proyecto	Si	Los AP deben de elaborar los documentos de diseño para realizar la estimación de tiempos de desarrollo
Diseño	Elaborar Documento de especificación técnica	Documento de Especificación Técnico	Uno por Proyecto	Analista Programador	Analista Funcional Arquitecto	Si/No	
Diseño	Definir Arquitectura Física y ambientes de desarrollo, calidad y producción	Documento de Especificación Funcional Documento de Especificación Técnico	Según lo requerido	Arquitecto	Jefe de Proyecto	Si	
Diseño	Coordinar la generación de los ambientes de Desarrollo, Calidad y Producción	Documento de Especificación Funcional Documento de Especificación Técnico	Según lo requerido	Gerente de Proyecto	Jefe de Proyecto	Si	
Diseño	Solicitar estimación de esfuerzo del equipo de desarrollo	Documento de Especificación Funcional Documento de Especificación Técnico	Según lo requerido	AP	Jefe de Proyecto Analista Funcional	Si	
Diseño	Coordinación y asignación de responsables para la generación de data de prueba para los ambientes de Desarrollo y Calidad	Documento de Especificación Funcional Documento de Especificación Técnico	Según lo requerido	AF Líder desarrollo Líder calidad Arquitecto	Jefe de Proyecto Analista Programador	Si	

Diseño	Solicitar recursos testers	Documento de Especificación Funcional Documento de Especificación Técnico	Según lo requerido	Líder de calidad	JdP	Si	Se debe solicitar los recursos testers o analistas pruebas que atenderán el proyecto en QA, antes del terminar el desarrollo el JdP debe de coordinar con los testers una reunión para explicar el detalle de proyecto y los planes de prueba
Diseño	Elaborar cronograma final de proyecto	Cronograma	Uno por proyecto	Analista Funcional	Jefe de Proyecto	Si/No	El Jefe de proyecto validará el cronograma con el Gerente de Proyecto
Diseño	Solicitar aprobación de cronograma del proyecto	Cronograma	Uno por proyecto	Gerente de Proyecto	Jefe de Proyecto	Si	
Diseño	Elaborar Plan de Pruebas Unitarias	Plan de Pruebas Unitarias	Uno por Proyecto	Analista Programador	Analista Funcional	Si/No	
Diseño	Elaborar Plan de Pruebas Integrales	Plan de Pruebas Integrales	Uno por Proyecto	Analista Funcional	Usuario solicitante	Si/No	
Diseño	Elaborar Plan de Pruebas de Usuario	Plan de Pruebas de Usuario	Uno por Proyecto	Analista Funcional	Usuario solicitante	Si/No	
Planificación	Elaborar Plan de Comunicaciones	Plan de comunicaciones	Uno por Proyecto	Analista Funcional	Jefe de Proyecto		
Planificación	Elaborar Plan de Gestión de Riesgos	Matriz de Riesgos	Uno por Proyecto	Analista Funcional	Jefe de Proyecto	Si/No	
Planificación	Registrar la información del proyecto en el PPM Clarity (Documentos de análisis + aprobaciones + documentos de planificación)	PPM Clarity	Según lo requerido	Jefe de Proyecto	Jefe de Proyecto	Si	
Planificación	Registrar las fechas del Proyecto en el Clarity. Crear las tareas en el Clarity, asignado tiempos y recursos.	PPM Clarity	Según lo requerido	Jefe de Proyecto	Jefe de Proyecto	Si	
Planificación	Registrar los aplicativos impactados en el Clarity.	PPM Clarity	Según lo requerido	Jefe de Proyecto	Jefe de Proyecto	Si	
Planificación	Registrar el equipo de trabajo en el Clarity.	PPM Clarity	Según lo requerido	Jefe de Proyecto	Jefe de Proyecto	Si	

Planificación	Crear los paquetes en Clarity	PPM Clarity	Según lo requerido	Jefe de Proyecto	Jefe de Proyecto	Si	
Planificación	Subir los Documentos de Análisis y Aprobaciones al SCM	PPM Clarity	Según lo requerido	Jefe de Proyecto	Jefe de Proyecto	Si	
Planificación	Solicitar al AP promover los paquetes de la fase de análisis a desarrollo	CA SCM	Según lo requerido	Jefe de Proyecto	Jefe de Proyecto Analista Funcional	Si	
Planificación	Subir los documentos de análisis y diseño del AP en los paquetes del SCM	CA SCM	Según lo requerido	Jefe de Proyecto	Jefe de Proyecto Analista Funcional	Si	
Planificación	Solicitar asignación de Analistas de Soporte (SOAP) a cada paquete	CA SCM	Según lo requerido	Jefe de Proyecto	JdP Jefe SOAP	Si	
<b>EJECUCIÓN</b>							
Construcción	Realizar Programación correspondiente		Según lo requerido	Analista Programador	Analista Funcional	Si	
Construcción	Soporte Funcional al Desarrollo		Según lo requerido	Analista Funcional	Jefe de Proyecto Analista Funcional		El AF debe aclarar las dudas y apoyar en la validación del avance de desarrollo para el cumplimiento de las tareas en las fechas programadas.
Construcción	Elaborar Evidencias de Pruebas Unitarias	Evidencia de Pruebas Unitarias	Según lo requerido	Analista Programador	Analista Funcional	Si	
Construcción	Elaborar Evidencias de Pruebas Integrales	Evidencia Pruebas	Según lo requerido	Analista Programador		Si	
Construcción	Realizar Pruebas Funcionales del Desarrollo	Pruebas Funcionales/Integrales	Según lo requerido	Analista Funcional	Jefe de Proyecto Analista Funcional	Si	
Construcción	Realizar Pruebas de Usuario del Desarrollo	Pruebas Funcionales/Integrales	Según lo requerido	Analista Funcional	Jefe de Proyecto Analista Funcional	Si	

Construcción	Ingresar y Actualizar los Entregables del proyecto en el Repositorio (CA SCM)	CA SCM	Según lo requerido	Analista Programador	Jefe de Proyecto Analista Funcional	Si	
Construcción	Elaborar documentación/entregables de desarrollo (Manual de Instalación, Manual Operación, Manual de Usuario, etc)	Entregables de desarrollo	Según lo requerido	Analista Programador	Jefe de Proyecto Analista Funcional	Si	
Transición	Priorización en QA	Formato de priorización QA	Uno por Proyecto	GdP	GdP	Si	Es responsabilidad de JdP coordinar previamente con la gerencia de desarrollo el orden de atención en QA de los paquetes que conforman el desarrollo del proyecto, debiendo de validar si aplica realizar alineación de fuentes según el orden de pases de paquetes
Transición / Pre - Certificación	Validación de documentación (MI, MO)		Según lo requerido	SOAP	AF SOAP AP	Si	
Transición / Pre - Certificación	Pre - Certificación	Promover el PQT de la etapa de Desarrollo a Certificación en CA SCM	Según lo requerido	AP	AF SOAP AP	Si	
Transición / Pre - Certificación	Pre - Certificación	Notificación formal del pase a certificación	Uno por Proyecto	JdP	AF SOAP AP	Si	El JdP debe enviar el correo de notificación de pase, adjuntando el formato de pase a QA, el correo es dirigido a los responsables del área de calidad
Transición / Certificación	Certificación	Instalación y configuración del ambiente de pruebas	Según lo requerido	AC	Líder Calidad		
Transición / Certificación	Certificación	Solicitar controles de arquitectura	Según lo requerido	Arquitecto	JdP	Si	Los AP y AF deberán de dar soporte durante las fases de certificación y producción
Transición / Certificación	Certificación	Realizar el seguimiento del proyecto en QA	Según lo requerido	JdP	JdP		

Transición / Certificación	Certificación	Pruebas en QA (AC, AF, AP, U) Pruebas de cada componente Pruebas de sistema Pruebas integrales	Según lo requerido	AC	JdP AF AP U AC	Si	El área o proveedor de Calidad, podrá hacer uso del plan de comunicaciones y de calidad, para hacer las consultas que se requieran
Transición / Certificación	Certificación	Soporte Funcional / Técnico las pruebas de Calidad	Según lo requerido	AC	AF JdP AP	Si/No	Los AF y AP deben dar soporte a las consultas de los AC durante la fase de certificación
Transición / Certificación	Certificación	Seguimiento y control de la ejecución de las pruebas de Calidad	Según lo requerido	Líder Calidad	Líder Calidad JdP AC	Si	El líder de calidad debe realizar el seguimiento directo con el equipo de QA o AC encargado de la certificación del proyecto revisando que el mismo está ejecutando las pruebas según lo planeado y coordinar las acciones necesarias en caso de retrasos. El líder de calidad debe informar al JdP de los temas que afecten a las pruebas.
Transición / Certificación	Certificación	Gestionar aprobaciones del AF y el Usuario por correo el plan de pruebas enviados por el QA.	Según lo requerido	AC	JdP AF U	Si	El plan de pruebas es aprobado por el AF y el Usuario
Transición / Certificación	Certificación	Validación y aprobación de documentación (MI, MO, MU)	Según lo requerido	AC	JdP SOAP Líder de Calidad Usuario Analista Calidad	Si	El equipo de Calidad o AC sólo deberá enviar los correos a los encargados de las aprobaciones. El seguimiento le corresponde al Jefe de Proyecto. Deben aprobar: -Plan de pruebas (AF y Usuario) -Manual de usuario (Usuario) -Manual de instalación - MI (SOAP) -Manual de operación - MO (SOAP)
Transición / Certificación	Certificación	Validar las aprobaciones en HARVEST	Según lo requerido	AC QA GdP Líder Soap	JdP AC Líder QA Soap GdP	Si	Los paquetes deberán tener las aprobaciones de: -AC -Líder QA -Soap -GdP

Transición / Certificación	Certificación	Validar que todos los pquetes se encuentren en COMITÉ DE PASES A PRODUCCIÓN	Según lo requerido	JdP	JdP	Si	El JdP debe asegurarse que todos los paquetes que componen su pase se encuentren en CPP para que los SOAP los sustenten en el comité de pase a producción del día
Transición / Certificación	Comité de pase a producción	Sustentación del PQT en el Comité de PaP	Según lo requerido	SOAP	JdP	Si	Los SOAPs asignados a los paquetes deberán de exponer la funcionalidad de cada uno y su impacto en los sistemas, en caso de no asistir, el JdP deberá de informarlo al Jefe Soap
Transición / Certificación	Comité de pase a producción	Informa al usuario la fecha programada para el pase a producción.	Según lo requerido	JdP	JdP	Si	
Transición / Producción	Producción	Realizar las pruebas Post Producción y obtener el OK del usuario sobre el pase	Según lo requerido	AF Usuario	JdP	Si	Se realizan una vez ejecutados los pases en el ambiente de Producción
Transición / Producción	Producción	Soporte funcional a pruebas en ambiente de producción	Según lo requerido	SOAP AP AF AC	JdP	Si	
Transición / Producción	Producción	Solicitar y ejecutar control de arquitectura	Según lo requerido	Arquitecto SOAP	JdP	Si	Los controles de arquitectura son actividades que realiza el arquitecto de solución para validar que el ambiente de PRODUCCIÓN cumple los requerimientos no funcionales . El JdP debe considerar ésta actividad en su cronograma y hacer seguimiento de la ejecución como parte de las pruebas post implementación
<b>SEGUIMIENTO Y CONTROL</b>							
Seguimiento y Control	Dirigir y Gestionar la Ejecución del Proyecto	Informe de salud del proyecto	Según lo requerido	Jefe de Proyecto	JdP AF	Si	El responsable de la ejecución del proyecto es el JdP, sin embargo este puede apoyarse en el Analista Funcional y delegarle actividades de control y seguimiento

	Ejecutar seguimiento y control del Proyecto	Actualización de cronograma	Según lo requerido	Jefe de Proyecto	JdP AF	Si	El responsable y ejecutor del seguimiento y control del proyecto es el JdP, sin embargo este puede apoyarse en el Analista Funcional y delegarle actividades de control y seguimiento
	Gestionar y Monitorear los Riesgos del Proyecto	Informe de Riesgos	Según lo requerido	Jefe de Proyecto	JdP AF	Si	El responsable y ejecutor del seguimiento y control del proyecto es el JdP, sin embargo este puede apoyarse en el Analista Funcional y delegarle actividades de control y seguimiento
	Actualizar Repositorio de Entregables del Proyecto (CA Clarity PPM)	CA Clarity PPM	Según lo requerido	Jefe de Proyecto	JdP AF	Si	
	Lecciones aprendidas	Registrar lecciones aprendidas en PPM	Según lo requerido	Jefe de Proyecto	JdP AF	Si	Se identifican escenarios de los cuales se puedan obtener conclusiones que sirvan como referencia e información para futuros proyectos.
<b>CIERRE</b>							
Cierre	Realizar cierre del proyecto	Acta de cierre Proyecto	Uno por Proyecto	Jefe de Proyecto	JdP	Si	
	Registrar Cierre de Proyecto en Clarity	Registrar Cierre de Proyecto en Clarity	Uno por Proyecto	Jefe de Proyecto	JdP	Si	

JdP Jefe de Proyecto  
 AF Analista Funcional  
 AP Analista Programador  
 SOAP Analista de Soporte de Aplicaciones  
 AQ Arquitecto  
 EF Especificación Funcional  
 RF Requerimiento Funcional  
 AC Analista de calidad  
 U Usuario

MI Manual de instalación  
MO Manual de Operación  
MU Manual de Usuario

Tabla 14. Actividades y roles del mecanismo propuesto



### 3.4. Mecanismo ágil de gestión y desarrollo de proyectos de software

El mecanismo ágil de gestión y desarrollo de proyectos de software en la presente investigación se encuentra enmarcada dentro de los cinco grupos de procesos que forman parte de la Guía del PMBOK: Inicio, Planificación, Ejecución, Seguimiento y Control, y Cierre. Los cuales cubren el alcance, los plazos, los recursos humanos, las comunicaciones, los riesgos y la calidad del proyecto, y donde cada uno cuenta con subprocesos o actividades que son necesarias para cumplir con los objetivos del proyecto y generar los entregables necesarios del mismo.

En todas las etapas y fases del mecanismo propuesto se toman en consideración las mejores prácticas de los métodos y/o metodologías para la gestión y desarrollo de software los cuales son: SCRUM, OpenUp, RUP y PMBOK como también se incluyen las buenas practicas observadas por la investigadora del presente trabajo obtenidas de su experiencia laboral.

La Figura 15. Ilustra las etapas y fases del mecanismo propuesto.

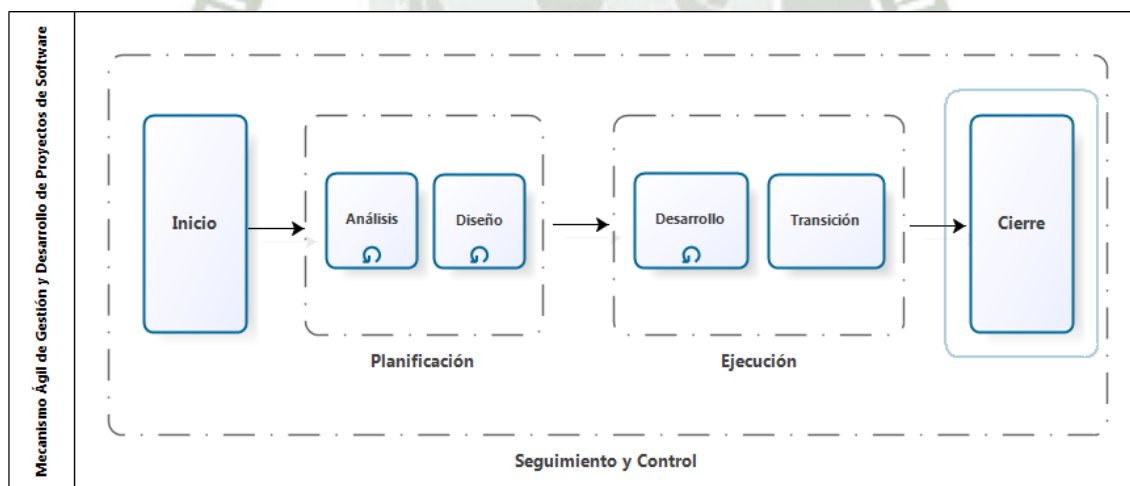


Figura 15. Etapas y fases del mecanismo ágil de gestión y desarrollo de proyectos de software.  
Fuente propia.

#### 3.4.1. Proceso de iniciación

En el proceso de iniciación el jefe de TI de una área de negocio específica (Project Manager), revisa y valida la solicitud de requerimiento de proyecto enviada por el usuario solicitante

(Documento aprobado por la Dirección respectiva). En esta etapa es importante que el Project Manager, valide el costo de implementación del proyecto contra los beneficios que este generará (estratégicos, tácticos y operativos), de ser rentable o beneficioso se procederá a asignar los recursos para el desarrollo del proyecto, en caso no proceda el Project Manager deberá de sustentar porque no es recomendable desarrollar el requerimiento solicitado.

En esta etapa se genera el documento de ficha de inicio del proyecto donde se registran la información del proyecto, alcance, tiempo de duración estimado, recursos (colaboradores, técnicos y proveedores) y las principales amenazas del proyecto.

El formato del documento se encuentra en el anexo A con la siguiente nomenclatura:

FI\_NombreProyecto\_FichaInicio\_CódigoProyecto\_versión\_Fecha.docx

#### **3.4.1.1. Asignar Líder del proyecto**

La iniciativa planteada por una área específica de la institución se registra en un documento denominado “Solicitud de requerimiento” el que debe ser aprobado por el Director del área solicitante, según la cultura organizacional de la institución; la solicitud de requerimiento aprobada se envía al Project Manager (Jefe).

El Project Manager revisa la cartera de proyectos vigente y realiza la asignación del Líder de Proyecto (Analista Funcional), este se encargará de elaborar el documento de “Ficha de inicio” con apoyo del usuario solicitante, el documento será aprobado por el Analista PMO (Validación de documento, deberá detallar claramente el objetivo del proyecto y sus requerimientos, validará que todos los apartados del documento se hallan registrado correctamente) y por el Project Manager.

En esta fase es importante que el Project Manager sea quien valide y agrupe las solicitudes de requerimientos de proyectos para su desarrollo y no los Analistas PMO asignados al Project Manager, debido a que en reiteradas ocasiones, los usuarios solicitantes registran y aprueban por

sus Direcciones solicitudes de requerimientos de proyectos que afectan a un mismo proceso de negocio, solicitudes que duplican el mismo requerimiento pero que han sido generadas por distintos usuarios y en fechas distintas, solicitudes que tienen requerimientos muy similares, solicitudes que dependen de otras debido a que están relacionadas. De esta manera el Project Manager es el responsable de agrupar y seleccionar las solicitudes de requerimientos para poder asignar al Líder de Proyecto (Analista Funcional).

#### **3.4.1.2. Elaborar Ficha de Inicio del Proyecto**

La Ficha de Inicio del Proyecto es realizada por el Líder de Proyecto o Analista Funcional, en ella se plasma el requerimiento del usuario solicitante como también información respecto al alcance, objetivos, estimación de tiempos y recursos necesarios para el planeamiento del proyecto.

El formato del documento se encuentra en el anexo A con la siguiente nomenclatura:

FI\_NombreProyecto\_FichaInicio\_CódigoProyecto\_versión\_Fecha.docx

#### **3.4.1.3. Registrar Proyecto en el PPM**

Se recomienda usar una herramienta de Administración de Portafolio de Proyectos (PPM), que permitirá a los Líderes de Proyectos, mantener toda la información del proyecto en un único lugar, teniendo a disposición información actualizada y de valor.

El registro del proyecto incluirá la publicación del documento de Ficha de Inicio del proyecto y las aprobaciones del Analista PMO y del Project Manager.

Para el presente trabajo de investigación se utilizará la herramienta CA Clarity™ PPM, la que incrementará la facilidad de acceso a la información y la optimización de la administración de proyectos. La herramienta PPM aportará agilidad durante la gestión y desarrollo del proyecto.

La figura 16 muestra los subprocessos del proceso de iniciación.

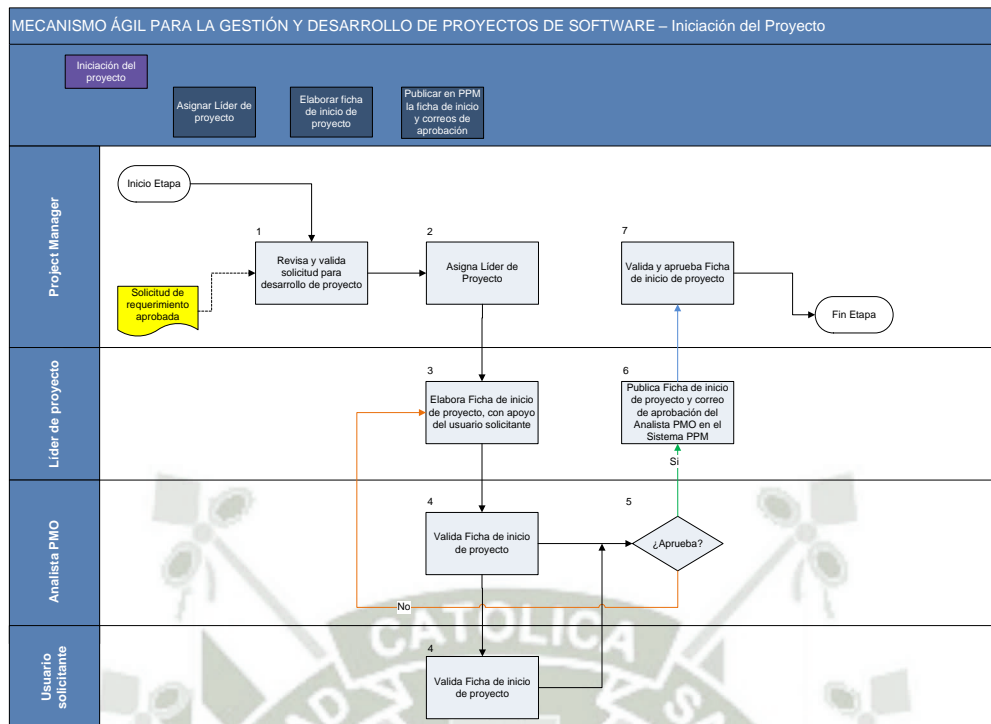


Figura 16. Subprocesos del proceso de iniciación.  
Fuente Propia

En la Figura 16. el proceso de color amarillo representa un documento de entrada.

### 3.4.2. Proceso de planificación

Este proceso corresponde a la planificación de los requerimientos y el alcance del proyecto, además de la estimación de los tiempos y recursos a utilizar.

El grupo de procesos que conforman la planificación del proyecto se encarga de generar el plan de trabajo (Especificación Funcional, Especificación Técnica, Plan de pruebas) así como el cronograma en el cual se detallan las tareas que serán ejecutadas para llevar a cabo el proyecto.

### 3.4.3. Reunión de Kick – Off

En su mayoría, un proyecto pequeño denominado también como requerimiento o tarea no requiere reuniones de Kick – Off, sin embargo para el desarrollo de proyectos medianos o grandes, es recomendable hacer una reunión de Kick – Off con el usuario solicitante y demás stakeholders

del proyecto como también los usuarios líderes de cada área que se verán impactados con la implementación del proyecto.

El Kick – Off no solo servirá para hacer la presentación del Proyecto según la solicitud inicial del usuario sino que también permitirá registrar otros requerimientos funcionales que surjan a partir del requerimiento inicial indicado.

El formato del documento se encuentra en el anexo A con la siguiente nomenclatura:

KO\_NombreProyecto\_Kick Off\_CódigoProyecto\_versión\_Fecha.ppt

#### **3.4.3.1. Acta de Reunión**

Registrar el acta de reunión y enviar vía correo electrónico a los stakeholders como evidencia de la conformidad de los puntos acordados en la reunión de Kick – Off.

El formato del documento se encuentra en el anexo A con la siguiente nomenclatura:

AR\_NombreProyecto\_Acta\_Reunion\_CódigoProyecto\_versión\_Fecha.docx

#### **3.4.3.2. Elaborar Checklist del Proyecto**

Pertenciente a la etapa de Planificación, el Checklist permitirá seleccionar que procesos aplican al Requerimiento o Proyecto en cuestión, brindando un resumen de las tareas a realizar durante las diferentes etapas del ciclo de vida del proyecto, sirviendo de guía para la dirección de proyectos.

El formato del documento se encuentra en el anexo A con la siguiente nomenclatura:

CL\_NombreProyecto\_CheckListProyecto\_CódigoProyecto\_versión\_Fecha.xlsx

### 3.4.3.3. Fase Análisis

En la fase de Análisis, se profundizará en el requerimiento del usuario que genero la idea de proyecto (¿Qué desea el usuario?, ¿Cuáles son los objetivos de la idea de proyecto?, ¿Se tienen restricciones?, ¿Se tendrá impacto en otros sistemas o externamente?). En esta etapa se llevan a cabo reuniones periódicas con el usuario a fin de tener claro su requerimiento y generar los requerimientos funcionales para el desarrollo.

En esta fase el recurso responsable es el Líder de proyecto o Analista Funcional, el que es responsable de recopilar información referente al requerimiento del usuario, generando el documento de especificación funcional el que debe de ser aprobado por el Jefe de área de TI, Arquitectura, Seguridad de la Información y por el usuario solicitante, este documento se enviará luego de ser aprobado al Líder técnico de desarrollo para la elaboración del documento de especificación técnica dando inicio a la fase de diseño.

Se recomienda que los Analistas funcionales no tengan más de cuatro proyectos asignados a su portafolio, ya que debido a las reuniones que mantienen con sus supervisores, líderes técnicos, analistas programadores, analistas de arquitectura, analistas de bases de datos, analistas de calidad, analistas de producción o analistas de soporte de aplicaciones (SOAP), analistas de control de proyectos, PMO, usuarios solicitantes, entre otros involucrados, no podrá gestionar y controlar adecuadamente sus proyectos, provocando demoras en el desarrollo del proyecto, uso deficiente de los recursos asignados y en el peor de los casos, el proyecto deberá de ser puesto en etapa de “stand by” (detenido) o cancelado, para liberar los recursos y asignarlos a otros proyectos.

Los inconvenientes de tener un proyecto en etapa de “Stand by”, es que cuando se desea retomar el proyecto es probable que los recursos que se tenían asignados, se liberen de sus proyectos después de varias semanas o inclusive meses, lo que supone asignar otros recursos nuevos al proyecto, de ser así, el analista funcional y el líder técnico deberán de considerar la curva de

aprendizaje de los nuevos recursos para continuar con el desarrollo, a fin de estimar tiempos de desarrollo correctos y evitar el incumplimiento de fechas de entrega del desarrollo.

#### **3.4.3.3.1. Elaborar Documento de Especificación Funcional**

El Documento de Especificación Funcional incluirá el requerimiento inicial indicado por el usuario y aquellos requerimientos funcionales que surgieron a partir del requerimiento inicial, también incluirá diagramas de procesos e información concerniente al análisis funcional. Este documento deberá de ser firmado o aprobado vía correo electrónico.

El formato del documento se encuentra en el anexo A con la siguiente nomenclatura:

EF\_NombreProyecto\_EspecificaciónFuncional\_CódigoProyecto\_versión\_Fecha.docx

En la Figura 17 se muestra los subprocesos involucrados en la fase de análisis.

#### **3.4.3.3.2. Línea base de alcance**

Una vez aprobado el documento de especificación funcional, esta información será tomada como línea base del alcance. Posterior a la gestión de aprobaciones, el documento de especificación funcional deberá de ser registrado en el PPM para ser consultado por los participantes según sus niveles de permisos cuando lo necesiten.

Una vez establecida la línea base del Alcance del Proyecto y hecha la gestión de aprobaciones, cualquier cambio que se tenga que hacer a la misma, se realizará formalmente a través de la de una solicitud de cambios.

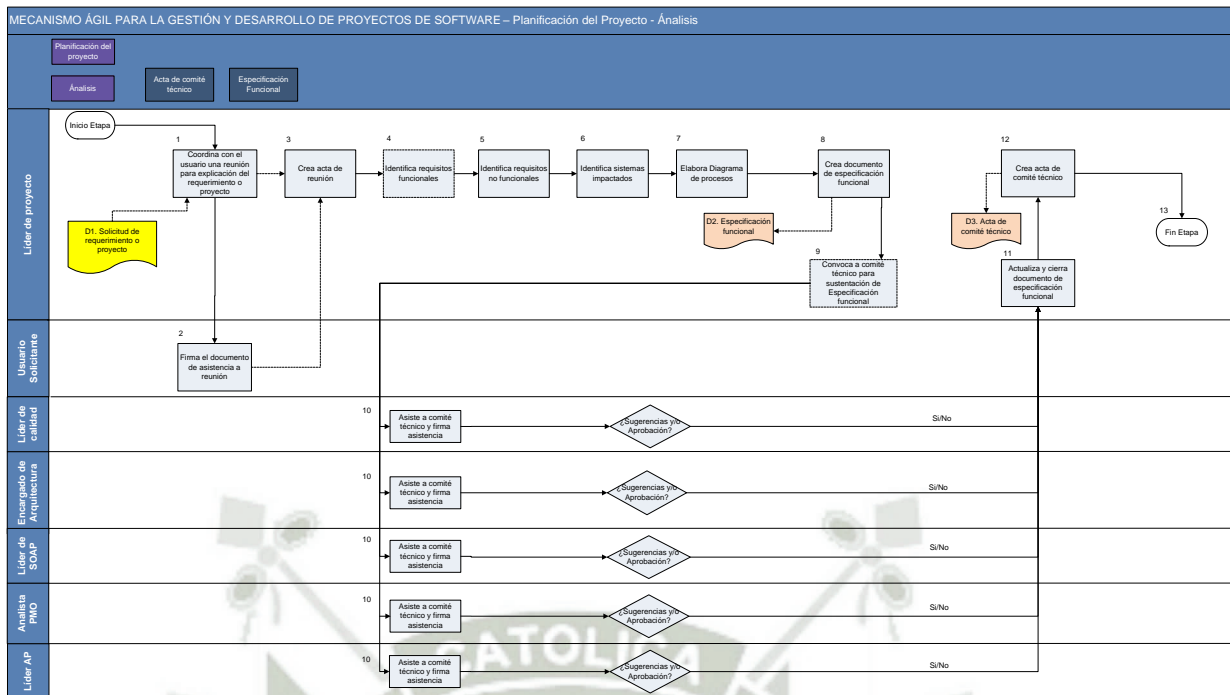


Figura 17. Subprocesos de la fase de Análisis.  
Fuente Propia.

#### 3.4.3.4. Fase de Diseño

En esta fase el documento principal a generar es la especificación técnica del proyecto de acuerdo a la especificación funcional, este documento es elaborado por el Analista Programador que es especialista en el aplicativo impactado, de tratarse de un nuevo aplicativo, el Project manager o Jefe de área de TI deberá de asignar al personal o a la empresa responsable del desarrollo del proyecto quien a su vez asignará al Líder técnico (Analista programador) para elaborar el documento técnico y para el liderazgo del desarrollo, el Líder técnico deberá de evaluar que recursos (Analistas programadores) se asignarán al desarrollo del proyecto, debiendo de tomar en cuenta su carga laboral y su experiencia en el aplicativo impactado.

Tanto el Líder Técnico (Analista programador) como el Líder de proyecto (Analista Funcional) seleccionados por el Project Manager o Jefe de área de TI, deberán tener comunicaciones periódicas para conocer el estado del desarrollo del proyecto. Ambos líderes deberán de apoyarse mutuamente teniendo como objetivo el buen término del proyecto.

Se recomienda que los líderes técnicos, sean asignados a un aplicativo específico, por ejemplo en una empresa con más de diez millones de clientes y veinte aplicativos de core de negocio, tendríamos veinte líderes técnicos cada uno con su respectivo backup, asignados a cada aplicativo core de negocio, esto nos asegura que el líder técnico conozca a profundidad el aplicativo que le corresponde de manera funcional y técnica, pudiendo explicar y delegar actividades al grupo de analistas programadores que tiene asignados, quienes tienen experiencia en el aplicativo y en la herramienta de programación en la que se implementó, de esta manera se logran los siguientes beneficios:

- Se agiliza la fase de diseño y desarrollo.
- Disminuyen las consultas o dudas hacia el Analista Funcional.
- Se evita que el desarrollo sea rechazado en QA por incumplimiento de normas y/o estándares de desarrollo.
- Disminuyen los errores funcionales por desconocimiento del aplicativo y de los procesos de negocio que este maneja.
- Se tiene un mayor control de los recursos (Analistas programadores) que se tienen asignados por aplicativo, pudiendo asignarlos a otros proyectos en cuanto terminen las actividades que se les dio.
- Se mejora la gestión de los proyectos en fases de diseño y desarrollo, ya que los líderes de proyecto y los líderes técnicos se reúnen periódicamente para revisar el estado de los proyectos, dar solución a problemas y mitigar posibles riesgos; a su vez el líder del proyecto informa sobre el estado del proyecto a su coordinador o supervisor.

#### **3.4.3.4.1. Elaborar Documento de especificación técnica**

El documento de especificación técnica registrará la propuesta técnica al proyecto solicitado, incluyendo el diagrama de flujo de la solución, requerimientos funcionales, requerimientos no funcionales, modelo de datos, especificación de componentes y arquitectura del sistema.

El formato del documento se encuentra en el anexo A con la siguiente nomenclatura:

ED\_NombreProyecto\_EspecificaciónTécnica\_CódigoProyecto\_versión\_Fecha.docx

Los subprocesos involucrados en la fase de diseño son se muestran en la figura 18.

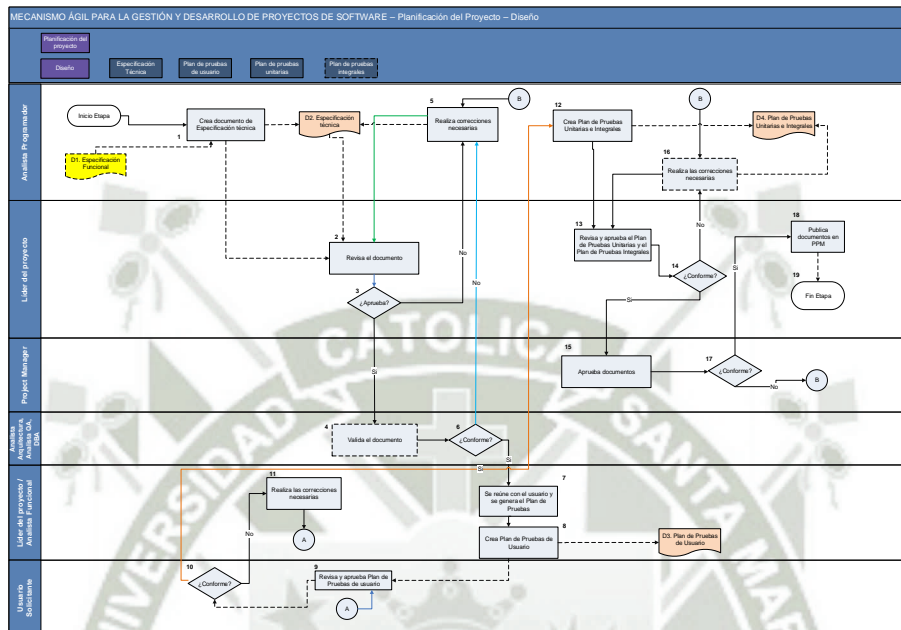


Figura 18. Subprocesos en la fase de Diseño.  
Fuente propia.

Documento de entrada en color amarillo y de documento de salida en color naranja.

### 3.4.3.4.2. Elaborar Plan de Pruebas Unitarias

El documento de Plan de Pruebas Unitarias registra el detalle de pruebas que aseguren el correcto funcionamiento de los componentes de código de manera aislada, incluye los datos del proyecto, participantes, ambiente de pruebas, listado de casos de pruebas funcionales, detalle de los casos de pruebas funcionales, detalle de los casos de pruebas técnicas, discrepancias y acuerdos y especificaciones del usuario.

#### **3.4.3.4.3. Elaborar Plan de Pruebas Integrales**

El documento de Plan de Pruebas Integrales registra el detalle de pruebas que aseguren el correcto funcionamiento de los componentes de código de manera conjunta, incluye los datos del proyecto, participantes, ambiente de pruebas, detalle de los casos de pruebas funcionales, detalle de los casos de pruebas técnicas, discrepancias y acuerdos y especificaciones del usuario.

El formato del documento se encuentra en el anexo A con la siguiente nomenclatura:

PI\_NombreProyecto\_PruebasIntegrales\_CódigoProyecto\_versión\_Fecha.docx

#### **3.4.3.4.4. Elaborar Plan de Pruebas de Usuario**

El documento de plan de pruebas de usuario registra el detalle de pruebas realizadas por el usuario líder para su confirmación y aceptación del proyecto desarrollado, registra el detalle de los casos de pruebas especificados por el usuario.

#### **3.4.3.5. Elaborar Plan de Gestión de Comunicaciones**

Perteneciente al proceso de Planificación, el plan de comunicaciones nos permitirá conocer a los participantes del proyecto, el medio de comunicación que se tendrá con cada uno, la frecuencia y la audiencia.

El formato del documento se encuentra en el anexo A con la siguiente nomenclatura:

PC\_NombreProyecto\_PlanComunicaciones\_CódigoProyecto\_versión\_Fecha.xlsx

#### **3.4.3.6. Elaborar Plan de gestión de riesgos**

Perteneciente al proceso de Planificación, el plan de gestión de riesgos se plasmara en el documento de matriz de riesgos, la que permitirá definir los riesgos, estimar su probabilidad de ocurrencia e impacto, como también el plan de mitigación, plan de contingencia, responsable y la fecha de registro del riesgo.

El formato del documento se encuentra en el anexo A con la siguiente nomenclatura:

MR\_NombreProyecto\_MatrizRiesgos\_CódigoProyecto\_versión\_Fecha.xls

### 3.4.3.7. Plan de gestión de cambios

El plan de gestión de cambios tiene por objetivo la evaluación y planificación del proceso de cambio para asegurar que, si este se lleva a cabo, se haga de la forma más eficiente, siguiendo los procedimientos establecidos y asegurando en todo momento la calidad.

El formato del documento se encuentra en el anexo A con la siguiente nomenclatura:

GC\_NombreProyecto\_GestiónCambios\_CódigoProyecto\_versión\_Fecha.docx

#### 3.4.3.7.1. Nivel de Impacto

Todo cambio en los componentes de TI, impacta en los recursos y servicios de la Institución es por ello que es importante identificar el nivel de impacto que tiene en relación a ellos. Los criterios a considerar para la evaluación de impacto son:

Categoría de Impacto	Criterio	
Alto	Hardware afectados	Deberá de ser especificado por las áreas correspondientes de la Institución
	Software afectados	Deberá de ser especificado por las áreas correspondientes de la Institución
	Servicios afectados	Deberá de ser especificado por las áreas correspondientes
Medio	Hardware afectados	Deberá de ser especificado por las áreas correspondientes de la Institución
	Software afectados	Deberá de ser especificado por las áreas correspondientes de la Institución

	Servicios afectados	Deberá de ser especificado por las áreas correspondientes de la Institución
Bajo	Hardware afectados	Cambios que no impactarán en las funciones o trabajo de los usuarios
	Software afectados	Cambios que no impactarán en las funciones o trabajo de los usuarios
	Servicios afectados	Cambios que no impactarán en las funciones o trabajo de los usuarios

### 3.4.3.7.2. Roles de la gestión y aprobación de cambios

Se deberá de especificar los roles que se necesitan para aprobar los cambios tecnológicos, como también el tipo de cambio, niveles de aprobación y canal de aprobación de cada persona asignada.

El cuadro 19, muestra el esquema para hacer el registro de los roles para la Gestión y Aprobación de cambios, donde el tipo de cambio puede ser:

- Normal: requerimiento de cambio planificado.
- Estándar: requerimiento de cambio pre-aprobado por la Gerencia de Sistemas o quien corresponda.
- Urgente: requerimiento de cambio para corregir un problema o error que impide la realización de las funciones de los usuarios y/o brindar los servicios de la institución de manera correcta.

Nombre del rol	Persona asignada	Tipo de Cambio	Nivel de Impacto	Niveles de aprobación	Canal de aprobación

Cuadro 19. Roles para la Gestión y Aprobación de Cambios.  
Fuente propia.

#### **3.4.3.7.3. Plan de contingencia ante solicitudes de cambio urgentes**

Es recomendable describir el plan de contingencia para atender las solicitudes de cambio urgentes que no pueden esperar a que se coordine con el comité de control de cambios. Por ejemplo ante una urgencia el canal de aprobación más directo sería vía telefónica con posterior regulación vía correo electrónico, este y demás mecanismos de acción deberán de ser registrados en el plan de contingencia ante solicitudes de cambios urgentes.

#### **3.4.3.7.4. Documento de solicitud de cambio**

El documento de solicitud de cambio es necesario para indicar el cambio requerido, el problema o situación actual, razón por la que se solicita el cambio, efectos en el proyecto, áreas y sistemas.

El formato del documento se encuentra en el anexo A con la siguiente nomenclatura:

SC\_NombreProyecto\_SolicitudCambio\_CódigoProyecto\_nroSRTI\_versión\_Fecha.docx

#### **3.4.3.8. Planeamiento de proyecto**

En este proceso se realizan las actividades necesarias para promover el proyecto de planificación al proceso de ejecución, entre ellas tenemos: comité técnico, gestión de aprobaciones, priorización de los desarrollos y la estimación de tiempos, elaboración del cronograma o calendario de actividades y elaboración de la matriz de riesgos. Los documentos de entrada son el documento de especificación funcional y el documento de especificación técnica.

El Flujo de subprocesos para el planeamiento del proyecto se muestra en la Figura 20.

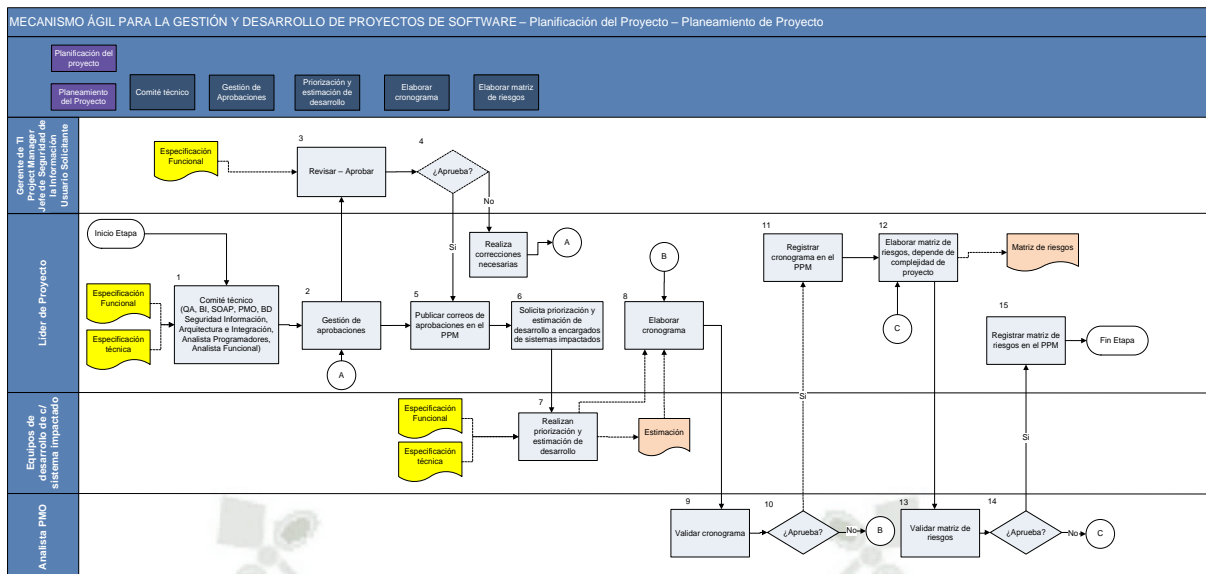


Figura 20. Flujo de subprocesos para la planificación del proyecto.  
Fuente propia.

Documento de entrada en color amarillo y de documento de salida en color naranja.

### 3.4.3.8.1. Comité técnico

El comité técnico es una reunión donde participan los encargados de los sistemas impactados y las áreas de responsables de que el proyecto cumpla con las normativas establecidas por la empresa.

Es recomendable que los participantes del comité técnico lean y revisen el documento de especificación técnica, de manera que aporten mejoras y correcciones, es por ello que se debe dar un plazo de 48 horas (2 días hábiles) para programar la reunión de comité técnico. Esta es la última reunión técnica en la que la propuesta de solución (especificación funcional y técnica) podrán ser modificadas sin generar una solicitud de cambio, luego de la gestión de aprobaciones cualquier adecuación deberá de ser gestionada con una solicitud de cambios.

Dependiendo de cultura organizacional de cada empresa, los participantes al comité técnico podrán ser:

- Jefe y/o Analista de BI (Inteligencia de negocios)

- Jefe y/o Analista SOAP (Soporte de Aplicaciones)
- Jefe o Analista de Seguridad de información
- Jefe o Analista de Arquitectura e Integración
- Líder técnico de cada sistema impactado
- Analistas programadores de cada sistema impactado
- Analista PMO
- Analista Funcional (Líder de proyecto)

#### 3.4.3.8.2. Gestión de aprobaciones

La gestión de aprobaciones es necesaria para promover el proyecto de la etapa de planificación a ejecución. Según la cultura de la institución se solicitarían las siguientes aprobaciones para el inicio del Desarrollo o ejecución del proyecto:

- Gerente de TI
- Seguridad de Información
- Usuario solicitante

Los documentos a aprobar son los siguientes:

- Documento de especificación funcional (Puede incluir anexo de arquitectura conceptual).

Internamente para el inicio del desarrollo del proyecto el Arquitecto de Solución solicitará la aprobación del documento de Arquitectura lógica (También se le denomina Documento de Especificación técnica) al Líder de Arquitectura e Integración, el cual será usado por los Analistas Programadores (AP) para elaborar su estimación de tiempos, diseñar los objetos de software y elaborar su documentación.

#### **3.4.3.8.3. Priorización y estimación de desarrollo**

Para empezar con la solicitud de priorización y estimación de desarrollo, es necesario contar primeramente con la gestión de aprobaciones, luego se enviará un correo a cada líder de los equipos de desarrollo involucrados en el proyecto para que prioricen el requerimiento o proyecto y generen las estimaciones de tiempos de desarrollo.

#### **3.4.3.8.4. Registrar documentos en el sistema PPM**

Todos los documentos aprobados y generados en el proceso de planeamiento se deben de ingresar al Sistema de administración de portafolio de proyectos (PPM), de igual manera, los correos de aprobación de los jefes, se deben de publicar en la herramienta, de esta manera se asegura la disposición de la información útil y actualizada.

#### **3.4.3.8.5. Elaborar matriz de riesgos**

La matriz de riesgos permite definir los riesgos, estimar su probabilidad de ocurrencia e impacto, como también el plan de mitigación, plan de contingencia, responsable y la fecha de registro del riesgo. Si la complejidad del proyecto es baja, no se recomienda elaborarla, sin embargo si el proyecto está definido con complejidad media o alta, se deberá de elaborar el documento de matriz de riesgos.

El formato del documento se encuentra en el anexo A con la siguiente nomenclatura:

MR\_NombreProyecto\_MatrizRiesgos\_CódigoProyecto\_versión\_Fecha.xls

#### **3.4.3.8.6. Línea Base del Tiempo**

La Línea Base del Tiempo se establecerá utilizando como herramienta el CA Clarity TM PPM, en el cual se ingresará el calendario del proyecto (cronograma) y se considerará la lista completa de todas las actividades que se desarrollarán.

La duración de las actividades serán estimadas en base a juicio de expertos y de proyectos anteriores.

Una vez establecida la línea base de tiempo, cualquier cambio que se tenga que hacer a la misma, se realizará formalmente a través de una solicitud de cambios.

La línea base del tiempo o cronograma se elaborará con las estimaciones de tiempos enviadas por cada líder de equipo de desarrollo de los sistemas impactados por el proyecto.

#### **3.4.4. Proceso de ejecución**

En el proceso de ejecución se lleva a cabo la fase de desarrollo del proyecto, el Líder de Proyecto deberá de tomar especial atención y seguimiento a esta fase, debiendo de solucionar a la brevedad cualquier inconveniente o dependencia que surja; en caso se le requiera información o apoyo para explicar los requerimientos funcionales deberá de estar presto y disponer a los Analistas programadores la información que requieran para evitar demoras y se re-planifique la fecha de entrega del desarrollo.

##### **3.4.4.1. Fase de Desarrollo**

El desarrollo o programación estará a cargo del Analista Programador este a su vez deberá respetar las normas y estándares de programación de la Empresa.

La figura 21 nos muestra los subprocesos involucrados en la etapa de desarrollo.

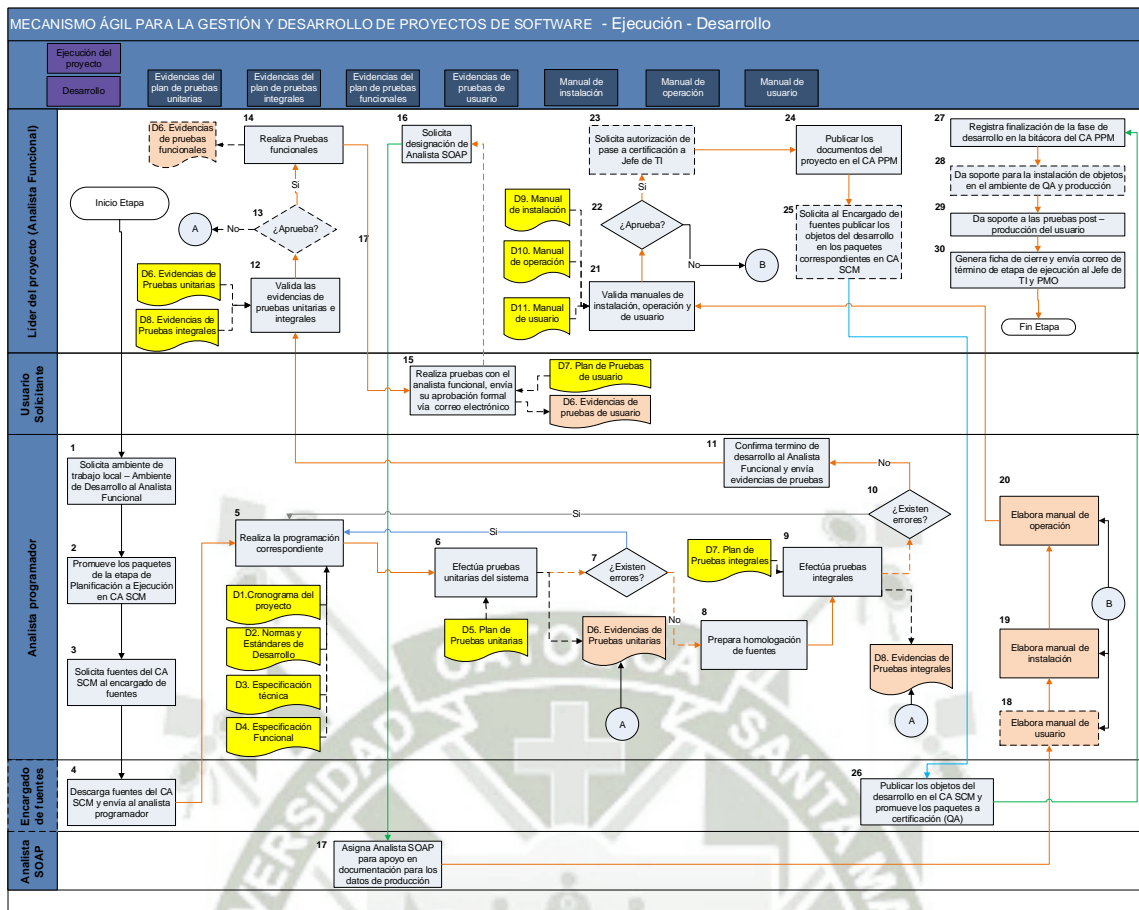


Figura 21. Subprocesos en la fase de Desarrollo.  
Fuente propia.

Documento de entrada en color amarillo y de documento de salida en color naranja.

### 3.4.4.1.1. Caso de estudio

#### 3.4.4.1.1.1. Explicación de proyecto

Se solicita implementar un proyecto cuya finalidad es realizar consultas de saldo del Cliente.

#### 3.4.4.1.1.2. Incidencias en fase de desarrollo

- Cambios en la Especificación Funcional.
- Cambios en la Especificación Técnica.
- Falta de comunicación con el Usuario técnico.
- Demora en la solicitud de data de prueba.

### 3.4.4.1.1.3. Recomendaciones y/o lecciones aprendidas

1. Es importante que todos los encargados del desarrollo de un proyecto (Analistas funcionales y analistas programadores), se reúnan, para verificar el alcance de la especificación técnica y su detalle, ya que el desarrollo involucrada tres plataformas tecnológicas. La reunión debe de ser coordinada por el Líder de proyecto antes de realizar el comité técnico. En esta reunión de verificación se podrán adecuar los requerimientos técnicos o agregar nuevos objetos a desarrollar, dependiendo de las necesidades de datos de cada aplicativo impactado.
2. Hacer seguimiento y control para agilizar el desarrollo, solucionar prestamente inconvenientes o dudas, a fin de entregar el desarrollo dentro de las fechas del cronograma, ya que en algunas situaciones si el desarrollo no es continuo y no cumple con las fechas de entrega especificadas en el cronograma, en caso surjan proyectos con mayor prioridad, se pondrá el proyecto en estado de “stand by” (se detiene proyecto y se liberan recursos) con el fin de reasignar los recursos a otros proyectos prioritarios, como sucedió en este caso de estudio.
3. Generar actas de reunión con los puntos acordados y reenviar a todos los participantes, de esta manera, el Analista Funcional, podrá sustentar que se han convocado reuniones técnicas y que en ninguna de ellas se solicitó adecuaciones o nuevos objetos a desarrollar. Se debe de minimizar las adecuaciones técnicas posteriores al comité técnico y a la gestión de aprobaciones. La figura 23 ilustra una acta de reunión de comité técnico:

AGENDA		
ACTIVIDAD	RESPONSABLE	TIEMPO
Revisión de Especificación Funcional.	-----	00:45 min.
DOCUMENTACIÓN		
QUÉ SE DEBE PRESENTAR EN LA REUNIÓN	RESPONSABLE	
Especificación Funcional y anexo de Arquitectura conceptual	Analista Funcional y Arquitecto de Solución	

CONCLUSIONES			
1	Se generará Solicitud de cambios (SC) .		
2			
ACTION ITEM	RESPONSABLE	FECHA LÍMITE	OBSERVACIONES
Elaborar y firmar SC N°001	Usuario	16/12	-----
Elaborar Acta de reunión	Analista Funcional	12/12	-----
Actualizar Especificación funcional	Analista Funcional	13/12	-----
Actualizar Arquitectura Lógica	Arquitecto de aplicación	13/12	-----

Figura 23. Acta de reunión.  
Fuente propia.

4. Es recomendable que durante la etapa de planificación (Diseño), se utilice el plan de pruebas unitarias, integrales y de usuario según corresponda, para detallar los requerimientos de data de prueba (generación de líneas, actualización de data de desarrollo con data de producción, generación de ambiente de pruebas propio, actualización de datos de bolsas, etc.) que necesitarán los equipos de desarrollo (analistas programadores), el analista funcional (pruebas integrales y funcionales) y los requerimientos de data y ambientes que requiere el usuario para hacer sus pruebas.
  - a. Esto permitirá que el requerimiento de data y ambientes de pruebas solicitados a los encargados correspondientes se vayan gestionando mientras se ejecuta el desarrollo del proyecto.
  - b. Para dar mayor importancia a los documentos elaborados se recomienda enviarlos vía correo electrónico a los usuarios o personal que hará las pruebas respectivas como también al analista de control de proyectos asignado, se recomienda enviar el correo cuando el proyecto sea promovido de la etapa de planificación a ejecución, en caso se tengan nuevos requerimientos de data de prueba, estos podrán ser agregados y gestionados con las áreas correspondientes mientras el proyecto se encuentra en la etapa de ejecución y en fase de desarrollo.

#### **3.4.4.1.2. Elaborar Evidencias de Pruebas Unitarias**

El Analista Programador deberá de evidenciar los casos o escenarios de pruebas definidos por el conjuntamente con el Analista Funcional.

El objetivo de las Pruebas Unitarias es:

- Se prueba el correcto funcionamiento del código generado de manera aislada.
- Los escenarios o casos de pruebas deben de tener independencia, debiendo dividir en pequeños módulos el código generado para tener pruebas más específicas y seguras.
- De haber problemas o errores en la ejecución del código generado, estos podrán ser solucionados de manera inmediata por el Analista Programador ya que en esta etapa es más sencillo identificar y dar solución a los errores que se presenten; tener en cuenta que solo se podrán identificar errores en cuanto al código generado mas no se tendrán errores de integración ya que se están probando módulos independientes de código.

#### **3.4.4.1.3. Elaborar Evidencias de Pruebas Integrales o Funcionales**

El Analista Programador deberá de evidenciar los casos o escenarios de pruebas definidos por el Analista Funcional.

El objetivo de las Pruebas Integrales es:

- Probar el correcto funcionamiento del código generado al integrarlo al sistema, servicio y/o componente.
- Probar la funcionalidad del Sistema en conjunto con sus elementos, interfaces y demás procesos.

#### **3.4.4.1.4. Publicar los documentos del proyecto en el PPM**

Para evitar que la información del proyecto que se desarrolla se encuentre en distintas fuentes como por ejemplo, correos electrónicos, PCs, file server, sistemas o aplicaciones de

versionamiento, impresiones, etc; es recomendable centralizar en una sola fuente todos los documentos relacionados al proyecto, una de las herramientas de software usada para ello es el CA Clarity™ PPM (Administración de portafolio de proyectos), por lo que se recomienda su uso dependiendo de las necesidades de cada empresa.

Los aportes de la herramienta PPM son:

- Mejorar el acceso a documentos,
- Evitar confusiones con el versionamiento de documentos,
- Optimizar de la gestión del proyecto,
- Involucrar a los actores del proyecto y permitirles acceso a los recursos publicados en el PPM según sea rol,
- Centralizar en una sola fuente todos los documentos relacionados al proyecto,
- Visualizar y hacer seguimiento al cronograma del proyecto.

#### **3.4.4.1.5. Publicar los objetos del desarrollo en el SCM**

El Software Change Management (SCM) permite la administración de las fuentes de la organización permitiendo la descarga, actualización, versionamiento como también la diferenciación de las fuentes que se encuentran en Desarrollo, Certificación, Producción y en Cierre.

Los documentos Funcionales del proyecto y las aprobaciones del mismo serán registradas por el Analista Funcional y los objetos del desarrollo o programación por el Analista Programador ubicándolos en la etapa del ciclo de vida del proyecto que corresponda (Análisis, Desarrollo, Certificación, Producción)..

Los aportes de la herramienta SCM son:

- Fortalecer la integridad de las fuentes,

- Mejorar el acceso a documentos técnicos, aprobaciones, fuentes y ejecutables de los proyectos
- Evitar confusiones con el versionamiento de documentos, fuentes y ejecutables,
- Mejorar la visibilidad del ciclo de vida del producto en desarrollo, certificación y producción,
- Facilitar la gestión de cambios en las aplicaciones,
- Optimizar de la gestión del proyecto.

#### 3.4.4.1.6. Elaborar Manual de usuario

El manual de usuario es documento sencillo sin tecnicismos, orientado a explicar el funcionamiento del desarrollo desde el sistema, servicio o herramienta usada por el usuario:

El manual contiene lo siguiente:

- Introducción
  - Objetivos del sistema
  - Funciones del sistema
- Acceso al sistema y descripción de la pantalla
  - Ingreso al sistema
  - Referencias de uso de la simbología
  - Descripción de las pantallas
- Funcionalidad del sistema
  - Descripción Funcional (Mapa del sistema)
  - Descripción detallada de las opciones del menú
- Glosario de términos y abreviaturas
- Anexos
- El formato del documento se encuentra en el anexo A con la siguiente nomenclatura:

MU\_NombreProyecto\_ManualUsuario\_CódigoProyecto\_versión\_fecha.docx

### **3.4.4.2. Transición**

La etapa de transición incluye la certificación, puesta en producción y estabilización del proyecto, siendo gestionadas por los jefes de las áreas de Certificación - QA y Producción respectivamente, es por ello, que no se profundizará su estudio en el presente trabajo de investigación.

#### **3.4.4.2.1. Caso de estudio**

##### **3.4.4.2.1.1. Explicación de proyecto**

Se solicita implementar un proyecto de automatización de procesos cuyo objetivo sea el de agilizar, optimizar y mejorar la calidad de atención y respuesta de una licitación, así como también la disponibilidad de información para la toma de decisiones de la gerencia.

##### **3.4.4.2.1.2. Incidencias en fase de Certificación (QA)**

Durante la etapa de certificación, el área de seguridad de información indica que se debe de cambiar los protocolos usados para el envío de correos desde el aplicativo.

Ante este inconveniente, se propone la creación de un desarrollo adicional siendo de quince días hábiles la estimación de tiempo de desarrollo y pruebas de la funcionalidad.

El usuario solicitante es informado de lo ocurrido y se acuerda re - planificar la fecha de pase a producción debido a que el término de la fase de certificación depende del desarrollo de España.

Otro retraso a considerar fue el retraso en la instalación de los ambientes de QA.

##### **3.4.4.2.1.3. Recomendaciones y/o lecciones aprendidas**

- Esta incidencia se pudo evitar, si los encargados de las áreas de seguridad de información y soporte de aplicaciones, hubieran tenido conocimiento de las consideraciones técnicas del proyecto indicadas en la especificación técnica, cuando asistieron a la reunión de comité técnico.

- Las reuniones de comité técnico deben de ser convocadas con anticipación adjuntando en el correo electrónico la especificación funcional y técnica, en este periodo de tiempo (24H o 48 H según cada empresa), los analistas designados por los jefes de área deberán de leer las especificaciones, de manera especial la especificación técnica, para que asistan a la reunión de comité con conocimiento del proyecto que sustentará el Analista Funcional o Líder del Proyecto, de esta manera podrán hacer aportes significativos y se evitarán las solicitudes de cambio en fase de desarrollo.
- Es importante hacer seguimiento de los requerimientos que se hacen a otras áreas de Tecnologías de información (habilitar ambientes de certificación y producción, generación de data de prueba, etc.) a fin de disponer de los recursos en el tiempo adecuado y no fuera de cronograma.

#### **3.4.4.2.2. Certificación del proyecto**

La certificación del proyecto es una actividad realizada por el área de certificación y calidad, siendo los analistas de calidad (QA) (según la cultura organizacional de cada institución) los encargados de la validación del desarrollo y de las pruebas del mismo; el Analista Funcional y/o el Jefe de Proyecto (Project manager) no tienen mayor participación sino la de hacer seguimiento y absolver dudas o consultas que surjan, es por ello que no se detallarán los a profundidad los procesos y documentos referentes a esta fase.

### **3.4.4.2.3. Producción y estabilización del proyecto**

La producción del proyecto es una actividad realizada por el área de producción o soporte de aplicaciones (según la cultura organizacional de cada institución), siendo los analistas de producción o los SOAP (analistas de soporte de aplicaciones) los encargados de la puesta en marcha del proyecto; tanto el líder del proyecto o Project manager como el líder de los analistas programadores deberán de apoyar durante este proceso en caso surja algún inconveniente o para absolver consultas, es por ello que no se detallarán los procesos y documentos referentes a esta etapa.

### **3.4.5. Seguimiento y Control**

El seguimiento y control es el proceso que se lleva a cabo durante todo el ciclo de vida del proyecto, es decir desde la iniciación hasta el cierre.

#### **3.4.5.1. Dirigir y Gestionar la Ejecución del Proyecto**

El líder del proyecto o analista funcional deberá de dirigir y gestionar la ejecución del proyecto, deberá de reunirse semanalmente con el líder de desarrollo del proyecto y con el líder de analistas programadores (AP) y consultar por el status del desarrollo, de haber consultas deberá de absolverlas, el status del desarrollo será enviado por correo electrónico por el líder de desarrollo con copia al PMO, Jefe de desarrollo, Analistas programadores y Líder del proyecto.

Posteriormente según la fecha acordada en el cronograma el líder de proyecto deberá de elaborar el informe de salud del proyecto, el que será presentado al Jefe de TI (según área de negocio), Jefe de desarrollo, Líder de desarrollo del proyecto y PMO.

El formato del documento de informe de salud del proyecto se encuentra en el anexo A con la siguiente nomenclatura:

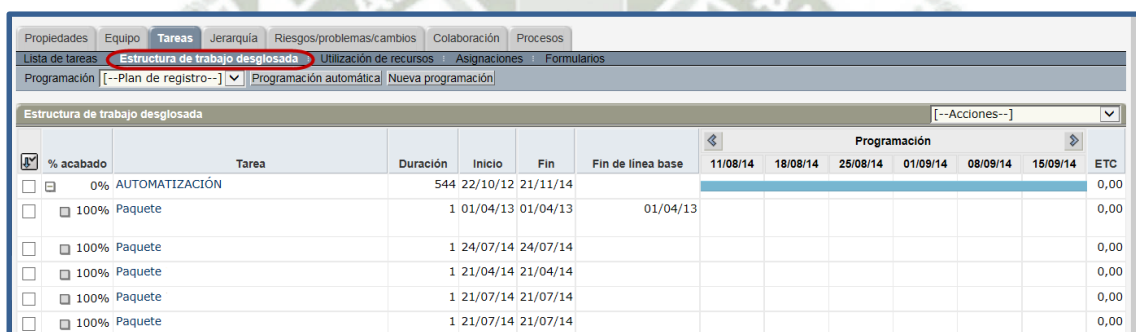
IS\_NombreProyecto\_InformeSaludProyecto\_CódigoProyecto\_versión\_Fecha.pptx

### 3.4.5.2. Ejecutar seguimiento y control del Proyecto

El líder del proyecto deberá de hacer seguimiento al cumplimiento de las fechas establecidas en el cronograma registrado en el PPM, controlando el término de actividades y la entrega de documentos en las fechas acordadas en el cronograma.

El recurso usado para la ejecución del seguimiento y control del proyecto es el cronograma registrado en la herramienta de administración de proyectos (CA Clarity PPM) desde donde será controlado por los interesados del proyecto y por los Analistas de control de proyectos.

La figura 24 ilustra la herramienta Clarity PPM y la opción para la visualización y actualización del cronograma del proyecto.



%	Tarea	Duración	Inicio	Fin	Fin de línea base	Programación						ETC
						11/08/14	18/08/14	25/08/14	01/09/14	08/09/14	15/09/14	
0%	AUTOMATIZACIÓN	544	22/10/12	21/11/14								0,00
100%	Paquete		01/04/13	01/04/13	01/04/13							0,00
100%	Paquete		1 24/07/14	24/07/14								0,00
100%	Paquete		1 21/04/14	21/04/14								0,00
100%	Paquete		1 21/07/14	21/07/14								0,00
100%	Paquete		1 21/07/14	21/07/14								0,00

Figura 24. Cronograma del proyecto (Estructura de trabajo desglosada).  
Fuente propia.

### 3.4.5.3. Gestionar y Monitorear los Riesgos del Proyecto

Para la gestión y monitoreo de riesgos del proyecto se tomará como base inicial la matriz de riesgos usada durante la planeación del proyecto, sin embargo durante la ejecución del proyecto y dependiente de la complejidad del proyecto y la necesidad, adicionalmente se elaborará un informe de riesgos que será presentado por el Líder del proyecto al PMO, Jefe de TI, Jefe de desarrollo, Líder de desarrollo del proyecto y usuario líder solicitante de ser necesario.

El informe contiene lo siguiente:

- Riesgos actuales potenciales
  - Revisión de triggers para los riesgos identificados inicialmente
  - Revisión de adecuación de respuestas planificadas para los riesgos identificados inicialmente
  - Revisión de planes de contingencia para los riesgos identificados inicialmente
- Riesgos actuales sucedidos
  - Valoración de impacto real vs impacto estimado
  - Revisión de planes de contingencia
  - Evaluación de necesidades de acciones correctivas o solicitudes de cambio
- Nuevos riesgos detectados
  - Definición de triggers de riesgos
  - Evaluación y categorización de riesgos
  - Definición de respuestas planificadas
  - Definición de planes de contingencia
  - Programación de ejecución de respuestas planificadas
- Aceptación del documento

El formato del documento de informe de riesgos del proyecto se encuentra en el anexo A con la siguiente nomenclatura:

IR\_NombreProyecto\_InformeRiesgos\_CódigoProyecto\_versión\_Fecha.docx

Los riesgos expuestos a los líderes y jefes involucrados en el proyecto se ingresarán a la herramienta PPM, para hacerles seguimiento, la figura 25 ilustra la sección en la herramienta PPM para registrar los riesgos.

ID	Nombre del Riesgo	Responsable del Riesgo	Estatus
RISG-1165	Tiempo de Desarrollo retrasado	Inga	Abierta

Figura 25. Riesgos del proyecto.  
Fuente propia.

### 3.4.5.4. Ingresar y Actualizar Entregables del Proyecto

Los entregables del proyecto se almacenarán por medio del CA Clarity PPM, esta actividad será hecha por el Líder del proyecto, esto permitirá a los interesados del proyecto disponer de la información de manera oportuna y actualizada.

La figura 26 ilustra la herramienta PPM y la opción para el registro y actualización de la documentación del proyecto.

Nombre	Tamaño	Tipo	Estado	Modificado	Acciones
2. Planificación				26/11/13 11:28 AM	[-- Acciones --]
Comité Técnico				13/11/13 03:43 PM	[-- Acciones --]
1. FR_Locución 136.pdf	485 KB	Adobe Acrobat		12/11/13 02:48 PM	[-- Acciones --]
2. PS_Locución 136_V1.docx	188 KB	Documento		12/11/13 02:48 PM	[-- Acciones --]
3. Aprobacion Jefe TI.msg	222 KB	Documento		12/11/13 02:48 PM	[-- Acciones --]
4. Aprobacion Gerente TI.msg	241 KB	Documento		13/11/13 03:43 PM	[-- Acciones --]
5. Aprobacion Usuario.msg	251 KB	Documento		26/11/13 11:27 AM	[-- Acciones --]
6. Aprobacion Seguridad.msg	252 KB	Documento		26/11/13 11:28 AM	[-- Acciones --]

Figura 26. Documentación del proyecto.  
Fuente propia.

### 3.4.5.5. Ingresar y actualizar bitácora del proyecto

El ingreso y actualización de la bitácora del proyecto es una actividad importante que nos permite conocer el status real del proyecto, debe registrarlo y actualizarlo el Analista Funcional o Líder del proyecto de manera directa en la herramienta CA Clarity PPM

### 3.4.5.6. Atención de solicitud de cambios

El líder del proyecto deberá de atender las solicitudes de cambios; para iniciar la gestión de cambios, el líder del proyecto deberá de recepcionar el requerimiento de solicitud de cambio de parte del usuario solicitante con firma del Jefe y Director del área, luego elaborará el documento de solicitud de cambios que deberá de ser aprobada por el Jefe de TI, finalmente se enviará la solicitud de cambio a los jefes y líderes de desarrollo para su adecuación.

El formato del documento se encuentra en el anexo A con la siguiente nomenclatura:

SC\_NombreProyecto\_SolicitudCambio\_CódigoProyecto\_nroSRTI\_versión\_Fecha.docx

Las solicitudes de cambio se ingresarán a la herramienta PPM, para hacerles seguimiento y tener de conocimiento las modificaciones que ha sufrido el proyecto durante la etapa de ejecución.

La figura 28 ilustra la sección en la herramienta PPM para registrar las solicitudes de cambio.

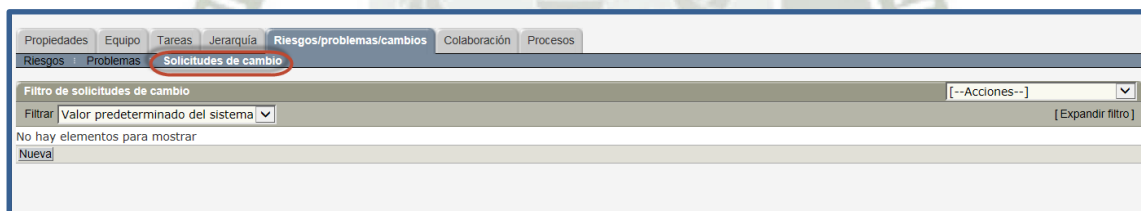


Figura 28. Solicitud de cambio del proyecto.  
Fuente propia.

### 3.4.5.7. Registrar lecciones aprendidas

Las lecciones aprendidas son el conocimiento adquirido durante la gestión y desarrollo del proyecto, debido a una buena o mala práctica realizada en el mismo. Es conveniente documentarlas y poner a disposición de la institución, para evitar cometer los mismos errores o para aplicar aquellas prácticas que dieron buenos resultados en otros proyectos a futuro.

### 3.4.6. Cierre

#### 3.4.6.1. Realizar cierre del proyecto

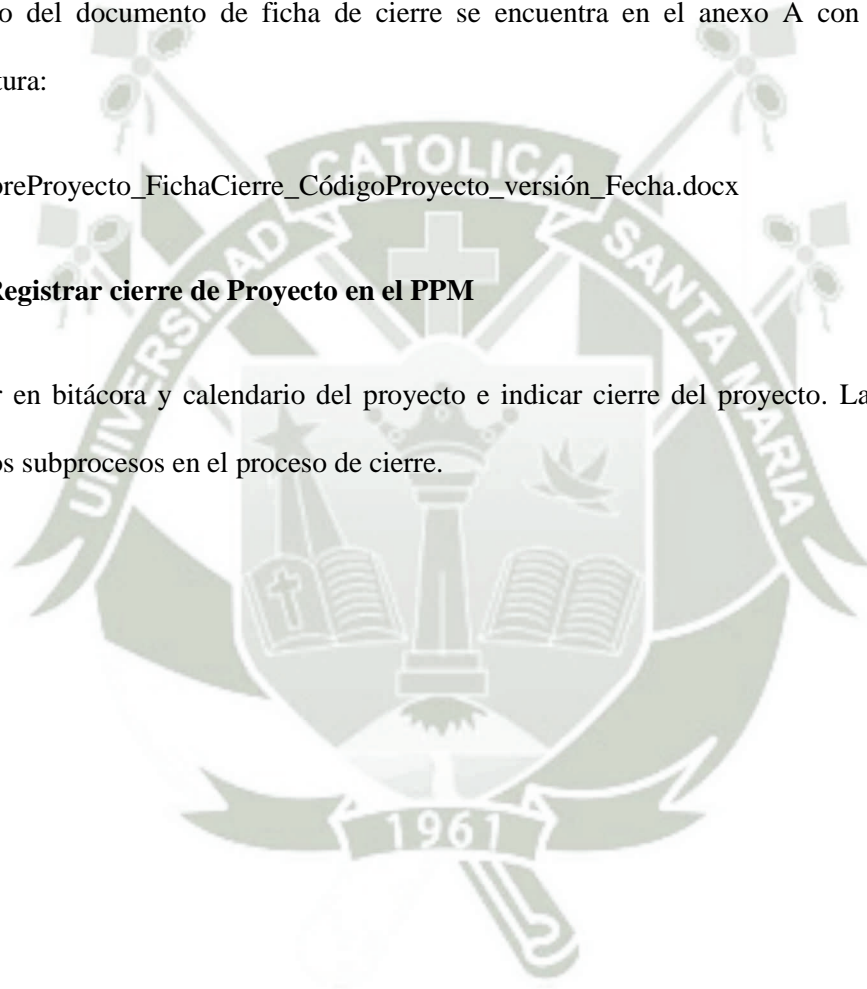
El cierre del proyecto indica la finalización de la ejecución del proyecto y la conformidad del resultado de parte del usuario solicitante, por lo que, debe de contar con la aprobación del Jefe de TI y el usuario solicitante, esta aprobación será vía correo electrónico, donde se adjuntará el documento de ficha de cierre.

El formato del documento de ficha de cierre se encuentra en el anexo A con la siguiente nomenclatura:

FC\_NombreProyecto\_FichaCierre\_CódigoProyecto\_versión\_Fecha.docx

#### 3.4.6.2. Registrar cierre de Proyecto en el PPM

Actualizar en bitácora y calendario del proyecto e indicar cierre del proyecto. La figura 3.16 muestra los subprocesos en el proceso de cierre.



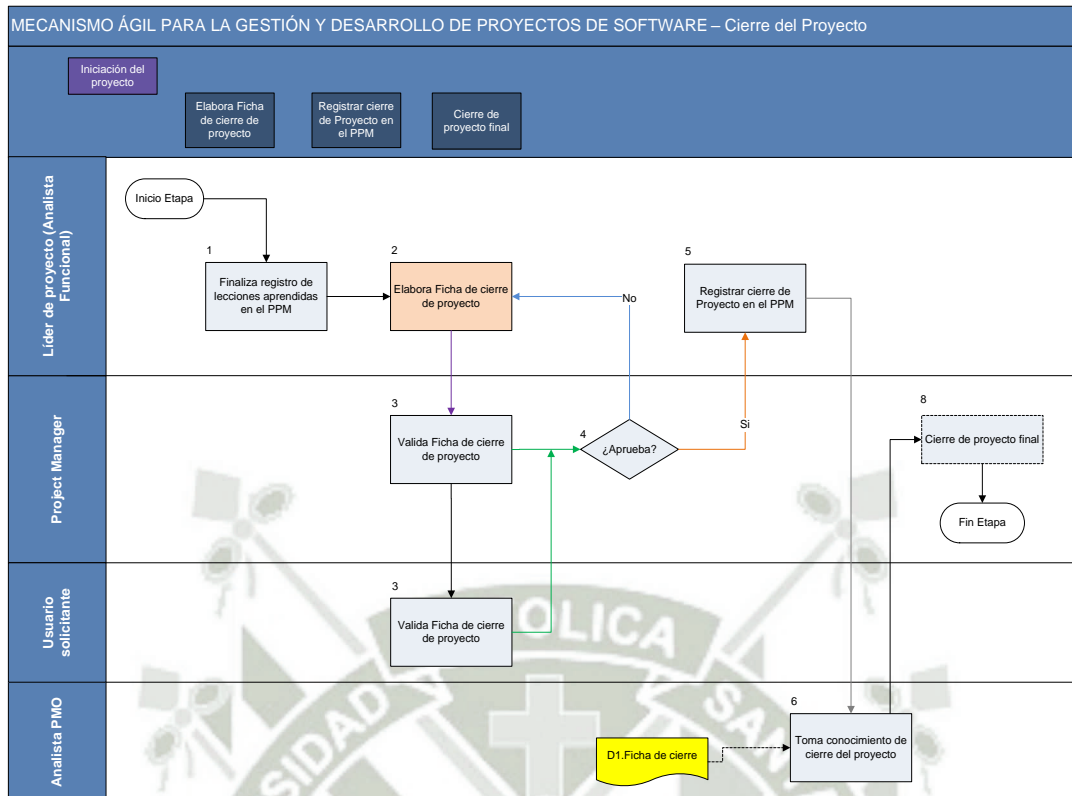


Figura 31. Subprocesos de cierre.  
Fuente Propia.

Documento de entrada en color amarillo y de documento de salida en color naranja.

## CAPITULO 4: APLICACIÓN DE MECANISMO

### 4.1. Proyecto de Automatización

#### 4.1.1. Introducción

El proyecto en el que se explicarán los procesos y actividades más relevantes del mecanismo propuesto está referido al desarrollo de un tablero de control que permita gestionar las tareas y actividades.

El objetivo del proyecto es agilizar, optimizar y mejorar la calidad de atención y respuesta de una licitación, así como también la disponibilidad de información para la toma de decisiones de la gerencia.

#### 4.1.2. Proceso de iniciación

##### 4.1.2.1. Asignar Líder del proyecto

Se buscan los recursos necesarios para el proyecto y se hace la adjudicación de recursos al proyecto.

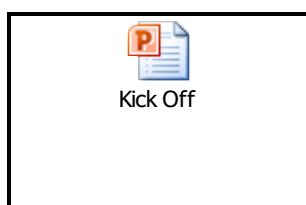
##### 4.1.2.2. Registrar Proyecto en el CA Clarity PPM

Registrar proyecto en el Sistema de Administración de Portafolio (CA Clarity PPM).

#### 4.1.3. Proceso de planificación

##### 4.1.3.1. Reunión de Kick – Off

La presentación del proyecto se encuentra en el siguiente archivo:

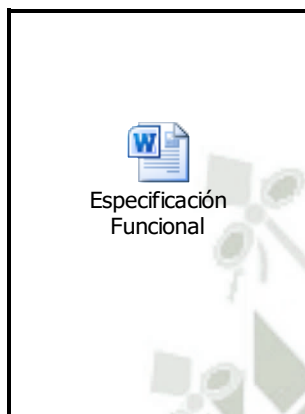


Ver también en el Anexo B.

#### **4.1.3.2. Etapa Análisis**

##### **4.1.3.2.1. Elaborar Documento de Especificación Funcional**

El documento de análisis funcional se encuentra en el siguiente archivo:

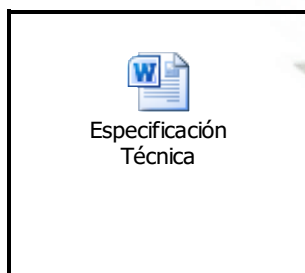


Ver también en el Anexo B el documento.

#### **4.1.3.3. Etapa de Diseño**

##### **4.1.3.3.1. Elaborar Documento de especificación técnica**

El documento de especificación técnica se encuentra en el siguiente archivo:



Ver también en el Anexo B el documento.

#### **4.1.3.4. Elaborar Plan de gestión de riesgos**

Para la gestión de riesgos utilizaremos como documento base, la matriz de riesgos.

El documento de la Matriz de riesgos se encuentra en el siguiente archivo:



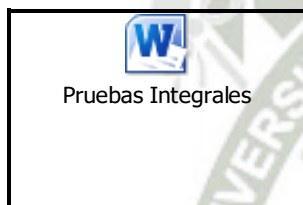
Ver también en el Anexo B el documento.

#### 4.1.4. Proceso de ejecución

##### 4.1.4.1. Etapa de Desarrollo

##### 4.1.4.1.1. Elaborar Evidencias de Pruebas Integrales

El documento de pruebas integrales se encuentra en el siguiente archivo:



Ver también en el Anexo B el documento.

##### 4.1.4.1.2. Publicar los documentos del proyecto en el PPM

Se deberá de subir la Propuesta de solución al CA Clarity PPM que incluye la especificación funcional y arquitectura, como también las aprobaciones de ambos documentos para dar inicio a la etapa de desarrollo. La siguiente Figura (38) ilustra la información del Proyecto en la herramienta de Administración de Proyectos CA Clarity.

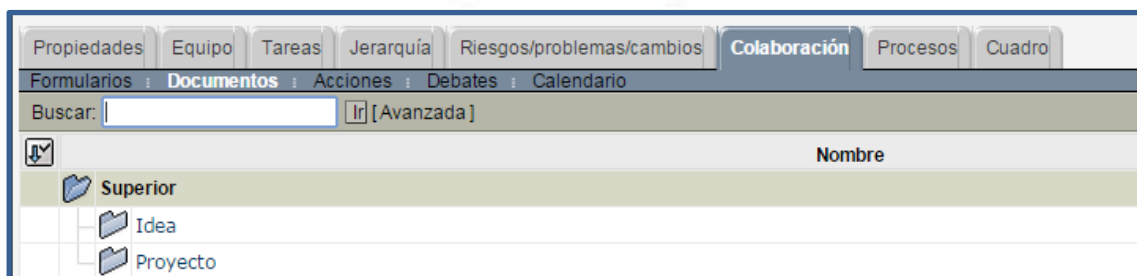


Figura 38. Carpetas para los documentos de la Idea y Proyecto.  
Fuente propia.

#### 4.1.4.1.3. Publicar los objetos del desarrollo en el SCM

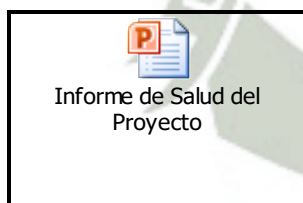
El encargado de las fuentes deberá de subir los objetos a los paquetes generados por el Analista Funcional desde el CA Clarity, a su vez se encargará de promover los paquetes a las siguientes etapas cuando el Analista Funcional o Líder del proyecto lo autorice y se cumplan las validaciones de cada etapa.

El seguimiento y control es el proceso que se lleva a cabo durante todo el ciclo de vida del proyecto, es decir desde la iniciación hasta el cierre.

#### 4.1.4.2. Dirigir y Gestionar la Ejecución del Proyecto

Para la dirección y gestión de la ejecución del proyecto se utilizará el Informe de Salud del Proyecto, el que nos permitirá conocer el estado actual del proyecto, se recomienda que la frecuencia de presentación de dicho documento sea semanal.

El documento de Informe de Salud del Proyecto se encuentra en el siguiente archivo:



Ver también en el Anexo B el documento.

Dependiendo de la solicitud del usuario y de la sencillez de la información a exponer, se puede elaborar un informe de salud del proyecto más gráfico y práctico, este tipo de formato es solicitado mayormente por los gerentes de área quienes mayormente no disponen de tiempo.

A continuación se ilustra un informe de salud o informe de estado de proyecto usado por una empresa transnacional de tecnologías de información para exponer el estado de la fase de desarrollo del proyecto al Gerente).

El informe consta dos estructuras:

- Planificación y Estado del proyecto.
- Puntos pendientes.

#### **4.1.4.3. Ejecutar seguimiento y control del Proyecto**

En recurso que nos ayudará a controlar las actividades planificadas para el Proyecto es el cronograma o Gantt.

#### **4.1.4.4. Gestionar y Monitorear los Riesgos del Proyecto**

La gestión y monitoreo de riesgos podrá ser registrado y expuesto usando los documentos “Informe de salud del proyecto” e “Informe de riesgos”, esta selección dependerá del usuario principal del proyecto a quien se le expone el estado y riesgos de su proyecto.

Los riesgos detectados podrán ser registrados en la herramienta CA Clarity PPM, para su visualización y recordatorio de los interesados del proyecto y de los analistas de control de proyectos; las siguientes figuras muestran las secciones en el CA Clarity para registrar los riesgos, problemas y solicitudes asociados al proyecto.

#### **4.1.4.5. Ingresar y actualizar bitácora del proyecto**

Es recomendable registrar la bitácora dos veces por semana, de manera que el Project Manager y los analistas de control de proyectos conocen el estado actual del proyecto, se deberá de registrar la etapa, fase, actividades pendientes y terminadas y riesgos relevantes.

## CAPITULO 5: EVALUACIÓN DE MECANISMO

En este capítulo se sustenta mediante la evaluación de expertos la viabilidad de implementar el mecanismo propuesto en la cultura organizacional de una Institución para la gestión y desarrollo de sus proyectos de software.

### 5.1. Método de evaluación

El método de evaluación usado es:

1. Juicio de expertos.

Se consultó con expertos en gestión y desarrollo de proyectos de software y se les aplicó una encuesta, obteniendo respuestas positivas para su uso e implementación.

La encuesta es la siguiente:

#### ENCUESTA

- ¿Cree usted que el mecanismo planteado incrementa la productividad de los recursos durante la gestión y desarrollo de proyectos de software?
- ¿Cree usted que el mecanismo planteado mejora la calidad del proyecto y su producto?
- ¿Cree usted que el mecanismo planteado permite la flexibilidad para la gestión de cambios durante el proyecto?
- ¿Según su experiencia, cuáles fueron los aportes que trajo el uso de la herramienta de Administración de Portafolio de Proyectos, el que también es recomendado en el mecanismo (CA Clarity PPM)?

- ¿Según su experiencia, cuáles fueron los aportes que trajo el uso de la herramienta de Administración de Cambios de Software, el que también es recomendado en el mecanismo (CA SCM)?

## 5.2. Resultados de evaluación

Según el Juicio de expertos se confirma lo siguiente:

1. El uso del mecanismo planteado incrementa la productividad
  - Explicación:
    - Los subprocesos pertenecientes al proceso de seguimiento y control enmarcados en el mecanismo planteado permitieron:
      - Gestionar eficientemente el proyecto, reconociendo riesgos asociados al proyecto a tiempo y de manera proactiva, resolviendo dudas o conflictos dentro de los equipos de trabajo, por medio de las reuniones con los equipos de trabajos y los informes de salud del proyecto que se presentan al jefe inmediato o a los interesados del proyecto.
      - Controlar la asignación y uso de recursos por medio de las ventanas de la herramienta CA PPM (Adjudicaciones de rol/recurso, Buscador de recursos/Habilidades de recursos, Lista de recursos, Tareas/Estructura desglosada de trabajo, Tareas/Utilización de recursos, Bitácora de proyecto, Fechas del proyecto, Plan de comunicaciones).
      - Cumplimiento de hitos y actividades,
      - Monitoreo de riesgos,
      - Mejorar el Control de recursos,

- Verificación de actualización de repositorios (CA PPM y CA SCM)
  - La herramienta CA PPM facilitó la gestión, planificación y control del portafolio de proyectos, disponiendo de los documentos y objetos del proyecto y su producto actualizados, también facilitó la visualización del ciclo de vida del proyecto y su producto, conociendo el estado del proyecto en el momento que se precise.
  - La herramienta CA SCM, facilitó el acceso a fuentes de sistemas informáticos, para el trabajo de los equipos de desarrollo de software.
2. Mejora la calidad del proyecto y su producto
- Explicación:
    - Los lineamientos del proceso de seguimiento y control mejoraron la calidad del proyecto.
    - La estandarización de los documentos (plantillas) generados para el Proyecto y su producto, permitió contar con información ordenada, clara, útil y no repetitiva en otros documentos.
    - La herramienta de CA PPM permitió el acceso de usuarios según sus permisos y roles para la visualización y actualización de información.
    - La herramienta CA PPM se usó como repositorio estándar de los documentos y objetos del Proyecto facilitando el uso y gestión de los mismos.
    - La herramienta CA SCM se usó como repositorio estándar de los documentos y objetos del Producto facilitando el uso y gestión de los mismos.
    - Se estandarizo el documento de Acta de reunión la cual es registrada y distribuida luego de una reunión de comité técnico.

- Para comprobar la aprobación de los documentos del Proyecto y del Producto se subieron a los repositorios CA PPM y CA SCM los correos electrónicos de aprobación de los documentos funcionales, técnicos, solicitud de requerimiento, solicitud de cambios, actas de reunión de comité entre otros.
  - La herramienta CA SCM fortaleció la integridad de las fuentes de código ya que se permite hacer reserva, actualización, lectura, sincronización de la fuente, teniendo un manejo de fuentes más transparente.
  - El subproceso de Construcción verificó el cumplimiento de los estándares de programación establecidos por la institución.
  - Las actividades de “Análisis y Ejecución de Pruebas de Certificación y QA” y “Aprobación de Pase a Producción” pertenecientes al subproceso de Transición aseguraron el cumplimiento de las normas y estándares de calidad y programación establecidos por la institución para el Proyecto y su Producto.
3. Permite la flexibilidad para la gestión de cambios durante el proyecto, sin mucho esfuerzo
- Explicación:
    - Las solicitudes de cambios son transmitidas vía correo electrónico y vía la herramienta CA PPM para el conocimiento de todos los interesados y visualización de los documentos referentes al cambio y su aprobación; de esta manera se agilizo la solicitud de requerimiento, su gestión y desarrollo.
    - La herramienta CA SCM facilito la obtención y actualización de fuentes.
    - Los informes de salud del proyecto se generan rápidamente porque se cuenta la información (documentos funcionales y documentos de diseño detallado aprobados, actas de reunión, cronogramas actualizados, solicitudes de cambio, correos electrónicos de aprobación o de puesta en “stand by” del proyecto) relacionada al proyecto en un único lugar y

actualizada (Lunes, Miércoles y Viernes), todo ello se encuentra en el repositorio del CA PPM.

- Se agilizó el pase de control a certificación, ya que se tiene el apoyo de la herramienta CA SCM para promover las fuentes al ambiente de Certificación y posteriormente a Producción.

4. ¿Cuáles son los aportes del uso de la herramienta de Administración de Portafolio de Proyectos (CA PPM)?

○ Explicación:

- Facilitó la disposición de información y la gestión del proyecto.
- Optimizó el control del ciclo de vida de desarrollo del proyecto.
- Permitted contar con información actualizada del proyecto.
- Permitted visualizar a los encargados o líderes de cada proyecto y verificar su carga laboral.
- Permitted centralizar la documentación del proyecto.
- Permitted involucrar a los interesados según sus roles en los distintos procesos y subprocesos de la gestión y desarrollo de Proyectos.

5. ¿Cuáles son los aportes del uso de la herramienta de Administración de Cambios de Software (CA SCM)?

○ Explicación:

- Optimizó el proceso de seguimiento y control del ciclo de vida del producto.
- Permitted contar con información actualizada y útil sobre el producto.
- Fortaleció la integridad de fuentes.
- Mejoró el acceso y control de los documentos, fuentes y ejecutables del producto.
- Mejoró la gestión de versiones de fuentes de código y de paquetes.

### 5.3. Información de Expertos

1. (PMP. Melvin Chavez, Gerente de Proyectos, Voicesat – Proyecto Interseguro, Enero 11, 2016).

*Project Manager en Hewlett Packard Enterprise; MBA Executive (España), Ingeniero Industrial, experto en Gestión, Dirección soporte comercial y pre-venta.*

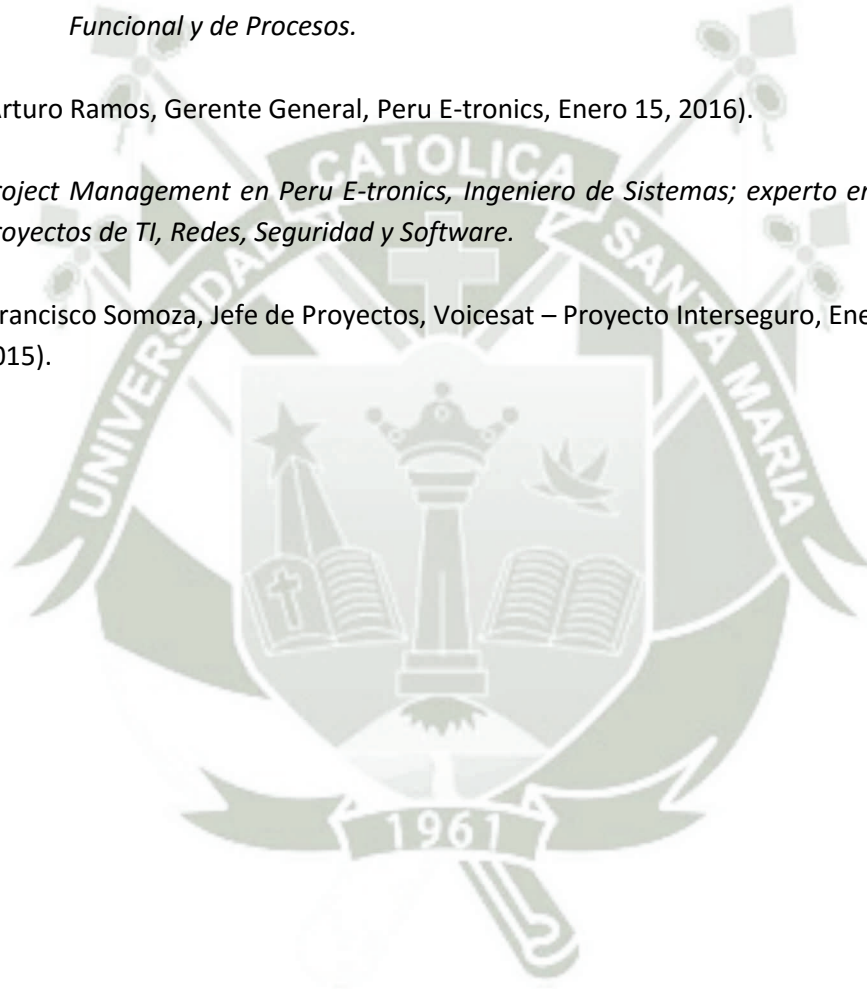
2. (PMP. Lucy Camacho, Jefe de Proyectos, Cosapi, Enero 13, 2016).

*Jefe de Proyectos en Cosapi; Magister en Gestión de Tecnologías de la Información, Ingeniero de Sistemas; experto en Gestión de Proyectos, Análisis Funcional y de Procesos.*

3. (Arturo Ramos, Gerente General, Peru E-tronics, Enero 15, 2016).

*Project Management en Peru E-tronics, Ingeniero de Sistemas; experto en Gestión de Proyectos de TI, Redes, Seguridad y Software.*

4. (Francisco Somoza, Jefe de Proyectos, Voicesat – Proyecto Interseguro, Enero 11, 2015).



## CONCLUSIONES

1. En resumen el mecanismo propuesto junto con sus cinco Etapas y cinco Fases permitió tener mapear y/o registrar las actividades que permitirán optimizar la gestión del desarrollo de proyectos de software disminuyendo los riesgos.
2. Se propuso un flujo de actividades a seguir considerando también Hitos y entregables importantes a considerar durante el ciclo de vida del proyecto.
3. Se estandarizó el uso de documentación, utilizando secciones sugeridas tanto por PMBOK como por SCRUM uniéndolas en documentos únicos quitando secciones repetitivas y dejando las de valor para la gestión y desarrollo de software.
4. Se estandarizó los entregables de la Etapa de Control y seguimiento de Proyectos, etapa importante que se lleva a cabo desde el inicio del Proyecto hasta el final, sugiriendo documentos y plantillas para presentaciones de informes entre el equipo de desarrollo y ante la Gerencia.
5. La estandarización de documentos, actividades y roles de cada fase e involucrado permitió agilizar la gestión de cambios otorgándole flexibilidad viéndose repotenciado con el uso de las herramientas PPM y SCM.

## BENEFICIOS

1. Incremento de la productividad de los recursos durante la gestión y desarrollo de proyectos de software.
2. Mejora de la calidad del proyecto y su producto.
3. Flexibilidad para la gestión de cambios durante el proyecto.
4. La herramienta CA PPM aporta facilidad de acceso a documentos y optimización de la gestión de proyectos.
5. La herramienta CA SCM aporta facilidad de acceso a fuentes y optimización de la gestión de versiones de fuentes y objetos del producto o desarrollo.

## RECOMENDACIONES

Se tienen las siguientes recomendaciones:

1. Se recomienda dividir el requerimiento inicial del usuario en RF medibles, de manera que cuando se deban de hacer pruebas estas se basen en los requerimientos funcionales, los cuales implementan porciones pequeñas del requerimiento del usuario, facilitando la fase de pruebas por parte del AP, AF, SOAP, QA, según corresponda.
2. Es recomendable dividir el desarrollo del proyecto por equipos o áreas, como por ejemplo, Bases de datos, Aplicativos o Servicios.
3. Se recomienda que la propuesta de solución cuente con la revisión y Feedback de Arquitectos de Aplicaciones, Seguridad de la Información, Soporte de Aplicaciones, Equipo de Configuraciones, de esta manera se asegura que todas las áreas conozcan el impacto del Proyecto y den aportes de mejora.
4. Se recomienda el uso de Documentos de Arquitectura Lógica, este será un documento de soporte técnico a la Propuesta de Solución Funcional y será el input para la elaboración del Documento de Especificación Técnica por parte de los Analistas Programadores.
5. Se recomienda el uso de cronogramas de trabajo para el seguimiento y control del proyecto y de sus recursos.
6. Se recomienda el uso de Actas de reunión para formalizar los acuerdos de los actores de cada proyecto de manera que se cumplan las responsabilidades asignadas.
7. Se recomienda el uso de herramientas de PPM y SCM para obtener la información del proyecto y su producto de manera ágil y simple.
8. Se recomienda la planificación de reuniones semanales de equipo para la revisión de status de proyectos.
9. Se recomienda la planificación de escenarios de prueba y DATA a usar desde la Etapa de Planificación, para solicitar la generación o preparación de data de prueba, de manera que no se tengan contratiempos y retrasos.

10. Se recomienda centralizar la información Arquitectónica de cada aplicativo (manual o automatizada) de manera que en cada desarrollo se conozca que recursos se pueden reutilizar, que datos se encuentran en sesión.
11. Contar con la Arquitectura del Aplicativo permite reaccionar ágilmente en caso se requiera identificar procesos críticos para optimizarlos.



## REFERENCIAS

- <sup>1</sup> PMI. (2013). *Project Management Body Of Knowledge (PMBOK GUIDE) (Quinta Edición)*.
- <sup>2</sup> Juan Palacio, Claudia Ruata (2011). Manifiesto ágil. *Scrum Manager Gestión de Proyectos*, 20 - 21. España: Safe Creative.
- <sup>3</sup> Juan Palacio, Claudia Ruata (2011). Gestión de Proyectos ágil". *Scrum Manager Gestión de Proyectos*, 41-43. España: Safe Creative.
- <sup>4</sup> Charles G. COBB. (2011). Agile History and Overview. *Making Sense of Agile Project Management: Balancing Control and Agility*, 40-43. New Jersey, Estados Unidos: Jhon Wiley & Sons.
- <sup>5</sup> Mike Cohn. (2005). An Agile Approach. *Agile Estimating and Planning*, 21-25. Massachusetts, Estados Unidos: Pearson Education.
- <sup>6</sup> Juan Palacio, Claudia Ruata (2011). Elementos de Scrum. *Scrum Manager Gestión de Proyectos*, 60, 61, 74, 75, 76,85. España: Safe Creative.
- <sup>7</sup> S.Chandramouli, Saikat Dutt. (2012). Monitoring and Controlling (Iterative cycle review,daily stand-up). *PMI Agile Certified Practitioner*. India: Dorling Kindersley.
- <sup>8</sup> S.Chandramouli, Saikat Dutt. (2012). What is adaptive Planning. *PMI Agile Certified Practitioner*. India: Dorling Kindersley.
- <sup>9</sup> S.Chandramouli, Saikat Dutt. (2012). Iteration and Realease Planning. *PMI Agile Certified Practitioner*. India: Dorling Kindersley.
- <sup>10</sup> S.Chandramouli, Saikat Dutt. (2012). Daily Planning. *PMI Agile Certified Practitioner*. India: Dorling Kindersley.
- <sup>11</sup> S.Chandramouli, Saikat Dutt. (2012).Agile Project Tracking. *PMI Agile Certified Practitioner*. India: Dorling Kindersley.
- <sup>12</sup> S.Chandramouli, Saikat Dutt. (2012). Product Quality. *PMI Agile Certified Practitioner*. India: Dorling Kindersley.

<sup>13</sup> S.Chandramouli, Saikat Dutt. (2012). Communications. *PMI Agile Certified Practitioner*. India: Dorling Kindersley.

<sup>14</sup> S.Chandramouli, Saikat Dutt. (2012). Agile, Metrics and Estimations. *PMI Agile Certified Practitioner*. India: Dorling Kindersley.

<sup>15</sup> S.Chandramouli, Saikat Dutt. (2012). Agile Project Risk Management. *PMI Agile Certified Practitioner*. India: Dorling Kindersley.

<sup>16</sup> Jacobson Sten. (2002). The Rational Objectory Process - A UML - based Software Engineering Process.



## ANEXO A

### FICHA DE INICIO

Nombre de documento:

FI\_NombreProyecto\_FichaInicio\_CódigoProyecto\_versión\_Fecha.docx

Detalle de documento:

CONTROL DE VERSIONES				
Versión	Elaborado por	Aprobado por	Fecha	Motivo

### FICHA DE INICIO DEL PROYECTO

1. INFORMACIÓN GENERAL	
NOMBRE DEL PROYECTO	CÓDIGO DEL PROYECTO
DESCRIPCIÓN DEL PROYECTO: <i>QUÉ, QUIÉN, CÓMO, CUÁNDO Y DÓNDE?</i>	
CASO DE NEGOCIO: <i>¿QUÉ RAZONAMIENTO SEGUIMOS PARA INICIAR EL PROYECTO Y QUE ESPERAMOS LOGRAR COMO BENEFICIOS CUANTITATIVOS O CUALITATIVOS?</i>	

<b>2. ALCANCE</b>			
<b>OBJETIVOS DEL PROYECTO:</b> <i>METAS HACIA LAS CUALES SE DEBE DIRIGIR EL TRABAJO DEL PROYECTO</i>			
<b>FUERA DE ALCANCE:</b> <i>AQUELLOS OBJETIVOS QUE NO SE CONSIDERARÁN EN ÉSTE PROYECTO</i>			
<b>2.1 DESCRIPCIÓN DEL ALCANCE DEL PRODUCTO</b>			
<b>REQUISITOS:</b> <i>CONDICIONES O CAPACIDADES QUE DEBE POSEER O SATISFACER EL PRODUCTO PARA CUMPLIR CON CONTRATOS, NORMAS, ESPECIFICACIONES, U OTROS DOCUMENTOS FORMALMENTE IMPUESTOS.</i>			
1.			
2.			
3.			
4.			
5.			
<b>RESTRICCIONES:</b>			
<b>SUPUESTOS:</b>			
<b>3. TIEMPO</b>			
<b>DURACIÓN DEL PROYECTO</b>			
<b>FECHA DE INICIO</b>		<b>FECHA DE FIN</b>	
<b>PLANIFICACIÓN GENERAL DEL PROYECTO</b>			
<b>Etapa</b>	<b>Ejecutor</b>	<b>Fecha Inicio</b>	<b>Fecha Fin</b>
<i>INICIO</i>			
<i>PLANIFICACIÓN</i>			
<i>EJECUCIÓN</i>			
<i>CIERRE</i>			

4. RECURSOS	
DESIGNACIÓN DEL LÍDER DEL PROYECTO.	
<i>NOMBRE</i>	
<i>REPORTA A</i>	

COLABORADORES USUARIOS QUE INTERVIENEN EN EL PROYECTO.	
<i>USUARIOS</i>	<i>ROL QUE DESEMPEÑA</i>

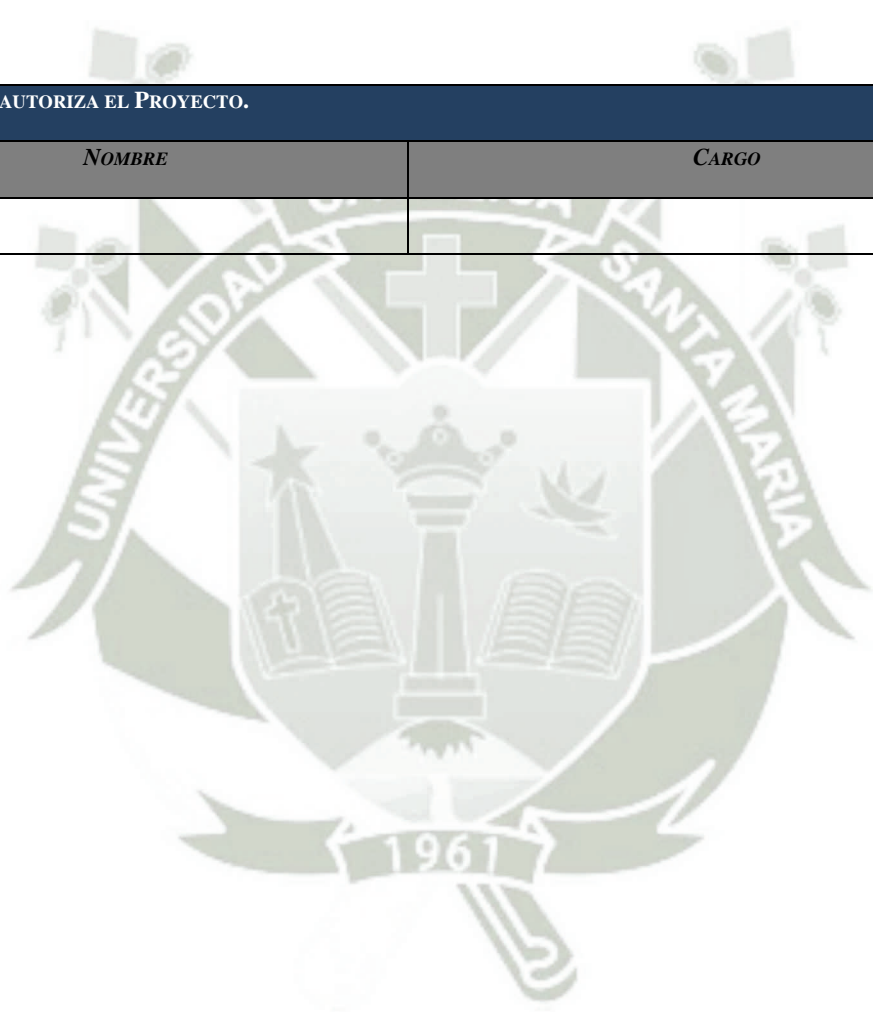
COLABORADORES TÉCNICOS QUE INTERVIENEN EN EL PROYECTO.	
<i>TÉCNICOS</i>	<i>ROL QUE DESEMPEÑA</i>

PROVEEDORES QUE INTERVIENEN EN EL PROYECTO.	
<i>ORGANIZACIÓN O GRUPO ORGANIZACIONAL</i>	<i>ROL QUE DESEMPEÑA</i>

**PRINCIPALES AMENAZAS DEL PROYECTO (RIESGOS).**


**SPONSOR QUE AUTORIZA EL PROYECTO.**

<i>NOMBRE</i>	<i>CARGO</i>



## KICK OFF

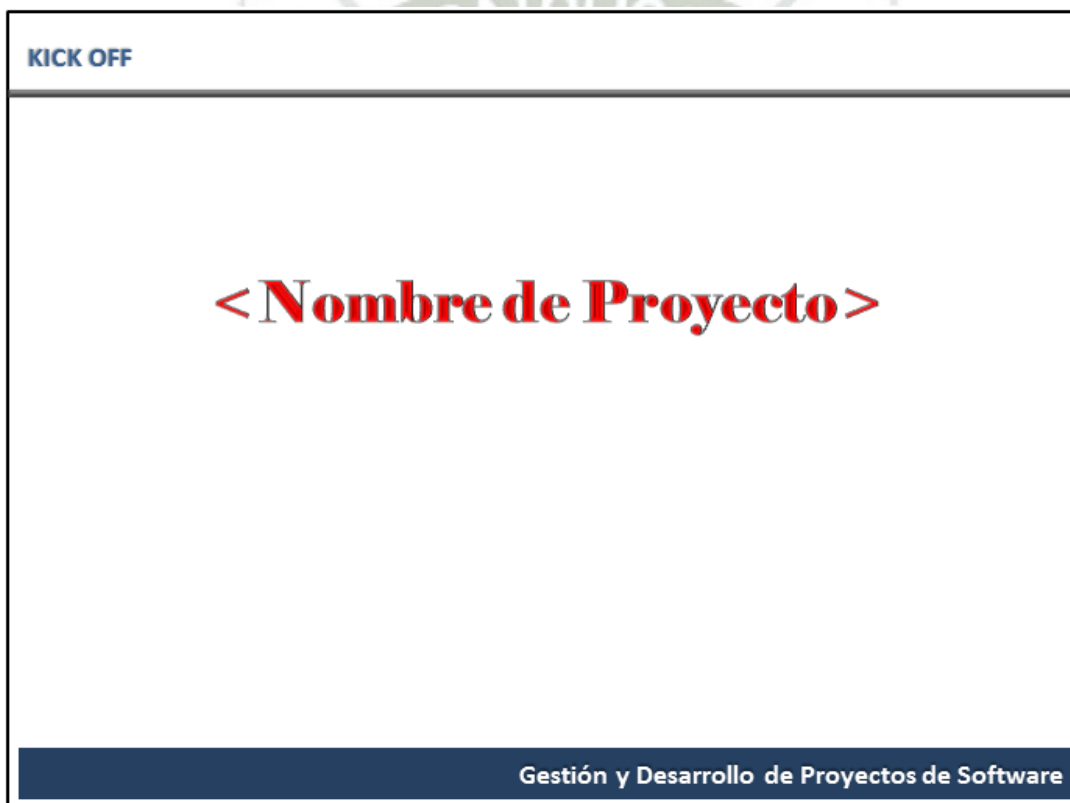
Nombre de documento:

KO\_NombreProyecto\_Kick Off\_CódigoProyeccto\_versión\_Fecha.ppt

Detalle de documento:

## *KICK OFF*

Detalle de documento:



KICK OFF

## AGENDA

1. Problemática actual

4. Estrategia

2. Objetivo

5. Equipo de trabajo

3. Alcance

6. Riesgos

7. Plan de comunicaciones

Gestión y Desarrollo de Proyectos de Software

KICK OFF

## PROBLEMÁTICA ACTUAL



Gestión y Desarrollo de Proyectos de Software

KICK OFF

## OBJETIVO



Gestión y Desarrollo de Proyectos de Software

KICK OFF

## ALCANCE



Gestión y Desarrollo de Proyectos de Software

KICK OFF

## ESTRATEGIA



Gestión y Desarrollo de Proyectos de Software

KICK OFF

## EQUIPO DE TRABAJO

Project Manager (PM)	Usuario Líder (UL)	Analista Funcional Proveedor (AF)	Analista Funcional Cliente (AF)	Analistas Programadores (AP)	Analista Calidad (AQ)	Analista de Control de Proyectos
----------------------------	--------------------------	--	--	------------------------------------	-----------------------------	--

Gestión y Desarrollo de Proyectos de Software

KICK OFF

## RIESGOS


Gestión y Desarrollo de Proyectos de Software

KICK OFF

## PLAN DE COMUNICACIONES

Nombre	Rol	Forma de comunicación	Frecuencia

Gestión y Desarrollo de Proyectos de Software

## ACTA DE REUNIÓN

Nombre de documento:

AR\_NombreProyecto\_Acta\_Reunion\_CódigoProyecto\_versión\_Fecha.docx

Detalle de documento:

## ACTA DE REUNIÓN

CONTROL DE VERSIONES				
<i>Versión</i>	<i>Elaborado por</i>	<i>Aprobado por</i>	<i>Fecha</i>	<i>Motivo</i>

<b>PROYECTO</b>		<b>CÓDIGO</b>	
<b>FASE</b>			
<b>FECHA Y HORA</b>			
<b>LUGAR</b>		<b>FACILITADOR</b>	
<b>OBJETIVO</b>			

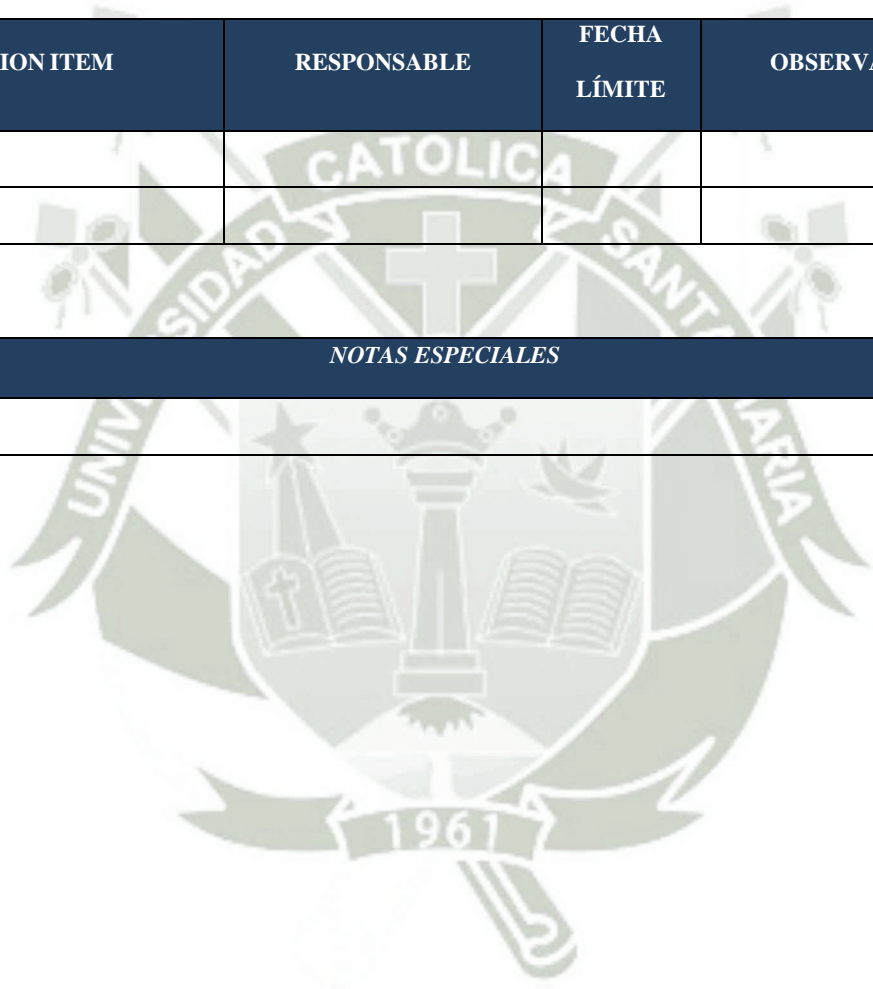
ASISTENTES			
<i>PERSONA</i>	<i>CARGO</i>	<i>ÁREA</i>	<i>ASISTIÓ</i>

AGENDA		
<i>ACTIVIDAD</i>	<i>RESPONSABLE</i>	<i>TIEMPO</i>

DOCUMENTACIÓN	
<i>QUÉ SE DEBE PRESENTAR EN LA REUNIÓN</i>	<i>RESPONSABLE</i>

CONCLUSIONES			
1			
2			
ACTION ITEM	RESPONSABLE	FECHA LÍMITE	OBSERVACIONES

NOTAS ESPECIALES	



## CHECK LIST

Nombre de documento:

CL\_NombreProyecto\_CheckListProyecto\_CódigoProyecto\_versión\_Fecha.xls

Detalle de documento:

### CHECK LIST

Etapa	Proceso	Plantilla / Actividad	Frecuencia	Rol de Responsable	Participantes	Aplica a Proyecto	Observaciones
<b>INICIO</b>							
Inicio	Asignar Líder de Proyecto	Aplicativo PPM Clarity	Uno por Proyecto	Jefe de Proyecto	Gerente de proyecto Jefe de Proyecto	Si	
Inicio	Reunión con el usuario para realizar el levantamiento de información sobre las necesidades del usuario.	Documento de acta de reunión	Según lo requerido	Jefe de Proyecto	Jefe de Proyecto	Si	
Inicio	Reunión con las áreas involucradas al proyecto.	Documento de acta de reunión	Según lo requerido	Jefe de Proyecto	Jefe de Proyecto	Si	
Inicio	Elaborar Ficha de Inicio de Proyecto	Documento de Ficha Inicio Proyecto	Uno por Proyecto	Jefe de Proyecto	Jefe de Proyecto Analista Funcional	Si	
Inicio	Aprobar Ficha de inicio de proyecto	Documento de Ficha Inicio Proyecto	Uno por Proyecto	Gerente de proyecto	Gerente de proyecto Jefe de Proyecto	Si	
<b>PLANIFICACIÓN</b>							
Planificación	Reunión de Kick – Off	Presentación Kick - Off	Según lo requerido	Jefe de Proyecto	Gerente de proyecto Jefe de Proyecto Analista Funcional Arquitecto	Si/No	
Planificación	Acta de Reunión Kick – Off	Documento de acta de reunión	Según lo requerido	Líder de Proyecto (Analista Funcional)	Jefe de Proyecto Analista Funcional	Si/No	
Planificación	Elaborar CheckList (Entregables del Proyecto)	CheckList	Uno por Proyecto	Líder de Proyecto (Analista Funcional)	Jefe de Proyecto Analista Funcional	Si/No	

Análisis	Reunión con el usuario para realizar el levantamiento de información sobre el proyecto.	Documento de acta de reunión	Según lo requerido	Jefe de Proyecto	JdP Arquitecto AF Usuario Soap Líder desarrollo Líder calidad	Si	
Análisis	Reunión con las áreas de TI para evaluar el impacto del proyecto	Documento de Arquitectura Conceptual Documento de Especificación Funcional	Uno por Proyecto	Analista Funcional	JdP Arquitecto AF Usuario Soap Líder desarrollo Líder calidad Seguridad de la Información	Si	
Análisis	Elaborar Documento de Especificación Funcional	Documento de Especificación Funcional	Uno por Proyecto	Analista Funcional	Jefe de Proyecto Arquitecto Analista Funcional	Si/No	
Análisis	Elaboración de la arquitectura conceptual	Documento de Arquitectura Conceptual	Uno por Proyecto	Arquitecto	Jefe de Proyecto Arquitecto Analista Funcional	Si	
Análisis	Validar Especificación Funcional con el Usuario, Seguridad y Gerente	Documento de Especificación Funcional	Según lo requerido	Jefe de Proyecto	JdP Arquitecto Usuario AF Soap Líder desarrollo Líder calidad Seguridad GdP	Si	
Análisis	Convocar a comité técnico	Documento de Especificación Funcional	Uno por Proyecto	Jefe de Proyecto	JdP Arquitecto AF Soap Líder desarrollo Líder calidad Seguridad GdP	Si	

Análisis	Sustentación de la EF en el Comité Técnico	Documento de Especificación Funcional	Uno por Proyecto	AF AQ	JdP Arquitecto AF Soap Líder desarrollo Líder calidad Seguridad GdP	Si	
Análisis	Elaborar acta de reunión del comité técnico	Documento de Acta de Reunión	Según lo requerido	Jefe de Proyecto	JdP Arquitecto AF Soap Líder desarrollo Líder calidad Seguridad GdP	Si	
Análisis	Enviar por mail la EF, acta de reunión y el acta de asistencia a todos los participantes del comité técnico	Documento de Especificación Funcional	Según lo requerido	Jefe de Proyecto	Jefe de Proyecto	Si	
Análisis	Elaborar documento de Arquitectura Lógica	Documento de Arquitectura Lógica	Uno por Proyecto	Arquitecto	Jefe de Proyecto Arquitecto Analista Funcional	Si/No	
Análisis	Solicitar las aprobaciones de la PS a las siguientes personas: - Usuario Solicitante - Seguridad de la Información - Gerente de Proyecto	Documento de Especificación Funcional	Según lo requerido	Jefe de Proyecto	Jefe de Proyecto Analista Funcional	Si	
Análisis	Solicitar estimación de esfuerzo inicial en QA (vía email)	Documento de Especificación Funcional	Según lo requerido	Jefe de Proyecto	Jefe de Proyecto	Si	
Análisis	Solicitar asignación de analistas programadores	Documento de Especificación Funcional	Según lo requerido	Jefe de Proyecto	Jefe de Proyecto	Si	
Diseño	Elaborar Documento de especificación técnica	Documento de Especificación Técnico	Uno por Proyecto	Analista Programador	Analista Funcional Arquitecto	Si/No	

Diseño	Definir Arquitectura Física y ambientes de desarrollo, calidad y producción	Documento de Especificación Funcional Documento de Especificación Técnico	Según lo requerido	Arquitecto	Jefe de Proyecto	Si	
Diseño	Coordinar la generación de los ambientes de Desarrollo, Calidad y Producción	Documento de Especificación Funcional Documento de Especificación Técnico	Según lo requerido	Gerente de Proyecto	Jefe de Proyecto	Si	
Diseño	Solicitar estimación de esfuerzo del equipo de desarrollo	Documento de Especificación Funcional Documento de Especificación Técnico	Según lo requerido	AP	Jefe de Proyecto Analista Funcional	Si	
Diseño	Coordinación y asignación de responsables para la generación de data de prueba para los ambientes de Desarrollo y Calidad	Documento de Especificación Funcional Documento de Especificación Técnico	Según lo requerido	AF Líder desarrollo Líder calidad Arquitecto	Jefe de Proyecto Analista Programador	Si	
Diseño	Solicitar recursos testers	Documento de Especificación Funcional Documento de Especificación Técnico	Según lo requerido	Líder de calidad	JdP	Si	
Diseño	Elaborar cronograma final de proyecto	Cronograma	Uno por proyecto	Analista Funcional	Jefe de Proyecto	Si/No	
Diseño	Solicitar aprobación de cronograma del proyecto	Cronograma	Uno por proyecto	Gerente de Proyecto	Jefe de Proyecto	Si	
Diseño	Elaborar Plan de Pruebas Unitarias	Plan de Pruebas Unitarias	Uno por Proyecto	Analista Programador	Analista Funcional	Si/No	
Diseño	Elaborar Plan de Pruebas Integrales	Plan de Pruebas Integrales	Uno por Proyecto	Analista Funcional	Usuario solicitante	Si/No	
Diseño	Elaborar Plan de Pruebas de Usuario	Plan de Pruebas de Usuario	Uno por Proyecto	Analista Funcional	Usuario solicitante	Si/No	
Planificación	Elaborar Plan de Comunicaciones	Plan de comunicaciones	Uno por Proyecto	Analista Funcional	Jefe de Proyecto		
Planificación	Elaborar Plan de Gestión de Riesgos	Matriz de Riesgos	Uno por Proyecto	Analista Funcional	Jefe de Proyecto	Si/No	
Planificación	Registrar la información del proyecto en el PPM Clarity (Documentos de análisis + aprobaciones + documentos de planificación)	PPM Clarity	Según lo requerido	Jefe de Proyecto	Jefe de Proyecto	Si	
Planificación	Registrar las fechas del Proyecto en el Clarity. Crear las tareas en el Clarity, asignado tiempos y recursos.	PPM Clarity	Según lo requerido	Jefe de Proyecto	Jefe de Proyecto	Si	
Planificación	Registrar los aplicativos impactados en el Clarity.	PPM Clarity	Según lo requerido	Jefe de Proyecto	Jefe de Proyecto	Si	
Planificación	Registrar el equipo de trabajo en el Clarity.	PPM Clarity	Según lo requerido	Jefe de Proyecto	Jefe de Proyecto	Si	

Planificación	Crear los paquetes en Clarity	PPM Clarity	Según lo requerido	Jefe de Proyecto	Jefe de Proyecto	Si	
Planificación	Subir los Documentos de Análisis y Aprobaciones al SCM	PPM Clarity	Según lo requerido	Jefe de Proyecto	Jefe de Proyecto	Si	
Planificación	Solicitar al AP promover los paquetes de la fase de análisis a desarrollo	CA SCM	Según lo requerido	Jefe de Proyecto	Jefe de Proyecto Analista Funcional	Si	
Planificación	Subir los documentos de análisis y diseño del AP en los paquetes del SCM	CA SCM	Según lo requerido	Jefe de Proyecto	Jefe de Proyecto Analista Funcional	Si	
Planificación	Solicitar asignación de Analistas de Soporte (SOAP) a cada paquete	CA SCM	Según lo requerido	Jefe de Proyecto	JdP Jefe SOAP	Si	
<b>EJECUCIÓN</b>							
Construcción	Realizar Programación correspondiente		Según lo requerido	Analista Programador	Analista Funcional	Si	
Construcción	Soporte Funcional al Desarrollo		Según lo requerido	Analista Funcional	Jefe de Proyecto Analista Funcional		
Construcción	Elaborar Evidencias de Pruebas Unitarias	Evidencia de Pruebas Unitarias	Según lo requerido	Analista Programador	Analista Funcional	Si	
Construcción	Elaborar Evidencias de Pruebas Integrales	Evidencia Pruebas	Según lo requerido	Analista Programador		Si	
Construcción	Realizar Pruebas Funcionales del Desarrollo	Pruebas Funcionales/Integrales	Según lo requerido	Analista Funcional	Jefe de Proyecto Analista Funcional	Si	
Construcción	Realizar Pruebas de Usuario del Desarrollo	Pruebas Funcionales/Integrales	Según lo requerido	Analista Funcional	Jefe de Proyecto Analista Funcional	Si	
Construcción	Ingresar y Actualizar los Entregables del proyecto en el Repositorio (CA SCM)	CA SCM	Según lo requerido	Analista Programador	Jefe de Proyecto Analista Funcional	Si	
Construcción	Elaborar documentación/entregables de desarrollo (Manual de Instalación, Manual Operación, Manual de Usuario, etc)	Entregables de desarrollo	Según lo requerido	Analista Programador	Jefe de Proyecto Analista Funcional	Si	
Transición	Priorización en QA	Formato de priorización QA	Uno por Proyecto	GdP	GdP	Si	
Transición / Pre - Certificación	Validación de documentación (MI, MO)		Según lo requerido	SOAP	AF SOAP AP	Si	
Transición / Pre - Certificación	Pre - Certificación	Promover el PQT de la etapa de Desarrollo a	Según lo requerido	AP	AF SOAP AP	Si	

		Certificación en CA SCM					
Transición / Pre - Certificación	Pre - Certificación	Notificación formal del pase a certificación	Uno por Proyecto	JdP	AF SOAP AP	Si	
Transición / Certificación	Certificación	Instalación y configuración del ambiente de pruebas	Según lo requerido	AC	Líder Calidad		
Transición / Certificación	Certificación	Solicitar controles de arquitectura	Según lo requerido	Arquitecto	JdP	Si	
Transición / Certificación	Certificación	Realizar el seguimiento del proyecto en QA	Según lo requerido	JdP	JdP		
Transición / Certificación	Certificación	Pruebas en QA (AC, AF, AP, U) Pruebas de cada componente Pruebas de sistema Pruebas integrales	Según lo requerido	AC	JdP AF AP U AC	Si	
Transición / Certificación	Certificación	Soporte Funcional / Técnico las pruebas de Calidad	Según lo requerido	AC	AF JdP AP	Si/No	
Transición / Certificación	Certificación	Seguimiento y control de la ejecución de las pruebas de Calidad	Según lo requerido	Líder Calidad	Líder Calidad JdP AC	Si	
Transición / Certificación	Certificación	Gestionar aprobaciones del AF y el Usuario por correo el plan de pruebas enviados por el QA.	Según lo requerido	AC	JdP AF U	Si	
Transición / Certificación	Certificación	Validación y aprobación de documentación (MI, MO, MU)	Según lo requerido	AC	JdP SOAP Líder de Calidad Usuario Analista Calidad	Si	
Transición / Certificación	Certificación	Validar las aprobaciones en HARVEST	Según lo requerido	AC Líder QA Soap GdP	JdP AC Líder QA Soap GdP	Si	
Transición / Certificación	Certificación	Validar que todos los pquetes se encuentren en COMITÉ DE PASES A PRODUCCIÓN	Según lo requerido	JdP	JdP	Si	
Transición / Certificación	Comité de pase a producción	Sustentación del PQT en el Comité de PaP	Según lo requerido	SOAP	JdP	Si	

Transición / Certificación	Comité de pase a producción	Informa al usuario la fecha programada para el pase a producción.	Según lo requerido	JdP	JdP	Si	
Transición / Producción	Producción	Realizar las pruebas Post Producción y obtener el OK del usuario sobre el pase	Según lo requerido	AF Usuario	JdP	Si	
Transición / Producción	Producción	Soporte funcional a pruebas en ambiente de producción	Según lo requerido	SOAP AF AP AC	JdP	Si	
Transición / Producción	Producción	Solicitar y ejecutar control de arquitectura	Según lo requerido	Arquitecto SOAP	JdP	Si	
<b>SEGUIMIENTO Y CONTROL</b>							
Seguimiento y Control	Dirigir y Gestionar la Ejecución del Proyecto	Informe de salud del proyecto	Según lo requerido	Jefe de Proyecto	JdP AF	Si	
	Ejecutar seguimiento y control del Proyecto	Actualización de cronograma	Según lo requerido	Jefe de Proyecto	JdP AF	Si	
	Gestionar y Monitorear los Riesgos del Proyecto	Informe de Riesgos	Según lo requerido	Jefe de Proyecto	JdP AF	Si	
	Actualizar Repositorio de Entregables del Proyecto (CA Clarity PPM)	CA Clarity PPM	Según lo requerido	Jefe de Proyecto	JdP AF	Si	
	Lecciones aprendidas	Registrar lecciones aprendidas en PPM	Según lo requerido	Jefe de Proyecto	JdP AF	Si	
<b>CIERRE</b>							
Cierre	Realizar cierre del proyecto	Acta de cierre Proyecto	Uno por Proyecto	Jefe de Proyecto	JdP	Si	
	Registrar Cierre de Proyecto en Clarity	Registrar Cierre de Proyecto en Clarity	Uno por Proyecto	Jefe de Proyecto	JdP	Si	

JdP            Jefe de Proyecto  
 AF            Analista Funcional  
 AP            Analista Programador  
 SOAP        Analista de Soporte de Aplicaciones  
 AQ            Arquitecto  
                   Especificación  
 EF            Funcional

RF	Requerimiento Funcional
AC	Analista de calidad
U	Usuario
MI	Manual de instalación
MO	Manual de Operación
MU	Manual de Usuario



## ESPECIFICACIÓN FUNCIONAL

Nombre de documento:

EF\_NombreProyecto\_EspecificaciónFuncional\_CódigoProyecto\_versión\_Fecha.docx

Detalle de documento:

### *ESPECIFICACIÓN FUNCIONAL*

- Nombre de proyecto

*(Describir el nombre del proyecto a ejecutar)*

- Código de proyecto

*(Describir el código del proyecto asignado)*

- Necesidad del Negocio u Oportunidad a aprovechar

*(Describir las limitaciones de la situación actual y las razones por las cuáles se emprende el proyecto)*

--

- Objetivos del Proyecto

*(Definir con claridad los objetivos del negocio y del proyecto)*


- Requisitos Funcionales

*(Describir procesos del negocio, información, interacción con el producto, etc.)*

Stakeholder	Prioridad otorgada por el Stakeholder	Requisitos	
		Código	Descripción

- Requisitos no Funcionales

*(Describir requisitos tales cómo nivel de servicio, performance, adecuación, etc.)*

Stakeholder	Prioridad otorgada por el Stakeholder	Requisitos	
		Código	Descripción

- Requisitos de Seguridad

*(Describir requisitos relativos a seguridad, se define el nivel de confidencialidad de la información que manejará el sistema, información que debe ser enmascarada, etc.)*

Stakeholder	Prioridad otorgada por el Stakeholder	Requisitos	
		Código	Descripción

- Modelado del Proceso Actual

- Descripción del proceso actual
- Funcionalidades del proceso actual
- Diagramas de flujo de procesos
- Modelado del Proceso Propuesto
  - Descripción del proceso propuesto
  - Funcionalidades del proceso propuesto
  - Diagramas de flujo de propuesto
- Reglas de negocio

- Impacto en otras Áreas

- Impacto en otras Entidades

- Impacto en otros sistemas

- Impacto legal

- Requerimientos de Soporte y Entrenamiento

---

- Supuestos

- Restricciones

- Aceptación del documento



## ESPECIFICACIÓN TÉCNICA

Nombre de documento:

ET\_NombreProyecto\_EspecificaciónTécnica\_CódigoProyecto\_versión\_Fecha.docx

Detalle de documento:

### *ESPECIFICACIÓN TÉCNICA*

- Introducción
  - Alcance
- Listado de reglas del negocio
- Tabla Reglas del negocio para el requerimiento funcional

Número	Regla

- Modelo de persistencia
  - Diagrama Entidad Relación Físico
  - Diccionario de Datos

Nombre de la tabla

Col#	Nombre Columna	Tipo de Dato	Llave PK / FK	Nulo	Valor por Defecto	Comentario

	Nombre del índice
	Nombre de la tabla
Col	Nombre Columna

- Especificación de Componentes de Base de datos

COMPONENTE					
NOMBRE LÓGICO					
OBJETIVO					
MÉTODOS					
NOMBRE			DESCRIPCION		
PARÁMETROS					
NOMBRE		TIPO DE DATO		IN / OUT	DESCRIPCION
DESCRIPCIÓN DE ACTIVIDADES					
BASE DE DATOS					
OBJETO		SEL	INS	DEL	UPD
					Otro (Triggers)

--	--	--	--	--	--

- Esquemas de la base de datos
- Determinar los roles de bases de datos
- Descripción y de los usuarios de la base de datos
- Especificación del proyecto de software
  - Integrantes del equipo

Función	Nombre(s) y Apellidos	Rol	Desde	Hasta

- Registro de Integrantes relevados

Función	Nombre(s) y Apellidos	Rol	Desde	Hasta

- REPOSITORIO DEL PROYECTO
  - Análisis
  - Desarrollo
  - Certificación
  - Producción
- Información específica
  - Solución técnica
  - COMPONENTES DEL MODULO
  - Componente / Aplicación → MOD01 - Nombre del Módulo
  - Diagrama de Estados
  - FUNCIONALIDAD EXCLUIDA O FUERA DE ALCANCE PARA EL MÓDULO
- Aceptación

## PLAN DE COMUNICACIONES

Nombre de documento:

PC\_NombreProyecto\_PlanComunicaciones\_CódigoProyecto\_versión\_Fecha.xlsx

Detalle de documento:

### *PLAN DE COMUNICACIONES*

PLAN DE COMUNICACIONES					
<Logo Empresa>	Dirección Tecnología de la Información				
	Jefatura de Control de Proyectos				
CONTROL DE VERSIONES					
Versión	Elaborado por	Aprobado por	Fecha	Motivo	
Código del Proyecto			Nombre del Proyecto		
Comunicación	Medio de Comunicación	Responsable	Frecuencia	Contenido de la Comunicación	Audiencia

## MATRIZ DE RIESGOS

Nombre de documento:

MR\_NombreProyecto\_MatrizRiesgos\_CódigoProyecto\_versión\_Fecha.xls

Detalle de documento:

### *MATRIZ DE RIESGOS*

MATRIZ DE RIESGOS							
<Logo Empresa>	Gerencia de Tecnología de la Información						
	Jefatura de Control de Proyectos						
	Clasificación de la Información: Confidencial / Uso Interno						
CONTROL DE VERSIONES							
Versión	Elaborado por	Aprobado por	Fecha	Motivo			
Código del Proyecto				Nombre del Proyecto			
Descripción	Prob.	Imp.	Exposición	Plan de	Plan de	Responsable	Fecha
Riesgo	(1 al 3)	(1 al 5)	(P*I)	Mitigación	Contingencia		

Probabilidad
Bajo = 1
Medio = 2
Alto = 3

Impacto
Bajo = 1
Alto = 5

## PRUEBAS FUNCIONALES

Nombre de documento:

PI\_NombreProyecto\_PruebasIntegrales\_CódigoProyecto\_versión\_Fecha.docx

Detalle del documento:

### *PRUEBAS FUNCIONALES*

- Información del Proyecto

<b>Nombre del Proyecto</b>	
<b>Objetivo del Proyecto</b>	
<b>Fecha de Inicio</b>	
<b>Fecha de Término Estimado</b>	
<b>Proyectos Relacionados</b>	

- Especificación de los escenarios de pruebas
- Requerimientos de data de prueba
- Casos de prueba
- Aceptación



*Estándar*

*Medio*

*Urgente)*

*Bajo)*

3. Plan de contingencia ante solicitudes de cambio urgentes

4. Aceptación del documento



## SOLICITUD DE CAMBIOS

Nombre de documento:

SC\_NombreProyecto\_SolicitudCambio\_CódigoProyecto\_nroSRTI\_versión\_Fecha.docx

Detalle de documento:

### *SOLICITUD DE CAMBIOS*

#### 1. NOMBRE DEL PROYECTO

#### 2. FECHA DE SOLICITUD

#### 3. USUARIO SOLICITANTE

#### RESPONSABLE DEL PROYECTO

<input type="text"/>	<input type="text"/>
----------------------	----------------------

**4. Definición del Problema o Situación Actual:** *(Defina y acote el problema que se va a resolver, distinguiendo el problema de sus causas, y de sus consecuencias.)*

**5. Descripción detallada del Cambio Solicitado:** *(Especifique con claridad el cambio solicitado, precisando el qué, quién, cómo, cuándo y dónde.)*

---

**6. Razón por la que se solicita el Cambio:** (Especifique con claridad porque motivos o razones solicita el cambio, qué sucedería si el cambio no se realiza.)

**7. Impacto en el Proyecto** (Explicar el impacto de realizar el cambio en tiempo y costo)

**8. Impacto en otros proyectos u operaciones**

**9. Observaciones y Comentarios adicionales**

**10. FECHA DE APROBACIÓN (FECHA DE APROBACIÓN DE LA SOLICITUD)**

**11. ACEPTACIÓN**

## MANUAL DE USUARIO

Nombre de documento:

MU\_NombreProyecto\_ManualUsuario\_CódigoProyecto\_versión\_Fecha.docx

Detalle de documento:

### *MANUAL DE USUARIO*

#### **1. Introducción**

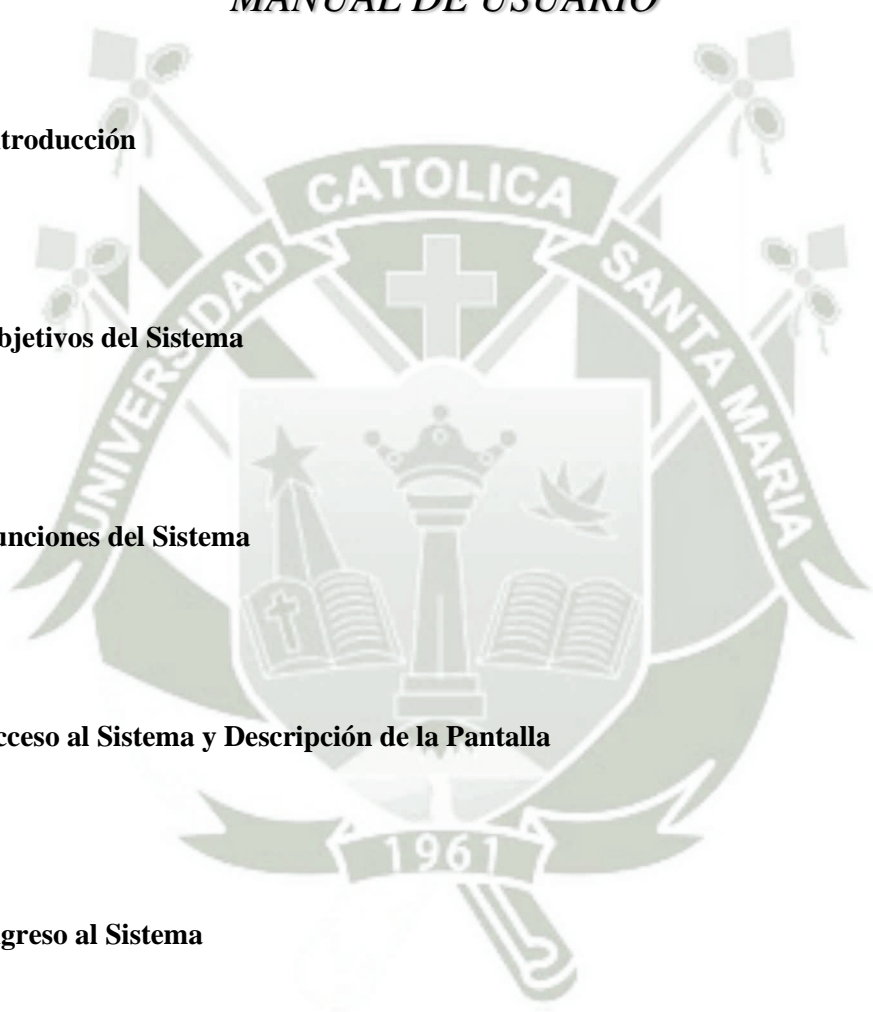
##### **1.1. Objetivos del Sistema**

##### **1.2. Funciones del Sistema**

#### **2. Acceso al Sistema y Descripción de la Pantalla**

##### **2.1. Ingreso al Sistema**

##### **2.2. Referencias de Uso de la Simbología**



### **2.3. Descripción de las Pantallas**

## **3. Funcionalidad del Sistema**

### **3.1. Descripción Funcional (Mapa Del Sistema)**

### **3.2. Descripción detallada de las opciones del menú**

## **4. Glosario de Términos y Abreviaturas**

## **5. Anexos**



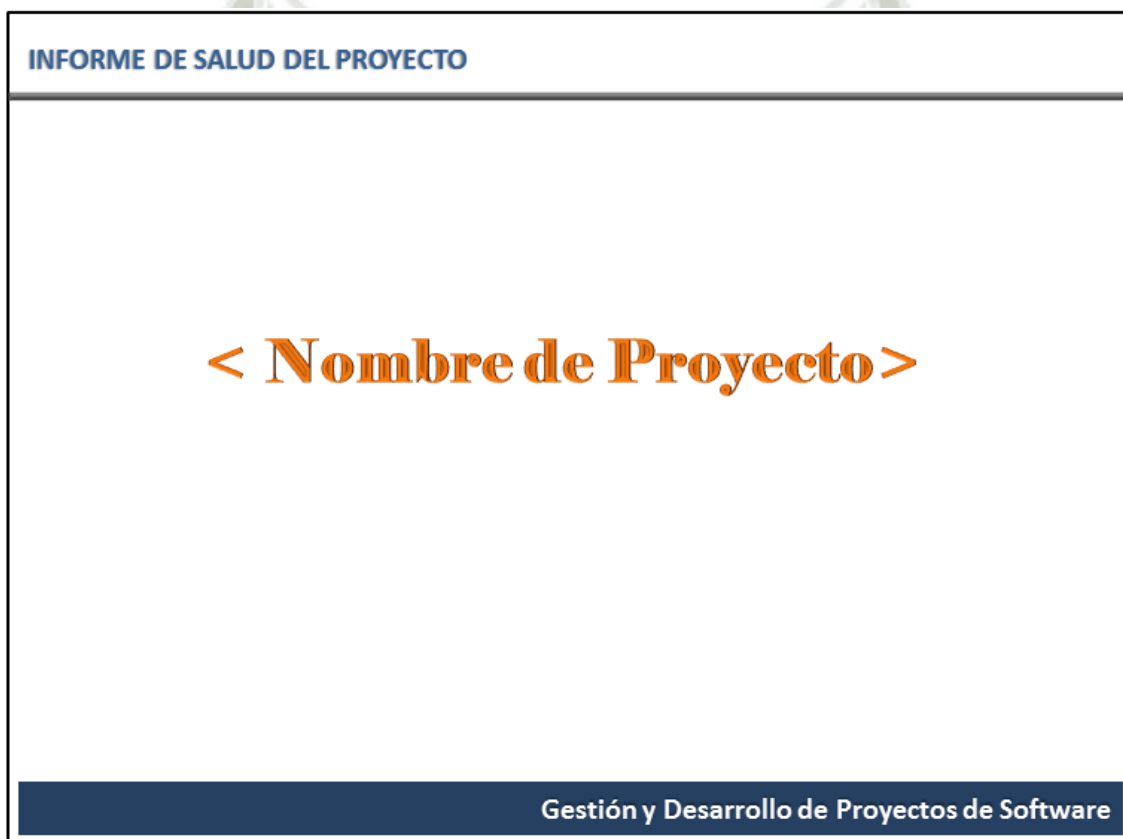
## INFORME DE SALUD DEL PROYECTO

Nombre de documento:

IS\_NombreProyecto\_InformeSaludProyecto\_CódigoProyecto\_versión\_Fecha.pptx

Detalle de documento:

### *INFORME DE SALUD DEL PROYECTO*



**INFORME DE SALUD DEL PROYECTO**

Agenda


- Tablero de Control
- Riesgos y Problemas
- Actividades Desarrolladas
- Tareas Pendientes
- Próximos Pasos


Gestión y Desarrollo de Proyectos de Software


**INFORME DE SALUD DEL PROYECTO**

Tablero de Control

Proyecto	F. Planificada Término	F. Estimada Término	% Avance Real	% Avance Planificado	Estado

 Riesgo **BAJO** de retrasos en cronograma y rebase de presupuesto

 Riesgo **MEDIO** de retrasos en cronograma y rebase de presupuesto

 Riesgo **ALTO** de retrasos en cronograma y rebase de presupuesto

Gestión y Desarrollo de Proyectos de Software

**INFORME DE SALUD DEL PROYECTO**

Riesgos y Problemas



FEP: Fecha Planificada Término

FEE: Fecha Estimada Término

Gestión y Desarrollo de Proyectos de Software

**INFORME DE SALUD DEL PROYECTO**

Actividades Desarrolladas



Gestión y Desarrollo de Proyectos de Software

**INFORME DE SALUD DEL PROYECTO**

Tareas Pendientes



Gestión y Desarrollo de Proyectos de Software

**INFORME DE SALUD DEL PROYECTO**

Próximos pasos



Gestión y Desarrollo de Proyectos de Software

## INFORME DE MONITOREO DE RIESGOS

Nombre de documento:

**IR\_NombreProyecto\_InformeRiesgos\_CódigoProyecto\_versión\_Fecha.docx**

Detalle de documento:

### *INFORME MONITOREO DE RIESGOS*

#### **1. Riesgos actuales potenciales**

##### **1.1. Revisión de triggers para los riesgos identificados inicialmente**

##### **1.2. Revisión de adecuación de respuestas planificadas para los riesgos identificados inicialmente**

##### **1.3. Revisión de planes de contingencia para los riesgos identificados inicialmente**

#### **2. Riesgos actuales sucedidos**

##### **2.1. Valoración de impacto real vs impacto estimado**

##### **2.2. Revisión de planes de contingencia**

## **2.3. Evaluación de necesidades de acciones correctivas o solicitudes de cambio**

### **3. Nuevos riesgos detectados**

#### **3.1. Definición de triggers de riesgos**

#### **3.2. Evaluación y categorización de riesgos**

#### **3.3. Definición de respuestas planificadas**

#### **3.4. Definición de planes de contingencia**

#### **3.5. Programación de ejecución de respuestas planificadas**

### **4. Aceptación del documento**

## FICHA DE CIERRE DEL PROYECTO

Nombre de documento:

**FC\_NombreProyecto\_FichaCierre\_CódigoProyecto\_versión\_Fecha.docx**

Detalle de documento:

### *FICHA DE CIERRE DEL PROYECTO*

CONTROL DE VERSIONES				
<i>Versión</i>	<i>Elaborado por</i>	<i>Aprobado por</i>	<i>Fecha</i>	<i>Motivo</i>

NOMBRE DEL PROYECTO		CÓDIGO DEL PROYECTO
DESCRIPCIÓN DEL PROYECTO: <i>QUÉ, QUIÉN, CÓMO, CUÁNDO Y DÓNDE?</i>		
OBJETIVOS DEL PROYECTO		
OBJETIVO INICIAL	OBJETIVO LOGRADO	OBJETIVO NO LOGRADO

DURACIÓN REAL DEL PROYECTO			
<i>FECHA DE INICIO</i>		<i>FECHA DE FIN</i>	
FECHAS REALES AL TÉRMINO DEL PROYECTO			
<i>Etapa</i>	<i>Ejecutor</i>	<i>Fecha Inicio</i>	<i>Fecha Fin</i>
<i>INICIO</i>			

<i>PLANIFICACIÓN</i>			
<i>EJECUCIÓN</i>			
<i>CIERRE</i>			

PLATAFORMAS / SISTEMAS IMPACTADOS:	
<i>SISTEMA</i>	<i>DESCRIPCIÓN DE IMPACTO</i>

SECCIÓN DE ACEPTACIÓN DEL PROYECTO

DECLARACIÓN DE LA ACEPTACIÓN FORMAL	
OBSERVACIONES ADICIONALES	
ACEPTADO POR	
<i>NOMBRE DEL SPONSOR U OTRO FUNCIONARIO</i>	<i>FECHA</i>
SPONSOR QUE AUTORIZA EL PROYECTO.	
<i>NOMBRE</i>	<i>CARGO</i>

## ANEXO B

### KICK OFF



KICK OFF

**PROYECTO**

Gestión y Desarrollo de Proyectos de Software

The slide features a white background with a dark blue header and footer. The word 'KICK OFF' is in the top left corner. The word 'PROYECTO' is centered in large, bold, red letters. The footer contains the text 'Gestión y Desarrollo de Proyectos de Software'.



**KICK OFF**

**AGENDA**

1. Problemática actual

4. Estrategia

2. Objetivo

5. Equipo de trabajo

3. Alcance

6. Riesgos

7. Plan de comunicaciones

Gestión y Desarrollo de Proyectos de Software



## ESPECIFICACIÓN FUNCIONAL

### 1. Nombre de proyecto

*(Describir el nombre del proyecto a ejecutar)*

Automatización

### 2. Código de proyecto

*(Describir el código del proyecto asignado)*

1111

### 3. Necesidad del Negocio u Oportunidad a aprovechar

*(Describir las limitaciones de la situación actual y las razones por las cuáles se emprende el proyecto)*

Automatización del Proceso.

### 4. Objetivos del Proyecto

*(Definir con claridad los objetivos del negocio y del proyecto)*

Gestión ágil de los procesos por parte de las áreas involucradas.

Control y seguimiento de los tiempos, estados y asignaciones.

Agilizar, optimizar y mejorar la calidad de los tiempos de respuesta y comunicación de cada una de las áreas involucradas.

Mejorar la gestión de la información necesaria para las bases integradas.

Obtención de información para la toma de decisiones de la gerencia.

### 5. Requisitos Funcionales

*(Describir procesos del negocio, información, interacción con el producto, etc.)*

Stakeholder	Prioridad otorgada por el Stakeholder	Requisitos	
		Código	Descripción
----	---	---	--
----	---	---	--

## 6. Requisitos no Funcionales

*(Describir requisitos tales como nivel de servicio, performance, adecuación, etc.)*

Stakeholder	Prioridad otorgada por el Stakeholder	Requisitos	
		Código	Descripción
Cliente	ALTA	0001	La herramienta deberá ser parametrizable.
Cliente	ALTA	0002	La herramienta deberá de implementar un mecanismo de LOG para generar las tramas de eventos satisfactorios o causas de error en los procesos.

## 7. Requisitos de Seguridad

*(Describir requisitos relativos a seguridad, se define el nivel de confidencialidad de la información que manejará el sistema, información que debe ser enmascarada, etc.)*

Stakeholder	Prioridad otorgada por el Stakeholder	Requisitos	
		Código	Descripción
Cliente	ALTA	0001	Se consultará información de Clientes.

Cliente	ALTA	0002	Se realizará notificaciones a los usuarios mediante el Servidor de Correos.
Cliente	ALTA	0003	Se autenticará utilizando el servidor Active Directory.

## 8. Modelado del Proceso Actual

- **Descripción del proceso actual**

Manual.

- **Funcionalidades del proceso actual**

Proceso manual.

- **Diagramas de flujo de procesos**

No aplica, debido al alcance del presente proyecto de investigación.

## 9. Modelado del Proceso Propuesto

- **Descripción del proceso propuesto**

Desarrollar una aplicación que permita automatizar la administración automatizada del proceso.

- **Funcionalidades del proceso propuesto**

a) Notificaciones

a. La aplicación permite enviar notificaciones como respuesta a los eventos que se producen durante el ciclo de vida de las peticiones.

## 10. Diagramas de flujo de propuesto

No aplica.

## 11. Reglas de negocio

No aplica.

## 12. Impacto en otras Áreas

Consultoría y Diseño, Área técnica, Legal, Facturación, Cobranzas y Centro de Atención Empresarial.

## 13. Impacto en otras Entidades

No aplica.

## 14. Impacto en otros sistemas

Servidor de correos.

## 15. Impacto legal

No aplica.

## 16. Requerimientos de Soporte y Entrenamiento

No aplica.

## 17. Supuestos

Se asume que los interesados tienen un sólido conocimiento de los procesos que intervienen en el proyecto.

---

**18. Restricciones**

No aplica.

**19. Aceptación del documento**



## ESPECIFICACIÓN TÉCNICA

### 1. Introducción

#### a. Alcance

Automatización de procesos.

### 2. Listado de reglas del negocio

#### a) Tabla Reglas del negocio para el requerimiento funcional

Número	Regla
0001	No aplica.

### 3. Modelo de persistencia

#### a. Diagrama Entidad Relación Físico

Producto enlatado.

#### b. Diccionario de Datos

No aplica (La información de nombres de tablas y sus campos correspondientes son manipulados por el proveedor no teniendo acceso a la BD).

Nombre de la tabla						
Col#	Nombre Columna	Tipo de Dato	Llave PK / FK	Nulo	Valor por Defecto	Comentario

	<b>Nombre del índice</b>
	<b>Nombre de la tabla</b>
<b>Col</b>	<b>Nombre Columna</b>

**c. Especificación de Componentes de Base de datos**

No aplica.

COMPONENTE					
NOMBRE LÓGICO					
OBJETIVO					
MÉTODOS					
NOMBRE			DESCRIPCION		
PARÁMETROS					
NOMBRE		TIPO DE DATO		IN / OUT	DESCRIPCION
DESCRIPCIÓN DE ACTIVIDADES					
BASE DE DATOS					
OBJETO		SEL	INS	DEL	UPD
Otro (Triggers)					
Esquema Oracle					

**d. Esquemas de la base de datos**

Oracle Database 11g (JIRADB)

Arquitectura Intranet:

No aplica.

**e. Determinar los roles de bases de datos**

No aplica.

**f. Descripción y de los usuarios de la base de datos**

No aplica.

**4. Especificación del proyecto de software**

**a. INTEGRANTES DEL EQUIPO**

Función	Nombre(s) y Apellidos	Rol	Desde	Hasta
<b>AF001</b>	Gabriela Torres	Analista Funcional	01/03/2014	01/03/2015
<b>AP001</b>	Mateo Soto	Analista Programador	01/03/2014	01/03/2015
<b>AP002</b>	Joaquín Del Valle	Analista Programador	01/03/2014	01/03/2015
<b>AQ001</b>	Carolina Ortega	Analista Calidad	01/03/2014	01/03/2015

**b. Registro de Integrantes relevados**

No se tienen.

Función	Nombre(s) y Apellidos	Rol	Desde	Hasta

## 5. REPOSITORIO DEL PROYECTO

No aplica.

### 6. Análisis

Promover a Desarrollo el paquete:

Paquete Harvest PQT-168345

### 7. Desarrollo

Promover a Certificación el paquete:

Paquete Harvest PQT-168345

### 8. Certificación

Promover a Producción el paquete:

Paquete Harvest PQT-168345

### 9. Información específica

### 10. Campos a medida TDR

Campos son referenciales.

Campo	Tipo	Descripción	Valores	Obligatorio	Pantalla Creación	Pantalla Edición	Pantalla Vista
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

### 11. Campos a medida LIC

Campo	Tipo	Descripción	Valores	Obligatorio	Pantala Creación	Pantalla Edición	Pantalla Vista
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

### 12. COMPONENTES DEL MODULO

Aplicación:

- a) JIRA
- b) Apache Tomcat 7.0.55

### 13. Componente / Aplicación → MOD01 - Nombre del Módulo

No aplica.

### 14. Diagrama de Estados

No aplica.

### 15. FUNCIONALIDAD EXCLUIDA O FUERA DE ALCANCE PARA EL MÓDULO

No aplica.

## MATRIZ DE RIESGOS

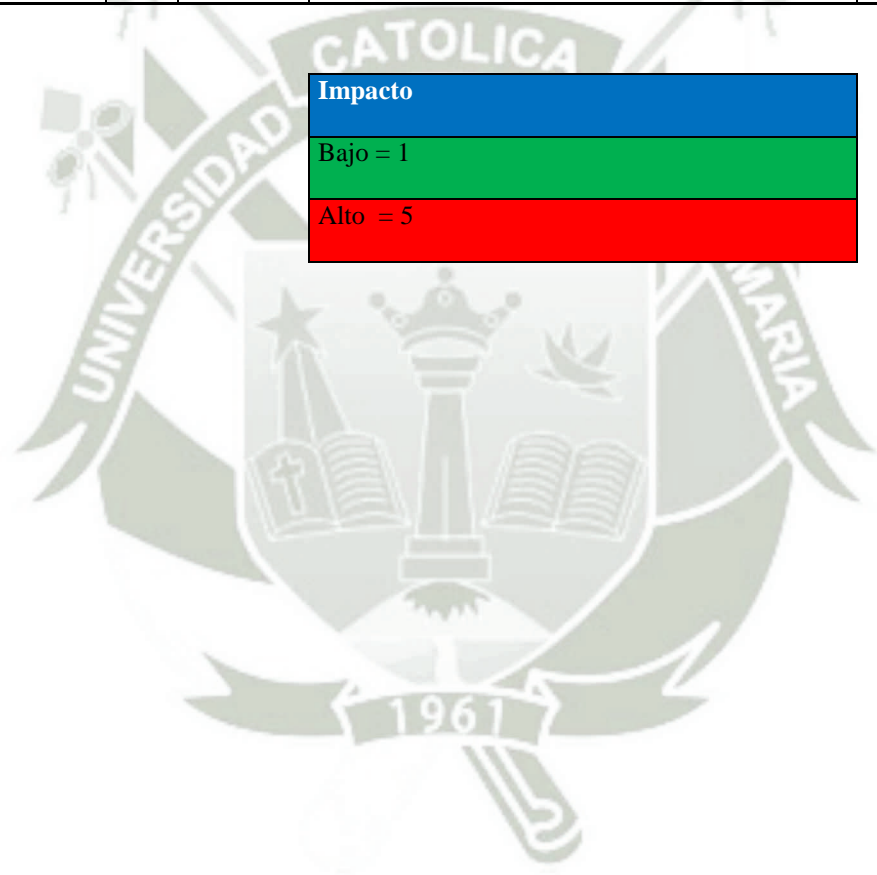
<LOGO EMPRESA>		MATRIZ DE RIESGOS						
		Gerencia de Tecnología de la Información						
		Jefatura de Control de Proyectos						
CONTROL DE VERSIONES								
Versión	Elaborado por	Aprobado por	Fecha	Motivo				
v1.0	Gabriela Torres (LP)	Carlos Estrada (PM)	10/10/214	Riesgos que podrían impactar en la entrega del proyecto de Automatización				
Código del Proyecto			Nombre del Proyecto					
PROY 111			AUTOMATIZACIÓN					
Descripción Riesgo	Prob. (1 al 3)	Imp. (1 al 5)	Exposición (P*I)	Plan de Mitigación	Plan de Contingencia	Responsable	Fecha	

No cumplir con el cronograma inicial debido a que en la etapa de análisis se requiera mayor tiempo del esperado en elaborar y validar los documentos funcional y técnico	2	5	10	Asegurar la aprobación de la documentación y su entendimiento y un acuerdo entre las partes de modificar el cronograma considerando como inicio de la construcción una fecha posterior a la aprobación del análisis.	Cronograma transparente	Gabriela Torres	
Retraso en la etapa de desarrollo, Certificación y Implantación, debido a la disponibilidad de Entornos (ambientes) de trabajo	2	5	10	Solicitar la creación de entornos (ambientes) a soporte IT con anticipación	Cronograma transparente	Gabriela Torres	
No cumplir con el cronograma inicial por integración con otras áreas.	2	5	10	Coordinar con un soporte en las fechas en las que se espera hacer Integración	Desplazar el equipo técnico Cronograma transparente.	Gabriela Torres	
No cumplir con el cronograma inicial debido a la disponibilidad de los usuarios para reuniones en el transcurso del proyecto	1	5	5	Hacer partícipes del proyecto y coordinar anticipadamente las reuniones a agenda Outlook	Cronograma transparente	Gabriela Torres	

No alcanzar la fecha final comprometida del proyecto, debido a que a los tiempos que QA asigne al proyecto.	2	5	10	Convocar a reunión al equipo de QA para brindarles un mayor capacitación en el uso del aplicativo.	Actualizar el cronograma considerando el tiempo estimado para establecer la fecha final.	Gabriela Torres
---	---	---	----	--	--	-----------------

<b>Probabilidad</b>
Bajo = 1
Medio = 2
Alto = 3

<b>Impacto</b>
Bajo = 1
Alto = 5



## PRUEBAS INTEGRALES

### 1. Información del Proyecto

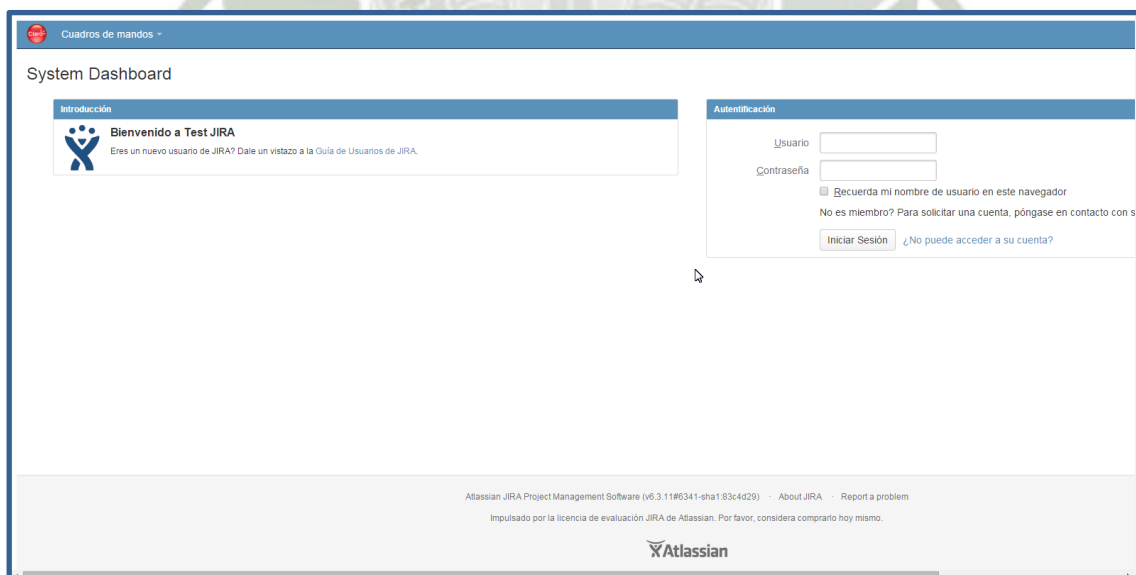
<b>Nombre del Proyecto</b>	Automatización
<b>Objetivo del Proyecto</b>	Automatizar Procesos
<b>Fecha de Inicio</b>	01/09/2014
<b>Fecha de Término Estimado</b>	28/10/2014
<b>Proyectos Relacionados</b>	Ninguno

#### 1.1. Especificación de los escenarios de pruebas

### 2. Requerimientos de data de prueba

No aplica.

**Figura 2.1. Pantalla Principal del aplicativo**



**Figura 2.2. Login del usuario**

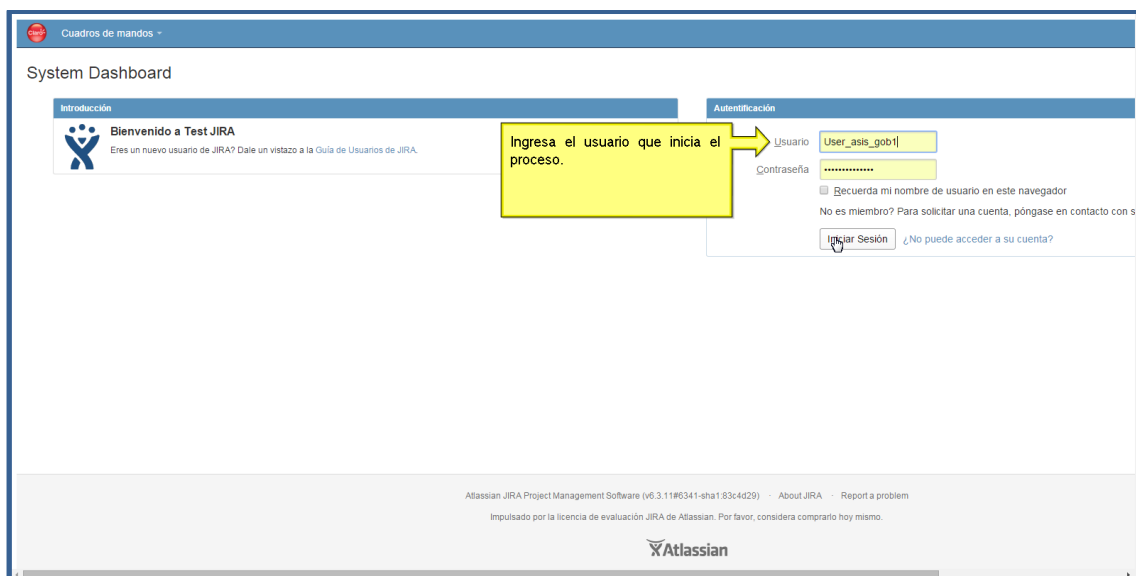


Figura 2.3. Gráfico de barras horizontales – Peticiones por estado

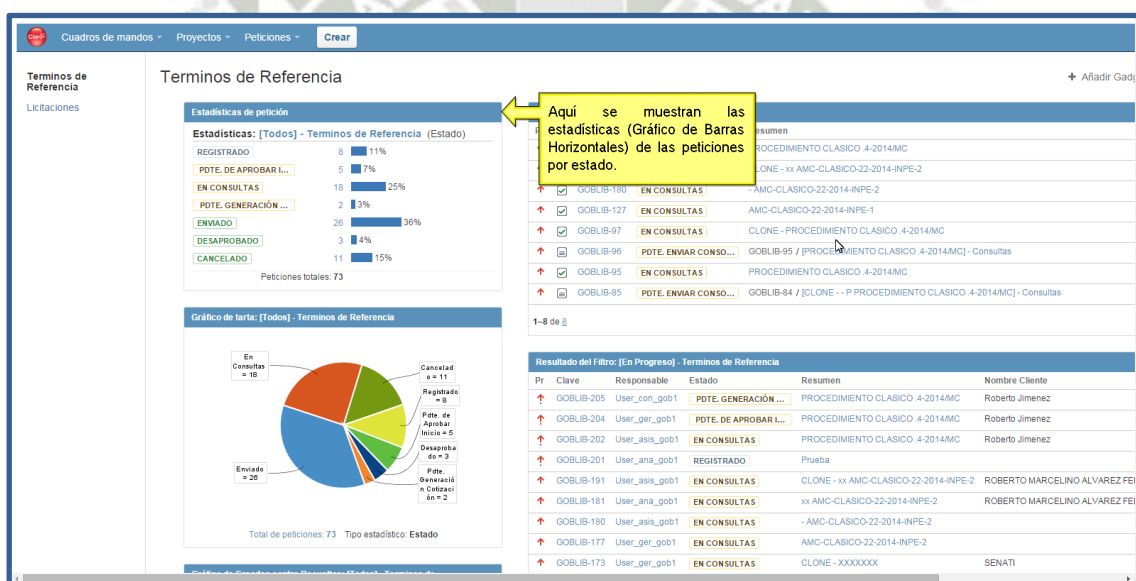


Figura 2.4. Grafico circular – Peticiones por estado

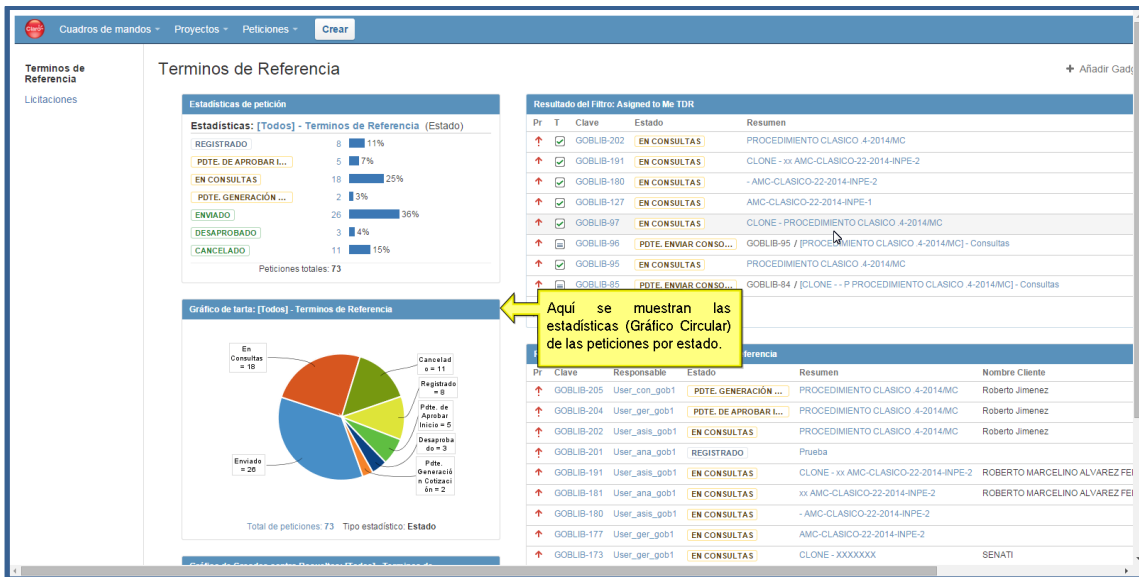


Figura 2.5. Peticiones que tengo asignadas

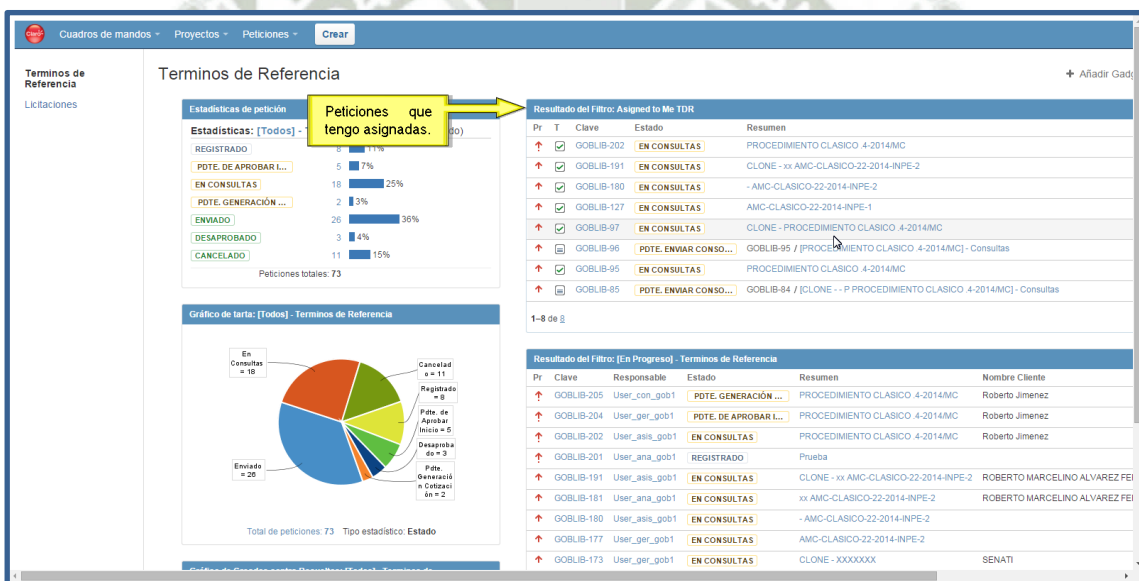


Figura 2.6. Peticiones en progreso

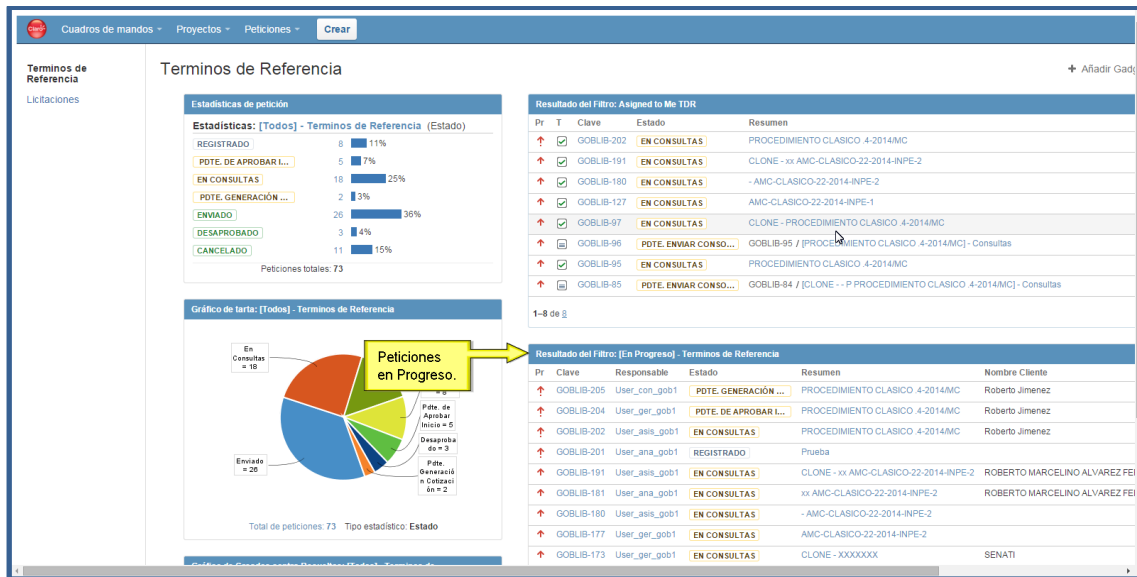


Figura 2.7. Gráfico de Peticiones Creadas contra las Peticiones Resueltas

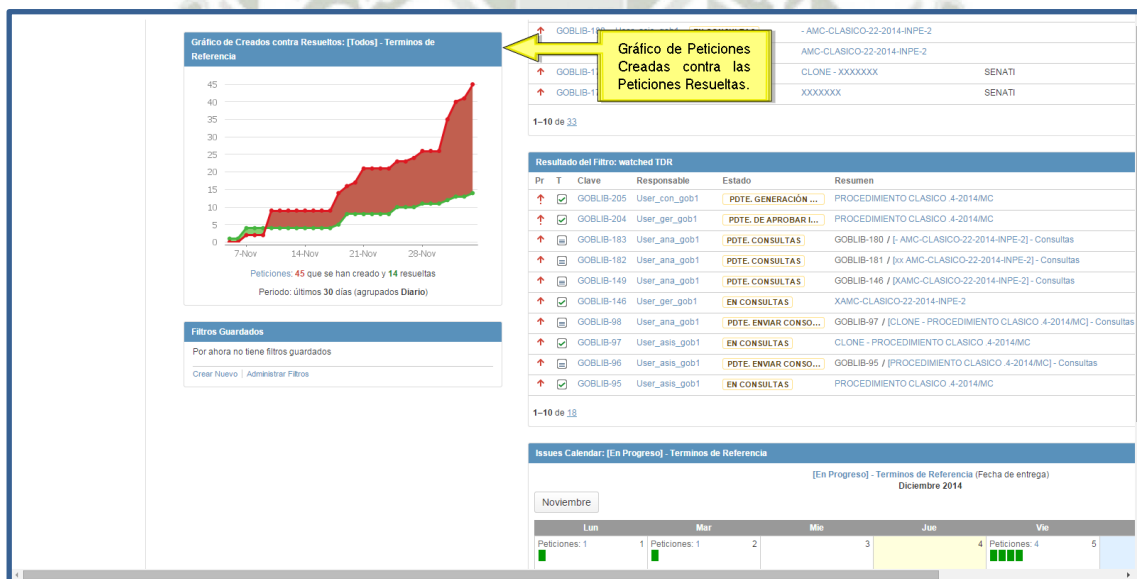


Figura 2.8. Filtros creados por el usuario

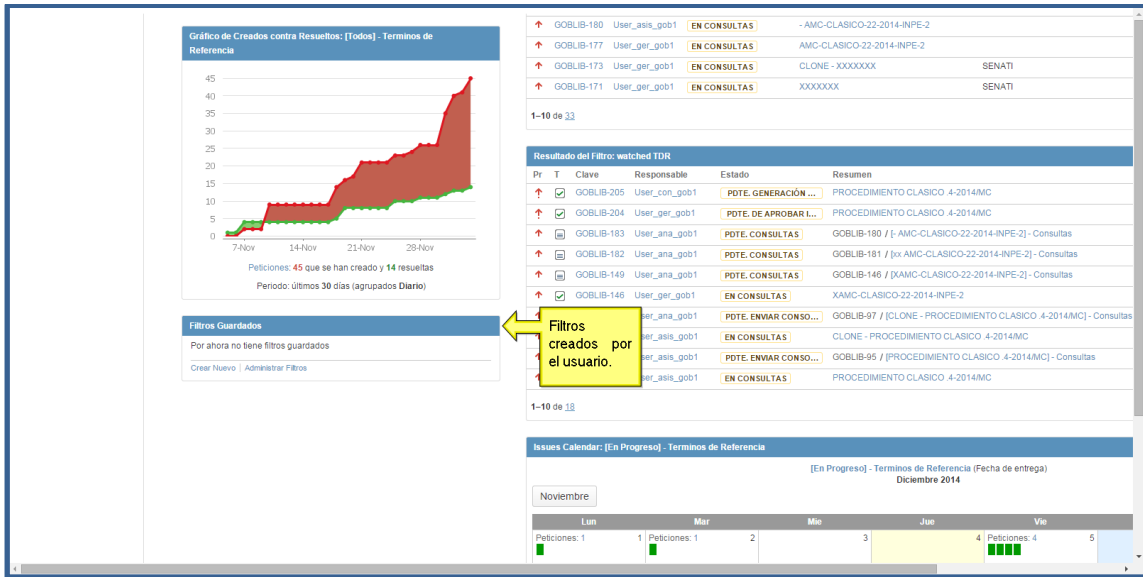


Figura 2.9. Peticiones en las que me encuentro como observador

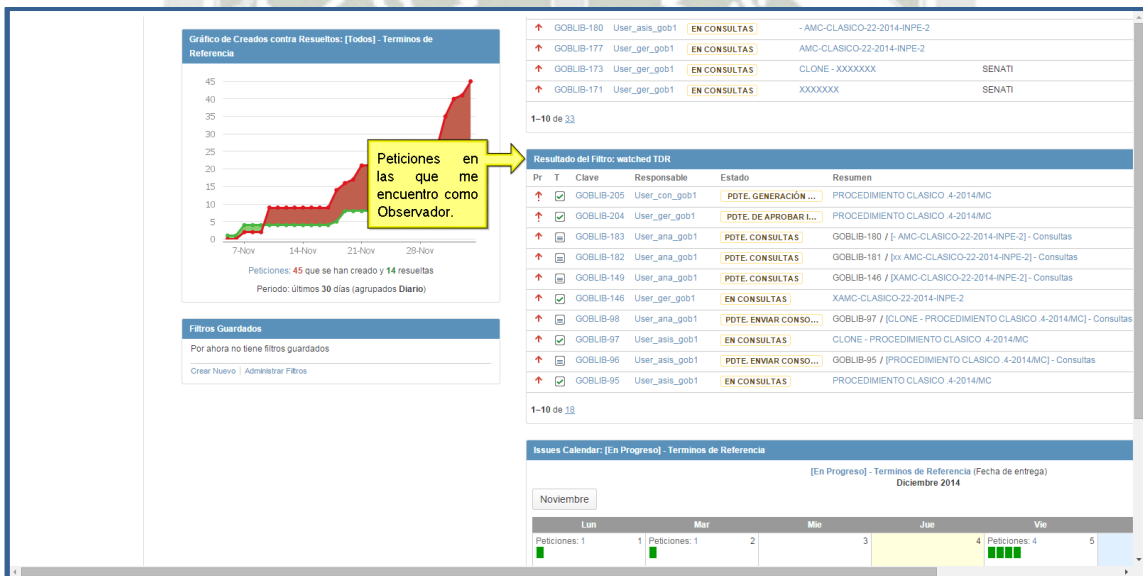


Figura 2.10. Peticiones en progreso según calendario

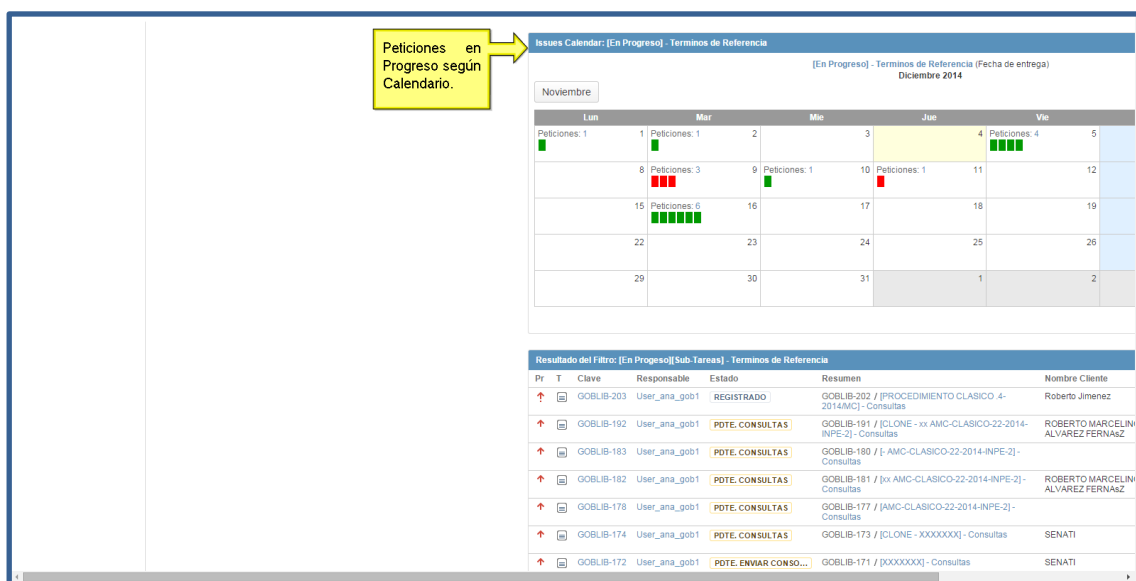


Figura 2.11. Sub - Tareas de las peticiones en progreso

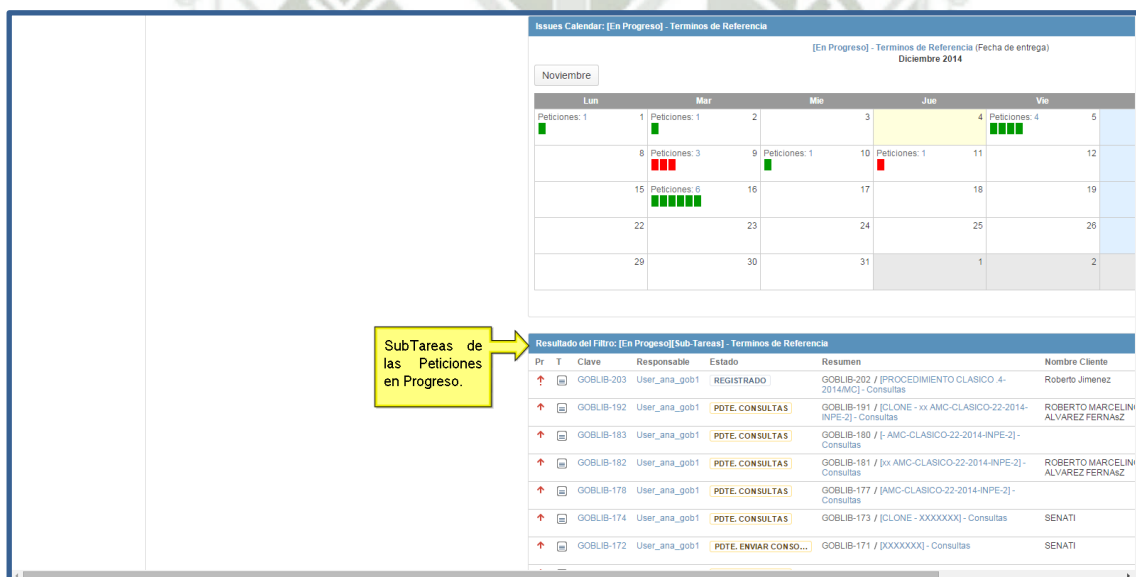


Figura 2.12. Sub-tareas en progreso según calendario

Issues Calendar: [En Progreso][Sub-Tareas] - Terminos de Referencia

[En Progreso][Sub-Tareas] - Terminos de Referencia (Fecha de entrega)  
Diciembre 2014

Noviembre

Lun	Mar	Mie	Jue	Vie
Peticiones: 15 1	Peticiones: 3 2	Peticiones: 4 3		
8	Peticiones: 1 9	10	11	12
15	Peticiones: 3 16	17	18	19
22	23	24	25	26
29	30	31	1	2

Atlassian JIRA Project Management Software (v6.3.11#6341-sha1:83c4d29) - About JIRA - Report a problem  
Impulsado por la licencia de evaluación JIRA de Atlassian. Por favor, considera comprarlo hoy mismo.

Atlassian

## INFORME DE SALUD

INFORME DE SALUD DEL PROYECTO

PROYECTO

Gestión y Desarrollo de Proyectos de Software

## INFORME DE SALUD DEL PROYECTO

### Agenda


- Tablero de Control
- Riesgos y Problemas
- Actividades Desarrolladas
- Tareas Pendientes
- Próximos Pasos


Gestión y Desarrollo de Proyectos de Software


## INFORME DE SALUD DEL PROYECTO

### Tablero de Control

Proyecto	F. Planificada Término	F. Estimada Término	% Avance Real	% Avance Planificado	Estado
Automatización del tablero de control de licitaciones	02/01/2015	16/02/2015	54%	75%	

 Riesgo **BAJO** de retrasos en cronograma y rebase de presupuesto

 Riesgo **MEDIO** de retrasos en cronograma y rebase de presupuesto

 Riesgo **ALTO** de retrasos en cronograma y rebase de presupuesto

Gestión y Desarrollo de Proyectos de Software

## INFORME DE SALUD DEL PROYECTO

### Riesgos y Problemas

Disponibilidad del ambiente de Calidad en las fechas esperadas para la implementación del proyecto.

Disponibilidad del ambiente de Producción en las fechas esperadas para la implementación del proyecto.

FEP: Fecha Planificada Término

FEE: Fecha Estimada Término

Gestión y Desarrollo de Proyectos de Software

## INFORME DE SALUD DEL PROYECTO

### Actividades Desarrolladas

Se configuro ambiente de desarrollo.

Se elaboro el plan de pruebas, manual de usuario, manual de instalación y operaciones.

Gestión y Desarrollo de Proyectos de Software

## INFORME DE SALUD DEL PROYECTO

### Tareas Pendientes

Pruebas Integrales en el ambiente de desarrollo.

Gestión de Solicitud de Servidores para Producción por el Equipo de soporte de aplicaciones.

Gestión y Desarrollo de Proyectos de Software

## INFORME DE SALUD DEL PROYECTO

### Próximos pasos

Revisión de formatos con el Área de Soporte.

Revisión de formatos con el Área de Calidad.

Gestión y Desarrollo de Proyectos de Software