



FACULTAD DE POSTGRADO

TESIS DE POSTGRADO

**GESTIÓN DE GOBERNABILIDAD DE DESARROLLO DE
SOFTWARE EN BANCO DE AMERICA CENTRAL 2015**

SUSTENTADO POR:

**EMY GRISSOL LÓPEZ ALY
SHIRLEY ARACELY VALLE MÉNDEZ**

**PREVIA INVESTIDURA AL TÍTULO DE
MÁSTER EN GESTIÓN DE TECNOLOGÍA DE LA
INFORMACIÓN**

TEGUCIGALPA, FRANCISCO MORAZÁN, HONDURAS, C.A.

MARZO 2015

**UNIVERSIDAD TECNOLÓGICA DE HONDURAS
UNITEC**

**FACULTAD DE POSTGRADO
AUTORIDADES UNIVERSITARIAS**

**RECTOR
LUIS ORLANDO ZELAYA MEDRANO**

**SECRETARIO GENERAL
ROGER MARTÍNEZ**

**VICERRECTOR ACADÉMICO
MARLON BRÉVE REYES**

**VICERRECTORA CAMPUS SPS
ANA LOURDES LAFFITE**

DECANO DE LA FACULTAD DE POSTGRADO

**GESTIÓN DE GOBERNABILIDAD DE DESARROLLO DE
SOFTWARE EN BANCO DE AMERICA CENTRAL 2015**

**TRABAJO PRESENTADO EN CUMPLIMIENTO DE LOS
REQUISITOS EXIGIDOS PARA OPTAR AL TÍTULO DE
MÁSTER EN**

GESTIÓN DE TECNOLOGÍA DE LA INFORMACIÓN

ASESOR METODOLÓGICO

CINTHIA ISELA CANO ACOSTA

ASESOR TEMÁTICO

OSCAR GILBERTO ZOCÓN ALVA

MIEMBROS DE LA TERNA:

IDALIA CÁRCAMO

HÉCTOR BERRIOS

CARLOS ZELAYA OVIEDO



FACULTAD DE POSTGRADO
GESTIÓN DE GOBERNABILIDAD DE DESARROLLO DE SOFTWARE EN BANCO
DE AMERICA CENTRAL 2015

AUTORES:

Emy Grissol López Aly y Shirley Aracely Valle Méndez

RESUMEN

En el Banco de América Central de Honduras durante el año 2014 se realizó la investigación donde el enfoque principal se centra en la falta de gobernabilidad en desarrollos de software, la programación no está centralizada solamente en el área de sistemas de TI sino que existen desarrollos de software realizados por programadores de las áreas de negocio y programadores subcontratados que son utilizados por usuarios de todas las áreas de negocio del banco, por lo anterior el objetivo del trabajo es identificar soluciones para eliminar la falta de gobierno y falta de seguimiento de una metodología de los programadores de negocio dado que representa riesgos operativos y monetarios para el banco, se aplicaron encuestas y entrevistas a la población involucrada así mismo se utiliza una investigación descriptiva y explicativa, en base a los resultados obtenidos por medio del cruce de variable podemos rechazar la hipótesis nula que indica que no es necesaria una metodología ágil de desarrollo acoplada con un gobierno de TI para la gestión de los proyectos de desarrollo de software de las áreas de negocio, de acuerdo a las características de los proyectos y entorno del banco se recomienda aplicar la metodología ágil Crystal clear y acoplarla con un gobierno de TI en hardware y gestiones relacionadas con el software que sean parte de la metodología.

Palabras claves: Gobierno TI, metodología de desarrollo ágil, programadores de negocio, proyectos de desarrollo de software, programadores externos.



GOVERNANCE MANAGEMENT SOFTWARE DEVELOPMENT BANK CENTRAL AMERICA 2015

BY:

Emy Grissol López Aly y Shirley Aracely Valle Méndez

ABSTRACT

In the Bank of Honduras Central America during the 2014 research where the main focus is the lack of governance in software development is performed, the programming is not centered only in the area of IT systems but there are developments software made by programmers of the business areas and contract programmers who are users of all business areas of the bank used by the above, the aim of this work is to identify solutions to eliminate the lack of governance and lack of monitoring methodology Programmers business as it represents operating and monetary risks to the bank, surveys and interviews with the population were applied likewise a descriptive and explanatory research is used, based on the results obtained by crossing variable can reject the hypothesis null indicating that an agile development methodology coupled with IT governance for managing software development projects in the areas of business is not necessary, according to the characteristics of the projects and the bank environment recommend applying Crystal clear agile methodology and couple it with a government IT hardware and software-related efforts that are part of the methodology.

Keywords: Government IT, agile development methodology, programmers, business software development projects, contract programmers.

DEDICATORIA

Emy López: A mis padres Joel López López y María Antonieta Aly porque desde el inicio de estudios de la maestría en Gestión de Tecnologías de Información hasta las fases finales me brindaron el apoyo incondicional de siempre, sin su cariño y comprensión nunca lo hubiera logrado, es un esfuerzo de conjunto, los quiero.

Shirley Valle: quiero dedicar este trabajo primeramente a Dios por darme la sabiduría y salud para luchar y alcanzar mis metas de vida, en segundo lugar a mis amados padres Salomón Ruiz, Sonia Méndez y Cleotilde Contreras por el apoyo incondicional brindado durante cada fase de mi vida, por formar mi carácter inculcando valores éticos, morales y espirituales, por educarme con su sabiduría y experiencia para ser cada día mejor persona e impulsarme a crecer como ser humano y profesional.

AGRADECIMIENTO

Emy López: A Dios por guardar de mí y darme las fuerzas necesarias para lograr llegar al final de este proceso. Al personal del banco que proporcionó la información requerida y participó en las encuestas y entrevistas para la elaboración de este trabajo, entre ellos se encuentran los programadores, el gestor de gobierno, los líderes de proyectos y los ejecutivos de negocio que dedicaron ese valioso tiempo así mismo agradecer en gran manera a nuestro asesor de tesis temático Oscar Zocón por ayudarnos a mejorar nuestro trabajo. A mis mejores maestros de estudios por inspirarme a superarme académicamente y por todas las lecciones de vida y académicas aprendidas.

Shirley Valle: agradezco primeramente a Dios por su guía, a mis amados padres por el apoyo durante todo el tiempo durante el cual curse mis estudios, mis hermanos por su apoyo y paciencia cuando he necesitado de su ayuda y comprensión, mis amigos y colegas por su apoyo y colaboración durante la ejecución y culminación de mi maestría mediante sus consejos y enseñanzas, a la institución Financiera Banco de América Central de Honduras y los compañeros de trabajo que nos brindaron la información y datos para poder realizar nuestro trabajo de investigación y por último a la familia en general por darme el ánimo y hacer crecer en mí un deseo de superación y sus muestras de cariño y buenos deseos.

Tabla de contenido

CAPÍTULO I. PLANTEAMIENTO DE LA INVESTIGACIÓN	1
1.1 INTRODUCCIÓN.....	1
1.2 ANTECEDENTES.....	2
1.2.1 ANTECEDENTES GLOBALES DEL GOBIERNO DE TI	2
1.2.2 ANTECEDENTES DEL GOBIERNO DE DESARROLLO DE SOFTWARE EN BAC HONDURAS	3
1.3 DEFINICIÓN DEL PROBLEMA	8
1.3.1 ENUNCIADO DEL PROBLEMA.....	8
1.3.2 FORMULACIÓN DEL PROBLEMA	9
1.3.3 PREGUNTAS DE INVESTIGACIÓN	9
1.4 OBJETIVOS DEL PROYECTO	9
1.4.1 OBJETIVO GENERAL	9
1.4.2 OBJETIVOS ESPECÍFICOS.....	10
1.5 JUSTIFICACIÓN	10
CAPÍTULO II. MARCO TEÓRICO.....	13
2.1 ANÁLISIS DE LA SITUACIÓN ACTUAL	13
2.1.1 ANÁLISIS DEL MACRO-ENTORNO.....	13
2.1.1.1 GOVERNABILIDAD EN DESARROLLO DE SOFTWARE.....	13
2.1.1.2 MODELOS DE DESARROLLO DE SOFTWARE DENTRO DE GOBIERNO DE TI.....	14
2.1.2 ANÁLISIS DEL MICRO-ENTORNO	16
2.1.2.1 GOVERNABILIDAD EN DESARROLLO DE SOFTWARE EN AMÉRICA CENTRAL	16
2.1.2.2 CRECIMIENTO DEL DESARROLLO DE SOFTWARE EN AMÉRICA CENTRAL .	17
2.1.2.3 GOBIERNO DE TI EN HONDURAS	18
2.1.3 ANÁLISIS INTERNO.....	20
2.2 TEORÍAS DE SUSTENTO	23
2.2.1 GOBIERNO DE TI	23
2.2.2 MARCO DE REFERENCIA COBIT	25
2.2.3 MODELO CMMI PARA SOFTWARE (MODELO DE MADUREZ DE LA CAPACIDAD INTEGRADO).....	28
2.2.4 NORMA ISO/IEC 12207	31
2.2.5 METODOLOGÍAS ÁGILES DE DESARROLLO DE SOFTWARE.....	33

METODOLOGÍA SCRUM	33
PROGRAMACIÓN EXTREMA XP	34
METODOLOGÍA CRYSTAL CLEAR	35
2.3 MARCO CONCEPTUAL	39
2.4 MARCO REFERENCIAL	41
2.4.1 IMPLEMENTANDO COBIT Y GOBIERNO DE TI EN CORPORACIÓN GPF (ECUADOR)	41
2.4.2 APLICACIÓN DE SCRUM EN UN PROYECTO DE SOFTWARE UNIVERSIDAD DE MONTEMORELOS DE MÉXICO	43
2.4.3 PROYECTO DE MEJORA CMMI, UN CASO DE ÉXITO EN ARGENTINA	45
CAPÍTULO III. METODOLOGÍA	47
3.1 CONGRUENCIA METODOLÓGICA.....	47
3.1.2 RELACIÓN DE LAS VARIABLES	48
3.1.3 HIPÓTESIS.....	55
3.2 ENFOQUE Y MÉTODOS	55
3.3 DISEÑO DE LA INVESTIGACIÓN.....	57
3.3.1 POBLACIÓN Y MUESTRA	58
3.3.2 UNIDAD DE ANÁLISIS.....	58
3.3.3 UNIDAD DE RESPUESTA.....	59
3.4 TÉCNICAS E INSTRUMENTOS APLICADOS	60
3.4.1 INSTRUMENTOS	60
3.4.2 TÉCNICAS.....	60
3.4.3 PROCEDIMIENTOS.....	61
3.5 FUENTES DE INFORMACIÓN	64
3.5.1 FUENTES PRIMARIAS	64
3.5.2 FUENTES SECUNDARIAS	64
CAPÍTULO IV. RESULTADOS Y ANÁLISIS	65
4.1 RESULTADOS ENCUESTAS.....	65
4.1.1 VARIABLE RESPONSABILIDADES EN EL FLUJO DE PROCESO	65
4.1.2 VARIABLE METODOLOGÍA ÁGIL DE DESARROLLO DE SOFTWARE	67
4.1.3 VARIABLE PROYECTOS DE SOFTWARE	83
4.1.4 VARIABLE PROCESO DE GOBIERNO DE TI.....	84

4.1.5 VARIABLE FALLAS EN EL PROCESO ACTUAL DE DESARROLLO	88
4.2 RESULTADOS ENTREVISTAS	97
ENTREVISTA PARA EJECUTIVOS DE NEGOCIOS QUE CONTRATAN PERSONAL EXTERNO A TI PARA DESARROLLO DE SOFTWARE.....	97
ENTREVISTA AL GESTOR DE GOBIERNO DE DESARROLLOS DE SOFTWARE DE TI	100
4.3 COMPROBACIÓN DE LA HIPÓTESIS.....	105
CAPÍTULO V. CONCLUSIONES Y RECOMENDACIONES	111
5.1 CONCLUSIONES.....	111
5.2 RECOMENDACIONES.....	112
CAPÍTULO VI. APLICABILIDAD.....	114
6.1 PLAN DE ACCIÓN PARA LA GESTIÓN DE GOBIERNO DE SOFTWARE PARA PROGRAMADORES DE NEGOCIO.....	114
6.2 INTRODUCCCION	114
6.3 METODOLOGÍA ÁGIL CRYSTAL CLEAR	115
6.3.1 CICLOS DEL PROCESO DE DESARROLLO A APLICAR	115
6.3.2 ESTABLECIMIENTO DE ROLES DENTRO DEL PROYECTO, FASE DE CONVENIO	116
6.3.3 ESTRATEGIAS Y TÉCNICAS CRYSTAL CLEAR A UTILIZAR EN PROYECTOS DE NEGOCIO DEL BANCO.....	117
6.3.3.1 ARTEFACTO DE REQUERIMIENTOS	117
6.3.3.2 MAPA DEL PROYECTO.....	119
6.3.3.3 ARQUITECTURA DEL PROYECTO.....	119
6.3.3.4 SEGUIMIENTO Y COMUNICACIÓN DEL PROYECTO.....	121
6.3.3.5 ACTIVIDADES DEL PROCESO DE IMPLEMENTACIÓN DE METODOLOGÍA ÁGIL	126
6.4 ACOPLAR METODOLOGÍA ÁGIL CON GOBIERNO DE TI UTILIZANDO COBIT 5.....	127
6.4.1 PRINCIPIOS Y POLITICAS	127
6.4.1.1 CONTROL DE SOFTWARE.....	127
6.4.1.2 CONTROL DE HARDWARE	130
6.4.2 ESTRUCTURAS ORGANIZATIVAS.....	133
6.4.3 CULTURA, ÉTICA Y COMPORTAMIENTO.....	133
6.4.4 INFORMACIÓN.....	135

6.4.5	SERVICIOS, INFRAESTRUCUTRA, Y APLICACIONES	136
6.4.6	EVALUAR, ORIENTAR Y SUPERVISAR DE COBIT	138
6.4.6.1	ASEGURAR EL ESTABLECIMIENTO Y MANTENIMIENTO DEL MARCO DE GOBIERNO	138
6.4.6.2	ASEGURAR LA ENTREGA DE BENEFICIOS	139
6.4.6.3	ASEGURAR LA OPTIMIZACIÓN DEL RIESGO	140
6.4.6.4	ASEGURAR LA OPTIMIZACIÓN DE LOS RECURSOS	143
6.4.6.5	ASEGURAR LA TRANSPARENCIA HACIA LAS PARTES INTERESADAS ...	144
6.4.7	ACTIVIDADES PARA MIGRACIÓN DE PROYECTOS EXISTENTES DE SOFTWARE	145
6.5	DIAGRAMA DE PROCESOS.....	148
6.6	CRONOGRAMA DE EJECUCIÓN.....	150
6.7	PRESUPUESTO.....	151
BIBLIOGRAFÍA	153

INDICE DE TABLAS

Tabla 1	Inventario de desarrollo de Software más utilizados realizados fuera del gobierno de TI 4	
Tabla 2	Procesos utilizados generalmente en la Norma ISO/IEC 12207 se clasifican en 3 tipos31	
Tabla 3	Comparación de las características de metodologías ágiles con metodologías tradicionales	38
Tabla 4	Matriz Metodológica.....	47
Tabla 5	Operacionalización de las variables.....	49
Tabla 6	Tipos de recursos involucrados en un proyecto.....	58
Tabla 7	Frecuencia proyectos asignados actualmente	65
Tabla 8	Frecuencia prioridad de los proyectos para programadores de negocio	66
Tabla 9	Frecuencia aspectos a mejorar en metodologías utilizadas por programadores de negocio.....	67
Tabla 10	Requerimientos nuevos una vez finalizado el proyecto, programadores de negocio... 69	
Tabla 11	Frecuencia respecto al reporte de avances del proyecto, usuario.....	70
Tabla 12	Frecuencia respecto de avances del proyecto.....	71
Tabla 13	Frecuencia tipo de metodología seleccionada por programadores de negocio o programadores externos	72
Tabla 14	Frecuencia ambiente de pruebas utilizado por programadores de negocio.....	73
Tabla 15	Frecuencia tiempo invertido en fases del proceso de desarrollo programadores de negocio o programadores externos.....	75

Tabla 16 Frecuencia tiempo invertido en fases del proceso de desarrollo programadores de negocio o programadores externos.....	76
Tabla 17 Frecuencia forma de trabajo programadores de negocio.....	78
Tabla 18 Frecuencia elementos que disminuyen costos para programadores externos.....	79
Tabla 19 Frecuencia grado de complejidad de los proyectos del banco.....	81
Tabla 20 Frecuencia grado de interacción con el usuario final en proyectos.....	82
Tabla 21 Frecuencia cumplimiento de tiempo de finalización de los proyectos.....	83
Tabla 22 Frecuencia tiempo invertido en la recolección de requerimientos	84
Tabla 23 Frecuencias satisfacción en las pruebas del proyecto	85
Tabla 24 Frecuencias cumplimiento en la fechas de cierre o entregables del proyecto.....	86
Tabla 25 Frecuencias uso de documentos de plan de proyecto	87
Tabla 26 Frecuencia de las causas de fallas en proyectos de software, programadores de negocio.....	88
Tabla 27 Frecuencia de las causas de fallas en proyectos de software, programadores de TI..	89
Tabla 28 Frecuencias fallas o ajustes realizados en los desarrollos de software.....	91
Tabla 29 Frecuencias calidad de los proyectos, usuarios de negocio.....	92
Tabla 30 Frecuencias Fallas en producción, usuarios de negocio.....	93
Tabla 31 Frecuencias mantenimiento a lo largo de ciclo de vida del aplicativo, usuarios de negocio.....	94
Tabla 32 Frecuencias solicitud de desarrollo de aplicaciones	95
Tabla 33 Frecuencias nivel de riesgo tecnológico asociados a los desarrollos de programadores de TI.....	96
Tabla 34 Análisis FODA de la entrevista a gestor de gobierno TI y ejecutivos de negocio	104
Tabla 35 Prueba de Chi cuadrado	105
Tabla 36 Hoja de tareas de un proyecto de desarrollo de software	119
Tabla 37 Objetos modificados en el proyecto	120
Tabla 38 Resultados y mejoras taller de reflexión proyectos de software	126
Tabla 39 Ejemplo formato de especificación de requerimientos	131
Tabla 40 Ejemplo formato de solicitud de cambio (RFC).....	132
Tabla 41 Ejemplo de un formato Plan de acción.....	134
Tabla 42 Asegurar los beneficios de la aplicación de metodología y gobierno de TI.....	139
Tabla 43 Periodicidad de la gestión de riesgos	141
Tabla 44 Identificación, Evaluación Cualitativa y Respuesta a riesgos.....	142
Tabla 45 Documento para Reporte de Error	145
Tabla 46 Presupuesto de plan de acción	151

INDICE DE FIGURAS

Figura 1 Inventario de aplicaciones y bases de datos críticas identificadas a la fecha en las áreas de Negocio.....	7
Figura 2 Gasto global en software y gasto en distribución de Tecnologías de Información y comunicaciones	15

Figura 3 Necesidades identificadas para ejercer Gobierno de TI en desarrollos de software.....	22
Figura 4 Modelo del Gobierno Corporativo de TI	24
Figura 5 Diagrama de las variables	49
Figura 6 Resultados proyectos asignados programadores de negocio y programadores externos	65
Figura 7 Resultados prioridad de los proyectos programadores de negocio.....	66
Figura 8 Resultado aspectos a mejorar en metodologías utilizadas por programadores de negocio.....	68
Figura 9 Resultado requerimientos nuevos una vez finalizado el proyecto, programadores de negocio.....	69
Figura 10 Resultado usuario de negocio respecto al reporte de avances del proyecto	70
Figura 11 Resultado usuario de negocio respecto al reporte de avances del proyecto	71
Figura 12 Resultado tipo de metodología seleccionada por programadores de negocio o programadores externos	72
Figura 13 Resultado ambiente de pruebas utilizado por programadores de negocio.....	74
Figura 14 Resultado tiempo invertido en fases del proceso de desarrollo programadores de negocio o programadores externos.....	75
Figura 15 Resultado elementos de enfoque dentro de un proyecto de software, programadores de negocio	77
Figura 16 Resultado forma de trabajo programadores de negocio	78
Figura 17 Resultado elementos que disminuyen costos para programadores externos	80
Figura 18 Resultado grado de complejidad de los proyectos del banco.....	81
Figura 19 Resultado grado de interacción con el usuario final en Proyectos.....	82
Figura 20 Resultado cumplimiento de tiempo de finalización de los proyectos.....	83
Figura 21 Resultado tiempo invertido en la recolección de requerimientos.....	84
Figura 22 Resultado satisfacción en las pruebas del proyecto.....	85
Figura 23 Resultado cumplimiento en la fechas de cierre o entregables del proyecto.....	86
Figura 24 Resultado uso de documentos de plan de proyecto.....	87
Figura 25 Resultado de las causas de fallas en proyectos de software, programadores de negocio.....	89
Figura 26 Resultado de las causas de fallas en proyecto de software, programadores TI.....	90
Figura 27 Resultado fallas o ajustes realizados en los desarrollos de software	91
Figura 28 Resultado calidad de los proyectos, usuarios de negocio	92
Figura 29 Resultado Fallas en producción, usuarios de negocio	93
Figura 30 Resultado mantenimiento a lo largo de ciclo de vida del aplicativo, usuarios de negocio.....	94
Figura 31 Resultado solicitud de desarrollo de aplicaciones.....	95
Figura 32 Resultado nivel de riesgo tecnológico asociados a los desarrollos de programadores TI	96
Figura 33 Cruce de variables metodologías y proceso de desarrollo de software	105
Figura 34 Cruce de variables metodologías y forma de trabajo a mejorar	106
Figura 35 Cruce de variables fallas en el proceso de desarrollo y proceso de gobierno de TI .	107

Figura 36 Cruce de variables tiempo de mantenimiento gestionado para las herramientas de software y pruebas previas realizadas en la puesta en producción	108
Figura 37 Cruce de variables metodologías que dan mayor valor a un proyecto y proceso de desarrollo en el cual se enfocan más los programadores.....	109
Figura 38 Cruce de variables elementos más relevantes en el desarrollo de un software y grado de complejidad en los proyectos del banco	110
Figura 39 Ciclos de Crystal clear	115
Figura 40 Artefacto de requerimientos de proyecto	118
Figura 41 Pizarra ó Lámina a utilizar para radiador de información.....	123
Figura 42 Gráfico de quemado de un proyecto de software.....	124
Figura 43 Gráfico de quemado Estimación trabajo pendiente en un proyecto de software	125
Figura 44 Diagrama Implementación Plan de acción gobierno TI.....	148
Figura 45 Diagrama Control Hardware y Software	149
Figura 46 Cronograma del Plan de acción.....	150

INDICE DE ANEXOS

Anexo 1 ENCUESTA DE SATISFACCIÓN AL USUARIO DE PROYECTOS FUERA DE GOBIERNO DE TI.....	156
Anexo 2 ENCUESTA PARA PROGRAMADORES DE DESARROLLO SOFTWARE QUE TRABAJAN CON LAS ÁREAS DE NEGOCIO.....	159
Anexo 3 ENCUESTA PARA PROGRAMADORES DE DESARROLLO SOFTWARE DEL ÁREA DE TI.....	163
Anexo 4 ENTREVISTA PARA EJECUTIVOS DE NEGOCIOS QUE CONTRATAN PERSONAL EXTERNO A TI PARA DESARROLLO DE SOFTWARE.....	166
Anexo 5 ENTREVISTA AL GESTOR DE GOBIERNO DE DESARROLLOS DE SOFTWARE DE TI.....	169
Anexo 6 DIAGRAMA DE ISHIKAWA DEL ENFOQUE DEL PROBLEMA.....	172
Anexo 7 MATRIZ DE ALINEACIÓN.....	173
Anexo 8 CARTA DE COMPROMISO PARA ASESORÍA TEMÁTICA.....	174

CAPÍTULO I. PLANTEAMIENTO DE LA INVESTIGACIÓN

1.1 INTRODUCCIÓN

En el trabajo de tesis “Gestión de gobernabilidad de desarrollo de software en Bac Honduras 2014” se estudian elementos como la satisfacción del usuario, metodologías, control de objetos, accesos a las aplicaciones y actividades en el proceso de desarrollo y mantenimiento de sistemas. El enfoque principal se centra en la falta de gobernabilidad en desarrollos de software, la programación no está centralizada solamente en el área de sistemas de TI, existen desarrollos de software realizados por programadores de las áreas de negocio y programadores externos, dichos desarrollos son utilizados por usuarios de todas las áreas de negocio del banco, por lo que se necesita identificar los riesgos que este tipo de desarrollos representa para el banco.

En el trabajo se investiga cuál es la metodología de software que mejor se adapta al banco de acuerdo a las consultas realizadas a ejecutivos del banco, programadores de TI, programadores de negocio y gestor de gobierno para conocer el ambiente, tipo de proyectos que se manejan, metodologías utilizadas por los diferentes programadores que no pertenecen a la gerencia de sistemas, fases del proceso en la que dedica más tiempo, satisfacción de los usuarios en el mantenimiento y funcionalidad de las aplicaciones, conocer cuáles son las necesidades de negocio y porque estas buscan soluciones por sí mismas sin contar con la asistencia de la gerencia de TI.

En la investigación realizada en Bac Honduras se pretende identificar el problema raíz, como se realizó dicho hallazgo, que ha hecho en la actualidad la organización tanto localmente como regionalmente para buscar una solución definitiva al problema y analizar la mejor propuesta que ayude a brindar una solución firme, estructurada y aplicable para las áreas de negocio que están siendo afectadas por la falta de un gobierno de TI que pueda aplicarse en el mismo proceso de desarrollo de software, ya que esto evitará sanciones o penalizaciones por entes regulatorios constituidos dentro y fuera de la organización.

El desarrollo de software por colaboradores ajenos a la gerencia de TI constituye un riesgo de seguridad, riesgo operativo y riesgo de tecnología en general no darle solución al problema puede traer consigo muchas repercusiones cuando las actividades y recursos no están bajo el control y administración exclusivo de la Gerencia de Tecnología de la Información bajo el concepto de gobierno de TI. Se necesita contribuir a un mejor gobierno de TI siendo que este está constituido también por el software y su forma de desarrollarlo, Bac Honduras es una empresa en donde existen mucho control, normas, procedimientos por lo que ningún flujo de trabajo debe quedar fuera de un proceso que ayude a obtener mejores resultados durante su ejecución.

1.2 ANTECEDENTES

1.2.1 ANTECEDENTES GLOBALES DEL GOBIERNO DE TI

El concepto de Gobierno de Tecnología de la Información (TI) tiene su origen en el año de 1993 como un derivado del concepto de gobierno corporativo, la finalidad del gobierno de TI es agrupar los recursos y administrarlos para alcanzar los objetivos estratégicos de la organización, significa que todos los riesgos de TI deben ser conocidos y gestionados, se ha utilizado para este tema el marco de referencia COBIT entre otros, mismo que ayuda a proveer una guía para alcanzar el alineamiento de TI con el negocio y trata de que TI haga el mejor uso posible de los recursos.

El software constituye servicios críticos y de valor para la empresa por lo que su interrupción puede derivar en pérdidas económicas y de tiempo, por ello ha sido importante para las empresas establecer un gobierno de TI en desarrollos de software este es la temática de este documento de tesis. A lo largo de los años algunas empresas han trabajado de forma desorganizada sin un proceso o metodología de trabajo ni metas específicas, por lo que los servicios de TI han presentado problemas para la empresas cuando dejan de existir la disponibilidad, confidencialidad e integridad de la información, por lo que se ha hecho necesario que las empresas cuenten con una metodología para proyectos de software, sin embargo no para todos los proyectos puede aplicar.

Los proyectos son diferentes y debería existir flexibilidad en el proceso para que todos los empleados puedan seguirlo. Lo anterior constituye parte del problema que las empresas de forma general tienen que gestionar cuando desarrollan software. Los problemas que se presentan para aplicar el gobierno TI de forma general (Boletín mensual de ámbito SAP e IT, 2010) pueden ser; mala gestión de proyectos, falta de un sistema de control de cambios, gestión de servicios inadecuada, falta de procesos de control y monitorización, métodos de software inadecuados, falta de alineamiento estratégico para las iniciativas de TI, teniendo riesgo de fracaso, por las actualizaciones de tecnología y la globalización del mercado que no permite desviarse de estos temas.

1.2.2 ANTECEDENTES DEL GOBIERNO DE DESARROLLO DE SOFTWARE EN BAC HONDURAS

Bac Honduras es una empresa en donde el proceso de desarrollo y mantenimiento de sistemas se encuentra estandarizado, cada cambio que se desee hacer a un objeto de software por requerimientos de negocio debe seguir este proceso por cada uno de los analistas del área de Desarrollo Banco y Tarjeta perteneciente a la Gerencia de Tecnología de la Información, sin embargo se identificó en el año 2011 la existencia de 97 aplicaciones en 11 áreas de negocio de Honduras que fueron desarrolladas por programadores que no pertenecen a la Gerencia de Tecnología de la información, motivo por el cual se ingresó un Producto no Conforme (PNC) que consiste en un documento electrónico ubicado en la herramienta de colaboración de la empresa Lotus.

Un PNC (producto no conforme) sirve para documentar la solución que se le dará a una falta de normativa, política, proceso o reglamento existente en la empresa que haya sido pasado por alto, en ella se justifica las razones y acciones a realizar para que no ocurra el mismo evento. La solución a este PNC fue poco significativa debido a la magnitud del problema, se observó que son muchas las aplicaciones desarrolladas internamente (áreas distintas a TI) y este hallazgo se identificó no solo en Honduras también se observó en otras áreas de negocio de los países de la región del grupo Bac Credomatic (Honduras, Guatemala, El Salvador, Nicaragua, Costa Rica, Panamá) las

cuales ejercen esta misma práctica, dicho hallazgo fue realizado mediante una auditoría de Bac Honduras.

Algunas de estas herramientas como por ejemplo: “Kiosco” fue elaborada por la gerencia de Recursos Humanos y sirve para información de recibos de planilla, cuenta de vacaciones e información relacionada con el puesto del empleado, donde todos los empleados de la empresa hoy en día utilizan esta herramienta y ni siquiera fue desarrollada por TI, otro ejemplo de estas aplicaciones es “Time to yes” elaborada por la gerencia de Crédito y Cobro de Honduras el cual consiste en un sistema de gestiones para préstamos ingresados por los oficiales de negocios.

Los desarrolladores de estos sistemas utilizan el lenguaje que mejor manejan para desarrollarla, los lenguajes base que utiliza la empresa son: RPG de IBM, Java de IBM, las bases de datos DB2 de IBM, SQLServer, además utilizan componentes ajenos a los que se encuentran con licencias autorizadas para Bac Honduras lo cual constituye una violación a la ley de derechos de autor que podría provocar sanciones para la institución por no utilizar herramientas autorizadas y con sus respectivas licencias.

A continuación se muestra la información referente a diferentes aplicaciones considerados importantes por su utilidad y por la cantidad de personas que lo usan y que han sido desarrollados sin la aprobación y autorización de la Gerencia de Tecnología de Bac Honduras, sin el seguimiento de ningún proceso:

Tabla 1 Inventario de desarrollo de Software más utilizados realizados fuera del gobierno de TI

Datos Generales de la aplicación	Gerencia	Lenguaje en que fue desarrollado	Tipo Base de datos	Tipo Aplicación
Nombre: Kiosco. Descripción: Muestra recibos de Pagos, vacaciones y beneficios de los colaboradores del banco. Concurrencia de usuarios: Todos los empleados activos.	Recursos Humanos	C# 2008	SqlServer 2008 R2	Web

Nombre: Sistema estadístico de créditos web para ventas. Descripción: Sistema Estadístico de Créditos web. Concurrencia de usuarios: Más de 100 empleados.	Crédito y Cobro	C#	MySQL, SQL Server, AS400B	Web
Nombre: Sistemas estadístico de créditos web para créditos. Descripción: Sistema Estadístico de Créditos web. Concurrencia de usuarios: Más de 100 empleados.	Crédito y Cobro	C#	MySQL, SQL Server, AS400B	Web
Nombre: Bac Móvil. Descripción: Precalificar clientes para ventas. Concurrencia de usuarios: Más de 100 empleados.	Crédito y Cobro	C#	MySQL, SQL Server, AS400B	Web
Nombre: Módulo de consulta para vendedores web. Descripción: Aplicación de consulta de ventas y devoluciones donde los usuarios (vendedores) pueden monitorear el movimiento de las solicitudes de tarjeta de crédito. Concurrencia de usuarios: 532	Ventas	Visual Basic 2010	SQL Server AS400	Web
Nombre: Sistema estadístico de ventas web. Descripción: Workflow de ventas, Estadísticas. Concurrencia de usuarios: 447	Ventas	Visual Basic 2010	SQL Server	Web
Nombre: Información de tarjetas. Descripción: brinda un detalle de todas las tarjetas de crédito que se ofrecen en el banco. Concurrencia de usuarios: 234	Mercadeo	Visual .net	SQL Server	Web
Nombre: MiniSiebel. Descripción: Simula cierta parte del funcionamiento de Siebel. Concurrencia de usuarios: 234	Mercadeo	Visual .net	SQL Server	Web
Nombre: Time to Yes. Descripción: Sistema de gestiones para préstamos ingresadas por los oficiales de negocios. Concurrencia de usuarios: 150	Crédito y Cobro	WPF Visual Studio 2008	SQL Server	Web
Nombre: Sistema principal de cuentas castigadas 2da versión. Descripción: Aplicación que maneja las cuentas 600, cuentas incobrables. Concurrencia de usuarios: 100	Crédito y Cobro	Asp.net, C#	SQL Server	Web
Nombre: Plataforma de cobros. Descripción: Aplicación que maneja las cuentas	Crédito y Cobro	Java	Postgresql	Web

600, cuentas incobrables. Concurrencia de usuarios: 100				
Nombre: Estrategias de Retención TSP/Asesor de productos de Financiamiento. Descripción: Ayuda a los oficiales a ofrecer productos cuando el cliente quiere bajar el límite o cancelar la tarjeta por el cobro que se realiza de la tasa de seguridad poblacional. Concurrencia de usuarios: 100	Mercadeo	C#	SQL Server 2008	Web

Fuente: (BAC Honduras, 2014)

En la Tabla 1 se muestra el detalle de las aplicaciones principales que existen dentro de la institución financiera y que fueron desarrolladas fuera de la Gerencia de TI, en ella se muestra algunas de las aplicaciones más importantes y que tienen una cantidad significativa de usuarios que trabajan a través de ellas, también se muestra información relevante como ser el tipo de lenguaje en las cuales fueron desarrollada, la bases de datos con las cuales trabaja, para que tipo de plataforma fue diseñada y a que gerencia dentro de la organización pertenece, sin embargo así como estos software existen más aplicaciones dentro de la empresa que deben ser investigadas e inventariadas.

A la fecha el área de Tecnología de la Información tiene identificado una parte de las aplicaciones de las áreas de Negocio y que han sido desarrolladas por programadores ajenos a los colaboradores que pertenecen a la Gerencia de TI, cabe mencionar que esta información la provee Bac Honduras, organización que brinda su apoyo con la información que se necesita para realizar la investigación y que es la principal interesada en practicar un buen gobierno de TI.

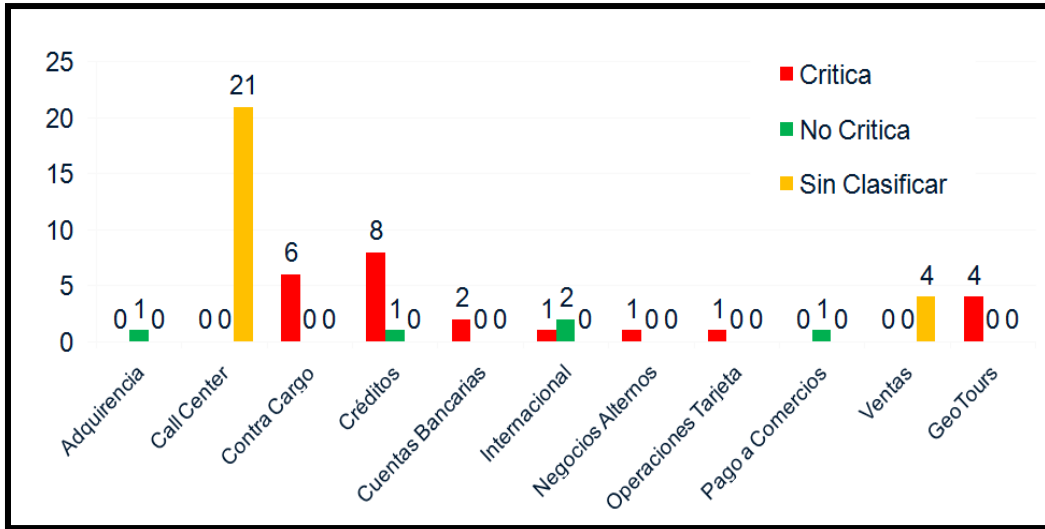


Figura 1 Inventario de aplicaciones y bases de datos críticas identificadas a la fecha en las áreas de Negocio

Fuente: (BAC Honduras, 2014)

Por medio de la Figura 1 proporcionada por el Bac Honduras, área de Desarrollo; se observa la cantidad de aplicaciones que se han desarrollado dentro de las gerencias de Negocios por los desarrolladores que pertenecen a cada una de ellas sin la aprobación de la Gerencia de Tecnología y sin seguir un proceso o metodología de desarrollo Estándar, en esta figura se muestra cuáles de las aplicaciones relacionadas son críticas, no críticas o aún no se ha identificado a qué grupo pertenece, esto para conocer cuáles de estos software tiene un mayor impacto dentro de las áreas de negocio y de la organización al momento de existir una falla .

La falta de control por parte de la Gerencia de Sistemas sobre aquellas aplicaciones y software que se desarrolla por parte de las gerencias de negocio constituye una serie de repercusiones por la falta de soporte que se pueda brindar por parte de TI hacia estos servicios de desarrollo, cuando existe codificación errónea y el desarrollador de la aplicación no puede dar asistencia técnica o algún tipo de mantenimiento, los especialistas técnicos de TI no pueden dar un soporte ya que carecen del conocimiento, accesos y herramientas para poder apoyar al negocio al momento que se presente la

falla, lo anterior lo pueden hacer también por medio de los artefactos que se dejarían en bitácora cuando se utilice una metodología para estos desarrollos de software.

Se ha identificado mediante hallazgos de auditoría y controles de la Ley Sarbanes Oxley (Sox) aplicados al área de Tecnología de la Gerencia de TI de forma mensual que esta falta de control puede provocar conflictos que afecten la seguridad de la información pues los servidores y las bases de datos utilizadas están fuera de un gobierno de seguridad perteneciente a la Gerencia de Tecnología que garantice que los datos sean íntegros, confiables y que estén disponibles en todo momento, además que falta la forma para garantizar que existe un mecanismo para asegurar que esta información se maneje de forma confidencial por parte de los programadores de las áreas de negocio lo que puede ocasionar sanciones por parte de entes regulatorios.

Al comenzar a utilizar aplicaciones externas a TI, auditoría interna del banco pudo identificar que existían este tipo de desarrollos en los cuáles ningún técnico de TI tenía nada que ver muchos menos un programador, el área de negocio inclusive cuenta con sus propios servidores, siendo así que no necesitaron recursos de infraestructura para llevar a cabo un proyecto de software, también se ha contratado personal externo para estos desarrollos como es el caso de la empresa SyTCorp que ofrece personal asignado dentro de la empresa para desarrollos y servicios de consultoría, este tipo de servicios ha tenido la desventaja que cuando se desea realizar un cambio o ejercer la escalabilidad de un objetos es necesario volverlos a contratar de nuevo, el soporte puede ser muy caro.

1.3 DEFINICIÓN DEL PROBLEMA

A continuación se presenta la definición del problema en el cual se basa el tema de tesis que será evaluada para verificar su aplicabilidad.

1.3.1 ENUNCIADO DEL PROBLEMA

El problema consiste en la existencia de aplicaciones de software que han sido analizadas, desarrolladas y probadas en ambientes de programación que no

pertenecen al área de desarrollo de sistemas de la Gerencia de Tecnología, careciendo de una metodología de desarrollo que contribuya a un mejor gobierno de TI. La gerencia de TI es precisamente la responsable de ejecutar las fases de un proyecto de software, carecer de los beneficios de seguir una metodología de desarrollo constituye un problema de desorganización, riesgo de pérdidas, falta de disponibilidad de las aplicaciones, riesgo de seguridad y de funcionalidad.

1.3.2 FORMULACIÓN DEL PROBLEMA

⇒ ¿Cuál es el procedimiento propuesto para adaptar una metodología de desarrollo que contribuya a un mejor gobierno de TI en desarrollos de software de BAC Honduras?

1.3.3 PREGUNTAS DE INVESTIGACIÓN

⇒ ¿Por qué las áreas de negocio de Bac Honduras recurren a desarrollar sus propias aplicaciones sin inclusión de TI?

⇒ ¿Qué pasos se deben realizar en cada etapa del desarrollo de software para erradicar los riesgos principales de la falta de gobernabilidad de TI?

⇒ ¿Qué parte del flujo de proceso a proponer debe ser seguido por los desarrolladores externos, que parte debe ser seguida por negocio y qué parte por la gerencia de TI?

⇒ ¿Cuál es la metodología de desarrollo de software ágil que podrían seguir los desarrolladores externos?

1.4 OBJETIVOS DEL PROYECTO

1.4.1 OBJETIVO GENERAL

Proponer a la gerencia de TI una metodología ágil de desarrollo de software en conjunto con un proceso que permita controlar todos los objetos de software, base de

datos e infraestructura de hardware que surjan de la metodología la cual será utilizada por los programadores de negocio en los proyectos que ellos lideren para lograr la calidad esperada en las áreas de negocio y mejorar la gestión de gobierno del banco.

1.4.2 OBJETIVOS ESPECÍFICOS

- ⇒ Identificar las causas del porqué las áreas de negocio desarrollan sin ayuda de la gerencia de TI, para definir prácticas de mejora en la gestión de gobernabilidad.
- ⇒ Establecer un procedimiento de gobierno para consolidar los objetos de software, base de datos e infraestructura de hardware a la gerencia de TI.
- ⇒ Determinar que fases del flujo de proceso serán realizadas por programadores de negocio, programadores externos y programadores de TI dentro del flujo de la metodología ágil propuesta.
- ⇒ Proponer la metodología de desarrollo ágil que mejor se adapte para cumplir con características de tiempo y calidad requeridas en un desarrollo de software para los programadores de negocio.

1.5 JUSTIFICACIÓN

Es necesario realizar esta investigación y formular una solución para mejorar la calidad, brindar una mayor seguridad y disponibilidad que debe poseer una aplicación ya sea desarrollada por colaboradores de Bac Honduras o por personal externo. Este tema adquiere relevancia porque la falta de solución puede incurrir en fraudes, sanciones y demás aspectos negativos para el banco como la mala gestión de tiempo de un proyecto de software, esto considerando que muchas aplicaciones son críticas para el negocio y que proporcionan valor para la organización, al necesitar un proceso de seguimiento de esas aplicaciones y un proceso del mantenimiento constituye un problema de gobierno de TI y un problema estratégico para la empresa en general.

Adicionalmente este tipo de desarrollos no garantizan la eficiencia y facilidad de uso, este es uno de los requisitos para catalogar un proyecto en la línea base formal de aplicativos del banco, para ello se deben realizar pruebas que midan el tiempo de respuestas y comparaciones contra la última versión anterior del objeto para conocer si el último cambio afectaría el rendimiento de la aplicación, este tipo de validaciones entre otras se pasan por alto cuando el desarrollo está fuera del alcance de TI.

Considerando que al momento de existir normas de programación, estándares de nomenclatura de base de datos, nombres de pantallas, programas, archivos, etc. dentro de la áreas de negocio que serán supervisadas por la Gerencia de TI se evitara un desorden a nivel operativo; también que en el momento en que un programador de una área de negocio se le impida asistir el soporte de una aplicación cuando falle generaría falta de disponibilidad del software ya que TI carece del conocimiento técnico y documentación sobre la misma para proveer el soporte necesario. Si un programador de un área de negocio dejara de laborar en el banco, invirtiendo tiempo para conocer el aplicativo y las herramientas en el cual esta soportado, generando atrasos por la falta de conocimiento.

Este trabajo se realiza con la finalidad de eliminar o mitigar dichos riesgos, ahorrar tiempo y recursos económicos a la empresa, es necesario realizar la investigación de las causas del problema y sus dimensiones, así mismo realizar una propuesta de solución basada en la maestría de Gestión de Tecnologías de Información para mejorar la productividad de la empresa en proyectos de software y mejorar el alineamiento que debe tener TI con los objetivos de las áreas de negocio, siendo una causa principal del desarrollo de esta tesis identificar y comunicar al área de Ingeniería de sistemas por qué las áreas de negocio prefieren contratar personal externo para el desarrollo de software y encontrar soluciones fuera de los servicios que brinda la gerencia de TI para satisfacer sus necesidades tecnológicas.

Se puede considerar que al realizar una metodología o proceso que deberán seguir los desarrolladores fuera de la Gerencia de TI, dará como ventaja tener un control sobre cada una de estas aplicaciones, garantizar el cumplimiento de los tiempos y la calidad

de cada desarrollo de software, y obtener así un mejor panorama para realizar mejoras continuas a los procedimientos de gestión de Proyectos para que estos sean ejecutados correctamente y con su respectiva documentación.

CAPÍTULO II. MARCO TEÓRICO

El gobierno de TI en desarrollos de software es necesario para medir el desempeño que tienen los proyectos, se debe controlar para poder medir, he ahí la importancia de seguir un proceso que ayude a dar seguimiento al cumplimiento de metas periódicas, un proceso que genere valor en los resultados que genera TI en relación a mantenimientos y desarrollos de software, es necesario estudiar las metodologías más prácticas y utilizadas para poder determinar qué aspectos de las mismas y que herramientas son las que más se adaptan la institución bancaria, a continuación se realiza un estudio de los beneficios de cada uno de los modelos escogidos y cómo se relacionan con la formulación del problema, el objetivo es escudriñar teorías que ayuden a la solución propuesta de este trabajo.

2.1 ANÁLISIS DE LA SITUACIÓN ACTUAL

2.1.1 ANÁLISIS DEL MACRO-ENTORNO

2.1.1.1 GOBERNABILIDAD EN DESARROLLO DE SOFTWARE

El ciclo de vida de desarrollo de sistemas sirve para trabajar de forma organizada los proyectos de software de empresas grandes y medianas y así satisfacer los requerimientos dentro de un período establecido para las áreas de negocio (INTECO, 2009). los programadores han buscado formas de organizar el código fuente de sus programas, se han presentado errores de programación que son difícilmente gestionados y se incurre en grandes pérdidas de tiempo al carecer de información organizada, antes de que estas metodologías existiera se dependía mucho de los programadores de la empresa porque cada uno conocía sus propios fuentes, donde se encontraban, el diseño lógico que tenían, para qué servía y cómo podía ser modificado, lo anterior era parte de la falta de gobierno de TI.

2.1.1.2 MODELOS DE DESARROLLO DE SOFTWARE DENTRO DE GOBIERNO DE TI

Los riesgos asociados al obviar una metodología de desarrollo de software son inherentes, la calidad no siempre es añadida a los resultados de un proyecto y los costos del mismo se pueden salir de control por lo que es necesario en los desarrollos de software comenzar a trabajar de forma ordenada para maximizar la productividad, toda empresa que busca ser competitiva y que necesita desarrollar software aplica alguna metodología “metodologías tradicionales” que se basan en un ciclo con etapas estándar de desarrollo y tienen la planeación y documentación de puntos críticos del proyecto como punto base, o alguna “metodología ágil” que se enfocan más en la gestión de comunicación de los involucrados de un proyecto para lograr mejor colaboración (Heredia, Almanza, Pons. 2010).

Las empresas a nivel mundial que desarrollan o dan mantenimiento al software utilizan metodologías que han ido evolucionado a través del tiempo como: las estructuradas, las metodologías orientadas a procesos, orientadas a objetos, orientadas a prototipos, CASE (Ingeniería de software asistido por computadora), soluciones ágiles etcétera, cada una de estas diferentes metodologías se usan con el fin de realizar procesos estandarizados que ayuden a los sistemas a funcionar en base a calidad y mejora continua.

(Cervantes y Gómez ,2012) indica que una de las metodologías de desarrollo más utilizadas es la estructurada o la tradicional ya que se divide en etapas de análisis, diseño, desarrollo, pruebas, implementación y seguimiento del sistema, los roles de quienes ejecutan cada etapa puede variar; una sola persona puede ejecutar cada etapa o puede existir una persona encargada para cada una de ellas, esto depende del tipo de empresa y tipo de proyecto, este método permite identificar la información y actividades críticas del proyecto para poder gestionarlo.

Las empresas que utilizan metodologías de desarrollo de software y que son exitosas han requerido estudiar la aplicación de las mismas mediante recursos propios o

capacitarse para lograr adaptar una metodología a los procesos más críticos de su empresa, en este punto ha sido importante el apoyo de la alta dirección hacia TI para poder implementarlas incluyendo a todos los involucrados, invirtiendo el tiempo necesario y recursos humanos que el área de ingeniería de sistemas necesite. Las personas que toman decisiones financieras respecto a la forma en que son manejados los proyectos eventualmente necesitan invertir dinero, a continuación se presenta cuáles continentes son los que están dispuestos a invertir dichos recursos y en qué porcentaje lo hacen.

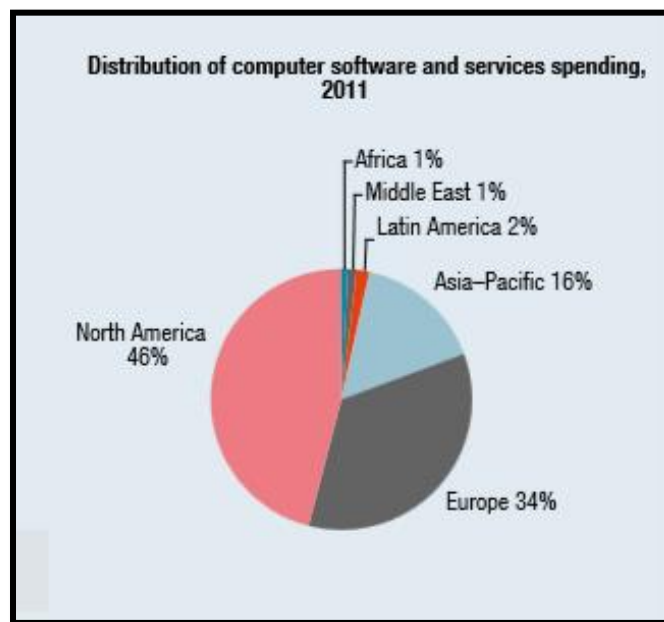


Figura 2 Gasto global en software y gasto en distribución de Tecnologías de Información y comunicaciones

Fuente: (Reporte de Economía de Información UNCTAD, 2012)

En la figura anterior se refleja en el gráfico circular el porcentaje en que los continentes invierten recursos monetarios para software y servicios computacionales, cabe mencionar que hasta el año 2011 América Latina era el país que menos invertía en estos temas según el Reporte de Economía de Información de 2012, sin embargo de acuerdo a este mismo estudio el mercado de software creció en América Latina, China, Rusia y la India.

2.1.2 ANÁLISIS DEL MICRO-ENTORNO

2.1.2.1 GOBERNABILIDAD EN DESARROLLO DE SOFTWARE EN AMÉRICA CENTRAL

Implementar el gobierno de TI a nivel centroamericano se ha convertido en un reto, en algunos países subdesarrollados de América Central aún se esfuerzan por implementar el uso de las TIC para sobresalir y obtener ventajas competitivas, por lo que se necesita incorporar dentro de las empresas una figura de autoridad de enfoque tecnológico para comenzar a disminuir la brecha entre los avances de desarrollo de software y uso de las TIC que poseen otros países. Según (José, 2012) se puede nombrar por ejemplo a Singapur quien incrementó su ventaja competitiva con un adecuado uso de las TIC, y en el 2014 el gobierno de Honduras firmó un fideicomiso con el Banco Atlántida por varios millones de dólares para proyectos de telecomunicaciones y proyectos de TI.

El uso en la región centroamericana de las TIC como herramientas tecnológicas ayuda a las empresas a mejorar la gestión administrativa interna y crear productos bajo estándares o metodologías que proporcionen control de calidad, aumento de la productividad, oportunidades de negocio y aumento de la ventaja competitiva a nivel internacional, sin embargo es difícil lograr conseguir estos beneficios cuando las empresas evitan dar su apoyo e invertir en temas asociados a plataformas o sistemas tecnológicos dando como resultado una pobre gestión por parte de los ejecutivos de TI de las organizaciones y la imposibilidad de brindar soporte cuando las áreas de negocio lo soliciten, incurriendo así en malas prácticas al poder satisfacer las necesidades de negocio y anticiparse al cambio en conjunto con TI.

La falta de reconocimiento sobre la importancia del liderazgo y el control interno que puede ejercer TI dentro de una organización imposibilita que surjan adelantos tecnológicos, se necesita que los productos de software en Centroamérica sean innovadores mediante procesos únicos establecidos para que se pueda medir indicadores bajo estándar adoptado por toda la empresa, ejecutar el mismo mecanismo a la hora de generar productos de software que satisfagan la necesidad de los usuarios

y clientes de la organización es una de las tareas a realizar para las empresas que deseen establecer un modelo de gobierno.

Es debido a la falta de aprovechamiento por parte de las empresas en temas de desarrollo Tecnológico que la región centroamericana se ha visto perjudicada ya que el poco interés hace que los países que integran a América Latina se encuentren mal posicionados, como una región que produce poco en base a lo que es desarrollo de software, uso e implementación de las TIC y la aplicación de infraestructura tecnológica de forma ágil y competente para aventajar y generar beneficios en un mercado altamente agresivo y competitivo.

(“En edición 2008-2009 del Reporte Global en Tecnologías de Información, de los países centroamericanos, Costa Rica se ubicó primero con la posición 56 en el global, seguido de Panamá (66), El Salvador (78), Guatemala (82), Honduras (95) y Nicaragua (125)”).(Ranking 2008-2009 en Tecnologías de Información – Central América Data , Central América Data, 2009).

Un beneficio que necesita el área centroamericana y que se logra al implementar un buen gobierno de TI en todas las empresas es evitar pérdidas por fraudes generados por la falta de seguridad informática, es necesario enfrentar adecuadamente los eventos que se presenten cuando surjan amenazas que eviten mantener la integridad de la información, para que los riesgos se minimicen y se reduzcan las vulnerabilidades que puedan producir un impacto negativo en la economía, objetivos y metas definidas en las organizaciones.

2.1.2.2 CRECIMIENTO DEL DESARROLLO DE SOFTWARE EN AMÉRICA CENTRAL

Debido a la creciente necesidad de sobresalir en un mercado tecnológico altamente competitivo, las empresas se ven forzadas a utilizar diferentes modelos que hagan uso de metodologías o procesos de desarrollo de software estructurados, existen asociaciones en algunos países de Latinoamérica como Argentina, Chile y Brasil que ayudan a impulsar el los desarrollos de software, el Observatorio de la Economía Latinoamericana (2004) afirma “Estas asociaciones tienen como objetivo propiciar políticas, mejorar los mercados y las cadenas de distribución, ayudar a sus asociados a mejorar sus capacidades

competitivas y buscar alternativas de desarrollo de programas conjuntos a partir del beneficio mutuo”, en cada uno de los países América Central debería existir una de estas asociaciones para explotar el talento profesional.

En Centroamérica también se utilizan diferentes modelos de desarrollo de software para gestionar los flujos de trabajo adecuados que se llevan a cabo en un proyecto liderado por las áreas de Tecnología, entre los modelos más utilizados se encuentran modelo cascada, modelos de procesos Incrementales, modelo DRA (Desarrollo Rápido de Aplicaciones), modelos evolutivos, construcción de prototipos, modelo espiral, modelo de desarrollo concurrente, proceso unificado. El Observatorio de la Economía Latinoamericana (2004) afirma “En Costa Rica las Tecnologías de la información representan el 1.3% del producto interno bruto” es decir existen iniciativas de apoyo en este país para el crecimiento de la industria del software, cabe mencionar que en América Latina Brasil es el país que más empresas de desarrollo de software posee.

2.1.2.3 GOBIERNO DE TI EN HONDURAS

En Honduras la adopción de las TIC y el crecimiento relacionado a temas de tecnología y desarrollo de software ha sido un rubro de difícil crecimiento debido al poco interés mostrado tanto por las empresas del sector privado y público hacia la inversión, no existe ninguna asociación que se encargue de apoyar en las mejoras competitivas y distribución de las empresas en relación al software como existe en otros países. Esta cultura por parte de los empresarios ha cambiado por la necesidad que hay para incursionar en la era tecnológica ya que esta se imponga cada día más y para poder competir a nivel global es importante trabajar y hacer uso de la tecnología para sacar el mayor provecho de ella.

Mediante estudios realizados por la AHTI en el año 2011 se observa un porcentaje en base al aporte hecho tanto por el género femenino como masculino en cuanto al crecimiento y uso de las TIC dentro del país, estos impulsados por el interés mostrado en muchos hondureños para realizar estudios en ramas de la Tecnología y que en cierta forma ha beneficiado al país y por ende a las empresas, las cuales han hecho

uso de estos profesionales para desarrollar sus propias aplicaciones y así mejorar la productividad generando mayores ganancias y ventajas competitivas.

Específicamente en el campo de la informática, los datos de la Asociación Hondureña de Tecnología de Información (AHTI) revela que la distribución por sexo en el campo laboral de las TIC en el año 2011 estuvo conformado principalmente por hombres (75.3%) y las mujeres constituyen casi una cuarta parte (24.7%). Sin embargo, el estudio revela que desde el año 2009 al 2011 la participación femenina ha reportado leves ascensos dentro del mercado laboral de las TIC. (Matamoros, 2011).

A nivel de gobierno de TI muchas empresas de Honduras han empezado a hacer inversión y así hacer uso de las buenas prácticas de marcos de referencias que les ayude a constituir dentro de la organización un ente regulador que garantiza que las actividades que se llevan a cabo dentro de cada una de sus áreas siguen estrategias alineadas con las gerencias de TI y que al momento de realizar algún cambio de índole tecnológica a nivel de infraestructura o desarrollo de software, la gerencia o los representantes a nivel de TI estén informados de dichos cambios o desarrollo a ejecutar y den su respectiva aprobación que formule productos y servicios que cumplan con los estándares de calidad.

Según (CentralAmericaData.com, 2009) algunas empresas de Honduras que han utilizado metodologías de desarrollo de software basadas en gobierno de TI y que ayudan a implementarlas son:

Diensa: Empresa dedicada a la Consultoría e implementación de sistemas de control de gestión: Estratégica, Operativa, Balance Score card, Riesgos, ITIL, COBIT, Inteligencia de Negocios y Marco Lógico para instituciones públicas.

Asociación Hondureña de Tecnología de Información: La AHTI agrupa y representa a las empresas del sector de Tecnología de Información de Honduras, para fortalecer el desarrollo de esta actividad y así contribuir al desarrollo económico y social.

Byte: Provee soluciones de software de clase mundial para las industrias bancaria/financiera y de telecomunicaciones.

Grupo Leitz: Outsourcing IT, Desarrollo Software, Web, Soporte de Aplicaciones, Soporte Técnico y Capacitación IT.

Soluciones DS S.A. (Grupo DS): Desarrollo y comercialización de software colaborativo para la automatización de procesos de negocios.

Es importante considerar que el área financiera es una de las áreas más interesadas en trabajar con procesos y estándares únicos y que sean internacionalizados ya que buscan sobresalir tanto nacional como internacionalmente es por eso el interés de grandes corporaciones como ser el grupo financiero Bac Honduras, Ficohsa y Atlántida por mencionar algunos que han ido impulsando la gestión de un buen gobierno de TI dentro de sus instituciones y con ello crear procesos o metodologías mediante la cual sus colaboradores pueden guiarse para desarrollar o crear productos que ayuden a incrementar su valor económico y ventaja competitiva.

2.1.3 ANÁLISIS INTERNO

El conjunto de acciones que realiza el área de TI ligado con la alta dirección para responder a requerimientos regulatorios, operativos o de negocio que conlleve al cumplimiento de los objetivos estratégicos que tiene la organización es el enfoque de gobierno de TI conocido por diferentes profesionales expertos del banco, cuando se carece de este enfoque TI tiene un problema que resolver para que los riesgos de pérdidas no se materialicen, evitar que se implementen penalizaciones, sanciones y existan pérdidas económicas millonarias, como se ha explicado en el capítulo I existen desarrollos de software que están fuera de este enfoque de gobierno, existen desarrollos de software realizados en áreas de negocio y los objetos están fuera de la línea base regional de objetos de software del banco.

En el banco existe un proceso de desarrollo de software, es el lineamiento al que cada desarrollo o mantenimiento debe someterse, está basado en estándares internacionales, se siguen las mejores prácticas en su flujo y adicionalmente es revisado periódicamente para cumplir con estándares de auditorías internacionales y locales así mismo responde a necesidades de mejora que expertos del banco identifican. En el proceso se siguen pasos específicos para el análisis de requerimientos, diseño, programación, pruebas, ambiente de pruebas, implementación, seguimiento, y evaluación final de un proyecto de software por el usuario final por lo que es una metodología de desarrollo tradicional (Bac Honduras, 2015).

En cada una de las fases del proceso se busca ahorrar el tiempo invertido, mitigar y eliminar riesgos y seguir un estándar regional para todo tipo de proyectos de software, al carecer de los beneficios que conlleva seguir una metodología de desarrollo constituye un problema de desorganización, riesgo de pérdidas, falta de disponibilidad de esas aplicaciones aprobadas, riesgo de seguridad que brindan estas aplicaciones y la funcionalidad puede inclusive estar dirigida a ejecutar un fraude. Las aplicaciones desarrolladas sin supervisión constituye la falta de gobernabilidad y falta de control por parte de la gerencia de TI misma que es el problema central actual, la alta directiva entiende la falla de gobierno y su criticidad según lo estudiado en el área de desarrollo.

Según la observación y la investigación realizada en el área de desarrollo de software y validada por el gestor de gobernabilidad el problema es también de carácter regulatorio, donde se involucran elementos como el control en la administración de aplicaciones, productos de software, autorizaciones y permisos de accesos a información sensible, así como prácticas de desarrollo de software sin estandarizar y regulado por una metodología previamente definida, documentada y formalizada. Adicionalmente las áreas de negocio de Bac Honduras aparte de estar recurriendo a empleados internos para estos desarrollos también solicitan servicios a empresas o recursos externos lo que conlleva otro riesgo en el manejo de la información y hacer crecer este problema.

Como desventaja cabe mencionar que el contar con aplicaciones sin gobierno de TI radica en que algunas de éstas son críticas para el negocio, tienen valor por la

frecuencia de uso y carecer de un proceso de seguimiento y soporte de esas aplicaciones constituye un problema estratégico para la empresa ya que nadie garantiza la disponibilidad de las mismas y su correcto funcionamiento, como segunda desventaja es la compatibilidad que tienen, en un futuro sería necesario combinar unos elementos de software con otros, hay aplicaciones en el inventario actual que trabajan con base de datos Postgresql por ejemplo siendo que los gestores con licencia autorizada de base de datos que puede utilizar la empresa son DB2 de IBM y SqlServer, por lo que para migrar se debe dedicar tiempo extra.

Los proyectos que son desarrollados fuera de TI tienen la limitación de definir y alcanzar resultados que puedan ser medidos y validados, así mismo la incapacidad de realizar mejoras continuas a los procedimientos de gestión de proyectos, lo que es necesario en un entorno de mucho cambio como Bac Honduras, para ayudar a controlar el problema la gerencia cuenta con un perfil de puesto para gobierno de TI, las funciones de este colaborador se enfocan en la identificación de estas aplicaciones y la gestión del proceso de catalogación de algunas de ellas, existen varios proyectos que se han pasado a la línea base de objetos de software regional del banco y se ha logrado documentar el diseño realizado y los requerimientos iniciales.

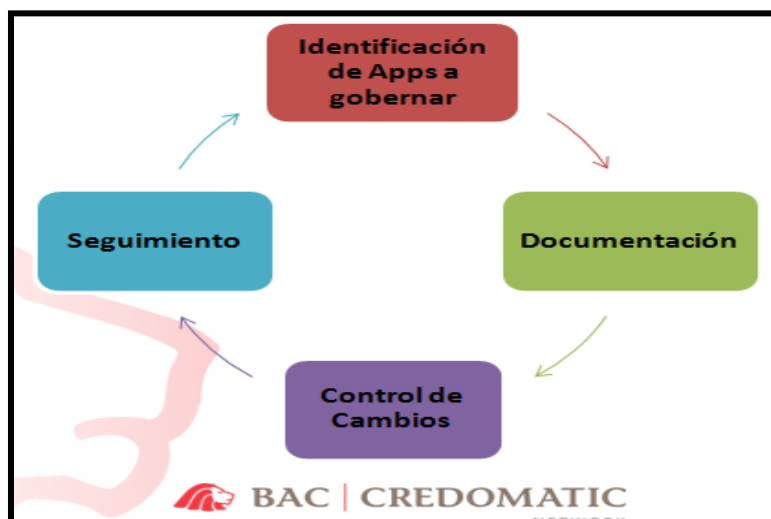


Figura 3 Necesidades identificadas para ejercer Gobierno de TI en desarrollos de software

Fuente: (Bac Honduras, 2014)

En la figura anterior se muestran las actividades identificadas por el gestor de gobierno de la gerencia de TI para comenzar a abarcar los riesgos relacionados con la falta de control de los desarrollos de software, estas actividades están orientadas a seguir el proceso de desarrollo actual que posee la empresa, es un comienzo orientado al enfoque del problema, identificar todas las aplicaciones, documentar las que son críticas e identificación de los riesgos de cada una de ellas es el punto de partida, sin embargo se hace necesario implementar la solución de raíz basado en un marco de referencia para el tema de gobierno.

2.2 TEORÍAS DE SUSTENTO

2.2.1 GOBIERNO DE TI

Para gestionar un adecuado gobierno de TI dentro de las organizaciones se debe tomar en cuenta la importancia que tienen las decisiones tomadas por la gerencia, se necesita utilizar la tecnología como un arma que genera ventaja competitiva, el gerente de TI debe darle importancia a la inversión que se necesite en la creación de diferentes plataformas y sistemas de tecnología, de esta forma mejorar las labores de los colaboradores y minimizar los riesgos a existir en los procesos críticos de negocio ejecutando planes de acción que den lugar a que en todo momento el negocio continúe funcionando y no se materialicen grandes pérdidas económicas, este tema de inversión necesita ser aplicada en herramientas que generen valor al negocio y no solo empeñarse en adquirir las últimas tecnologías.

(Network-Sec, 2014) indica que las estructuras dentro del gobierno de TI lo que realmente buscan es proveer la ayuda a las áreas de negocio para que se cumplan los objetivos y metas que estos establezcan, obteniendo un crecimiento en su desempeño que es directamente apoyado por TI, para ello se hace necesario el alineamiento de las estrategias de negocio con las estrategias de TI, adicionalmente un gobierno de TI bien implementado permite a las empresas adquirir nuevas ventajas competitivas, obtener beneficios y dar un valor agregado a los productos desarrollados con la gestión de TI, este último es el encargado de la correcta administración de la información de la

compañía, la gestión de calidad de los productos y soporte de los servicios que ofrece la empresa además del rendimiento de los recursos y la construcción de mejores prácticas de tecnología.

Establecer un buen gobierno de TI dentro de una organización es una forma de gestionar las herramientas, recursos, proyectos y productos de la empresa ordenadamente, para que los usuarios hagan uso de dichas herramientas automatizando tareas que pueden ser repetitivas y tediosas que consumen tiempo y esfuerzo, con el gobierno de TI se busca gestionar los procedimientos o metodologías adecuadas que garanticen que los programadores que no pertenecen al área siguen las fases de un proyecto de software correctamente mejorando el rendimiento laboral, siguiendo flujos de trabajos organizados con el fin de generar valor a través de una nueva forma de trabajo.



Figura 4 Modelo del Gobierno Corporativo de TI

Fuente: (ISACA, 2012)

En la Figura No. 4 se observa los elementos que el gobierno de TI provee para las estructuras que unen los procesos de TI, alineando las estrategias de negocio mediante un adecuado desempeño laboral incorporando actividades a aplicar en la planificación, implementación y operación de temas como estrategia, entrega de valor, gestión de recursos, medición de desempeño y gestión de riesgos, cada una de estas actividades

debe realizarse en el orden correcto con el fin de obtener resultados acorde a las estrategias que el negocio defina.

Siempre es útil considerar cuando se desee desarrollar software de forma controlada realizar el análisis, estudio preliminar y las fases que un proyecto de software debe contener, se debe recolectar toda la información y los datos necesarios para gestionar los requerimientos que los clientes o usuarios del sistema de información desean ver en su producto final, para cumplir con los objetivos por los cuales la aplicación o producto de software se va a desarrollar; ejecutando las tareas, funciones y conexiones correctamente con el fin de cumplir con las expectativas de las gerencias de negocio y sus usuarios finales y que estas estén acopladas con las metas y estrategias de la gerencia de TI y de la institución financiera Bac-Honduras.

“Para efectuar la planificación del proyecto se debería disponer de una serie de datos cualitativos y cuantitativos en forma de requisitos (tiempo máximo de transacción, número de usuarios máximos, interfaces gráficas)”. (Gutiérrez de Mesa & Pagés Arévalo, 2009, p. 43),

La cita anterior indica que una parte vital del desarrollo es el entendimiento de los requerimientos, además se debe dedicar el tiempo suficiente en la planificación del mismo, es importante investigar cuáles son las variables críticas o vitales que son parte de los requerimientos para darle un rumbo prometedor al proyecto, adicionalmente el modelo de gobierno de TI invita a evaluar cada fase crítica como la implementación de un proyecto, dirigirla significa controlar que los aspectos importantes sean ejecutados con diligencia, de la mejor manera, y monitorear significa validar que todo se ejecute de acuerdo a lo que se planeó.

2.2.2 MARCO DE REFERENCIA COBIT

COBIT es un marco de referencia que ayuda a gestionar las tareas y actividades que se llevan a cabo en una metodología para incorporar un adecuado gobierno en el desarrollo de software, metodología a partir de la cual los desarrolladores de las

diferentes áreas de negocio crearan sus propios productos de software. Se toma en cuenta este marco por sus diversos recursos, herramientas y procesos que sirven como guía y orientan sobre el camino a seguir para analizar, estructurar y ejecutar las fases de desarrollo a cumplir por los programadores de las áreas de negocio, es necesario que estos programadores estén bajo el gobierno y autoridad de la gerencia de TI para evitar que se realicen productos sin un estándar definido, que provoque la pérdida de integridad y calidad en los servicios y plataformas.

El poder de COBIT 5 se encuentra en su amplitud de herramientas, recursos y orientación. El valor de COBIT 5 se encuentra en la forma en que se aplica a su profesión. COBIT 5 ya está en línea, buscar usos específicos por área temática y llevar el poder y el valor de COBIT 5 para su organización. («COBIT 5: A Business Framework for the Governance and Management of Enterprise IT», 2014)

Al incorporar un buen gobierno de TI dentro de una empresa teniendo como referencia las buenas prácticas que COBIT 5 se satisface las necesidades en la administración de riesgos, el cumplimiento de los objetivos de control para las diferentes tecnologías de información y la optimización en la inversión de negocios, la misma gerencia de TI debe incursionar en temas como: la administración de recursos, tecnologías y desarrollo de software entre otros, soportando de esta forma los procesos de negocio y los requerimientos de la información para evitar impactos innecesarios en los recursos de TI que afecten directa o indirectamente a la organización.

De acuerdo a (ISACA, 2012) el marco de referencia COBIT 5 consta de varios objetivos de control ligados a cada uno de los procesos de TI existentes, los cuales se pueden agrupar en los siguientes dominios:

- Planificar y organizar la gestión de gobierno de TI
- Adquisición e implementación de las mejores prácticas de COBIT para ejercer un buen gobierno de TI
- Entrega de servicios eficaces en tiempos adecuados
- Soporte y monitorización de las actividades y procesos de TI

Uno de los objetivos que tiene Cobit según Isaca Cobit 5 (2012) es incrementar el valor que TI genera para la empresa, esto significa que es necesario abarcar problemas de áreas críticas del negocio con soluciones prácticas y eficientes, estas no necesariamente son soluciones con tecnologías altamente costosas sino más bien soluciones creativas e innovadoras, uno los temas a tratar debe ser la mejora de la satisfacción del usuario final de TI, esa satisfacción se logra con mejores tiempos, mejores servicios, no obligatoriamente con equipos costosos, las herramientas por sí solas no generan valor, genera valor el uso que los integrantes de una empresa le da a esas herramientas y generan valor los procesos que son mejorados a partir de la actualidad que la empresa vive.

Al utilizar este marco de referencia como enfoque de solución al problema planteado se necesita estudiar cómo optimizar los recursos con el uso de la tecnología, específicamente para el problema de este trabajo es importante utilizar recursos y herramientas que funcionen sin pérdidas de tiempo ni cuellos de botella, adicionalmente gestionar los riesgos asociados de tal manera que no representen pérdidas, el gobierno de TI debe asegurarse de cumplir con los objetivos mediante la toma de decisiones y el establecimiento de prioridades, ISACA (2012) afirma que “la administración planifica, construye, ejecuta y monitorea las actividades conforme a las directivas fijadas por el ente de gobierno” existen diferencias entre gobierno y administración, pero las dos están orientadas a optimizar y a mitigar los riesgos.

Los principios que plantea Cobit según Isaca Cobit 5 consisten en cubrir la organización de forma integral y aplicar un solo marco integrado entre otros, esto significa que la solución planteada deberá enfocarse en que todos sigan un mismo proceso, o que se adapten a partes del proceso según sea el rol que tengan en el proyecto, es importante considerar que se debe implementar una metodología que abarque cada uno de los roles de un proyecto de software y que se adapte a todo tipo de escenarios que exista en la organización, todos deben estar involucrados con la metodología de gobierno inclusive las áreas de negocio, adicionalmente el gobierno debe tomar decisiones basadas en las consecuencias que genera a cada involucrado del proyecto.

2.2.3 MODELO CMMI PARA SOFTWARE (MODELO DE MADUREZ DE LA CAPACIDAD INTEGRADO)

Este modelo consiste en una integración para mejorar los procesos y la ingeniería del software, permite identificar el grado de madurez que tiene la empresa respecto a la gestión de proyectos de software, identificando la ruta que debe seguir la empresa de acuerdo al nivel de madurez en el que se encuentra a nivel de gestión, el CMMI es un modelo muy útil para definir métricas en base a las metas que se desean alcanzar durante cada fase al ejecutar proyectos de desarrollo de software, evitando eliminar actividades que ya se encuentran implementadas en la empresa respecto a gestión de proyecto y que son útiles, sino que implementa mejoras en las etapas que realmente necesitan ser perfeccionadas, tratar de mejorar un proceso no implica eliminar todo lo que ya existe sino realizar un proceso de reevaluación.

Cuando se estudian las vulnerabilidades de la empresa se comienza a desarrollar un plan de mejora de procesos basado en los hallazgos encontrados y recomendaciones de expertos, se definen las prioridades, grupos de trabajo y métricas para dar seguimiento al avance de las soluciones establecidas en el plan. SEI indica que durante la ejecución del plan de mejoras se crean herramientas, políticas, formatos que ayuden al proceso de software y se comunica todos los cambios a realizar en toda la organización, durante la implementación de las mejoras se deben documentar las lecciones aprendidas y ponerlas en práctica en el inicio de nuevos procesos de mejora.

Ventajas del modelo CMMI según SEI Administrative Agent, (Noviembre 2010):

- Mejora al aumento de la productividad porque se corrigen errores que existen en el proceso de desarrollo actual que posee la empresa.
- Se cumplen con los acuerdos del tiempo de desarrollo del proyecto porque se eliminan cuellos de botella en el proceso y re trabajos innecesarios.
- Se reducen las correcciones que un programador debe realizar luego de la implementación o finalización el proyecto.

- La calidad del producto de software es mejorada por la reducción del número de defectos y el cumplimiento nítido de los objetivos.
- Se elimina la improvisación en cualquier parte de la gestión de proyectos
- Permite tener una visión amplia de la organización, para implementarlo hay que conocer a la empresa, conocer sus procesos más importantes.

Las ventajas anteriores son exitosas cuando toda la organización esté dispuesta a contribuir en el seguimiento de los pasos definidos para obtener mejoras en los procesos existentes, eliminar la resistencia al cambio puede mitigarse si se involucran a los interesados desde la fase de inicio de este modelo, “La investigación ha mostrado que el paso inicial más importante para la mejora de procesos es fomentar el apoyo de la organización mediante un fuerte patrocinio de la alta dirección” (SEI Administrative Agent, 2010, p.90), lo anterior aplica a todos los modelos cuando se quiere realizar un cambio en un proceso es importante que los ejecutivos entiendan los beneficios económicos y operativos de dichos cambios para que puedan apoyarlos.

El modelo CMMI tiene varios niveles con los cuales se representa la madurez de un desarrollo de software, estos niveles marcan las pautas para desarrollar productos y comienza por el nivel 1 o nivel inicial que se refiere a la etapa en la cual las empresas tiene un ambiente poco estable en cuanto a la forma de desarrollar una aplicación, luego prosigue el nivel 2 que se refiere a la gestión de proyectos mediante métricas para la administración de proyectos y la calidad con la que estos proyectos deben cumplir, en el nivel 3 se define la integración de varios grupos para que realicen proyectos de desarrollo de software mediante las métricas definidas en el nivel 2, capacitando al personal, entrenándolo y darle la correcta formación mediante una metodología de desarrollo.

Dentro de los últimos niveles que indica el CMMI para implementar un adecuado gobierno de TI en el desarrollo de software, se encuentra el nivel 4 que indica que se debe realizar una gestión cuantitativa referente a la calidad y productividad que ayudará a la toma de decisiones y una correcta gestión de riesgos mejorando los procesos de la

organización en el momento adecuado, por último se define el nivel 5 que se refiere a la optimización y mejora continua, gestionando procesos de innovación, analizando las causas de los problemas observados en las aplicaciones administrados dentro de la organización y aplicando las soluciones que resuelvan los errores identificados en los problemas encontrados.

“El modelo CMMI tiene dos representaciones denominadas continua y escalonada. En su representación escalonada está compuesto por cinco niveles de madurez: Nivel inicial, Nivel gestionado, Nivel definido, Nivel gestionado de manera cuantitativa, Nivel optimizado”. (Catherine Picazzo, Norha Villegas, & Gabriel Tamura, 2008, p. 93).

Con el modelo CMMI se pueden definir metas para cada gestión que se empieza a elaborar comenzando por el proceso de definición de requerimientos para luego analizar los problemas por el cual se solicita un producto de software, aplicando las medidas correctivas que se pueden tomar en consideración para solucionar los problemas encontrados y cumplir con las expectativas de los clientes y usuarios, una vez definido el problema se procede a planificar las actividades y tareas que se harán para llevar a cabo el proyecto de desarrollo, de esta manera se formaliza un compromiso real con el proyecto y los tiempos de entrega durante su ejecución, siempre asegurando que el producto final cumpla con la calidad exigida por las áreas de negocio.

Con el CMMI se garantiza la integridad que debe tener la información en todo momento durante el proceso de configuración ya que es importante manejar siempre información concreta y viable, que ayuda a los departamentos de Ingeniería del software a tener el control sobre los trabajos realizados por los desarrolladores que pertenecen tanto a la gerencia de TI como a las áreas de negocio que están siendo gobernadas o supervisadas por TI, verificando que las métricas o procesos definidos se cumple a cabalidad.

2.2.4 NORMA ISO/IEC 12207

Esta norma establece procesos por medio del cual se puede definir el ciclo de vida que puede alcanzar un software comenzando desde la definición de requisitos hasta la finalización o terminación de su vida útil. La norma ISO/IEC 12207 es un estándar que define una estructura a seguir por desarrolladores, proveedores, técnicos, analistas y gestores para que hagan uso de un mismo procedimiento de desarrollo utilizando un lenguaje de programación común con el cual debe estar bien definido en los procesos que la norma indica al iniciar un proyecto de desarrollo de software.

Tabla 2 Procesos utilizados generalmente en la Norma ISO/IEC 12207 se clasifican en 3 tipos

Procesos Norma ISO/IEC 12207		
Procesos Principales	Procesos de Soporte	Procesos de la Organización
<ul style="list-style-type: none">⊕ Adquisición.⊕ Suministro.⊕ Desarrollo.⊕ Explotación.⊕ Mantenimiento.	<ul style="list-style-type: none">⊕ Documentación⊕ Gestión de la configuración.⊕ Aseguramiento de calidad.⊕ Verificación.⊕ Validación.⊕ Revisión conjunta.⊕ Auditoria.⊕ Resolución de problemas.	<ul style="list-style-type: none">⊕ Gestión.⊕ Infraestructura.⊕ Mejora.⊕ Formación.

Fuente (ISO/IEC, 2008)

Una ventaja de esta norma es que la estructura que tiene definida para los estándares desarrollados a través de ella son muy flexibles dando la facilidad de poder aplicar una metodología de desarrollo de software que puedan seguir los diferentes analistas que pertenecen a las áreas de negocio para satisfacer las necesidades que quiera ejercer un buen gobierno de TI al momento de ejecutar un mecanismo para llevar un control de los proyectos de software que se inician dentro de la organización, este proceso se caracteriza por ser modular es decir que sus procesos se pueden acoplar con facilidad y al mismo tiempo establece la responsabilidad que recae sobre cada programador

garantizando de esta forma que se ejecute cada fase que contempla una metodología de desarrollo.

El estándar ISO/IEC 12207 no solo ayuda a poder gestionar proyectos de software, sino que da las herramientas mediante las cuales se puede mejorar o implementar los procesos adecuados para gestionar el desarrollo de aplicaciones de software o mejorar los procesos ya existen de acuerdo a la madurez de la organización e incorporando estos estándares a un adecuado Gobierno de TI que ayude a la administración y autoridad de la Gerencia de TI sobre las diversas áreas de negocio que pertenecen al grupo financiero Bac Honduras.

(INTECO, 2009) se refiere a: La norma ISO 12207 como un marco que ayuda a generar e implementar modelos para evaluar los procesos definidos dentro de una empresa para ver su nivel de madurez y tomar decisiones sobre cambios o mejoras que se deben hacer para mantener alineados estos procesos con el ciclo de vida que un producto de Software, en esta norma se considera que los programas están enfocados a la calidad del software, gestionando adecuadamente los riesgos cada vez que se desea comenzar un nuevo proyecto de desarrollo ya que esta parte es la que proporciona la información concreta para estimar los costos y controles que debe contener un proyecto exitoso.

Los ciclos de vida para realizar un producto basado en desarrollo de Software pasa por un cierto número de fases desde el momento en que se recolectan los requerimientos que el usuario indica desea que su aplicación tenga, hasta que la aplicación como tal deja de coexistir, con la norma ISO 12207 lo que se hace es analizar cuáles serán esas fases, para comenzar a desarrollar una aplicativo indicando los pasos a seguir para establecer una metodología que deben seguir todos los diferentes analistas desarrolladores cuando comiencen proyectos que crean nuevos productos de software, estandarizados mediante un mismo flujo de desarrollo de inicio a fin.

2.2.5 METODOLOGIAS ÁGILES DE DESARROLLO DE SOFTWARE

Las metodologías de desarrollo de software ágiles como Scrum se centran en procesos que apoyan la colaboración de los integrantes del proyecto, sus procesos son más flexibles que las metodologías de ciclo de vida de desarrollo ya que pueden existir cambios durante el proyecto, existen menos roles y menos artefactos o tipos de documentos en el proyecto, se enfocan más en desarrollar software con funcionalidad exitosa y de valor para el usuario que en seguir un proceso basado en marcos de referencia como ITIL, al utilizar este tipo de metodologías se pueden cambiar los requerimientos iniciales en colaboración con los usuarios de negocio, son flexibles cuando se presentan cambios nuevos durante el proyecto.

METODOLOGÍA SCRUM

Scrum es una metodología de desarrollo de software donde existen iteraciones de 30 días y en cada una de ellas se otorga una entrega de software al cliente para que éste la evalúe, existen reuniones diarias para coordinar el trabajo y para recibir retroalimentación entre todos los involucrados del proyecto, una metodología como esta o un proceso basado en estas prácticas podría servir para que los desarrolladores que no pertenecen al área de TI y los programadores externos sigan un proceso en donde los requerimientos y la programación sean el punto principal de enfoque de trabajo, eliminar actividades innecesarias que un programador deba realizar al seguir desarrollar y trasladar esta responsabilidad a un analista que beneficie el cumplimiento del tiempo y resultados de los proyectos.

Los proyectos con Scrum comienzan tiene una visión general del mismo y realizan las actividades prioritarias en un lapso de tiempo corto en los cuales se debe entregar una parte del proyecto, es decir en un ciclo de desarrollo de Scrum conocido también como Sprint debe existir una entrega al usuario que debe ser evaluada por el mismo. Palacio (2008) indica que: “Con Scrum el diseño y la estructura del resultado se construyen de forma evolutiva” esto significa que en los proyectos los diseños de un software pueden cambiar por lo tanto no vale la pena estarlos modificando en la documentación y eso

resulta cuando se hace en la primera fase en lugar de poder cambiar el diseño en el transcurso del mismo. La colaboración con esta metodología no depende del puesto de trabajo sino que todos pueden participar según las capacidades de las personas.

Los elementos de la funcionalidad de Scrum según Palacio (2008) se componen de:

- ✓ Sprint: Se planifica el lapso de tiempo, tareas se van a realizar durante ese lapso del cual debe surgir una entrega.
- ✓ Seguimiento diario: para identificar imprevistos para lograr alguna tarea, para informar lo realizado al día anterior.
- ✓ Se hace un documento de requisitos del sistema
- ✓ Se hace un documento de la lista de actividades a realizar en cada sprint o lapso de tiempo
- ✓ Revisión del sprint: para verificar los resultados de la entrega
- ✓ Trabajo en equipo
- ✓ Más importantes las responsabilidades que los roles del equipo
- ✓ Gráfico Burn Down: indica el avance del equipo, sirve para identificar si el proyecto está en tiempo.

PROGRAMACIÓN EXTREMA XP

La programación extrema es otra metodología ágil (Ruiz, Almanza, Linares, 2010) afirman que es “una metodología ligera de desarrollo de software que se basa en la simplicidad, la comunicación y la realimentación o reutilización del código desarrollado” orienta para seguir un flujo basado en pruebas unitarias que realiza el programador, enfocarse en reutilización que consiste en modificar el código de un fuente existente para adaptarlo a las necesidades del negocio, también es permitido trabajar en parejas de desarrolladores para agilizar el proyecto, una ventaja de este modelo es que los objetivos en tiempo, funcionalidad y costos pueden cambiar, cuando se sigue un flujo tradicional de ciclo de vida los programadores pueden desviarse de la calidad que debe tener el software por enfocarse más en la documentación del proyecto.

La programación extrema práctica el cambio de requerimientos en el transcurso del desarrollo de un proyecto, según (Bautista, 2012) el cambio de requerimientos iniciales es inevitable, por lo que se acepta programar requerimientos no planificados, es una metodología basada en prueba y error, dentro de la misma el cliente entrega al programador un documento de historias de usuario y el programador convierte esas historias en tareas, una vez terminado el código se entrega al “tester” o el encargado de pruebas, mismo que se encarga de validar las pruebas con el usuario, también existe el rol de “tracker” que es el que se encarga de darle seguimiento al cumplimiento de las tareas del proyecto, en general esta metodología se basa más en la adaptabilidad.

La funcionalidad de programación extrema se compone de:

- ✓ Planificación: poner en producción los requerimientos más importantes primero
- ✓ Diseño simple
- ✓ Pruebas continuas
- ✓ Modificar el código constantemente para asegurar de que queda lo más sencillo posible
- ✓ Programación en parejas
- ✓ Los programadores pueden cambiar partes del código en cualquier momento
- ✓ Se realiza documento de historias de usuario, actividades del programador
- ✓ Integración del código fuente

METODOLOGÍA CRYSTAL CLEAR

La metodología Crystal Clear es un tipo de metodología que se basa principalmente en la comunicación con el objetivo de aplicar la disciplina necesaria para lograr el éxito en el desarrollo de un proyecto de software, esta metodología es muy adaptable ya que ayuda a disminuir trabajos que se consideren innecesarios hacer dependiendo del tipo de proyecto que se esté llevando a cabo, brindando la flexibilidad de trabajar con los recursos y entregables realmente necesarios para finalizar un proyecto en un tiempo óptimo y con la calidad que debe ofrecer al usuario final en base a calidad y funcionalidad.

“En La serie Crystal las personas involucradas escogen aquellos principios que les resultan efectivos y mediante la aplicación de la metodología en diversos proyectos agregan o remueven principios en base al consenso grupal del equipo de desarrollo”. (Ruiz Heredia, Almanza Álvarez & Linares Pons, 2010, p. 8)

(Ruiz Heredia, Almanza Álvarez & Linares Pons, 2010) afirman que Crystal Clear establece un conjunto de prioridades y principios que sirven de guía para la toma de decisiones, como lo es:

- ✓ Eficiencia en el desarrollo: que habla sobre la rentabilidad en los proyectos
- ✓ Seguridad en lo que se entrega.
- ✓ Habitabilidad: esto se refiere al cumplimiento de las actividades asignadas a cada Recurso Humano involucrado en el proyecto.
- ✓ Frecuencia en las entregas: periodo de tiempo en el cual se estarán brindando entregas parciales o completas de los entregables.
- ✓ Comunicación: hacer uso de los diferentes mecanismos y herramientas de comunicación para dirigir el proyecto de desarrollo.

La metodología de Crystal Clear brinda varios principios necesarios para la documentación de requerimiento como ser: el diseño y planeamiento de los proyectos de software que se ejecutarán, reduciendo los entregables de acuerdo a la complejidad y accesos que se desee dar a este tipo de información por las personas que realmente requieran de ella y que se vean involucradas en el proyecto, logrando de esta forma un colaboración y trabajo en equipo entre todos los miembros del proyecto, todo esto mediante diversas técnicas según la conveniencia del proceso de desarrollo a utilizar en el proyecto.

La guía de trabajo que presenta Crystal Clear es altamente recomendable para equipos pequeños, da flexibilidad y prioriza la parte humana (como todas las Metodologías Ágiles), apuntando a lograr eficiencia, habitabilidad y confianza en los miembros del equipo. Presta especial importancia a la ubicación física del grupo, donde la

comunicación cumple el rol principal. La entrega frecuente de código confiable y "funcionando" mantiene el foco y evita distracciones. (Word Press, 2010)

Esta metodología ágil se centra principalmente en el equipo que integra el proyectos de software, la comunicación que debe existir entre cada miembro y los controles adecuados para lograr una adecuada gobernabilidad por parte de la Gerencia de TI en cada fase del proceso de desarrollo que se ejecuten desde el comienzo hasta el fin del proyecto incorporando los estrategias y técnicas necesarias para lograr las metas establecidas en tiempo de ejecución, calidad, minimización de costos y satisfacción del usuario final, mediante esta metodología ágil de desarrollo se tiene la flexibilidad de tener una estructura de programación incremental que se ajusten a los cambios necesarios para mejorar la funcionalidad de las aplicaciones de software.

En la metodología Crystal Clear se debe considerar en todo proyecto de software estrategias y técnicas necesarias para lograr que los proyectos de software sean exitosos, estos corresponden a la formulación de diferentes plantillas o entregables que ayudan a identificar el estado del proyecto, sus avances y la iteración entre los integrantes del equipo del proyectos y las actividades o tareas de las cuales son responsables

(Barrientos Oscar, 2011) Afirma que las estrategias de esta metodología son:

- ✓ Exploración de 360°
- ✓ Victoria temprana
- ✓ Esqueleto ambulante
- ✓ Rearquitectura Incremental
- ✓ Radiadores de Información

En cuanto a las técnicas que forman parte de la metodología ágil Crystal Clear se mencionan:

- ✓ Entrevistas de Proyecto
- ✓ Talleres de reflexión

- ✓ Planeamiento Blitz
- ✓ Estimación Delphi con estimaciones de pericia
- ✓ Encuentro diarios de pie
- ✓ Miniatura de procesos
- ✓ Gráficos de quemado
- ✓ Programación lado a lado

Tabla 3 Comparación de las características de metodologías ágiles con metodologías tradicionales

Características	Metodologías Ágiles	Metodologías Tradicionales
Enfoque	Adaptativa	Predictiva
Medidas de éxito	valor de negocio	conformada por planes
Tamaño del proyecto	Pequeña	Grande
Estilo de la administración	Descentralizada	Autocrática
Perspectiva al cambio	adaptación al cambio	sostenibilidad al cambio
Cultura	liderazgo-colaboración	comando-control
Documentación	Alto	Baja
Énfasis	orientada a personas	orientada a procesos
Ciclos	Numerosa	Limitada
Domino	impredecible, exploratoria	Predecible
Planificación	mínima,	desde el comienzo
Retorno de la Inversión	temprano en el proyecto	al final del proyecto
Tamaño de los equipos	pequeños/creativos	Grandes

Fuente: M. A. Awad(2005).

La tabla 3 permite identificar que las tecnologías ágiles generalmente están orientadas a equipos de trabajos más pequeños, por lo que se permite la oportunidad de que sea impredecible y descentralizada en sus flujos de trabajo mientras que los métodos tradicionales pueden aplicar a grupos de trabajo más grandes, comparando las características de las diferentes metodologías se puede concluir que no existe una mejor que la otra sino que en la empresa se debe implementar la que mejor se adapte al entorno de la misma y la que permite reformular flujos de trabajo de acuerdo a las necesidades que cambian por muchos factores, para el problema en estudio sería interesante tomar lo mejor de cada tipo de metodología para elaborar una solución que sea útil al gobierno de TI.

2.3 MARCO CONCEPTUAL

Los conceptos más importantes utilizados a lo largo del documento se explican a continuación en orden de prioridad y frecuencia de uso.

TI: Departamento de Tecnología de la Información, es el área que se encarga de ofrecer el soporte técnico y administrar las diferentes aplicaciones e infraestructura de tecnología que forman parte del grupo financiero Bac Honduras, también se asegura de que los diferentes sistemas y aplicaciones desarrolladas están disponibles para satisfacer las necesidades que tengan las áreas de negocio para la operativa normal del banco, incluyendo servicios informáticos, reportes, hardware, redes y comunicaciones.

TIC: Tecnología de Información de Telecomunicaciones, consiste en los recursos, procedimientos y técnicas que se utilizan para distribuir la información y la comunicación a través de la infraestructura que conforma la institución, también se refiere a la forma de manejar la información por los diferentes medios de comunicación, asegurando que estos viajen de forma íntegra y segura para evitar pérdidas que puedan generar molestia en base a los servicios ofrecidos.

Metodología: Es un marco de trabajo que se utiliza para estructurar y controlar un proyecto de desarrollo o mantenimiento de software, estos son marcos de trabajo por lo general se basan en estándares ya definidos que incluye los diferentes pasos que se deben realizar, como se debe ejecutar cada fase de un proyecto de desarrollo de software y cuáles son las herramientas a utilizar en cada una de las fases establecidas del proyecto.

Desarrollo de software: Es el conjunto de actividades que realiza un programador para desarrollar un aplicativo que cumpla con los requisitos previamente establecidos por el usuario del área de negocio, realizando cada uno de los pasos definidos en una metodología de desarrollo desde el levantamiento de requerimientos, pasando por el

desarrollo y pruebas del software y finalizando con la puesta en producción del producto final, asegurando de cumplir las expectativas de los usuarios y las áreas de negocio.

Gobierno de TI: Estructura que se implementa para dirigir y tener el control de la empresa en términos de alcanzar los objetivos definidos por las estrategias de negocio y añadir valor a los procesos que utilizan negocio para lograrlo, al mismo tiempo ayuda con la gestión de riesgos y el retorno de la inversión en TI, el término de Gobierno TI en este trabajo es utilizado para indicar el control que debe tener TI sobre el elemento de software, los gestores de base de datos, ejecutables e interfaces que tiene la empresa, también se refiere a la autoridad que TI debe ejercer sobre todo elemento de software o procesos utilizados dentro de cada una de las áreas de negocio, estableciendo las métricas de uso adecuadas para el desarrollo de software a seguir dentro de las unidades de la institución.

Proceso de Desarrollo de Software: Se considera que un proceso de desarrollo de Software son todas aquellas actividades que ejecutan siguiendo un modelo o estructura que se aplica para toda la etapa de desarrollo de un productos de software, generalmente estos procesos son creados para mejorar la calidad y productividad dentro de una empresa para formalizar actividades únicas y estándares que se ejecuten de forma organizada colaborando de esta forma en minimizar costos y aumentar los beneficios, cumpliendo con las entregas de los proyectos en el tiempo estimado.

Proceso de Negocio: es un conjunto de actividades que se realizan en una empresa para generar resultados en base los insumos proporcionados por la misma institución y que interactúan entre sí para ofrecer los productos y/o servicios con la mayor eficacia posible, agregando valor al negocio y estableciendo márgenes de satisfacción tanto al cliente interno como el externo, enfocando cada tarea o actividades en el cumplimiento de los objetivos de las gerencias que conforman el comité ejecutivo para aumentar la ventaja competitiva y mejorar los estándares de calidad de las funciones y roles que tiene cada recurso que componen las unidades de negocio.

Cobit: Marco de trabajo o de referencia de buenas prácticas utilizado para la implementación y mejora en las actividades que debe realizar un buen Gobierno de TI dentro de las empresas, COBIT es una guía que ayuda a las empresas a minimizar los riesgos ligados a la mala gestión de la Gerencia de TI dentro de las áreas de negocio, proporcionando herramientas, procesos, prácticas y principios que estimulan el deseo de mejora continua en temas relacionados a Gobierno de TI, la Tecnología, administración de riesgo y la gestión adecuada de la información crítica que está bajo el mando de negocio y debe pasar bajo la autoridad de la gerencia de Tecnología de Información.

CMMI: Es un Modelo de madurez que ayuda a la mejora y evaluación de los procesos de desarrollo de software incorporados en una empresa para identificar la capacidades de cada uno de los productos de software resultantes al momento de iniciar un proyecto de desarrollo, este modelo colabora a los analista de desarrollo para identificar las debilidades que puede existir al momento de generar un producto de desarrollo y así ver las mejoras que se puedan realizar en el momento oportuno antes de sacar la aplicación a producción, evitando la mala gestión en la funcionalidad de los servicios a ofrecer.

Objeto de Software: Un objeto de software es un aplicativo, sistema operativo o ejecutable con una serie de instrucciones y funciones para generar una salida con funcionalidad específica, para entregar un servicio o información útil.

2.4 MARCO REFERENCIAL

2.4.1 IMPLEMENTANDO COBIT Y GOBIERNO DE TI EN CORPORACIÓN GPF (ECUADOR)

Al momento de implementar una adecuada gestión de Tecnología en una corporación hay varios factores que se deben tomar en cuenta entre ellos se tiene que considerar: la necesidad de las empresas por incorporar plataformas tecnológicas, como desean administrarlas, las necesidades de los usuarios y clientes para automatizar herramientas de trabajo o plataformas de servicio, la cultura de la empresa para seguir

metodologías y procesos estandarizados que generen productos de calidad. Para enfatizar en estos aspectos se debe considerar que el marco de referencia que se implemente para gestionar el desarrollo de las aplicaciones de software sea el ideal y el que provee el diseño y construcción de procesos acordes a lo que se desee obtener como producto final.

La corporación GPF es una entidad ecuatoriana formada por un grupo de corporaciones farmacéuticas dedicadas al servicio en pro de la salud de los pobladores de Ecuador, esta corporación ha creado alianzas con diferentes instituciones para contribuir con el bienestar de sus colaboradores, familias y comunidad, ofreciendo programas que ayuden a los pobladores de más bajos recursos en temas como la educación, salud, protección del medio ambiente y mejora de los entornos laborales por medio de alianzas con instituciones motivadas y dispuestas a brindar sus servicios sociales y mejorar así la calidad de vida de la población en general. (Corporación GPF, 2014)

(Diseño de Procesos bajo Cobit 5.0, 2014) afirma que: para poder obtener su éxito actual, tuvo que hacer uso de metodologías con el objetivo de incorporar diferentes plataformas bajo un mismo estándar usando como pivote las mejores prácticas proporcionadas por el marco de referencia COBIT a través de un compañía consultora llamada DELTA, la cual ayudó a que GPF implementará procesos en base a la Metodología de Cascada de Métricas involucrando temas de; diseño, gestión de procesos, construcción de cronogramas para gestionar proyectos bajo un mismo gobierno de TI y una elaboración de planes fundamentados en procesos y fases estandarizados.

La corporación GFP vio la necesidad de implementar una metodología por medio de la cual pudieran mantener y unificar todas las diferentes plataformas y sistemas que utilizan para administrar los servicios que proveen a través de las compañías y organizaciones con las cuales mantiene sus alianzas de servicio, ya que necesitaba poder contar con un adecuado gobierno de TI que contará con procesos estándar que pudieran utilizar de forma que el desarrollo y la gestión de tecnología se hiciera siguiendo flujos de trabajo únicos y genéricos que ayuden a garantizar la calidad del

servicio a ofrecer y complacer de esta manera las necesidades y expectativas de los usuarios sobre el software desarrollado e implementado en la corporación.

Es debido al uso de un adecuado marco al momento de incorporar las mejores prácticas de gobierno TI que la corporación GPF no solo tiene un control sobre la gestión de tecnologías de Información sino que también puede medir y tener indicadores que le ayuden a obtener información importante sobre su trabajo social y el trabajo social realizado por sus compañías aliadas, teniendo la oportunidad de tomar decisiones en base a mejoras que sean necesarias realizar para mejorar el servicio brindado y la calidad que este debe mantener en todo momento.

Este es un ejemplo de cómo un adecuado análisis ayuda al momento de querer implementar una metodología que imponga la mecánica de trabajo correcta para incorporar un gobierno de TI eficiente, mejorando los procesos de la empresa y colaborando a su crecimiento no solo organizacional sino también a su crecimiento competitivo, aumentando los beneficios y el valor de las compañías usando los recursos adecuados con una considerable disminución en costos mediante la ejecución de proyectos exitosos apoyados por las diferentes áreas de negocio.

2.4.2 APLICACIÓN DE SCRUM EN UN PROYECTO DE SOFTWARE UNIVERSIDAD DE MONTEMORELOS DE MÉXICO

Se implementa la metodología Scrum en un equipo de desarrollo de software de un sistema financiero de la Universidad Montemorelos, México, anteriormente cada proyecto debía pasar por las etapas tradicionales de desarrollo análisis, diseño, programación, pruebas e implementación según el artículo el equipo de trabajo perdía mucho tiempo en las etapas de requerimientos, análisis y diseño teniendo problemas al cumplir con fechas de entrega de los proyectos, al invertir mucho tiempo en la documentación se corre el riesgo de no invertir suficiente tiempo en la parte de pruebas las que sirven para validar la funcionalidad de los requerimientos, siendo esta una de las etapas más importantes cuando se desarrolla software y en la que se debe invertir más tiempo.

Cuando el programador no invierte el suficiente tiempo en la etapa de pruebas se tiene el riesgo de obviar errores en la funcionalidad del código, corregirlo es re-trabajo e invertir tiempo en elementos que no constituyen entregas nuevas para el usuario. En el proyecto del sistema financiero los usuarios solicitaban cambios a los requerimientos o presentaban nuevos requerimientos los cuales también debían ser documentados, esto significaba regresar a una etapa que ya se había ejecutado en la parte de análisis del desarrollo, se debían modificar documentos ya existentes, para este entorno se necesitaba aplicar Scrum, Romero, Salinas(2014) afirman para “eliminar el círculo vicioso desde sus raíces gracias a la flexibilidad y respuesta ágil de Scrum ante el entorno complejo y dinámico”.

En el año 2008 el líder de proyectos de la universidad tuvo interés por la metodología de Scrum, investigó acerca del modelo para poderlo implementarlo debido a que los problemas de los proyectos estaban asociados a cuellos de botella, agilidad en los proyectos era un concepto necesario para él, la primera mejora derivada del modelo según este caso de estudio consistió en eliminar la figura del líder de proyecto, no se requería tener una persona ordenando a todo el equipo y ejerciendo presión para que los proyectos avanzaran en su lugar el líder se convirtió en un elemento de ayuda al proyecto eliminando las pérdidas de tiempo que existían en el proyecto a raíz de las reuniones diarias que se tenían con el equipo de trabajo donde se exponían los casos y oportunidades de mejora.

En las reuniones diarias que son parte de las metodologías de Scrum los programadores explican que habían realizado en día anterior que realizarían en el día y los problemas o necesidades que tenían las cuales eran abarcadas por integrante del proyecto, las tareas que se planificaban se lograban finalizar utilizando este método y los integrantes del equipo se encontraban motivados por la comunicación estable y efectiva que se tenía, se invertía el tiempo en reuniones que eran productivas, lo anterior es muy importante en cualquier proyecto, recibir retroalimentación y encontrarse en sintonía con un líder de proyecto para realizar mejoras o abarcar problemas en lugar de invertir mucho tiempo en actividades protocolarias que no generen valor a un proyecto de software.

Para dar por terminada una tarea se debía evidenciar las pruebas, la productividad aumentó cuando el equipo comenzó a trabajar reportando tareas completadas y se realizaba un análisis de la capacidad de trabajo del equipo para evitar incumplimientos, este equipo concluyó que para que Scrum funcione de manera exitosa se debe seguir un proceso de mejora donde cada uno de los roles aprenda a ejecutar sus tareas de manera correcta para que generen un valor al equipo, Scrum es flexible para poder adaptarse a un proceso de la empresa, sin embargo no se debe abusar del cambio de la metodología porque puede conllevar a perder los beneficios de la misma.

2.4.3 PROYECTO DE MEJORA CMMI, UN CASO DE ÉXITO EN ARGENTINA

Cecilia Boggi (2014) “El CMMI es un conjunto de buenas prácticas para lograr un objetivo específico, incluyendo que es lo que se debe hacer”, este concepto era necesario implementarlo para la empresa de consultoría de TI y desarrollo de software Millenniu3 la cual tiene negocios en Latinoamérica, Estados Unidos y Europa es una empresa que le ayuda a otras a utilizar la tecnología y la innovación para brindar un valor agregado en los servicios de la empresa, cuenta con más de 150 empleados dedicados al objetivo, algunos de los valores de esta empresa consistían en el trabajo en equipo, mejora continua, transparencia y agilidad, adicionalmente varios de sus clientes eran Movistar, Verizon, Telmex.

La empresa escogió la implementación de CMMI por la calidad que proveería a los proyectos, era necesario contar con gerentes de proyectos con certificaciones PMI y obtener certificaciones como el certificado CMMI nivel dos, otra de las razones para implementar CMMI consistía en el cumplimiento de los objetivos estratégicos que la empresa entre los cuales se encontraba consolidarse como una empresa de servicios de desarrollo y mantenimiento de software con alcance internacional, también optimizar los costos y la calidad para ser competitivos. esta metodología tiene un enfoque de ciclos de vida de desarrollo o metodología tradicional de desarrollo.

Los elementos de la metodología implementada para los proyectos eran los procesos, las plantillas, las guías, checklist, tipos de proyecto y línea del ciclo de vida del

proyecto, para comenzar a trabajar en cada uno de los elementos la empresa realizó una evaluación inicial para determinar qué procesos eran críticos para el proyecto y no se estaban ejecutando con eficiencia, luego se trabaja en la estrategia de las mejoras y detalle de las actividades, en la asignación de recursos y tiempo, el seguimiento del plan, la capacitación y políticas para la implementación. Como en todo proyecto se necesita el apoyo de la alta dirección y tener los recursos necesarios, se asignó a un responsable por área de proceso, reuniones de seguimiento semanales y un plan de capacitación del modelo CMMI.

Uno de los desafíos para implementar CMMI es cambiar la cultura de los clientes no todos los equipos de trabajo de la empresa estaban acostumbrados a seguir una metodología de trabajo que tenga actividades para cada fase de un proyecto, algunos equipos presentaban resistencia al cambio, sin embargo luego de reestructurar los procesos y adaptarlos para hacerlos más eficientes, y lograr involucrar directamente a todos los interesados dentro del modelo se obtuvieron beneficios según Cecilia Boggi (2014) como: la mejora de la relación con los clientes, mejora de la calidad de servicios, mejora de la comunicación de los equipos de trabajo, disminuyó la rotación de personal y ayuda a tomar mejores decisiones debido a la información organizada que se genera a partir de la documentación de los procesos.

CAPÍTULO III. METODOLOGÍA

El siguiente capítulo tiene como finalidad estudiar de manera detallada los elementos que contribuyan a identificar los puntos clave de la investigación por medio de los cuales se podrá establecer o sugerir oportunidades de mejora en el área de desarrollo de Bac Honduras, usando herramientas estadísticas que den validez a la información obtenida, considerando los proyectos como punto de estudio principal debido a que estos proveen datos relacionados con temas como la calidad, el tiempo de ejecución, control y la satisfacción del usuario final.

3.1 CONGRUENCIA METODOLÓGICA

3.1.1 LA MATRIZ METODOLÓGICA

Tabla 4 Matriz Metodológica

Titulo	Problema	Pregunta	Objetivo		Variable	
			General	Específico	Independiente	Dependiente
Gestión de Gobernabilidad de desarrollo de software en Bac Honduras 2014	¿Cuál es el procedimiento propuesto para adaptar una metodología de desarrollo que contribuya a un mejor gobierno de TI en desarrollos de software de BAC Honduras?	¿Porqué las áreas de negocio de Bac Honduras recurren a desarrollar sus propias aplicaciones sin inclusión de TI?	Plantear una metodología ágil de desarrollo de software que mejore la gestión de gobierno de TI y que sea utilizada por los	Identificar las causas del porqué las áreas de negocio desarrolla sin ayuda de la gerencia de TI.	Fallas en el proceso actual	Proyectos
		¿Qué pasos se deben realizar en cada etapa del desarrollo de software para solventar la falta de gobernabilidad de TI?	programadores externos, a la vez establecer un procedimiento para adherir a los programadores que pertenecen a Bac	Establecer un procedimiento para adherir a los programadores de negocio en las etapas del proceso de desarrollo de software existente en la gerencia de TI	Proceso del gobierno de TI	
		¿Qué parte del flujo de proceso a proponer debe ser	Honduras al proceso de desarrollo de software existente en	Determinar que fases del flujo de proceso serán realizadas por programadores de negocio, programadores externos y	Responsabilidades en el flujo del proceso	

	seguido por los desarrolladores externos, que parte debe ser seguida por negocio y qué parte por la gerencia de TI?	el área de desarrollo, determinando o actividades que mejoren la calidad de un proyecto.	programadores de TI dentro del flujo de la metodología ágil propuesta.		
	¿Cuál es la metodología de desarrollo de software ágil que podrían seguir los desarrolladores externos?		Proponer la metodología de desarrollo ágil que mejor se adapte para cumplir con características de tiempo y calidad requeridas en un desarrollo de software para los programadores de negocio.	Metodología ágil de desarrollo	

3.1.2 RELACIÓN DE LAS VARIABLES

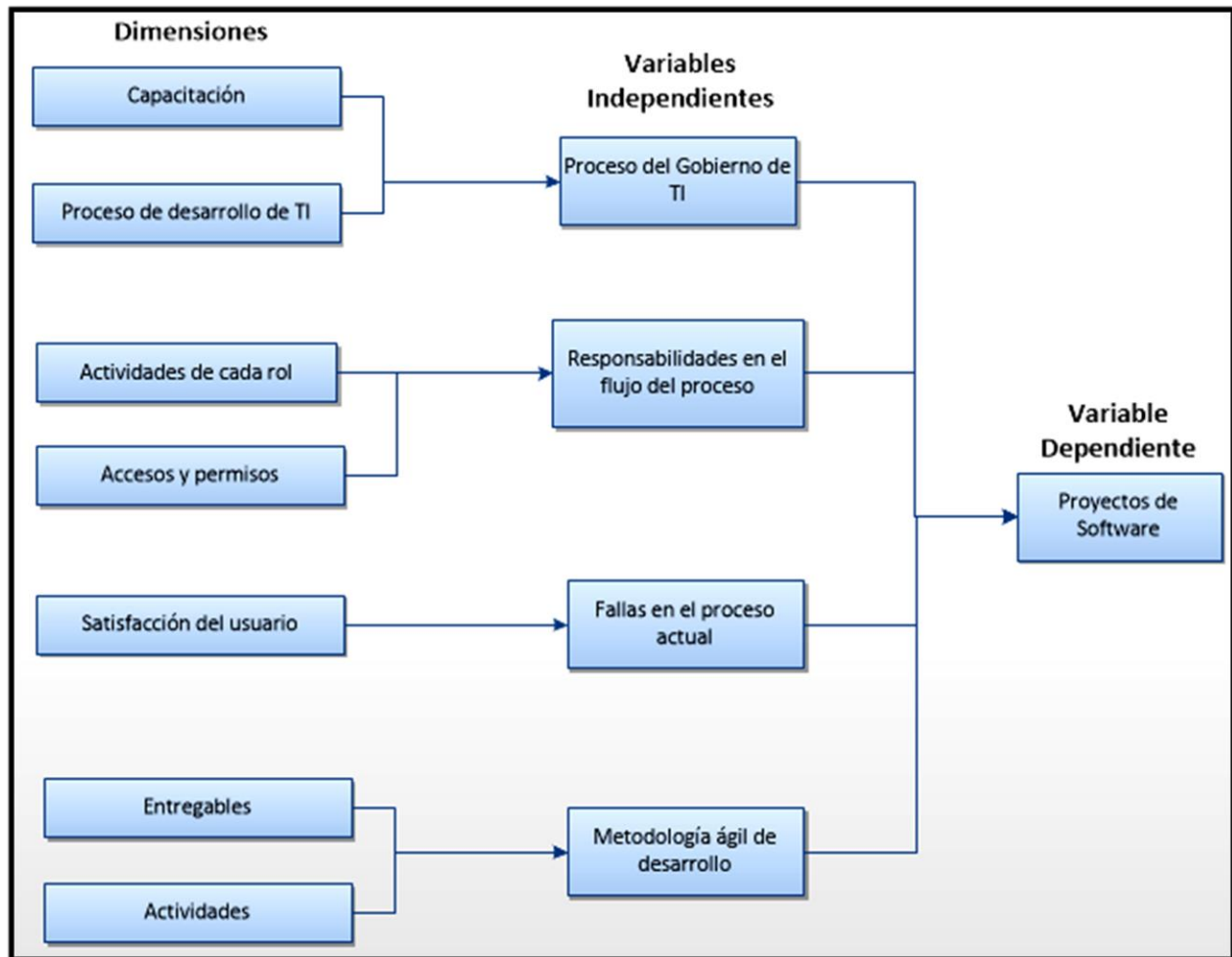


Figura 5 Diagrama de las variables

Tabla 5 Operacionalización de las variables

Variable	Definición		Dimensiones	Indicador	Unidades (Categorías)	Escala
	Conceptual	Operacional				
Variable Independiente: Proceso del Gobierno de TI	Son los pasos propuestos para adherir a los programadores de las áreas de negocio al proceso de desarrollo de TI.	Son las fases a ejecutar para lograr que los programadores que pertenecen al banco se adapten a la metodología de desarrollo existente.	Capacitación	Periodicidad	¿Cada cuánto se realizara n capacitaciones del proceso actual de desarrollo de software de TI a los programadores de negocio?	Frecuencia en meses a) b) c)
				Efectividad	¿Qué medios a utilizar para capacitar a los programadores de negocio sería más efectivo?	Tipos de instrumentos utilizados a) b)
				Efectividad	¿Cuál es el mejor método para enseñar el proceso de desarrollo actual?	Preferencia en método de enseñanza a) b) c)

Variable	Definición		Dimensi ones	Indicador	Unidades (Categorías)	Escala
	Conceptual	Operacional				
					¿Cuándo se puede definir si una capacitación es efectiva respecto a la asistencia?	Cantidad de programadores a) b)
			Proceso de desarrollo de TI	Documentación requerida	¿Debe realizarse la parte de documentación en la planificación y acuerdos de fechas para el ejecutivo de negocio?	Frecuencia de la documentación a) b) c)
					¿Qué fase del proyecto debe ser documentada por programadores de negocio?	Fases del desarrollo a) b) c) d)
				Administración de objetos de software	¿Cuáles objetos de software pueden migrarse a la línea base de TI?	Tipos de objetos de software a) b) c)

Variable	Definición		Dimensiones	Indicador	Unidades (Categorías)	Escala
	Conceptual	Operacional				
					¿Cuál será la prioridad para la catalogación de los objetos de software?	Orden de ejecución de los proyectos a) b) c)
Variable Independiente: Responsabilidades en el flujo de proceso	Son las actividades que realizan los involucrados de negocio en el proceso de desarrollo TI y las actividades que realizarán los programadores externos en la metodología ágil a proponer.	Identificar los roles que se necesitan para la funcionalidad de las metodologías.	Actividades de cada rol	Cantidad de desarrolladores de TI y negocio.	¿Cuántos proyectos pueden ejecutar cada programador?	Cantidad de proyectos asignados a) b) c)
					¿Qué tipo de proyectos se asignan primero?	Prioridad de ejecución de proyectos a) b) c)
			Control del software	¿Cuáles roles estarán involucrados en la migración del software?	Roles técnicos de la metodología a) b) c)	
				¿Quién debe dar visto bueno del diseño? Poner en encuesta	Roles administrativos de la metodología a) b) c)	
			Accesos y permisos	Tiempo	¿Quién debería dar los accesos a los programadores de negocio?	Roles dentro del proceso de desarrollo a) b) c) d)

Variable	Definición		Dimensiones	Indicador	Unidades (Categorías)	Escala
	Conceptual	Operacional				
					¿Cuánto tiempo debe durar el acceso según el tipo de proyecto para los programadores de negocio?	Tipos de autorización de acceso a) b) c) d)
				Privilegios de los accesos	¿Cuánto tiempo puede durar un acceso a un programador que no es parte de TI?	Cantidad de tiempo para acceso a) b) c)
					¿Cuántos proyectos de software se terminan según complejidad?	Cantidad de proyectos finalizados a) b) c) d)
Variable Independiente: Fallas en el proceso actual	Son las diferentes debilidades que tienen el proceso de desarrollo de software actual, que ayuden a identificar las causas del problema.	Características que disminuyen la calidad de los proyectos de software dentro del proceso actual.	Satisfacción del usuario	Cantidad de Proyectos terminados	¿En base a las solicitudes de programación cuál es su nivel de satisfacción?	Satisfacción en base a tipos de programadores del banco a) b) c)
					¿Quiénes son los responsables de que los	Proyectos de los diferentes tipos de programadores a) b)

Variable	Definición		Dimensiones	Indicador	Unidades (Categorías)	Escala
	Conceptual	Operacional				
					proyectos fallen?	c)
				Cantidad de Proyectos que fallan	¿Qué fases del proceso pueden originar la mayor cantidad de fallas?	Fases del proceso a enfocarse a) b) c) d)
					¿Qué áreas atienden más proyectos de desarrollo?	Cantidad de proyectos por área a) b) c) d) e)
				Cantidad de solicitudes de desarrollo TI y de Negocio.	¿Quién desarrolla proyectos más complejos?	Cantidad de proyectos complejos desarrollados a) b) c)
					¿Cuáles es el seguimiento que necesitan las áreas de negocio en el desarrollo de un proyecto de software?	Frecuencia del seguimiento a) b) c)
Variable Independiente Metodología Ágil de Desarrollo	Son las metodologías ágiles a estudiar para determinar cuál se necesita	Son las características requeridas en una metodología ágil para adaptarlas	Entregables	Cantidad de entregables por metodología ágil	¿Cuál metodología es más óptima para los programadores	Tipo de metodología a) b) c) d)

Variable	Definición		Dimensiones	Indicador	Unidades (Categorías)	Escala
	Conceptual	Operacional				
	más para los programadores externos contratados por negocio.	al gobierno de TI.			externos?	
				Complejidad	¿Cuál es la metodología que contiene actividad es más relevantes para mejorar la calidad de los proyectos?	
			Actividades	Calidad	¿Qué características disminuyen costos en un proyecto cuando los recursos son programadores externos?	Características de metodologías a) b) c)
				Costo	¿Cuáles proyectos son desarrollados en menor tiempo?	Tiempo de finalización por área a) b) c)
Variable Dependiente: Proyectos de Software	Es un conjunto de actividades que se interrelacionan para lograr un producto de	Es un objeto de software, como base de datos, aplicaciones, interfaces, etcétera para	Tiempo	Tiempo de ejecución del proyecto	¿A qué fase del proceso de desarrollo de software se le	Fases a enfocarse en el proceso a) b) c) d) e)

Variable	Definición		Dimensiones	Indicador	Unidades (Categorías)	Escala
	Conceptual	Operacional				
	software con funcionalidad específica.	satisfacer las necesidades de automatización de negocio.			dedica más tiempo?	

3.1.3 HIPÓTESIS

H₀: Una metodología ágil de desarrollo en conjunto con el proceso de gobierno de TI no es necesaria para el control de TI sobre los proyectos de software.

H_A: A mayor cantidad de proyectos de software mayor es la necesidad de acoplarlos al proceso de gobierno de TI en conjunto con una metodología ágil para el control de TI sobre los proyectos de software.

3.2 ENFOQUE Y MÉTODOS

Enfoque Mixto: El enfoque de la investigación en este trabajo de tesis es Mixto por el tipo de estudio realizado, ya que se definen las características y cualidades en las cuales está fundamentado el actual proceso de desarrollo de software dentro del grupo financiero Bac Honduras, tomando estas características para proponer a partir de lo que ya existe una metodología estándar que ayude en la gestión de proyectos realizados por los desarrolladores de negocio, para mejorar los tiempos de ejecución, la calidad y los flujos de procesos de desarrollo, así mismo se estudia variables cuantitativas como ser costos, cantidad de recursos de las metodologías a estudiar, cantidad de fallas de los proyectos de software, cantidad de proyectos desarrollados por tipo de área, por las investigaciones cualitativas y cuantitativas se determina que el enfoque es mixto.

Investigación Descriptiva: Se realizó una investigación descriptiva porque se hace uso de marcos de referencia y mejores prácticas ya establecidos y estandarizados para mejorar las actividades realizadas actualmente por los programadores de negocios y al iniciar proyectos de desarrollo de software, utilizando los procesos y lineamientos que sean útiles al momento de gestionar proyectos de desarrollo bajo un adecuado gobierno de TI, mejorando el rendimiento de las aplicaciones, controlando las actividades realizadas por los analistas y brindando las herramientas necesarias para dar el mantenimiento adecuado a las aplicaciones cuando lo requieran.

Investigación Explicativa: Se explica el porqué de la necesidad para implementar en el banco un flujo de proceso que ayude a administrar los proyectos realizados por los diversos desarrolladores de la institución fuera de gobierno de TI mediante una metodología de desarrollo con mejores prácticas, esto con el objetivo de resolver los problemas actuales, las aplicaciones desarrolladas fuera de la gerencia de TI no cuenta con la adecuada documentación, administración y soporte; cuando fallan o presentan errores que afecten las operaciones del banco, predisponiendo a los usuarios y afectando los servicios brindados al cliente externo.

Métodos:

Se aplicó el método científico porque se identificó y analizó las relaciones significativas de los resultados de los datos explicando las relaciones entre las variables para poder explicar las causas del problema. Así mismo se aplicó un método estadístico como ser la estadística descriptiva e inferencia porque se tabularon datos a través de encuestas y se pudieron analizar tendencias y elegir la ruta que más se adecua a una solución, el Alpha de Cronbrach permitió validar fiabilidad de algunas de las preguntas a través de la herramienta estadística SPSS mediante la cual también se elaboran tablas de frecuencia y los gráficos, adicionalmente los métodos teóricos utilizados fueron el análisis y síntesis para obtener la base objetiva de la realidad estudiada.

3.3 DISEÑO DE LA INVESTIGACIÓN

Para la investigación se tomó una muestra no aleatoria o no probabilística ya que la población con la cual se trabaja es una población definida en el campo de desarrollo de software. Los pasos para lograr los objetivos del trabajo de tesis fueron:

1. Determinar de acuerdo al análisis de la situación actual y problema de este trabajo las teorías a estudiar que puedan servir para lograr cumplir con los objetivos y presentar mejoras que la empresa pueda aplicar, identificar las características más importantes de dichas teorías para poder consultarlas cuando se esté elaborando el plan de aplicabilidad.
2. Realizar una encuesta por cada tipo de programador y rol involucrados para asegurarse de que las preguntas puedan ser contestadas de acuerdo a la información y experiencia particular de los programadores de negocio, subcontratados, de TI; así mismo los ejecutivos de negocio y gestor de gobierno, se realizó una reunión con licenciado de estadística de UNITEC para identificar oportunidades de mejora de los instrumentos realizados para recabar la información, una vez obtenido el visto bueno se realizan las encuestas en herramienta online encuesta.com para que puedan ser llenadas por los participantes de forma electrónica.
3. Llevar a cabo una investigación no experimental, no se pone en práctica las recomendaciones realizadas y el plan de aplicabilidad de este trabajo para poder demostrar su eficacia sino que se utilizó un método transversal para recabar la información en determinado tiempo, tabular la misma y examinar la incidencia y los valores de las variables para poder comprobar la hipótesis.
4. Realizar análisis Foda de las entrevistas realizadas para poder recabar oportunidades de mejora, causas de las debilidades de la metodología tradicional actual e identificar falta de control operativo para poder proponer las mejoras respecto al control de gobierno en el plan de aplicabilidad.

3.3.1 POBLACIÓN Y MUESTRA

Cantidad de recursos involucrados en un proyecto de software: 103 recursos de Honduras, la muestra es igual a la población para llegar a resultados más exactos y objetivos y considerando que la cantidad no es tan grande para Bac de Honduras.

Tabla 6 Tipos de recursos involucrados en un proyecto

Recursos	Cantidad
Programadores TI	40
Programadores de negocio	26
Programadores Externos u outsourcing	29
Gestor de Gobernabilidad	1
Ejecutivos de negocio	7
Total	103

La población se compone de los recursos que están involucrados en un proyecto de desarrollo de software, se debe estudiar la forma de trabajo por medio del cual ejecutan dichos proyectos de mantenimientos o desarrollo de software para áreas específicas de negocio. Los proyectos son realizados por programadores que son contratados por los ejecutivos de las gerencias de negocio por lo que no están ligados a TI, estos programadores constituyen otro elemento de estudio; los mismos pueden ser colaboradores internos o externos con el conocimiento suficiente para cumplir con la funcionalidad que debe contener la aplicación mediante sus propias herramientas.

3.3.2 UNIDAD DE ANÁLISIS

Satisfacción de los ejecutivos de negocio de los proyectos: Aquí se debe definir los diferentes motivos que tiene la alta gerencia de negocio para recurrir a ejecutar sus propios proyectos de desarrollo de software sin la intervención, apoyo y colaboración de la gerencia de sistemas y medir el nivel de satisfacción en las diferentes fases del proyectos cuando no se sigue una metodología de desarrollo. Se establecen diferentes niveles de satisfacción para identificar el criterio de los ejecutivos en términos de alcance, tiempo y costos de un proyecto fuera del gobierno de TI.

Forma de trabajo de los desarrolladores fuera del gobierno: se necesita conocer cuáles son las actividades que la mayoría de estos desarrolladores utiliza para ejecutar un proyecto de software, en qué fases del proyecto dedican más tiempo y porque, cómo gestionan los elementos de calidad, tiempo, costos y alcance del proyecto para poder identificar mejoras en las actividades, identificar qué principios del proyecto son más importantes para estos desarrolladores de tal forma de proponer una metodología a la que se puedan adaptar fácilmente.

3.3.3 UNIDAD DE RESPUESTA

Cantidad de Proyectos: Estará determinada por la cantidad de proyectos que un área de negocio ha implementado con los programadores externos, se analizará la cantidad de proyectos que ha sido exitosos y la cantidad que ha fallado sin una metodología, se necesita conocer la cantidad de proyectos que recurren a realizar mucho mantenimiento y soporte a lo largo de un tiempo determinado.

Cantidad de Programadores: Para identificar cuantos programadores tienen las áreas de negocio que forman parte del banco, cuantos programadores son externos y cuantos programadores son parte de TI para determinar si las cantidades de programadores existentes satisfacen la demanda de proyectos del banco.

Cantidad de usuarios: Cantidad de usuarios que utilizan las aplicaciones que han sido desarrolladas en las diferentes gerencias de negocio donde la gerencia de TI no ha tenido participación, y medir la satisfacción de los usuarios que usan estas aplicaciones a diario dando así un diagnóstico sobre la calidad de las mismas y de esta forma buscar el establecimiento de un proceso que ayude a la mejora continua de los proyectos de software.

3.4 TÉCNICAS E INSTRUMENTOS APLICADOS

3.4.1 INSTRUMENTOS

Para esta fase se hará uso de instrumentos tanto cualitativos como cuantitativo, necesarios para recolectar toda aquella información que ayude a formular las respuestas por medio de las cuales se obtendrá los resultados para dar solución a la problemática de la falta de una metodología de desarrollo en las áreas de negocio bajo un gobierno de TI, por ejemplo; se realizarán cuestionarios basados en preguntas abiertas y cerradas para poder medir los indicadores que definimos en cada una de nuestras dimensiones de estudio, partiendo de los resultados obtenidos para formular posibles soluciones o mejoras favorables para la institución financiera Bac-Honduras.

Es importante mencionar que estos cuestionarios estarán contenidos en diferentes formatos de entrevistas y encuestas a realizar a los recursos involucrados en un proyecto de desarrollo, que va desde el Gerente que aprueba y hace la contratación de los programadores para que desarrollen, hasta la opinión de los mismos programadores sobre el uso de las herramientas y recursos proporcionados por la gerencia de negocio para que ejecuten proyectos de desarrollo de software. Considerando también la satisfacción de los usuarios respecto al producto terminado y puesto en producción, como la disponibilidad de estas aplicaciones cuando realizan sus tareas cotidianas.

3.4.2 TÉCNICAS

Entrevista: Se utiliza un instrumento cualitativo como ser la entrevista para recabar la mayor parte de la información requerida, estas entrevistas se realizarán al gestor de gobernabilidad del área de desarrollo, a ejecutivos de negocio que contratan a los programadores fuera de la gerencia de TI y a los programadores que han ejecutado algunos de los proyectos para las áreas de negocio.

Encuesta: Un instrumento cuantitativo será la encuesta para conocer la satisfacción de los usuarios finales de Bac Honduras, se utilizará una encuesta predeterminada por el área de desarrollo de sistemas que se utiliza para conocer la satisfacción del usuario en

las diferentes etapas del proceso de desarrollo que se sigue dentro de Gobierno de TI y se realizará una encuesta específica para los usuarios que hayan participado en un proyecto de software fuera de gobierno de TI.

Mediante estos dos instrumentos se necesita conocer indicadores de medición siguientes relacionados con requerimientos, código fuente, pruebas y calidad, cantidad de desarrolladores de TI y negocio, administración de los objetos, tiempo de proyectos, privilegio de accesos, proyectos terminados, cantidad de proyectos que fallan, alertas de aplicaciones, disponibilidad de los servicios, estándares y mejores prácticas implementadas.

3.4.3 PROCEDIMIENTOS

Los pasos realizados para obtener la información para demostrar la hipótesis fueron:

1. Se investigó qué tipo de herramienta estadística es útil para recolectar la información necesaria para dar validez a la hipótesis planteada según el problema definido en el trabajo de investigación, se determina realizar las encuestas online y utilizar el Alpha de Cronbach para comprobar la hipótesis.
2. Una vez definidas las herramientas a aplicar: encuestas y entrevistas, se define cuáles deben ser las variables e indicadores para formular las preguntas y posibles respuestas que se desean validar para luego obtener respuestas adecuadas.
3. Se analiza y concluye que las herramientas estadísticas se deben aplicar a tres diferentes sub-poblaciones: Programadores, ejecutivos de negocio y usuarios a quienes se les aplica las encuestas y entrevistas para que ellos brinden sus opiniones en base a las diferentes experiencias que tenga cada uno en el proceso de un proyecto de software.
4. Una vez definida las sub-poblaciones se toma una muestra de 15 programadores a los cuales se les aplica las encuestas, para validar que la información contenida en

las encuestas fuera efectiva y acorde a los datos que se necesitan obtener para hacer una comprobación de hipótesis.

5. Una vez aplicada la muestra se tabulan los datos en la herramienta estadística SPSS, por medio de la cual se validan las respuestas mediante el Alfa de Cronbach utilizado para medir la efectividad de la información y verificar que las preguntas realizadas ayuden a proporcionar los datos necesarios, para poder concluir y recomendar una posible solución al problema planteado en la investigación.
6. Al aplicar el alfa de Cronbach se identifican que las encuestas son válidas sin embargo se deben modificar algunas de las preguntas de las encuestas dado que presentan un índice de validación bajo indicando que las preguntas no están bien formuladas o la información obtenida a través de ellas no es fiable, en este caso las preguntas a reformular fueron las siguientes:

Encuestas aplicadas a programadores de negocio, se modifican y dejar para encuestas finales las siguientes preguntas:

¿Qué tipo de metodologías le dan mayor valor a un proyecto, las metodologías tradicionales (Enfoque en planificación y documentación) o las metodologías ágiles (enfoque en comunicación y resultados del proyecto)?

¿En qué parte del proceso de desarrollo de software se enfoca más, da más prioridad o dedica más tiempo?

¿Cuántos proyectos de desarrollo de software finaliza por trimestre?

¿Qué aspectos de su forma de trabajo o la metodología adoptada en base a sus conocimientos le gustaría mejorar?

¿Qué elementos son más relevantes para enfocarse dentro de un proyecto de software?

Encuestas aplicadas a los usuarios de las aplicaciones de software, se modifican las preguntas:

¿Cuántos avances del proyecto o entregables presentó el analista programador a lo largo del proyecto?

¿Las pruebas realizadas previas a la puesta en producción fueron claras, el programador explicó las funcionalidades nuevas de manera concisa?

¿Continuaría solicitando desarrollos de software de programadores adheridos a negocio que no forman parte de TI?

7. Una vez reformuladas las preguntas se vuelve a validar las encuestas haciendo uso del SPSS y verificando por medio del alfa de Cronbach que estas tuvieran un porcentaje mayor en la formulación de preguntas dando como resultado porcentajes más altos: 1. Encuestas aplicada a programadores de Negocio y subcontractados (0.729), 2. Encuestas aplicada a programadores de TI (0.636), 3. Encuestas aplicadas a los usuarios (0.686).
8. Para validar las encuestas y entrevistas realizadas se enviaron a un asesor estadístico mismo que revisó el contenido y forma de ambas herramientas y valido la información contenida en ellas, de igual forma el asesor reviso la población y muestra definida concluyendo que para tener una visión más clara al momento de validar las hipótesis se debe tomar el total de la población definida, la cual consiste en aplicar las encuestas a 103 personas (Programadores de negocio, programadores subcontractados, programadores de TI y usuarios) y así obtener la información necesaria para realizar una propuesta que dé solución al problema planteado.
9. Se aplican las encuestas validadas de forma online y se revisa la cantidad de encuestas finalizadas y los resultados obtenidos por medio de ellas, luego se tabulan los resultados en la herramienta estadística y generar por medio de los

mismos gráficos que muestren tendencias que ayuden a validar la hipótesis de la investigación.

10. Comprobar la hipótesis por medio del análisis de los resultados obtenidos en las encuestas y entrevistas finalizadas.

11. Hacer los planes de acción de las recomendaciones identificadas, necesarias y que estén orientadas hacia la solución del problema expuesto y el cumplimiento de los objetivos planteados.

3.5 FUENTES DE INFORMACIÓN

3.5.1 FUENTES PRIMARIAS

Las fuentes primarias del trabajo se enfocan en abordar el tema de Gobernabilidad en desarrollos de software, para encontrar una propuesta óptima a Bac Honduras que resuelva el problema de raíz se estudia referencia como Cobit como marco de referencia, CMM como un modelo de madurez para la empresa, Norma ISO/IEC 12207 y las prácticas de las metodologías ágiles de desarrollo, a partir de ellas se busca encontrar la falta de mejoras en los procesos que sigan las áreas de negocio cuando implementan proyectos de software fuera del gobierno de TI.

3.5.2 FUENTES SECUNDARIAS

Como fuentes secundarias se consideran varias técnicas y fuentes de información como ser: entrevistas, encuestas, revistas científicas, cuestionarios entre otros, por medio de las cuales se trata de obtener información relevante para apoyar y validar el éxito que se puede lograr gestionando proyectos de desarrollo, cuando se implementa dentro de una empresa una metodología de desarrollo de software administrada por medio de un adecuado Gobierno de TI, incorporando marcos de referencias estandarizados y realizados para este fin, estas fuentes ayudan a conocer que es lo que la empresa realmente.

CAPÍTULO IV. RESULTADOS Y ANÁLISIS

4.1 RESULTADOS ENCUESTAS

4.1.1 VARIABLE RESPONSABILIDADES EN EL FLUJO DE PROCESO

⇒ ¿Cuántos proyectos considera puede tener asignados actualmente?

Tabla 7 Frecuencia proyectos asignados actualmente

Opciones	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
a) Tres proyectos asignados por diferentes niveles de complejidad	42	43,8	43,8	43,8
b) Solamente un proyecto, se asigna el siguiente hasta terminado el anterior	13	13,5	13,5	57,3
c) Dos proyectos en paralelo	41	42,7	42,7	100,0
Total	96	100,0	100,0	

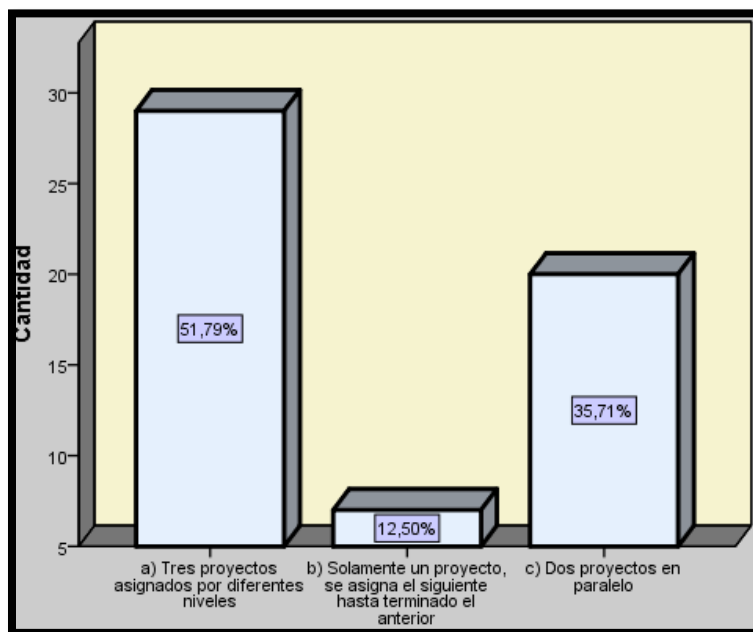


Figura 6 Resultados proyectos asignados programadores de negocio y programadores externos

Tanto los programadores de negocio como los programadores de TI pueden tener más de un proyecto asignado, la metodología ágil de desarrollo se debe adaptar en base a

la cantidad de trabajo asignada a un recurso, se identificó que los programadores de negocio pueden manejar hasta tres proyectos asignados por diferente complejidad, este hallazgo es importante para identificar una forma de trabajo que no afecte la calidad y tiempos de entrega cuando se tienen varios proyectos en paralelo, por otro lado la mayoría de los programadores de TI que siguen la metodología de desarrollo tradicional trabajan con dos proyectos a la vez, por lo que sería importante manejar dos tipos de metodologías que se adapten a la complejidad de los diferentes proyectos.

⇒ ¿Qué tipo de proyectos considera resolver como prioridad, quién lo establece?

Tabla 8 Frecuencia prioridad de los proyectos para programadores de negocio

Opciones	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos a) Lo decide negocio, dependiendo de la necesidad	50	89,3	89,3	89,3
b) Regulatorios	3	5,4	5,4	94,6
c) Por elección propia	3	5,4	5,4	100,0
Total	56	100,0	100,0	

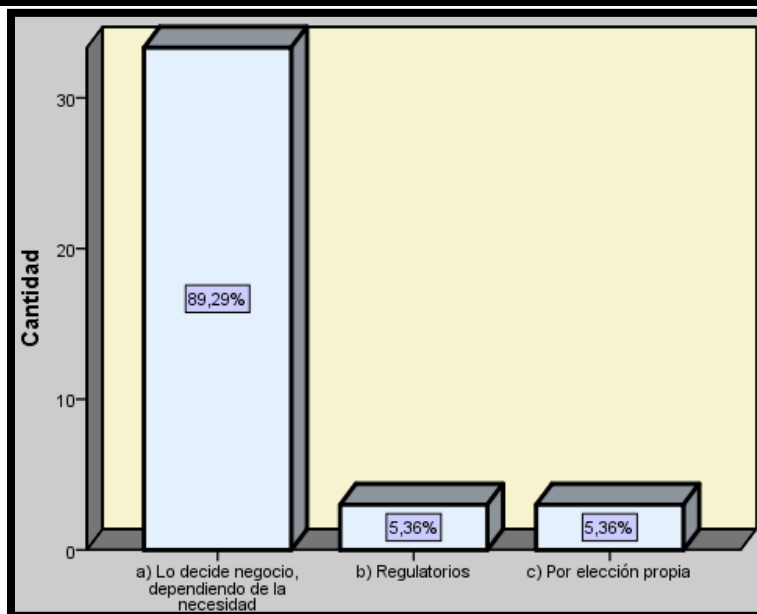


Figura 7 Resultados prioridad de los proyectos programadores de negocio

Se identifica que los programadores de negocio desarrollan aplicaciones estratégicas, aplicaciones que son parte de la operativa del negocio, los proyectos regulatorios no son la prioridad para ellos sino que constituyen una prioridad para los programadores de TI, por medio de los resultados se indica que los programadores de TI atienden con

más frecuencia requerimientos de entes nacionales como CNBS, esta es una de las causas por la que los ejecutivos de negocio contratan sus programadores para satisfacer sus necesidades de automatización e inteligencia de negocios porque los programadores de TI se enfocan en aplicaciones orientadas a satisfacer requerimientos regulatorios y dejan por último requerimientos que ayudan a mejorar la operativa del negocio.

4.1.2 VARIABLE METODOLOGÍA ÁGIL DE DESARROLLO DE SOFTWARE

⇒ ¿Qué aspectos de su forma de trabajo o la metodología adoptada en base a sus conocimientos le gustaría mejorar?

Tabla 9 Frecuencia aspectos a mejorar en metodologías utilizadas por programadores de negocio.

Opciones	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos				
a) Reducir el número de fases	12	21,4	21,4	21,4
b) Reducir el número de documentos a realizar	16	28,6	28,6	50,0
c) Mejorar las herramientas de colaboración con los usuarios	28	50,0	50,0	100,0
Total	56	100,0	100,0	

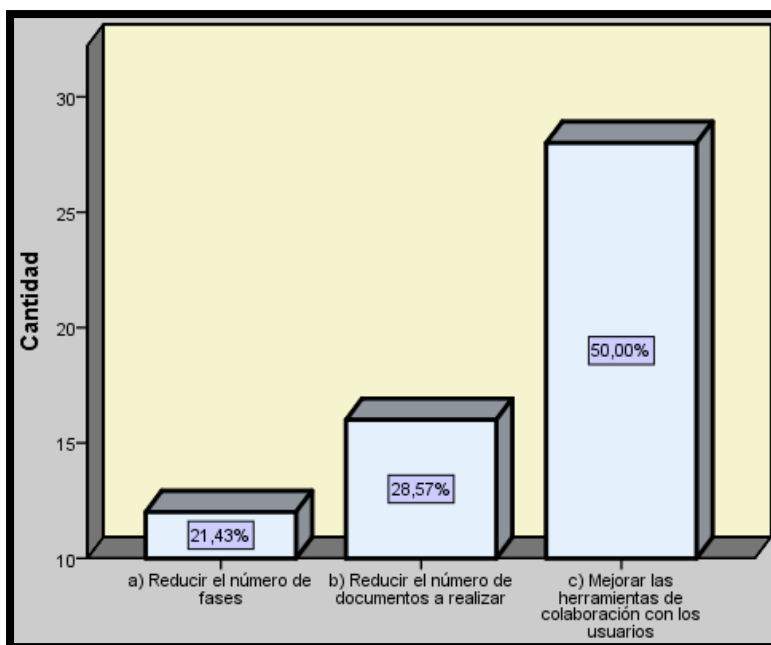


Figura 8 Resultado aspectos a mejorar en metodologías utilizadas por programadores de negocio

La mayoría de programadores de negocio o externalizados necesitan mejorar la forma en que se comunican con los usuarios, una de las características de las metodologías ágiles de desarrollo es precisamente mejorar la colaboración del equipo por lo que este resultado ayuda a determinar si realmente una metodología ágil podría ser útil para ejecutar diferentes proyectos del banco, adicionalmente los programadores que trabajan con negocio consideran que necesitan reducir el número de documentos que realizan actualmente, en donde la documentación no debe ser la prioridad sino los resultados y la forma de trabajo orientada a resultados óptimos.

⇒ ¿Cada cuánto el usuario adiciona requerimientos que no estaban establecidos inicialmente como parte del alcance?

Tabla 10 Requerimientos nuevos una vez finalizado el proyecto, programadores de negocio

Opciones	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos	a) Siempre	28	50,0	50,0
	b) Casi nunca	5	8,9	58,9
	c) Muy pocas veces	23	41,1	100,0
	Total	56	100,0	100,0

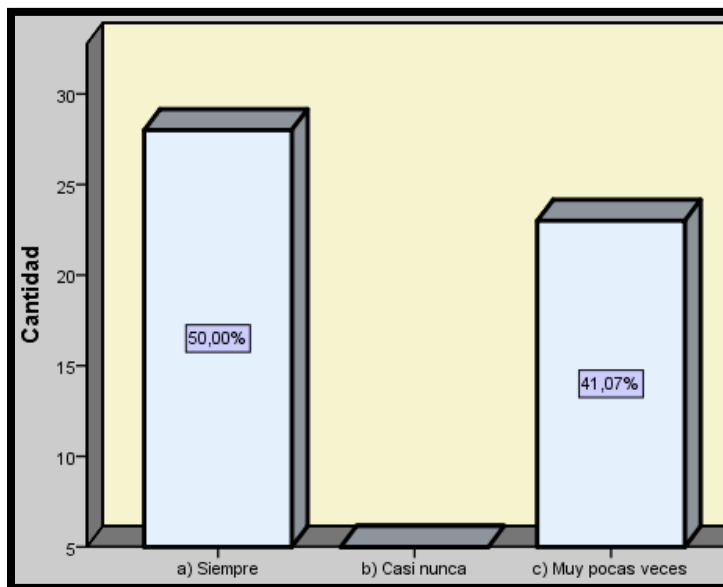


Figura 9 Resultado requerimientos nuevos una vez finalizado el proyecto, programadores de negocio

Los programadores de negocio indican que los usuarios agregan requerimientos nuevos una vez que termina la programación del proyecto mismos que no estaban establecidos en el plan de proyecto inicial, lo cual indica que el banco maneja proyectos que suelen cambiar mucho teniendo un nivel de complejidad alto, una de las características de las metodologías ágiles es la flexibilidad que tienen al cambiar requerimientos a lo largo de la ejecución del proyecto, si el usuario pide algo nuevo o una modificación esta es atendida debido a que el seguimiento es alto en estas metodologías de desarrollo, es por eso que al tomar en cuenta estos resultados se

puede tener una mejor orientación sobre que metodología sería más favorables aplicar para satisfacer las necesidades de negocio y sus usuarios.

¿Consideró importante la retroalimentación de avances durante un proyecto?

Tabla 11 Frecuencia respecto al reporte de avances del proyecto, usuario

Opciones	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos	a) Sí, siempre deben existir	6	85.7	85.7
	b) No, se espera la finalización del proyecto completo.	1	14.3	100.0
	c) Si el proyecto es grande si es necesario.			
Total	7	100.0	100.0	

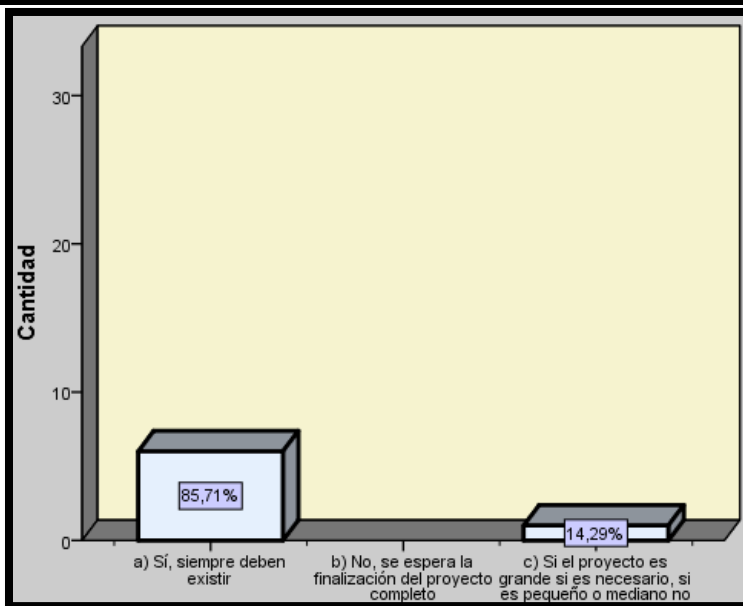


Figura 10 Resultado usuario de negocio respecto al reporte de avances del proyecto

⇒ ¿Con qué frecuencia presentó avances o entregables el analista programador a lo largo del proyecto?

Tabla 12 Frecuencia respecto de avances del proyecto

Opciones	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos a) Avances diarios	1	14.3	14.3	14.3
b) Entrega total (solamente un avance)	1	14.3	14.3	28.6
c) Avances periódicos (dependiendo del tipo de proyecto)	5	71.4	71.4	100.0
Total	7	100.0	100.0	

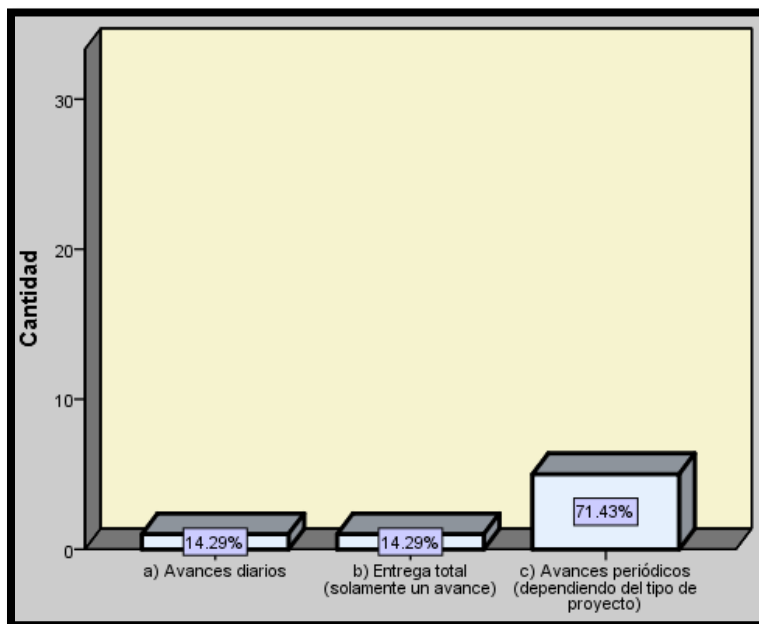


Figura 11 Resultado usuario de negocio respecto al reporte de avances del proyecto

Los usuarios o ejecutivos de negocio del banco consideran que siempre es necesario que existan reuniones de avances del proyecto y según las entrevistas realizadas a los ejecutivos esto depende de la complejidad y tipo de proyecto, si es un proyecto pequeño quizá necesiten la entrega final sin reuniones periódicas de avance, en este punto es importante identificar la complejidad de la mayoría de proyectos para identificar que metodología ágil se adapta al tipo de retroalimentación que los usuarios necesitan, inclusive en una de las entrevistas se mencionó que para que un proyecto

fuera finalizado siempre se debe dar seguimientos a los programadores de negocio asignados, caso contrario el proyecto se podría retrasar en tiempo y ejecución.

⇒ ¿Qué tipo de metodologías le dan mayor valor a un proyecto, las metodologías tradicionales (Enfoque en planificación y documentación) o las metodologías ágiles (enfoque en comunicación y resultados del proyecto)?

Tabla 13 Frecuencia tipo de metodología seleccionada por programadores de negocio o programadores externos

Opciones	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
a) Metodologías ágiles	63	65,6	65,6	65,6
b) Metodologías tradicionales	24	25,0	25,0	90,6
c) No estoy seguro	9	9,4	9,4	100,0
Total	96	100,0	100,0	

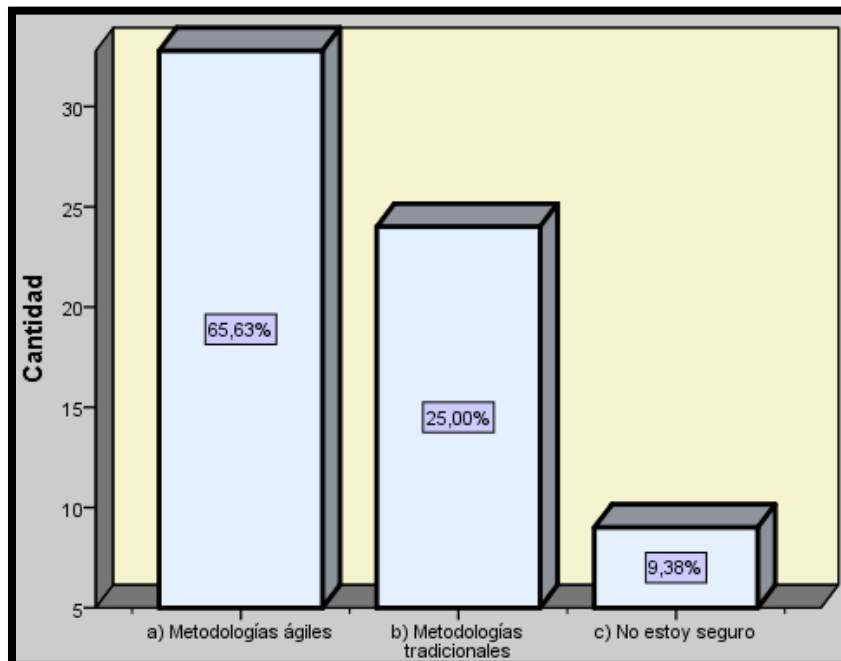


Figura 12 Resultado tipo de metodología seleccionada por programadores de negocio o programadores externos

La mayoría de los programadores prefieren enfocarse en la comunicación y resultados de un proyecto de software en lugar de la documentación y planificación del mismo, inclusive los programadores de TI que trabajan con la metodología tradicional que incluye como parte de su flujo la documentación de cada una de las fases de desarrollo de software y documentación de vistos buenos y aprobaciones de expertos. Es importante recalcar que la metodología puede llegar a depender del tipo de proyecto a desarrollar (de su tamaño, complejidad, entorno cambiante), el objetivo del estudio es identificar una metodología que puedan utilizar todos los programadores al mismo tiempo.

⇒ ¿Qué ambiente de pruebas utiliza para replicar los escenarios de prueba?

Tabla 14 Frecuencia ambiente de pruebas utilizado por programadores de negocio

Opciones	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
a) Negocio brinda uno	15	26,8	26,8	26,8
b) Creo mi propio ambiente de pruebas	37	66,1	66,1	92,9
c) Hago las pruebas directamente en producción	4	7,1	7,1	100,0
Válidos				
Total	56	100,0	100,0	

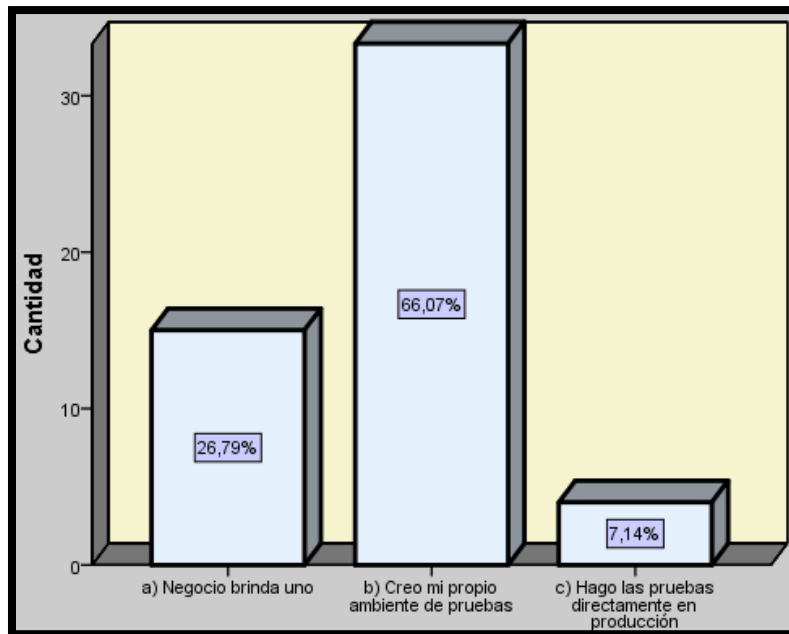


Figura 13 Resultado ambiente de pruebas utilizado por programadores de negocio

Los programadores de negocio y externos utilizan un ambiente de pruebas que es propio, este es un riesgo operativo grande para el banco ya que contar con ambientes de prueba creados por el mismo programador sin ningún estándar puede conllevar a la obtención de objetos de software mal probados afectando el flujo normal de trabajo del negocio cuando fallen, adicionalmente esto le añade tiempo de terminación al proyecto de software, montar un ambiente de pruebas con las tablas, archivos, aplicativos, base de datos, etc. consideran un tiempo mayor en la ejecución del proyecto, por lo que se identifica que se debe realizar un plan de acción en base a este hallazgo, adherir estos ambientes a los de la gerencia de TI puede ser una buena forma de mitigar este riesgo.

Al consultar si le gustaría seguir un flujo de trabajo establecido por la empresa para cual labora Los programadores de negocio si consideran importante seguir un patrón de trabajo para los proyectos de software, no se encuentran renuentes a seguir con su forma de trabajo, por lo que se determina que la resistencia al cambio no sería un problema al ejecutar la aplicabilidad de este trabajo de tesis, la idea de implementarlo para la empresa no solamente es eliminar riesgos, ahorrar costos, organizar, controlar y mejorar el gobierno de TI en desarrollos de software sino que también los

programadores se beneficien ya que sus habilidades y productividad sería aprovechada al máximo, recomendando un flujo en el que inviertan su tiempo en las fases donde se necesita más experiencia de su parte.

⇒ ¿En qué parte del proceso de desarrollo de software se enfoca más, da más prioridad o dedica más tiempo?

Tabla 15 Frecuencia tiempo invertido en fases del proceso de desarrollo programadores de negocio o programadores externos

Opciones	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
a) Requerimientos	12	12,5	12,5	12,5
b) Análisis	42	43,8	43,8	56,3
c) Diseño	8	8,3	8,3	64,6
d) Programación	22	22,9	22,9	87,5
e) Pruebas	10	10,4	10,4	97,9
f) Seguimiento y soporte	2	2,1	2,1	100,0
Total	96	100,0	100,0	

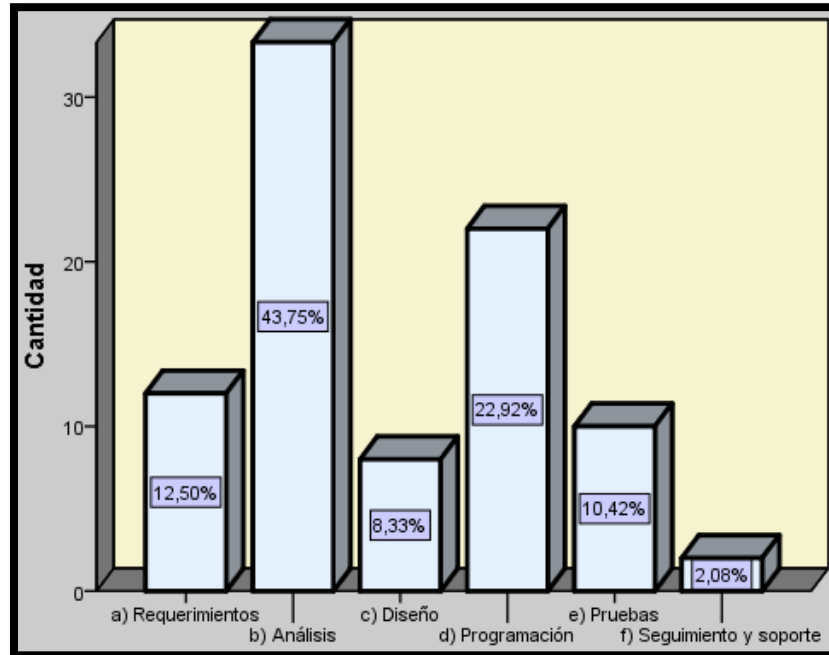


Figura 14 Resultado tiempo invertido en fases del proceso de desarrollo programadores de negocio o programadores externos

La mayoría de programadores invierten más tiempo en la fase de análisis y en la de programación, es importante que la solución en la parte de aplicabilidad tenga un artefacto que ayude a almacenar los aspectos más importantes del análisis y diseño de un mantenimiento o desarrollo de software, sería interesante que los programadores externos realicen un solo documento que ayude a documentar los puntos clave del proyecto y que sirva para el futuro en caso de que otro programador necesite consultar características de un proyecto ya cerrado como ayuda para un proyecto en ejecución, el objetivo sería evitar que se invierta tanto tiempo documentando características que no son tan importantes sino que lo inviertan precisamente analizando y diseñando.

⇒ ¿Qué elementos son más relevantes para enfocarse dentro de un proyecto de software?

Tabla 16 Frecuencia tiempo invertido en fases del proceso de desarrollo programadores de negocio o programadores externos

Opciones	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos				
a) Documentación	12	12,5	12,5	12,5
b) Comunicación con los involucrados del proyecto	49	51,0	51,0	63,5
c) Retroalimentación o seguimiento	23	24,0	24,0	87,5
d) Enfoque en programación (Prueba y error)	12	12,5	12,5	100,0
Total	96	100,0	100,0	

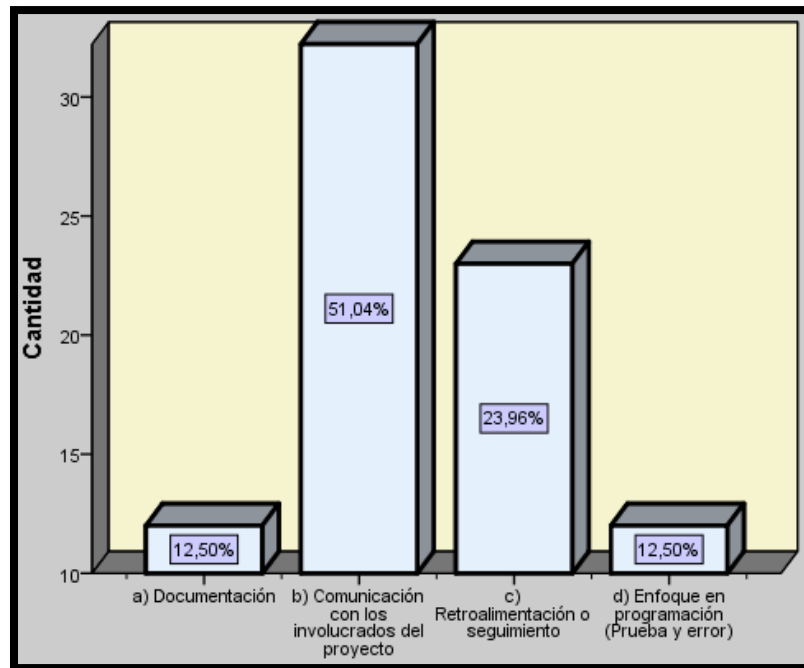


Figura 15 Resultado elementos de enfoque dentro de un proyecto de software, programadores de negocio

Se identifica que los programadores consideran más importante enfocarse en la comunicación con los involucrados del proyecto así como en la retroalimentación o seguimiento del mismo en lugar de enfocarse en la documentación, las posibles opciones de respuesta de estas preguntas estaban redactadas con características de las diferentes metodologías ágiles escogidas en el marco teórico, Scrum estaba relacionada con la comunicación, Crystal Clear estaba relacionada con la retroalimentación y programación extrema con el enfoque de a prueba y error, lo anterior con la idea de identificar que metodología se adaptaría mejor en base a la forma de trabajo del banco pero también en base al tipo, complejidad y magnitud de proyectos que manejan.

En las figuras anteriores se observa que los programadores de TI también consideran que la retroalimentación o seguimiento entre los involucrados del proyecto es clave para éxito del mismo, lo anterior ayuda también a determinar el rumbo de metodología ágil que se debe seleccionar, los resultados de los programadores de TI son similares, por lo que podría aplicarse esta metodología en el área de sistemas también en caso de

que el tipo de proyecto aplique de otra forma los programadores de TI podrían continuar utilizando su actual metodología de desarrollo de software que es tradicional y aplicar la nueva solamente a los programadores de negocio y los externalizados que son a quienes se enfoca el problema principal de la investigación.

⇒ ¿Qué forma de trabajo es más importante en un proyecto?

Tabla 17 Frecuencia forma de trabajo programadores de negocio

Opciones	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos	a) Reuniones diarias, equipos organizados, respuesta al cambio, motivación del equipo	20	35,7	35,7
	b) Comunicación, disminución de trabajos innecesarios, entrega frecuente, flexibilidad de trabajo	30	53,6	53,6
	c) Garantizar la calidad del software, forma de trabajo a prueba y error, cambio de requerimientos iniciales durante el proyecto	6	10,7	100,0
Total	56	100,0	100,0	

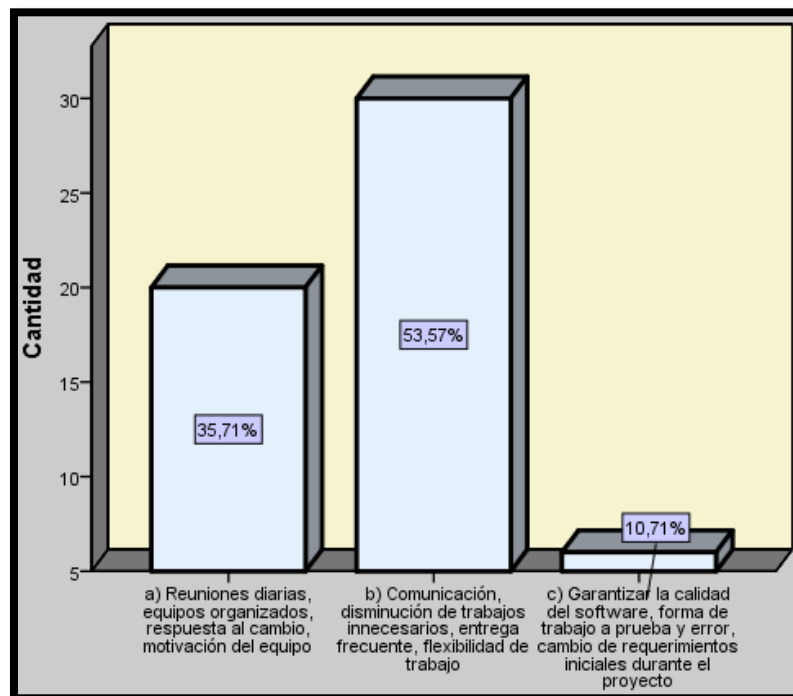


Figura 16 Resultado forma de trabajo programadores de negocio

La mayoría de los programadores considera la metodología ágil Crystal Clear como una metodología que se adaptaría mejor de acuerdo a sus características de disminución de trabajos innecesarios, flexibilidad en el cambio y comunicación, en los resultados el segundo lugar es para Scrum que es una metodología que plantea un seguimiento más frecuente y entregables más periódicos, en combinación con resultados anteriores en donde se descubre el enfoque en análisis, enfoque en retroalimentación y ahora flexibilidad en el cambio de requerimientos se tiene un panorama más claro de la opinión que tiene los programadores respecto a lo que podría mejorar su forma de trabajo en un proyecto en base a la experiencia que tienen en ellos.

⇒ ¿Qué características disminuyen costos en un proyecto cuando los recursos son Outsourcing?

Tabla 18 Frecuencia elementos que disminuyen costos para programadores externos

Opciones	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
	40	41,7	41,7	41,7
a) Menos tiempo en documentación	10	10,4	10,4	52,1
b) Trabajo en equipo	18	18,8	18,8	70,8
Válidos c) Adaptabilidad de acuerdo al proyecto	16	16,7	16,7	87,5
d) Mayor enfoque en programación	12	12,5	12,5	100,0
Total	96	100,0	100,0	

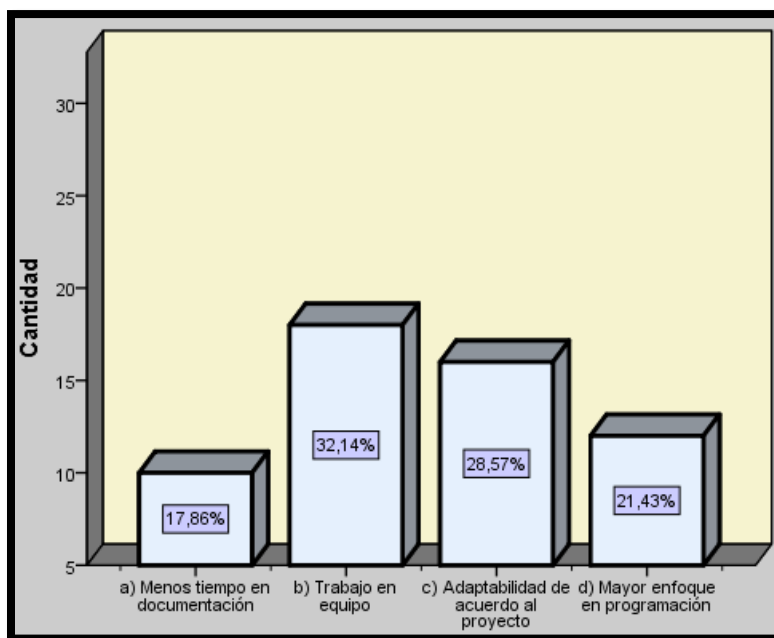


Figura 17 Resultado elementos que disminuyen costos para programadores externos

Según los programadores de negocio y externos el trabajo en equipo es el elemento más importante para disminuir costos o tiempo invertido en un proyecto, este resultado ayuda a identificar que ciertas partes del proceso de desarrollo pueden ser delegadas, en la gráfica anterior se indica que cuando se trata de trabajar con personal externo se puede trabajar en equipo, en la parte de aplicabilidad se podría establecer un procedimiento que implique la participación de los roles necesarios para hacer que el trabajo en equipo refleje una mayor productividad en los proyectos del banco, ese trabajo en equipo puede distribuirse en la parte de documentación y focalizar de manera más profunda al programador de negocio a la parte de programación y adaptabilidad del proyecto que fueron elementos con porcentajes similares en los resultados.

⇒ ¿Cuál es el grado de complejidad en la mayoría de proyectos del banco?

Tabla 19 Frecuencia grado de complejidad de los proyectos del banco.

Opciones	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos a) Alto	79	82.3	82.3	82.3
b) Bajo	17	17.7	17.7	100.0
Total	96	100.0	100.0	

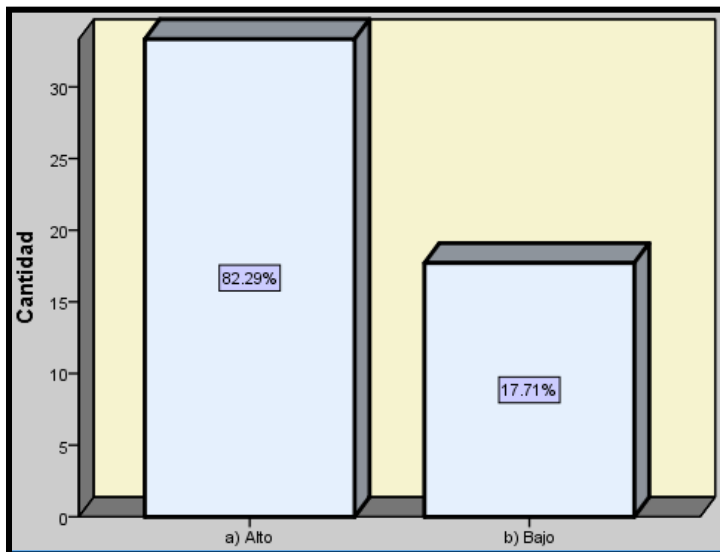


Figura 18 Resultado grado de complejidad de los proyectos del banco.

Se identifica que los proyectos que maneja el banco en su mayoría son complejos o largos porque tienen varios requerimientos que necesitan tiempo para su ejecución en análisis, diseño programación y pruebas, existen metodologías ágiles que no funcionan con proyectos muy grandes y complejos, así como también existen metodologías que han sido orientadas para trabajar con proyectos que involucran una mayor cantidad de tiempo y análisis, por lo que la metodología ágil que el banco necesite debe contar con la característica de manejo de proyectos según su tamaño y complejidad, dado que algunos de los programadores realizan sus proyectos de forma individual, no comparten tareas que ayude a facilitar en ciertos tipos de proyecto su terminación eficaz.

⇒ ¿Cuál es el grado de interacción con el usuario final en sus proyectos?

Tabla 20 Frecuencia grado de interacción con el usuario final en proyectos.

Opciones	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos a) Alta	52	54.2	54.2	54.2
b) Bajo	44	45.8	45.8	100.0
Total	96	100.0	100.0	

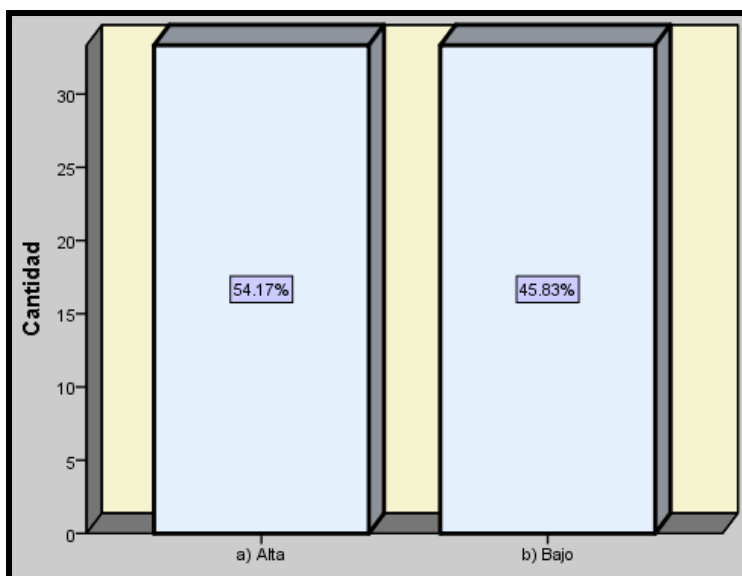


Figura 19 Resultado grado de interacción con el usuario final en Proyectos

El grado de interacción que los programadores de negocio tienen con los usuarios que harán uso de sus plataformas es alta dando como resultado una mejor satisfacción del usuario con las plataformas desarrolladas, sin embargo la interacción con el usuario final en los proyectos de desarrollo realizados por programadores de TI es relativamente baja lo cual puede dar como resultado que las aplicaciones desarrolladas tenga poca satisfacción una vez finalizadas, la metodología ágil a proponer debe tener la característica de verificar que al momento de iniciar un proyecto y durante todo el proceso de desarrollo del proyecto de software se asegure la participación activa de todos los interesados para canalizar los requerimientos adecuadamente.

4.1.3 VARIABLE PROYECTOS DE SOFTWARE

⇒ ¿En qué nivel se cumple con los tiempos de finalización de proyecto acordados con los usuarios?

Tabla 21 Frecuencia cumplimiento de tiempo de finalización de los proyectos

Opciones	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos	a) Siempre	25	44,6	44,6
	b) Casi nunca	3	5,4	50,0
	c) A veces	28	50,0	100,0
	Total	56	100,0	100,0

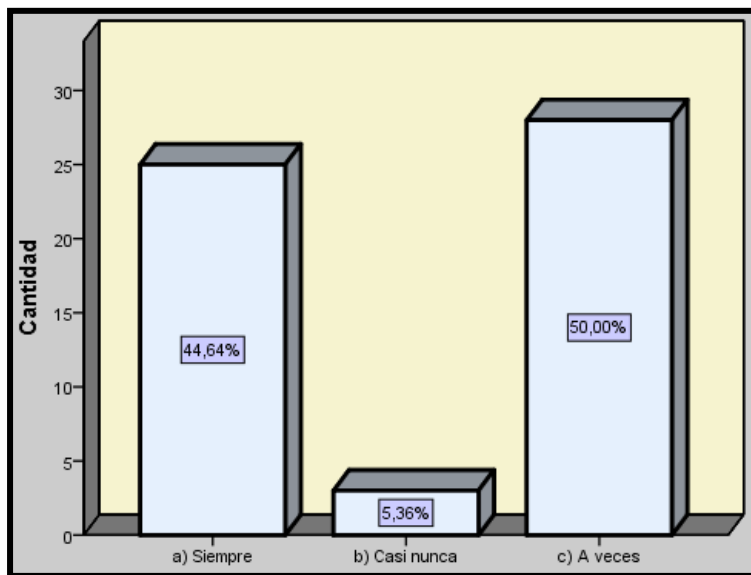


Figura 20 Resultado cumplimiento de tiempo de finalización de los proyectos.

Se muestra que existe un problema en base al tiempo en el cual los proyectos de software son finalizados donde se logra evidenciar que la mayoría de los proyectos de software ejecutados por los programadores de Negocio y programadores de TI no son finalizados en el tiempo, causando molestias en base a las fechas de finalización de las aplicaciones solicitadas, esta información es útil al momento de analizar la metodología ágil de desarrollo que pueden usar los programadores que de garantía a que los proyectos se ejecuten en el tiempo acordado satisfaciendo las necesidades de negocio y de los usuarios.

4.1.4 VARIABLE PROCESO DE GOBIERNO DE TI

⇒ ¿Cuánto tiempo dedicó el analista programador para la recolección de requerimientos?

Tabla 22 Frecuencia tiempo invertido en la recolección de requerimientos

Opciones	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos				
a) De 3 a 8 horas	2	28.6	28.6	28.6
b) De 8 a 16 horas	4	57.1	57.1	85.7
c) Ninguna hora	1	14.3	14.3	100.0
Total	7	100.0	100.0	

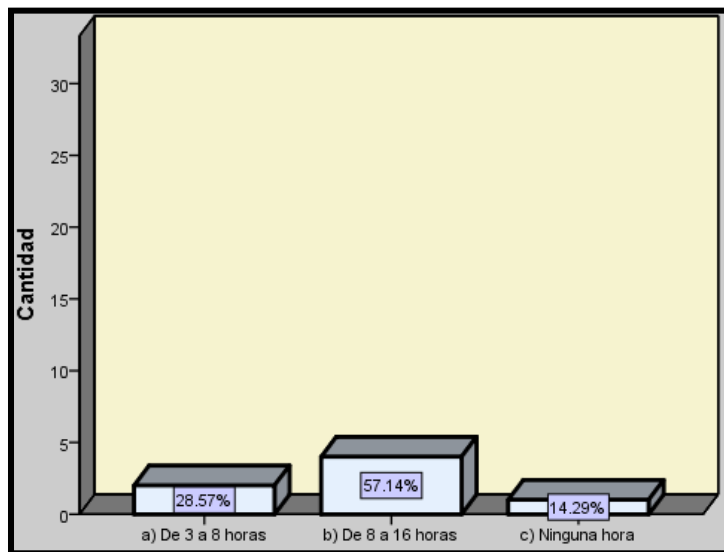


Figura 21 Resultado tiempo invertido en la recolección de requerimientos

El tiempo dedicado al levantamiento de requerimientos por parte de los programadores de negocio y programadores subcontractados por las gerencias de negocio es relativamente bajo ya que se considera que el tiempo dedicado para entender lo que el usuario o ejecutivo de negocio quiere en la nueva aplicación no suficiente para que el software cumpla con las funcionalidades solicitadas, la metodología ágil de desarrollo a utilizar debe tomar consideración que el tiempo dedicado a esta fase en un proyecto de desarrollo sea suficiente para comprender la funcionalidad de la aplicación, garantizando que al finalizarla no se tenga fallas por falta de comprensión en los

requerimiento iniciales o porque no se tomó en cuenta todos los elementos necesarios para que el proyecto fuera exitoso.

⇒ ¿Las pruebas realizadas previas a la puesta en producción fueron claras, el programador explicó las funcionalidades y estas fueron exitosas?

Tabla 23 Frecuencias satisfacción en las pruebas del proyecto

Opciones	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos a) Parcialmente exitosas	3	42.9	42.9	42.9
b) Totalmente exitosas	4	57.1	57.1	100.0
c) No fueron exitosas	0	0	0	0
Total	7	100.0	100.0	

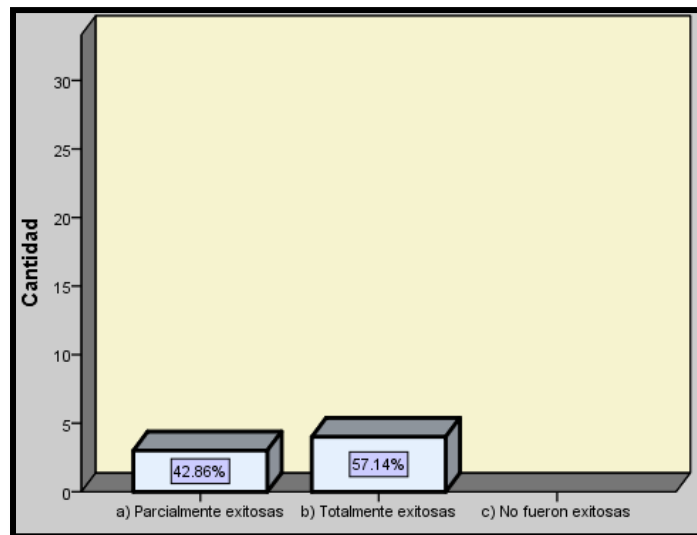


Figura 22 Resultado satisfacción en las pruebas del proyecto

Se muestra que las pruebas realizadas al momento de poner la aplicación a producción no son totalmente exitosas y por consiguiente no siempre son claras dando como resultado un margen de error lo cual provoca retraso en la entrega del proyecto final, este resultado indica que existe una debilidad al momento de realizar la fase de pruebas y puesta en producción de la aplicación lo cual proporciona información importante que ayude a comprender mejor que tipo de metodología ágil es recomendable a usar por los programadores de negocio y programadores

subcontratados para fortalecer esta fase en un proyecto de desarrollo, garantizando la calidad de las aplicaciones y su tiempo de vida útil.

⇒ ¿El programador cumplió con las fechas de cierre o entregables del proyecto?

Tabla 24 Frecuencias cumplimiento en la fechas de cierre o entregables del proyecto

Opciones	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
a) Sí	3	42.9	42.9	42.9
b) No	4	57.1	57.1	100.0
Válidos c) No existían acuerdos	0	0	0	0
Total	7	100.0	100.0	

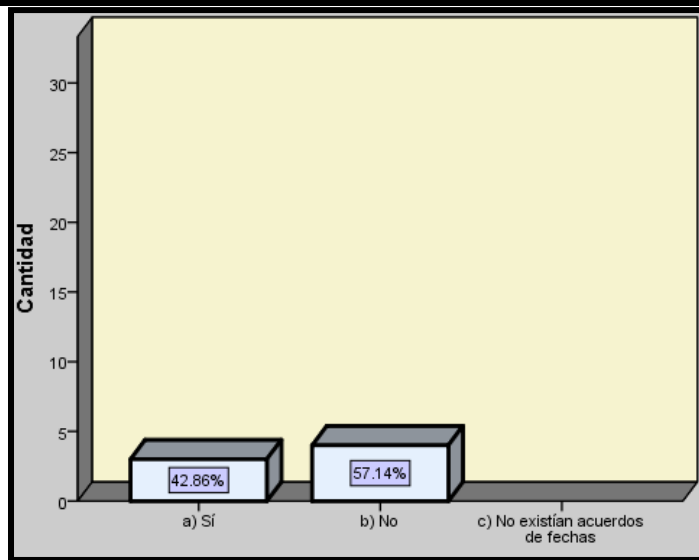


Figura 23 Resultado cumplimiento en la fechas de cierre o entregables del proyecto

Con respecto a las fechas de cierre y entregables que los programadores deben realizar el cada proyecto de software se puede validar inconformidad de los usuarios respecto a estas fechas de entrega de los proyectos desarrollados por programadores de negocio y programadores de TI, indicando que existe un problema al momento de considerar las fechas de ejecución de un proyecto de software, este dato da la apertura para que al momento de considerar una metodología ágil de desarrollo a usar por los programadores del banco se considere que una de sus características sea el adecuado

manejo de los tiempos en el cual un proyecto debe finalizar exitosamente ajustándose a las fechas de entrega sin perder de vista la calidad en ejecución, costos y efectividad.

⇒ ¿Se documentó un plan de proyecto donde se estipulara las fechas de cumplimiento para el cierre de proyecto?

Tabla 25 Frecuencias uso de documentos de plan de proyecto

Opciones	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos a) Sí	7	100.0	100.0	100.0
b) No	0	0	0	0

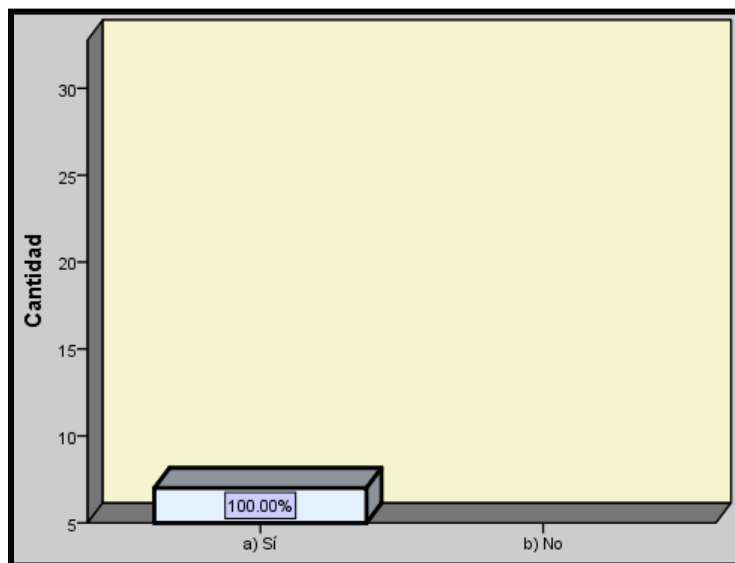


Figura 24 Resultado uso de documentos de plan de proyecto

Al iniciar un proyecto de desarrollo de software los programadores deben documentar un plan de proyectos donde se realiza un plan de acción con las fechas de cumplimiento de cada una de las actividades que intervienen en un proyecto de software, esta información indica la importancia que el usuario da a la ejecución de un plan de proyecto desde su fase inicial hasta la fase final al desarrollar una aplicación considerando todas las actividades, recursos, costos y procesos necesarios para la entrega final del software, información útil a considerar una metodología ágil de desarrollo de software que ayude a mantener dicho indicador y evitar que esta actividad

pierda su valor, por ser una de las principales tareas a realizar el momento de emprender un proyecto de desarrollo.

4.1.5 VARIABLE FALLAS EN EL PROCESO ACTUAL DE DESARROLLO

⇒ ¿Cuáles son las razones más recurrentes por las cuáles fallan los proyectos de software?

Tabla 26 Frecuencia de las causas de fallas en proyectos de software, programadores de negocio

Opciones	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos a) Tiempo invertido en muchos proyectos a la vez	10	17,9	17,9	17,9
b) Falta de un ambiente de pruebas fiable	22	39,3	39,3	57,1
c) Entendimiento erróneo de los requerimientos iniciales	5	8,9	8,9	66,1
d) Disponibilidad de herramientas y accesos necesarios	6	10,7	10,7	76,8
e) Nivel de experiencia	1	1,8	1,8	78,6
f) Al finalizar el proyecto el usuario cambia requerimientos o incluye nuevos	12	21,4	21,4	100,0
Total	56	100,0	100,0	

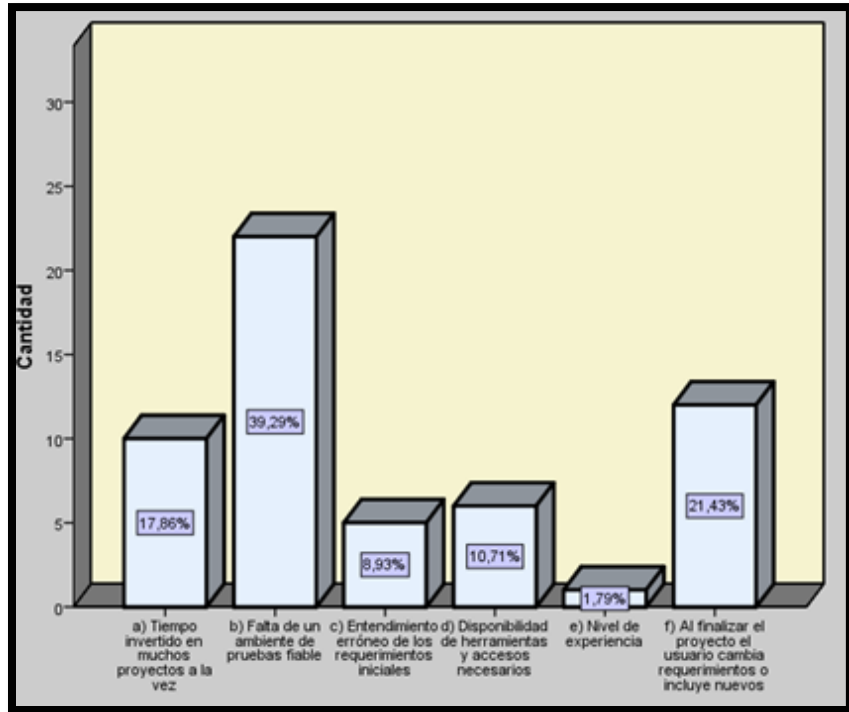


Figura 25 Resultado de las causas de fallas en proyectos de software, programadores de negocio.

Tabla 27 Frecuencia de las causas de fallas en proyectos de software, programadores de TI

Opciones	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
a) Tiempo invertido en muchos proyectos a la vez	6	15.0	15.0	15.0
b) Entendimiento erróneo de los requerimientos iniciales	12	30.0	30.0	45.0
c) Nivel de experiencia	4	10.0	10.0	55.0
d) Al finalizar el proyecto el usuario cambia requerimientos o incluye nuevos	18	45.0	45.0	100.0
Total	40	100.0	100.0	

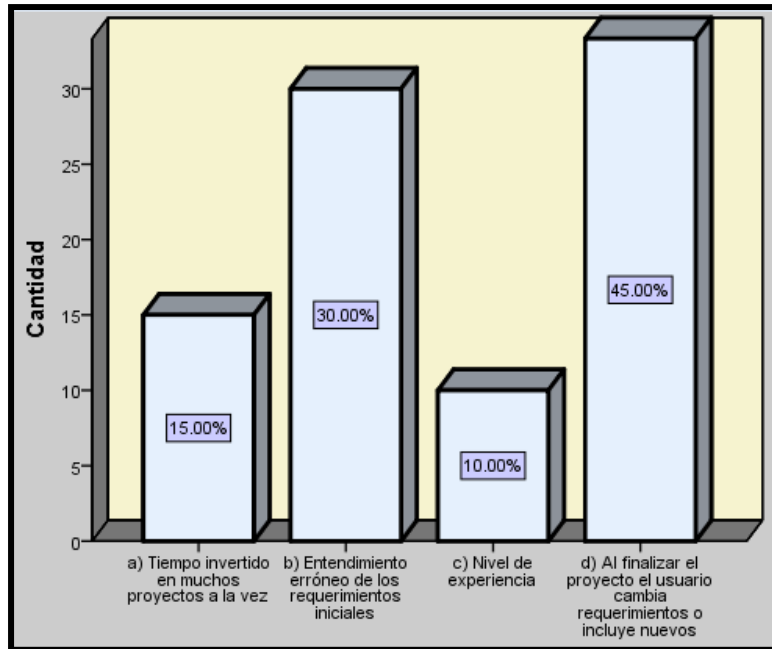


Figura 26 Resultado de las causas de fallas en proyecto de software, programadores TI

Los programadores de negocio como programadores externos y los programadores de TI concuerdan que las fallas más comunes al momento de desarrollar una aplicación de software se da porque los usuarios cambian o agregan nuevos requerimientos al proyecto una vez que este es finalizado lo cual da como resultado reprogramar la aplicación y dedicar más tiempo para su puesta en producción, también se observa que al no contar con un ambiente de pruebas fiable para realizar la etapa de pruebas puede causar fallas que eviten la finalización exitosa de un proyecto de software, este es un indicador clave que ayude a analizar cuál debe ser la metodología de desarrollo ágil a considerar para el manejo flexible de requerimientos durante la fase de desarrollo de un proyecto.

⇒ ¿Qué tan seguido sus desarrollos de software fallan en producción o una vez finalizados se identifican muchos ajustes?

Tabla 28 Frecuencias fallas o ajustes realizados en los desarrollos de software

Opciones	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos				
a) Siempre	6	6.3	6.3	6.3
b) Muy pocas veces	80	83.3	83.3	89.6
c) Nunca	10	10.4	10.4	100.0
Total	96	100.0	100.0	

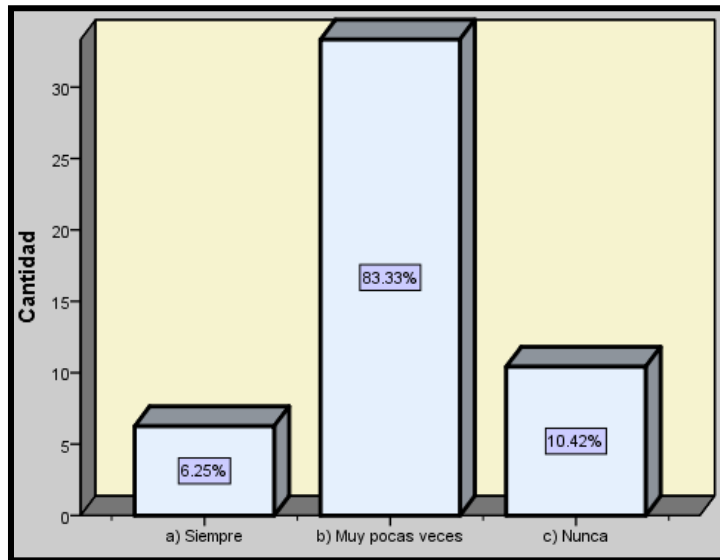


Figura 27 Resultado fallas o ajustes realizados en los desarrollos de software

Los programadores del banco concuerdan en que al momento de finalizar un proyecto y ponerlo en producción las aplicaciones casi no necesitan ajustes y son muy pocas las ocasiones en las que fallan o necesitan ajusten por causa de algún problema en la lógica del programa, estas fallas o ajustes en su mayoría se dan porque el usuario desea agregar nuevos requerimientos antes no mencionados, esta información es útil para identificar que metodología de desarrollo ágil puede garantizarnos que el indicador permanezca con la misma tendencia o que mejore su valor en base a la calidad de las aplicaciones de software ajustándose a los requerimientos del usuario final y cumpliendo con los objetivos estratégicos de negocio regidos bajo la correcta supervisión de la gerencia de tecnología.

⇒ ¿Cuáles son los proyectos que cumplen con todos los requisitos de calidad la mayor parte del tiempo?

Tabla 29 Frecuencias calidad de los proyectos, usuarios de negocio

Opciones	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
a) Desarrollos de negocio	6	85.7	85.7	85.7
Válidos b) Desarrollos de TI	1	14.3	14.3	100.0
Total	7	100.0	100.0	

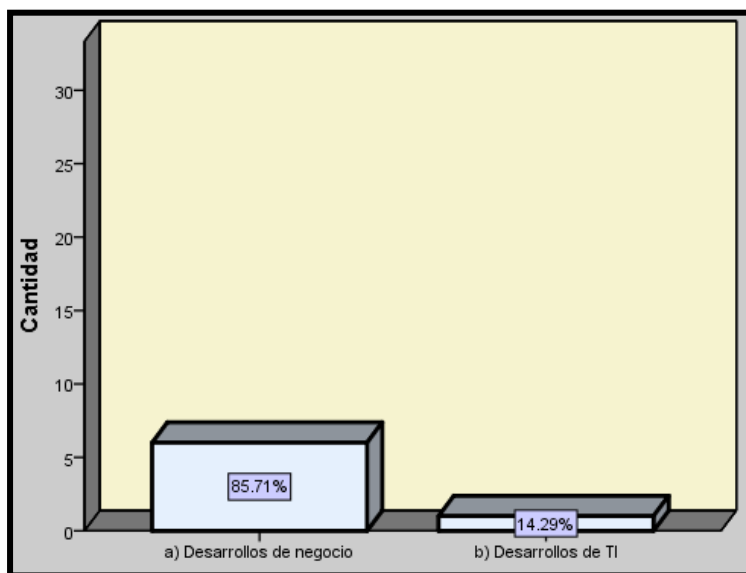


Figura 28 Resultado calidad de los proyectos, usuarios de negocio

Se logra identificar que los usuarios tienen una mayor satisfacción con las aplicaciones que desarrollan los programadores de negocio o programadores externos, que con las aplicaciones desarrolladas por los programadores de TI, esto a causa de que los programadores de negocio conocen mejor la operativa de la gerencia y ajustan sus aplicaciones en base a la automatización de las actividades que realizan los usuarios que harán uso de sus programas y los objetivos de los ejecutivos para dar valor agregado a las estrategias de las gerencias de negocio, la metodología ágil de desarrollo a utilizar debe considerar un adecuado análisis tomando en cuenta la prioridad de la aplicación para el área que hará uso de ella, satisfaciendo los requerimientos del usuario.

⇒ ¿Cuántas fallas o necesidades de ajustes se presentaron en producción cuando se instaló el proyecto final?

Tabla 30 Frecuencias Fallas en producción, usuarios de negocio

Opciones	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos	a) De 1 a 2	3	42.9	42.9
	b) De 2 a 5	3	42.9	85.7
	c) Más de cinco	0	0	0
	d) Ninguna	1	14.3	100.0
	Total	7	100.0	100.0

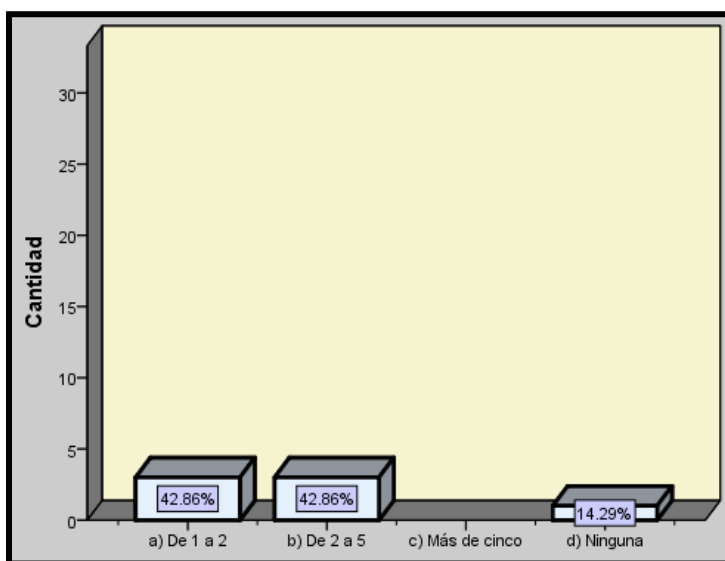


Figura 29 Resultado Fallas en producción, usuarios de negocio

El realizar ajustes a las aplicaciones desarrolladas por los programadores de negocio, programadores subcontratados y programadores de TI, se da debido a que aún finalizado el programa siempre se deben realizar ajustes por las fallas que se presentan en la etapa final del proyecto de desarrollo, y aunque no existan fallas en el programa se debe considerar la mejora continua para el ciclo de vida de cada programa de software, la metodología ágil a considerar debe tener la característica de realizar una fase efectiva en el análisis y desarrollo de la aplicación, para que al momento de ejecutar la fase de pruebas esta sea exitosa evitando ajustes por errores en la fase final del proyecto de software al momento de ponerla en producción dando como resultado un proyecto confiable e íntegro.

⇒ ¿Una vez terminado el proyecto cuánto tiempo de mantenimiento solicitó el encargado programador durante el uso activo de la herramienta?

Tabla 31 Frecuencias mantenimiento a lo largo de ciclo de vida del aplicativo, usuarios de negocio

Opciones	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos				
a) Cada dos meses	6	85.7	85.7	85.7
b) Anualmente	0	0	0	0
c) Más de dos años	1	14.3	14.3	100.0
Total	7	100.0	100.0	

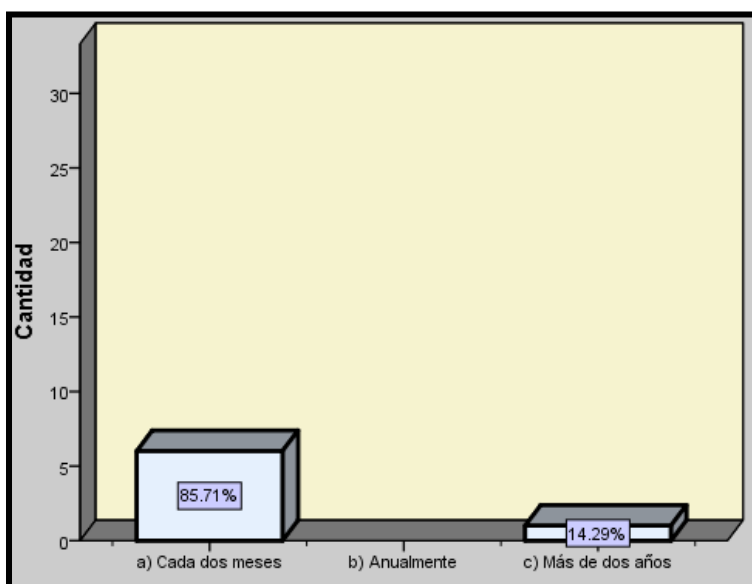


Figura 30 Resultado mantenimiento a lo largo de ciclo de vida del aplicativo, usuarios de negocio

Por medio de la opinión de los encuestados se observa que siempre es necesario estar realizando mantenimientos de forma periódica en las aplicaciones desarrolladas esto para alargar el ciclo de vida de la aplicación y realizar las mejoras continuas en el momento adecuado, este resultado ayuda a proporcionar información importante sobre cuál es la metodología ágil de desarrollo que mejor se adapte a la mejora continua de las aplicaciones de software y ayude a los sistemas a funcionar de acuerdo a las necesidades de negocio, manteniendo su disponibilidad al momento de brindar el apoyo

a los usuarios que hagan uso de su funcionalidad disminuyendo los tiempos en la ejecución de tareas y aumentando la productividad en las diferentes áreas del banco.

⇒ ¿Considera seguir solicitando desarrollos de software a programadores adheridos a negocio que no forman parte de TI?

Tabla 32 Frecuencias solicitud de desarrollo de aplicaciones

Opciones	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos a) Dependiendo del tipo de proyecto	6	85.7	85.7	85.7
b) Dependiendo de la disponibilidad de recursos	1	14.3	14.3	100.0
c) Dependiendo de los recursos monetarios	0	0	0	0
d) Dependiendo de las políticas del banco	0	0	0	0
Total	7	100.0	100.0	

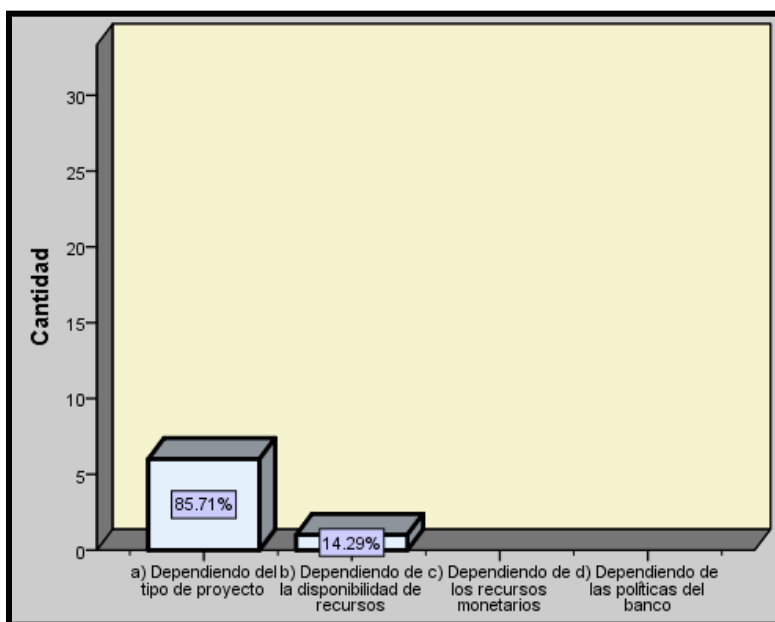


Figura 31 Resultado solicitud de desarrollo de aplicaciones

Los proyectos a ser asignados a los programadores de negocio y programadores subcontratados depende mucho del tipo de proyecto que desean sea desarrollado y la necesidad del negocio, en este caso para proyectos estratégicos de sus áreas prefieren asignarlos a los programadores contratados por sus propias gerencias, para

aplicaciones más complejas o de carácter regulatorio prefieren asignarlos a programadores de la gerencia de tecnología por los estándares de calidad que esta gerencia maneja, la metodología ágil de desarrollo a utilizar debe ayudar a que los programadores de negocio y programadores subcontractados cumplan con los objetivos y necesidades de negocio y que se puedan llevar a cabo proyectos de desarrollo en base a mejores prácticas.

⇒ ¿Cuál es el nivel de riesgo tecnológico asociado en la mayoría de proyectos que usted desarrolla?

Tabla 33 Frecuencias nivel de riesgo tecnológico asociados a los desarrollos de programadores de TI

Opciones	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos	a) Alto	23	57.5	57.5
	b) Bajo	17	42.5	100.0
	Total	40	100.0	100.0

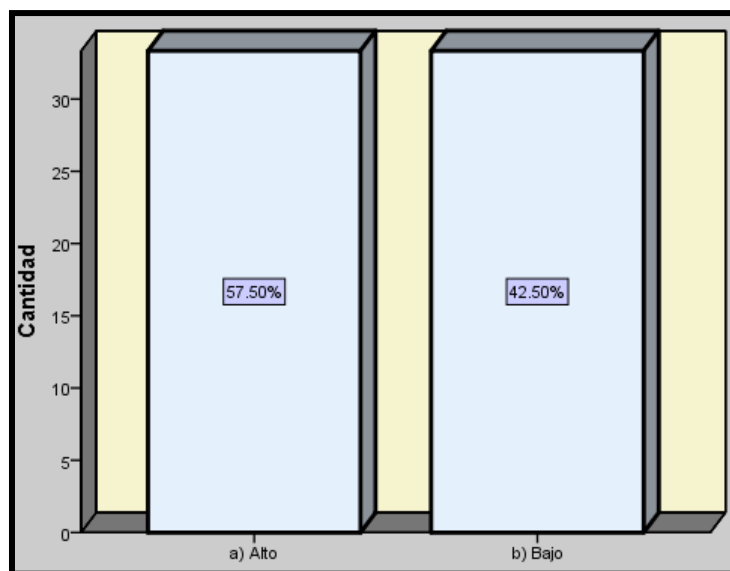


Figura 32 Resultado nivel de riesgo tecnológico asociados a los desarrollos de programadores TI

Los programadores de negocio y programadores de TI indican que en todo proyecto de software que está bajo su responsabilidad el nivel de riesgo que manejan en su

mayoría es relativamente alto, debido a que sus aplicaciones forman parte de la operativa diaria de estas áreas y son usadas por grandes cantidades de usuarios, sin embargo una parte de esta población indica que el nivel de riesgo en sus proyectos es baja, dando como resultado considerar todos los riesgos existente en un proyecto de software al momento de elegir una metodología ágil de desarrollo de software que ayude a ejercer un buen gobierno de TI para asegurar que siempre estén disponibles y tengan el soporte en el momento requerido.

4.2 RESULTADOS ENTREVISTAS

ENTREVISTA PARA EJECUTIVOS DE NEGOCIOS QUE CONTRATAN PERSONAL EXTERNO A TI PARA DESARROLLO DE SOFTWARE

Los ejecutivos consideran que el nivel de éxito que tienen los proyectos que desarrollan en su gerencia es alto, tienen un nivel de satisfacción elevado de los programadores que tienen en su gerencia sin embargo algunos de los proyectos importantes consideran delegarlos a la gerencia de TI, consideran que TI tiene un alto grado de calidad en sus proyectos, son profesionales y confían en las soluciones de automatizan que estos brindan. Al consultar cuál era el tipo de proyectos que más manejaban en el área las respuestas se diversificaron en proyectos medianos, grandes y pequeños, en todas las áreas existían estos tres tipos.

Cuando se consultó a los entrevistados quien desarrollaba la mayor cantidad de proyectos varios de ellos contestaron que la gerencia de negocio, otros la gerencia de TI, y otros ejecutivos de negocio no estaban seguros de quien desarrollaba más proyectos para su área, sin embargo todas la áreas habían seguido al menos un proyecto con TI y la mayoría tenía mayores requerimientos cumplidos con los programadores de negocio dado que estaban disponibles exclusivamente para requerimientos de dicha área. Sobre el servicio ofrecido por los programadores de negocio u externos, en base a calidad y tiempo de finalización de los proyectos; algunos ejecutivos manifestaron que debían dar seguimiento a los programadores de negocio para que terminaran con el proyecto asignado a tiempo.

Cabe mencionar que los ejecutivos están satisfechos con las habilidades profesionales de los programadores externos que contratan, así mismo uno de los ejecutivos no estaba totalmente satisfecho con el servicio ofrecido por TI, por otro lado al consultar por el nivel de importancia que tenía la parte de documentación de la planificación, acuerdos de fechas y diseño técnico del desarrollo de software se encontró que es importante documentar acuerdos de fecha de compromiso de finalización de proyecto, que es altamente importante planificar el proyecto y uno de ellos manifestó que los programadores estaban comenzando a documentar un archivo de diseño como parte de sus responsabilidades, aunque la mayoría de programadores de negocio no lo documentaban.

En base a la continuidad de negocio, se consultó si las aplicaciones que desarrollaban sus programadores permanecían siempre activas, las respuestas eran positivas ya que según los ejecutivos los programadores realizan mejoras y mantenimiento continuo, como se encuentran en las mismas áreas de negocio siempre están disponibles por cualquier mejora o problema que se presente, se puede concluir a partir de esta pregunta que los programadores de TI no siempre están disponibles para cada área, se podría recomendar asignar a un programador de Te haga proyectos para cada área así los ejecutivos de negocio no dependerían tanto de los programadores subcontratados que tienen.

Los programadores de TI no conocen a detalle los aplicativos que son desarrollados en negocio por lo que no pueden brindar soporte, respecto a las fallas más recurrentes que afectan el servicio sus aplicaciones los ejecutivos manifestaron que tiene problemas por fallas de red, problemas por rendimiento del servicio, y por parte de TI comentaron que a veces existe la falta de sentido de urgencia que tiene algunos especialistas para atender solicitudes, analizando esta respuesta se le puede atribuir la causa a la cantidad de especialistas en comparación con la cantidad de colaboradores de negocio contratados que estos tiene que atender, sería importante ver si fallan en los tiempos y calidad del servicio brindado por una sobre carga de trabajo.

Varios de los ejecutivos de negocio están satisfechos con la atención brindada por TI una vez que la solicitud de programación es atendida, sin embargo la distribución de recursos es un problema, para que se atienda una solicitud de programación esta debe entrar a una cola larga de atención por lo que existe la falta de disponibilidad para atenderlo en un tiempo óptimo, varios ejecutivos manifestaron esta debilidad por parte de la gerencia de TI, en estas colas de solicitudes de programación entran proyectos de todas las áreas y cada una de las áreas puede tener varios proyectos en espera de aprobación del comienzo.

Respecto al nivel de riesgo asociado a los proyectos desarrollados en el área los ejecutivos hablaron de la cantidad de aplicaciones caídas que se podía tener, en caso de que solo fuera una o pocas no lo consideraban un riesgo alto, otros manifestaron que las aplicaciones se encontraban alojadas en servidores virtuales por lo que lo tenían un riesgo bajo de pérdidas, sin embargo están consientes que siempre es necesario tener una gestión de identificación de riesgo para mitigarlos en tiempos oportunos evitando que el impacto sea mayor.

Entre las principales razones para hacer contratación de programadores en las áreas de negocio y no hacer uso de los recursos de TI se encontraba la falta de recursos en la gerencia de TI para atender solicitudes de programación y la urgencia y necesidad de tener el software o el requerimiento listo en el menor tiempo posible, sin embargo la satisfacción cuando TI atendía una solicitud de programación utilizando la metodología tradicional era buena. Entre las debilidades que los ejecutivos encontraron de TI al atender estas solicitudes se encontró que la duración de atención de las solicitudes de programación es de tres a cuatro meses, ese es el tiempo promedio en que un programador de TI se tarda en terminar un proyecto.

Entre los beneficios de contratar programadores externos o programadores adheridos a la gerencia sin enlace con TI los ejecutivos indicaron que se obtenía un software a la medida, calidad, posibilidad de desarrollar sin restricciones, recursos siempre disponibles, mayor grado de automatización, control de avances, disponibilidad de las aplicaciones, comunicación y mejor adaptación con requerimientos de clientes, el

conocimiento del negocio por parte del programador es más amplio debido a que se encuentra directamente en el área lo cual es una ventaja, se observa que podrían realizar la retroalimentación de los proyectos de una forma más seguida, facilitando así la adaptación de una metodología ágil que ayuda a realizar este seguimiento de forma creativa y organizada.

Existe un formulario desarrollado junto al gestor de gobierno de TI con clausula de confidencialidad que están firmando algunos de los programadores externos al momento del contrato que estipula las consideraciones correspondientes respecto a temas de patentes, derechos de autor de las aplicaciones para protección del banco, sin embargo solamente una de las áreas está utilizando el formulario. Adicionalmente al consultar por los proyectos que tienen un costo menor los ejecutivos de negocio manifiestan que los proyectos de negocio tienen un costo menor a los de TI porque se resuelven en menor tiempo, sin embargo el área debe proveerles las herramientas para desarrollar y TI ya tiene toda una infraestructura y áreas que se ocupan del control y administración de los objetos de software.

ENTREVISTA AL GESTOR DE GOBIERNO DE DESARROLLOS DE SOFTWARE DE TI

Dentro del plan de capacitaciones ofrecidas por del banco se incorporan mensualmente una capacitación referente al proceso actual de desarrollo de software que siguen los programadores de la gerencia de TI, estas capacitaciones son ofrecidas a los programadores de negocio residentes o nuevo, su objetivo es mostrar el proceso de desarrollo existente en TI para que entienda su flujo y logren aplicar las mejores prácticas en las fases de desarrollo de sus aplicaciones, tomando en consideración aquellas fases que son realmente necesarias, enfocándose más en la parte de programación que en la parte de documentación al desarrollar las aplicaciones satélites estratégicas en las áreas que han los han contratado.

Estas capacitaciones son realizadas por medio de conferencias presenciales prácticas que ayudan a adquirir conocimiento de forma más eficaz y rápido, sin embargo también

es útil realizar capacitaciones audiovisuales para una mejor comprensión de los programadores de negocio en base al proceso de desarrollo que siguen los programadores de la gerencia de tecnología y que se desea realicen los programadores de negocio al momento de adherir sus aplicaciones bajo el gobierno de TI.

Mediante una investigación profunda se observa que existen diferentes causas o motivos por las cuales las áreas de negocio deciden realizar sus propias aplicaciones sin la intervención de TI entre ellas se puede mencionar:

- ⇒ Proceso de desarrollo de software utilizado en la gerencia de TI es relativamente lento, hay muchas solicitudes de programación ingresadas al área de tecnología dando como consecuencia problemas al momento de asignarlas y darles prioridad por escases de recursos disponibles.
- ⇒ No existe un área en la gerencia de tecnología que desarrolle programas satélites en base a las estrategias y necesidad de automatización surgidas en cada una de las áreas que pertenecen a las gerencias de negocio.
- ⇒ No se puede obtener resultados de forma rápida cuando se emprende un proyecto de desarrollo de software debido a que el proceso actual de TI es demasiado largo lo cual da como consecuencia que las fases y sus actividades se realicen de forma lenta ocasionando atrasos en las fechas de entrega.
- ⇒ Existen problemas al momento de solicitar las aprobaciones del comité regional en desarrollos de TI, estos atrasos en las aprobaciones al brindar el visto bueno para poner en marcha un proyecto de software genera pérdidas de tiempo en la ejecución de un proyecto de software, este mismo problema persiste al momento querer incorporar la aplicación desarrollada en producción lo cual causa malestar y molestia al usuario final.

Debido a que las fases del procesos de desarrollo de software utilizado en la gerencia de tecnología puede llegar a ser largo y lento, es que las gerencia de negocio se han visto obligados a contratar programadores dentro de su áreas funcionales para satisfacer sus necesidades, se observa que las aplicaciones que más desarrollan fueran del gobierno de TI son de carácter estratégico ya que estos son prioridad para

los ejecutivos de negocio por encima de los proyectos estratégicos regulatorio del banco que son las aplicaciones de prioridad para la gerencia de tecnología.

Se considera que para implementar un adecuado gobierno de TI dentro de la institución financiera Bac Honduras no es recomendable adherir a los programadores de negocio al proceso actual de desarrollo de software por el tiempo de ejecución que este toma al desarrollar las aplicaciones, ya que la metodología de desarrollo actual usada en la gerencia de tecnología está más orientada a actividades correspondiente a la documentación del proyecto que se necesita para comenzar a programar lo cual lo convierte en un proceso largo y tedioso.

Existen varios riesgos asociados al tener programadores fuera del gobierno de TI entre los cuales se puede mencionar:

1. Que se realice un cambio en producción afectando la continuidad de negocio
2. Se realicen cambios en la data sin autorización
3. Falta de soporte por parte de la gerencia de TI al fallar las aplicaciones a consecuencia del desconocimiento de la herramientas y su funcionalidad
4. accesos no autorizados a usuarios sin visto bueno de gerencia
5. Falta de documentación o bitácoras de los cambios realizados en las aplicaciones.

Se considera adecuado que se deben migrar los objetos de software de las aplicaciones desarrolladas en negocio, sin embargo este es un proceso que debe ejecutarse poco a poco donde se da prioridad a migrar los objetos críticos (agregando componentes de seguridad – para dar permisos y accesos) con el objetivo de corregirlos y afinarlos, tarea a realizar en conjunto con los programadores que han desarrollado las aplicaciones esto por el conocimiento que ellos tienen, para que la gerencia de TI luego pueda adueñarse del código y así poder darles seguimiento y realizar la publicación en base a vistos buenos.

Es adecuado que los accesos a otorgar a los programadores de negocio deben ser manejados por la gerencia de tecnología para ejercer un adecuado gobierno de TI y

que estos accesos deben ser asignados por un experto de TI para que la actividad se realice de forma rápida y eficaz, disminuyendo tiempos de atención y aumentando los tiempos de validación, aprobación, permisos e ingresos necesarios a las herramientas que los programadores de negocio necesiten utilizar. Los tiempos que estos accesos debe durar dependen de la experiencia del programador ya que es esta uno de los principales motivos por el cual se puede prolongar la ejecución de una o varias actividades durante el proceso de desarrollo de una aplicación de software.

En base a la primicia de aplicar una metodología ágil de desarrollo a programadores que no pertenecen a la gerencia de TI esta sería la primicia más recomendable siempre y cuando los programadores se dediquen solo a la fase de programación ya que seguir el flujo de desarrollo de la gerencia de TI completamente puede llevar mucho tiempo para la finalización de los proyectos que tengan asignados, lo cual sería un fracaso para los ejecutivos de negocio causando atrasos significativos en las fechas de entrega final de la aplicación a desarrollar.

Al momento de catalogar los objetos de software de las aplicaciones desarrolladas en las áreas de negocio, se debe considerar que primero se deben de catalogar los proyectos de carácter regulatorio, continuando luego con los proyectos estratégicos de negocio. Se tratara de no perder códigos fuentes y documentación respectiva del sistema para conocer mejor su funcionalidad. La catalogación será de carácter investigativo a nivel local por medio de formularios enviados a las áreas de negocio que desarrollan fuera de un gobierno de TI.

Los roles que deben estar involucrado en la migración del software de negocio al gobierno de TI son el gestor de gobierno y el programador de negocio ambos perfiles son necesarios para llevar un control y monitoreo de las aplicaciones sobre las cuales se está trabajando para aplicar planes de acción en caso que alguna de esta aplicaciones fallen al momento de realizar la migración garantizando la continuidad del negocio. Es importante considerar que el visto bueno del diseño de un proyecto de software debe ser otorgado por un experto local asignado para este fin (como buena práctica se debe considerar crear un comité de aprobación donde participen las

unidades funcionales que deben medir los aspectos necesarios para que la aplicación funcione correctamente).

Tabla 34 Análisis FODA de la entrevista a gestor de gobierno TI y ejecutivos de negocio

Fortalezas	Oportunidades
<ul style="list-style-type: none"> • Buena opinión del servicio de TI y programadores de negocio. • Resolución de requerimientos ya sea por área de TI o por programadores del área de negocio. • Programador de negocio tiene más conocimiento de la operativa del área. • Capacitaciones periódicas del proceso de desarrollo de la gerencia de TI a los nuevos programadores del banco. • Adecuado manejo en el control de accesos y la seguridad de las aplicaciones bajo gobierno TI. • Generación de planes de acción que ayuden a la continuidad del negocio. 	<ul style="list-style-type: none"> • Adaptar una metodología ágil de desarrollo que ayude a cierto tipo de proyectos. • Adoptar un mejor control de objetos de software mediante la normativa de TI. • Enfocar la metodología ágil en las fases de desarrollo de un proyecto que en la documentación. • Mejorar los ambientes de prueba de las áreas de negocio. • Realizar una adecuada administración de los cambios realizados en las aplicaciones de negocio. • Validar los permisos y accesos otorgados a los programadores de desarrollo de negocio. • Validar la calidad de los objetos manejados en las aplicaciones de negocio. • Agilizar el proceso de desarrollo de software para las áreas de negocio.
Debilidades	Amenazas
<ul style="list-style-type: none"> • Falta de distribución correcta de recursos por parte de TI para atender solicitudes de programación. • Falta de flujo estándar de trabajo para todos los programadores de negocio. • Proceso de desarrollo actual es lento y largo. • El área de desarrollo de la gerencia de TI se centra más en proyectos de carácter regulatorio que en proyectos estratégicos de negocio. • Falta de Recursos disponible para satisfacer las necesidades de negocio al asignar 	<ul style="list-style-type: none"> • Ambientes de pruebas creados por programadores de negocio no tienen control. • Des estandarización en el proceso de desarrollo para las aplicaciones de negocio. • Incumplimientos regulatorios con entes legales como ser la CNBS o el marco de referencia ISO 9000. • Perdidas de información o ataques cibernéticos en aplicaciones de negocio que no están protegidas

solicitudes de programación.	adecuadamente.
------------------------------	----------------

4.3 COMPROBACIÓN DE LA HIPÓTESIS

Tabla 35 Prueba de Chi cuadrado

	Valor	Gf	Sig. asintótica (bilateral)	Sig. de Monte Carlo (bilateral)		
				Sig.	Intervalo de confianza al 99.9%	
					Límite inferior	Límite superior
Chi-cuadrado de Pearson	19,837 ^a	6	,003	,000 ^b	,000	,116
Razón de verosimilitudes	19,302	6	,004	,000 ^b	,000	,116
Estadístico exacto de Fisher	15,104			,000 ^b	,000	,116
N de casos válidos	56					

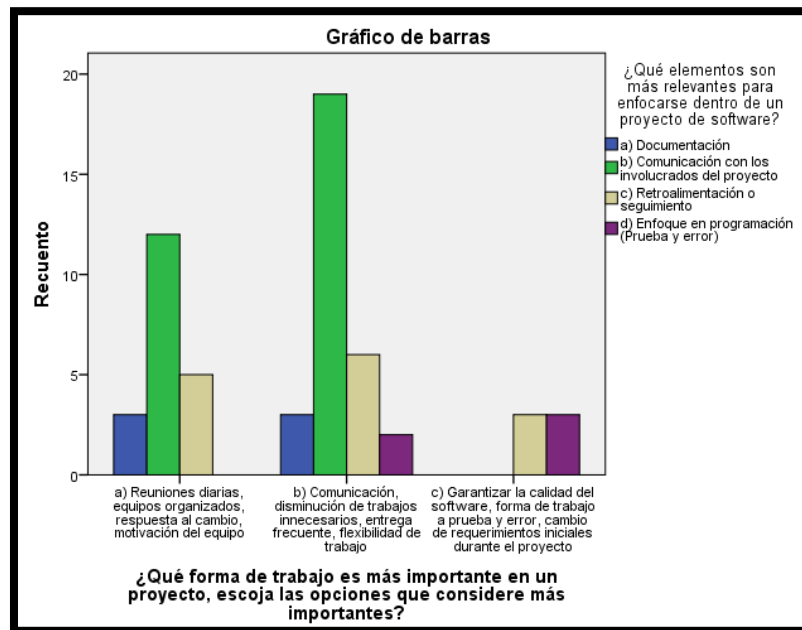


Figura 33 Cruce de variables metodológicas y proceso de desarrollo de software

La hipótesis nula indicaba que una metodología ágil de desarrollo en conjunto con el proceso de gobierno de TI no es necesaria para el control de TI sobre los proyectos de software, se comprobó de acuerdo a los resultados de los instrumentos aplicados que los programadores de negocio y outsourcing necesitan mejorar su forma de trabajo, seguir una metodología ágil de acuerdo al enfoque que le dan a sus desarrollos de

software y conociendo que negocio solicita desarrollos ajenos a TI por la urgencia en los tiempos de entrega, por la necesidad de la disponibilidad de las aplicaciones y por la espera que deben realizar para que TI atienda las solicitudes.

El valor del chi cuadrado que se obtiene al realizar este análisis es menor a 0.05 por lo que se puede rechazar la hipótesis nula, la flexibilidad necesitada en el cambio de requerimientos es alta por lo que es necesario enfocarse en la solución de la hipótesis debido a que una metodología tradicional no brinda las características encontradas que el banco necesita, con el objetivo de aumentar la capacidad y mejorar lo tiempos en que se desarrolla un proyecto de software.

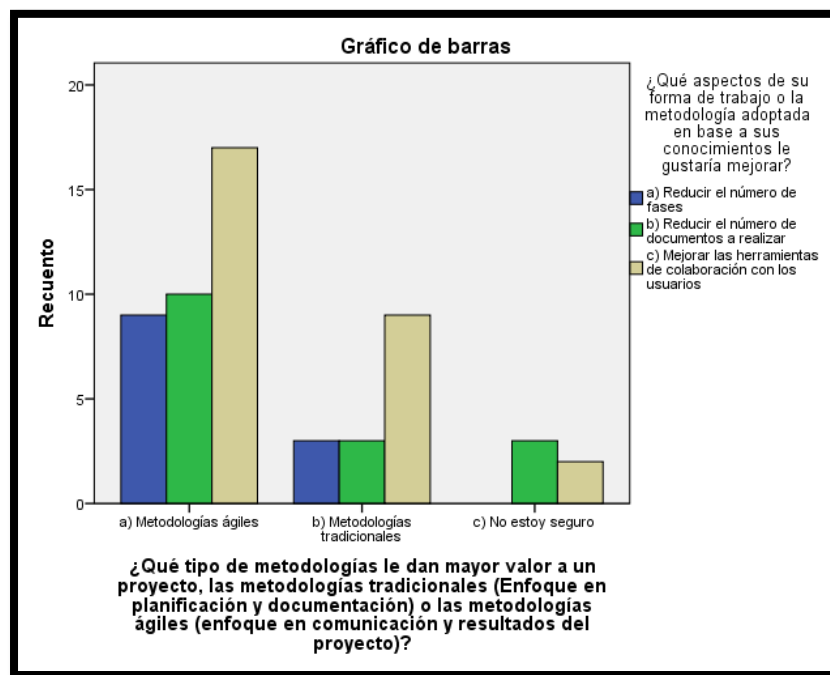


Figura 34 Cruce de variables metodologías y forma de trabajo a mejorar

En el cruce de variables anterior se observa que los programadores de negocio consideran que las metodologías ágiles darían mayor valor por el tipo de proyectos que manejan y consideran que el aspecto más importante a mejorar es la colaboración con los usuarios y reducir el número de documentos a realizar en el proceso de ejecución del software, aquí se identifica una oportunidad de mejora en la forma de trabajo que tienen por lo que las hipótesis que se plantean son válidas como mejoras para los

proyectos, cabe mencionar que los programadores de TI también consideraron importante aplicar las características de una metodología ágil para los proyectos.

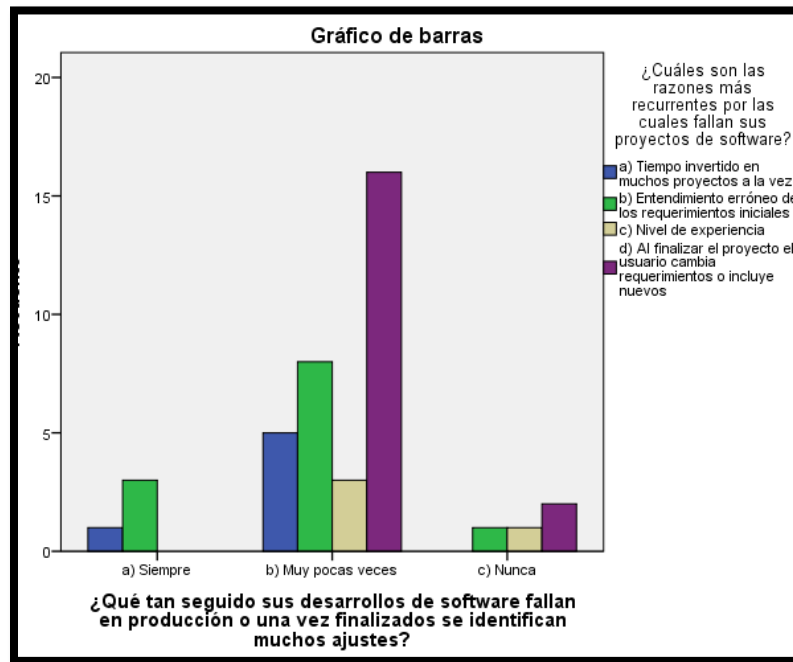


Figura 35 Cruce de variables fallas en el proceso de desarrollo y proceso de gobierno de TI

El cruce de variables anterior ayuda a comprobar la hipótesis porque los usuarios solicitan requerimientos nuevos una vez finalizado el proyecto o durante el ciclo de vida del mismo por lo que se identifica que los proyectos que manejan el banco necesitan un flujo que sea flexible respecto a los cambios y este es un elemento que lo puede proporcionar alguna metodología ágil, así mismo se identifica que los proyectos fallan en producción, con una metodología adicional para cierto tipo de proyectos se puede mejorar la frecuencia de los fallos porque al existir mayor colaboración, mayor enfoque en programación, mayor adaptabilidad del proyecto para el usuario los resultados tendrían que mejorar.

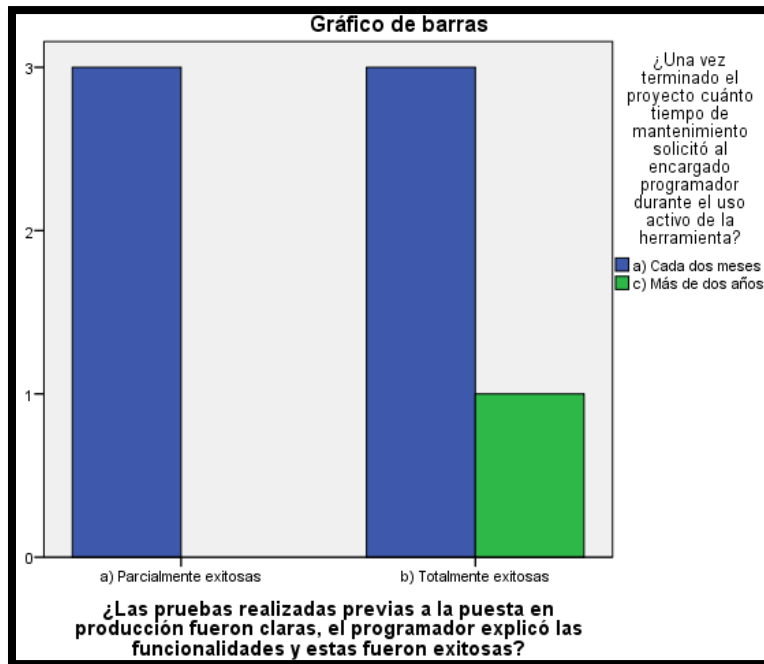


Figura 36 Cruce de variables tiempo de mantenimiento gestionado para las herramientas de software y pruebas previas realizadas en la puesta en producción

Se puede concluir que al momento de realizar pruebas cuando las aplicaciones se llevan a producción siendo estas exitosas o parcialmente exitosas, siempre se le deben estar dando un mantenimiento periódico esto con el objetivo de tener las aplicaciones controladas y que las pruebas y validaciones se realicen en un ambiente de prueba fiable administrado y brindado por la gerencia de tecnología asegurando que todos los proyectos de software desarrollados resulten exitosas al momento de ser usadas por los usuarios finales reduciendo los fallas y riesgos asociados a una falta de gobierno TI.

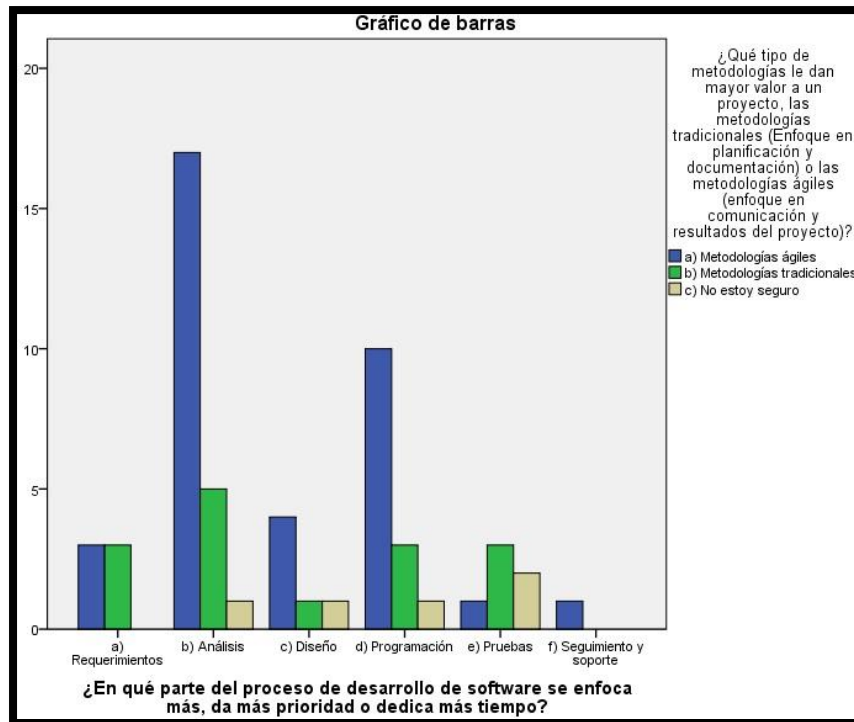


Figura 37 Cruce de variables metodologías que dan mayor valor a un proyecto y proceso de desarrollo en el cual se enfocan más los programadores

Se observa que para cada una de las diferentes fases ejecutadas dentro de un proyecto de software las etapas que más requieren tiempo como son la etapa de análisis y programación, con lo cual se define que la metodología ágil de desarrollo se le da un mayor valor al proyecto centrandose su validez en la prioridad que tenga el proyecto y el tiempo en el cual se realice, tomando en consideración que esta metodología debe estar gestionada y aprobada por la gerencia de tecnología aplicándose medidas de control que hagan que su diseño y aplicación sean efectivas.

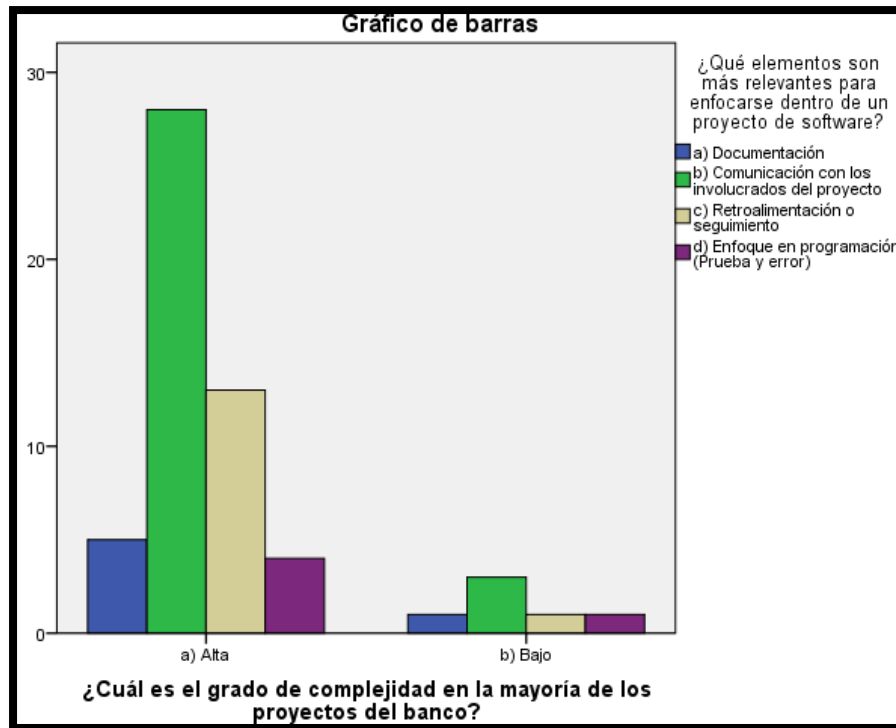


Figura 38 Cruce de variables elementos más relevantes en el desarrollo de un software y grado de complejidad en los proyectos del banco

Se observa que uno de los elementos a tomar en cuenta al momento de emprender un proyecto de software debe ser gestionar una adecuada comunicación con las personas involucradas en el proyecto, considerando este elemento y tomando en cuenta que para dicha actividad se considera que el grado de complejidad manejado en estos tipos de proyectos es alta se puede concluir que la metodología ágil Crystal Clear puede ser una opción muy favorable para el control adecuado de los proyectos de desarrollo ejecutados por negocio y regidos bajo un buen gobierno de TI para brindar productos de calidad.

CAPÍTULO V. CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

1. Las áreas de negocio recurren a elaborar sus propias aplicaciones sin inclusión de TI por la carga de trabajo de programadores de TI, negocio tiene necesidades de automatización y deben ingresar una solicitud de programación que entra a una cola de espera cuyos tiempos de respuesta pueden tardar más tiempo del que negocio puede esperar, adicionalmente por la disponibilidad, retroalimentación y mejor entendimiento de la operativa negocio que tiene un programador asignado directamente en el área; los programadores TI tiene asignados proyectos regulatorios, por lo que no dedican todo el tiempo a proyectos de la operativa del negocio.
2. Para erradicar los riesgos principales de la falta de gobierno en el desarrollo de software realizados por las áreas de negocio es necesario conocer la funcionalidad de las aplicaciones y adherirlas a la infraestructura de TI mediante el proceso de catalogación de proyectos, para los nuevos proyectos desarrollados por negocio establecer una metodología ágil de desarrollo de software que se acople a las necesidades y consolide el control de los objetos de negocio bajo el gobierno de TI.
3. La metodología ágil propuesta para los programadores de negocio ayudará a centralizar sus actividades en la etapa de programación y se enfoca en un método de colaboración con los involucrados en el proyecto para agilizar el avance del mismo, todas las fases que tenga dicha metodología de software serán seguidas por los programadores que están en las áreas de negocio y los programadores de TI pueden adoptar solamente ciertas fases de la metodología propuesta como la parte de seguimiento, esta sería la fase más importante que TI debería seguir.
4. La metodología de desarrollo ágil a seguir por los programadores de negocio y externos es la Metodología Crystal Clear debido a que las características de la

misma fueron las que obtuvieron un porcentaje más alto en los resultados de los instrumentos aplicados, las interrogantes relacionadas con la forma de trabajo más importante para enfocarse en un proyecto de software fueron la comunicación, disminución de trabajos innecesarios, entregables frecuentes, flexibilidad de trabajo, así como la disminución de documentación, características de esta metodología.

5.2 RECOMENDACIONES

1. Se recomienda que las áreas de negocio sigan desarrollando proyectos de software sin embargo se sugiere establecer una metodología ágil para todos los programadores de negocio y también el control de configuración debe estar de lado de la gerencia de TI, adicionalmente para distribuir de forma oportuna la carga de trabajo de proyectos de TI y negocio se sugiere que TI adopte algunas técnicas y estrategias de la metodología ágil enfocadas en la comunicación para mejorar esta fase.
2. Se recomienda para erradicar los riesgos principales establecer una metodología ágil que sigan los programadores que estén con las áreas de negocio, en la fase de programación es donde se deberán establecer más controles que contribuyan a que los objetos de software estén desarrollados bajo normas de programación del banco, es necesario que los objetos de software que surjan de los desarrollos sean almacenados en un repositorio proporcionado por TI así mismo que esta gerencia otorgue un ambiente de pruebas fiable a cada área de negocio para cumplir con el control de gobierno, la parte de administración de objetos debe encontrarse bajo la línea base de configuración de TI misma que no tendrá inclusión en la metodología utilizada.
3. Se recomienda que el proceso de desarrollo de software de la gerencia de TI sea seguido solo por programadores que estén contratados para TI, lo anterior debido a que el proceso de desarrollo de TI se basa en una metodología tradicional y los proyectos de los programadores de negocio o externos se beneficiarían más con una metodología ágil, la parte orientada a la comunicación en el plan de acción de la

metodología Crystal si puede ser utilizada por los programadores de TI porque es una oportunidad de mejora.

4. Para mejorar el gobierno de sistemas se recomienda que los programadores de negocio sigan la metodología ágil Crystal Clear ya es la metodología que se acopla más a la situación que tienen los proyectos del banco y al entorno de los involucrados, contribuye a manejar la flexibilidad de los cambios surgidos cuando los usuarios desean incorporar nuevos requerimientos, ayuda lograr una mejor comunicación y retroalimentación con las personas involucradas en el proyecto, contribuye a eliminar trabajos innecesarios a reducir los artefactos al máximo para enfocarse en resultados más rápidos y siempre con calidad y menor costo.
5. Se recomienda incorporar los programadores que actualmente se encuentran adheridos a negocio a la gerencia de TI, para fortalecer el gobierno de TI, cada programador debe tener asignados proyectos de un área específica de negocio para solventar las causas del porque actualmente negocio tiene programadores propios así mismo es importante que sigan la metodología ágil para lograr las oportunidades de mejora en tiempo, costos, colaboración y sentido de urgencia de algunos requerimientos para los ejecutivos de negocio, dicha metodología ágil con el plan de control de gobierno debe estar avalada por la alta directiva de la organización para que pueda ser implementada con todos los recursos y apoyo necesario permitiendo así obtener los beneficios de su aplicación.

CAPÍTULO VI. APLICABILIDAD

6.1 PLAN DE ACCIÓN PARA LA GESTIÓN DE GOBIERNO DE SOFTWARE PARA PROGRAMADORES DE NEGOCIO

6.2 INTRODUCCION

Se explican los pasos que se sugiere realicen los programadores de negocio para implementar la metodología ágil de desarrollo de software Crystal Clear, describiendo las fases que se eligieron de acuerdo a los resultados para la metodología y detallando los artefactos y actividades oportunas para establecer un estándar en los proyectos de las áreas de negocio con la idea de lograr aumentar la efectividad en cada proyecto, mejoras en los tiempos de entrega, aumento en la calidad de cada una de las actividades y mejorar la colaboración en la ejecución del proyecto, así mismo el plan de acción indica los pasos que los programadores realizaran para que TI pueda ejercer control sobre el hardware y software, logrando así mejorar el gobierno que debe existir en TI.

La metodología Crystal Clear recomienda que el equipo sea pequeño por lo que se adecua a la situación de proyectos de software del banco, también por el grado de interacción con el usuario que en los resultados de la investigación se observó era alto, el elemento más importante para la selección es la fiabilidad y calidad en menos tiempo y menor costo al aplicar este tipo de metodologías y una de las ventajas es la reducción de artefactos al mínimo que se considera es una de las mejoras para este tipo de proyectos y este tipo de programadores, se propone el rol del encargado de pruebas, esto es importante debido a que en los resultados se identificó que los programadores están creando sus propios ambientes de pruebas lo cual constituye un riesgo.

6.3 OBJETIVO

Implementar la metodología ágil Crystal Clear en el desarrollo y mantenimiento de software para los programadores de negocio para que esta contribuya a mejorar la productividad y que TI pueda mantener un gobierno sobre el software y hardware que surjan de los desarrollos de las áreas de negocio.

6.3 METODOLOGÍA ÁGIL CRYSTAL CLEAR

6.3.1 CICLOS DEL PROCESO DE DESARROLLO A APLICAR

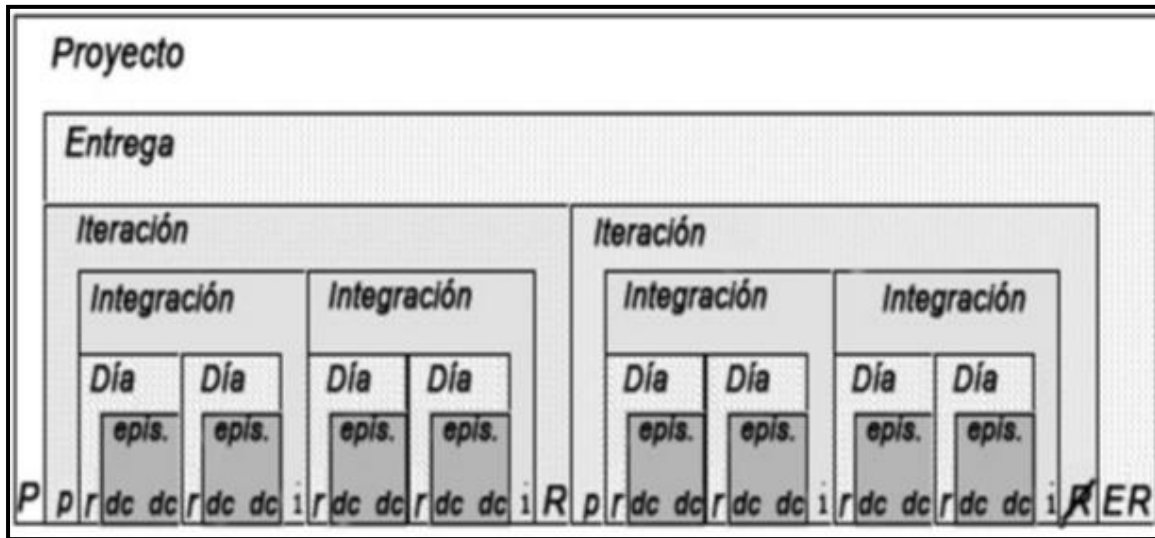


Figura 39 Ciclos de Crystal clear

Fuente: (Cockburn Alistar, 2005)

La figura anterior muestra las fases de la metodología Crystal Clear que se sugiere sean implementadas por los programadores de negocio, para cada una de ellas se explica en el plan de acción las estrategias y técnicas a utilizar cuando se está ejecutando cada una de ellas, los ciclos o fases de esta metodología se componen de:

- ⇒ El proyecto (P) es la etapa de preparación del mismo, se levantan requerimientos y establecen actividades, distribución de tareas, roles, viabilidad del proyecto con los roles involucrados.

- ⇒ Convenio (C): se establecen los acuerdos bajo los cuales se ejecutará el proyecto, como tiempos de entrega, involucrados.
- ⇒ El ciclo de la entrega (E): es un entregable que es una parte del proyecto funcional para el usuario que ya puede ser utilizado.
- ⇒ El planeamiento de la iteración (p): es la estimación de la duración de las actividades de desarrollo y prueba.
- ⇒ La integración (i): se juntan todos los aplicativos de proyecto y se prueban para verificar que la funcionalidad sea correcta, la integración será de 30 minutos a 3 días como máximo.
- ⇒ El día de trabajo (Día): periodo de un día utilizado para programar.
- ⇒ Reunión diaria de trabajo (s): de pie todos los integrantes se reúnen para hablar aspectos del proyecto.
- ⇒ Desarrollo (d): Programación de una parte del sistema.
- ⇒ Control (c): revisiones y comprobaciones de lo estimado que se realiza para la cada parte que ha sido entregada.
- ⇒ Taller de reflexión (R): reunión del equipo que se realiza después de cada entrega para identificar oportunidades de mejora para el proyecto.
- ⇒ La finalización del proyecto (w): Se entrega todos los requerimientos funcionales en software establecidos inicialmente con el experto de negocio.

6.3.2 ESTABLECIMIENTO DE ROLES DENTRO DEL PROYECTO, FASE DE CONVENIO

Los roles sugeridos a establecerse dentro del proyecto de software son:

- ⇒ Patrocinador: Será el jefe del área de banco que debe ser encargado de conseguir los recursos necesarios para que el proyecto se ejecute sin retrasos.

- ⇒ Experto en Negocios: es el usuario de negocio que conoce los requerimientos y la funcionalidad que deben tener los aplicativos, es responsable que dictar claramente cada uno de los requisitos del sistema al diseñador principal a través de las entrevistas de proyecto y de firmar el artefacto con la línea base inicial.
- ⇒ Diseñador Programador: será el encargado de participar en la entrevista para documentar los objetivos y requerimientos con el experto de negocios, así como diseñar la solución arquitectónica y programación del software y seguir las normas y reglas de programación establecidas en el banco.
- ⇒ Administrador de la configuración: será el encargado de solicitar y gestionar vistos buenos de los cambios de software u objetos de software nuevos requeridos para darle solución al desarrollo al área de TI encargado de la configuración.
- ⇒ Coordinador: encargado de producir el plan de entrega, convocar a las reuniones de seguimiento para determinar riesgos, estado y productividad del proyecto con la ayuda del equipo, el objetivo de este rol es dar seguimiento a la ejecución óptima del proyecto e informar cuando se de lo contrario para tomar acciones correctivas.
- ⇒ Escritor: encargado de redactar el manual de usuario del sistema previo reuniones con el diseñador programador, debe tener conocimiento técnico para ejecutar el aplicativo para la toma de interfaces para el manual.

6.3.3 ESTRATEGIAS Y TÉCNICAS CRYSTAL CLEAR A UTILIZAR EN PROYECTOS DE NEGOCIO DEL BANCO

6.3.3.1 ARTEFACTO DE REQUERIMIENTOS

Se deberá documentar el artefacto siguiente para poder documentar los requerimientos generales del proyecto, recalando que se pueden añadir más requerimientos a medida avance el desarrollo del proyecto, este documento deberá ser elaborado en conjunto con el diseñador programador, el usuario experto de negocio y el patrocinador en caso de que sea necesario para requerimientos que involucren recursos necesitados nuevos, se propone que estos documentos se adjunten en un repositorio online para que sirva

como fuente para proyectos futuros y para que todas los involucrados tengan disponible la última versión del documento.

No. de proyecto:	Descripción de proyecto:
Fecha de inicio:	
No.de requerimiento: 1	Descripción de requerimiento:
	Fecha en que debe estar listo:
No.de requerimiento: 2	Descripción de requerimiento:
	Fecha en que debe estar listo:
Nivel de complejidad del proyecto:	
Diseñador Programador:	
Firma de aprobación del experto de negocios:	
Patrocinador:	

Figura 40 Artefacto de requerimientos de proyecto

Entrevistas de proyectos:

En caso de que el proyecto haya sido calificado como complejo en el artefacto de requerimientos y tenga una duración estimada con tiempos grandes se deberá aplicar esta técnica de la metodología Crystal Clear para investigar qué actividades realizadas funcionaron de manera correcta o trajeron ventajas a un proyecto y que acciones no deben volverse a realizar según la experiencia de los entrevistados entre los cuales puede estar el diseñador programador, el patrocinador, usuarios expertos, coordinador, se recomienda hacer las entrevistas a más de una persona, se debe recolectar la siguiente información según 2007, (Sangolqui):

- ✓ Nombre del proyecto
- ✓ Fecha de inicio, tamaño del personal, el dominio y la tecnología que se va a utilizar
- ✓ Historia del proyecto
- ✓ Errores que no deberían cometerse
- ✓ Lo que se desea obtener

- ✓ Prioridades
- ✓ Otros datos relevantes

6.3.3.2 MAPA DEL PROYECTO

Se sugiere utilizar el siguiente artefacto de Hoja de tareas donde se pondrán solamente las tareas críticas que el equipo que conforma el proyecto de desarrollo de software debe realizar con el objetivo de identificar la secuencia del trabajo y responsables, en este artefacto se indica cuál es el esfuerzo requerido en tiempo para finalizar dichas tareas y conocer las auto asignaciones hechas por los mismos miembros del equipo, cabe mencionar que deberá realizarse en grupo con los involucrados y puede imprimirse en grande para pegarlo en las pizarras de información.

Tabla 36 Hoja de tareas de un proyecto de desarrollo de software

No.	Tarea	Responsable	Estimación
1	Tarea 1	Individuo 1	4 días
2	Tarea 2	Individuo 2	2 días
3	Tarea 3	Individuo 3	5 días
4	Tarea 4	Individuo 4	6 días
5	Tarea 5	Individuo 5	26 días

6.3.3.3 ARQUITECTURA DEL PROYECTO

Victoria Temprana:

El equipo y principalmente el diseñador programador deberán enfocarse en buscar entregas iniciales periódicas en lugar de completar toda la entrega del proyecto, esta estrategia de Crystal clear se basa en el premisa de que es mejor obtener un triunfo

pequeño temprano que un gran triunfo tardío, contribuye a mejorar la disposición del equipo para completar el proyecto, 2008 (Rogelio Barroso), por lo anterior en el mínimo de 3 meses el diseñador programador debe realizar una entrega luego de una serie de ciclos de iteración que se componen de desarrollos y resultados temprano, el código que es compilado y puede mostrar una funcionalidad debe considerarse como una victoria temprana de este manera el usuario y patrocinador pueden ver avances frecuentes.

Esqueleto Ambulante:

Las funcionalidades del proyecto se agregaran de forma incremental, se compone de transiciones simples y completas que surgen del código de programación 2008 (Rogelio Barroso), esta estrategia de Crystal clear a aplicar en este plan de acción se basa en la premisa de que el proyecto se irá completando por partes de esta forma los avances que pueden ser pequeños se muestran desde el comienzo y a la vez no es necesario escribir en firme el diseño de la solución sino que esta se va ir modificando conforme avance el proyecto. En cada proyecto el diseñador programador deberá presentar avances del esqueleto ambulante inicial en las reuniones periódicas de las que se hablará más adelante y también deberá ir modificando el diseño según sea necesario para el proyecto.

Rearquitectura Incremental:

El programador realizará el diseño o arquitectura inicial del software requerido, mismo que podrá cambiarse a lo largo del proyecto, el proyecto no deberá detenerse parara corregir el diseño en un artefacto o documento sino que se realizan los cambios en la marcha de la programación y cuando este finalice se le dará la siguiente plantilla con información de que objetos quedaron involucrados en el diseño final de las aplicaciones:

Tabla 37 Objetos modificados en el proyecto

No.	Aplicación	Versión	Tipo de	Plataforma	Criticidad	Objetivo
-----	------------	---------	---------	------------	------------	----------

			Objeto		del objeto	del cambio
1						
2						
	Tablas	Versión	Tipo de Objeto	Plataforma	Criticidad del objeto	
1						
2						
	Pantallas/Interfases	Versión	Tipo de Objeto	Plataforma	Criticidad del objeto	
1						
2						
	Otros objetos relacionados con la arquitectura.	Versión	Tipo de Objeto	Plataforma	Criticidad del objeto	
1						
2						

6.3.3.4 SEGUIMIENTO Y COMUNICACIÓN DEL PROYECTO

Reuniones diarias de pie:

Se sugiere realizar reuniones diarias, recalcando que solamente aplicarían en el caso del banco para proyectos críticos o estratégicos urgentes para el negocio y dependiendo también del nivel de experiencia de los diseñadores programadores a cargo del proyecto, se sugiere aplicarlas cuando el equipo de trabajo para un proyecto sea más grande a 8 personas adicionalmente dichas reuniones deben llevarse a cabo en el término de 5 a 10 minutos como máximo con la idea de identificar atrasos, determinar la siguiente actividad a realizar y lo que se ha realizado; lo anterior expuesto de manera breve para mantener el enfoque productivo del proyecto.

Radiadores de Información:

Este componente se utilizará por el equipo de trabajo del proyecto de software para mantenerse informado de aspectos relevantes del proyecto, consiste en mantener una lámina, pizarra o cualquier recurso visual pegada en un lugar estratégico donde el equipo de proyecto pueda verla fácilmente, debe ser simple y muy comprensible en la información a contener para que sea de fácil entendimiento y que llame la atención de las personas, este recurso debe ubicarse de forma estratégica para mantener la

atención de las personas con el objetivo de mostrar las actividades que se deben realizar, su estado, fechas de entregas, estado de los componentes físico y lógicos, resultado del taller de reflexión y todas aquellas tareas que se deben realizar para finalizar un proyecto de desarrollo de software.

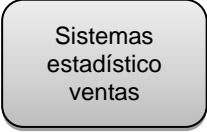
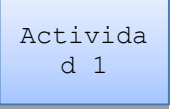



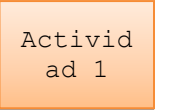



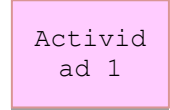

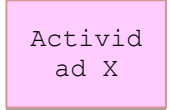
Descripción Proyectos asignados	Equipo de Trabajo	Actividades a realizar	Fecha entregables	Estado del proyecto
Proyecto A 	<i>Líder de Proyecto A</i> <i>Programador líder</i> <i>proyecto A</i> <i>Analista 1</i> <i>Programador externo 1</i>	  	21/05/ 2014 -> Actividad 1 13/06/2014 -> Actividad 2 08/10/2014 -> Actividad X	Proyecto finalizado
Proyecto B 	<i>Líder de Proyecto B</i> <i>Programador líder</i> <i>proyecto B</i> <i>Analista 2</i> <i>Programador externo 2</i>	  	21/05/ 2014 -> Actividad 1 13/06/2014 -> Actividad 2 X/X/X -> Actividad X	Proyecto en ejecución
Proyecto C 	<i>Líder de Proyecto C</i> <i>Programador líder</i> <i>proyecto C</i> <i>Analista 3</i> <i>Programador externo 3</i>	  	03/11/ 2014 -> Actividad 1 09/01/2015 -> Actividad 2 26/02/2014 -> Actividad X	Proyecto finalizado

Figura 41 Pizarra ó Lámina a utilizar para radiador de información

Fuente: (Cockburn Alistar, 2005)

Gráficos de quemado

Esta técnica se utilizará para generar estadísticas sobre el estado de los proyectos mediante gráficos para descubrir demoras o atrasos en las actividades del proceso de ejecución de un proyecto de software con el objetivo de realizar cambios preventivos para poner en marcha el proyecto en tiempos y costos, se realizan estimaciones de tiempo faltante por programar y se resta a los tiempos las actividades realizadas actualmente logrando de esta forma el dominio de los proyectos de desarrollo de software siendo flexibles a los cambios que se pueden generar de un momento a otro. Los gráficos de quemado ayudan a mostrar la velocidad de los procesos, analizando la diferencia entre las líneas proyectadas y efectividad en cada entrega del proyecto, como se ilustra en la siguiente figura:

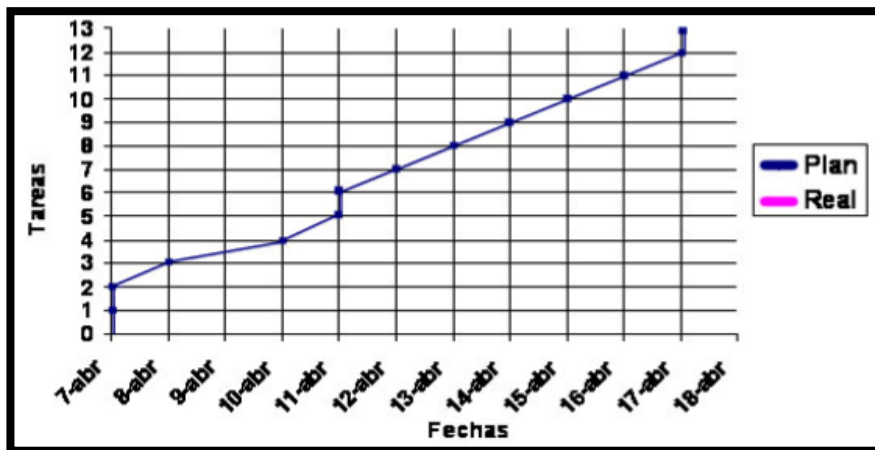


Figura 42 Gráfico de quemado de un proyecto de software

Fuente: (Barroso, 2008)

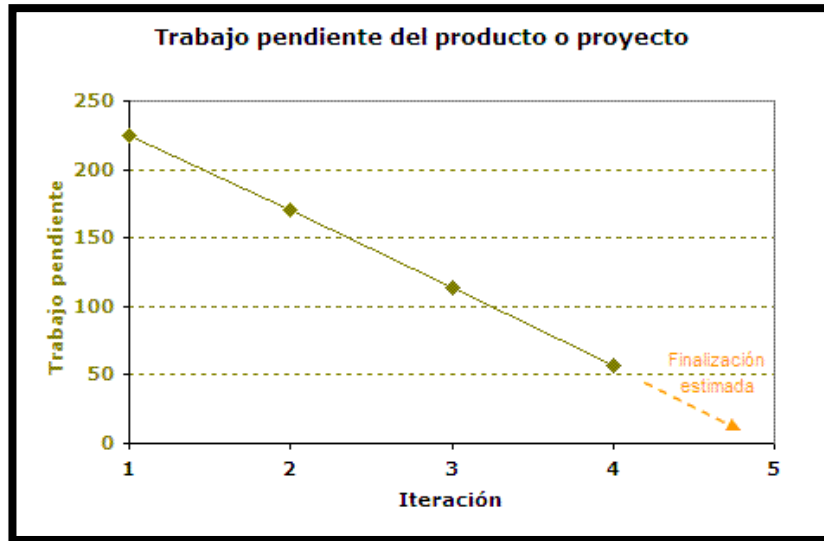


Figura 43 Gráfico de quemado Estimación trabajo pendiente en un proyecto de software

Fuente (proyectosagiles.org, 2014)

En la figura anterior se muestra un ejemplo del gráfico a utilizarse para proyecciones en base a los trabajos pendientes por realizar en un proyecto de software por medio del cual se puede analizar y estimar cuales deben ser las tareas o iteraciones a las cuales se les debe dar prioridad para poner el proyecto de nuevo en marcha y cumplir de esta forma con las entregas parciales y total del producto de software, estos gráficos son muy útiles para identificar debilidades y así poner un mayor empeño es mejorarlas mediante cambios o estrategias aplicadas en forma y tiempo adecuadamente.

Taller de reflexión:

En los talleres de reflexión los diferentes equipos que están involucrados en un proyecto de software deberán dedicar un poco de su tiempo, por lo general de 30 minutos a 1 hora para analizar, consultar y reflexionar sobre las diferentes actividades que deben realizar durante el proceso de desarrollo de una aplicación de software, discutir entre ellos inconvenientes y mejoras, con el objetivo de planear el período y las siguientes actividades y entregas forma ordenada y controlada.

Tabla 38 Resultados y mejoras taller de reflexión proyectos de software

Resultados del Taller de reflexión	
Mantener:	Iteraciones y periodos de cambio:
Problemas:	
Mejoras:	

6.3.3.5 ACTIVIDADES DEL PROCESO DE IMPLEMENTACIÓN DE METODOLOGÍA ÁGIL

1. Identificar programadores involucrados en la metodología, realizar una lista de todos los programadores de negocio y programadores externos (Outsourcing que son parte de negocio también) que deban seguir la metodología para desarrollo.
2. Clasificar los tipos de proyectos, la metodología Crystal establece identificación por color dependiendo del tamaño del equipo de trabajo, en este caso se considera oportuno clasificarlo de 3 a 8 personas
3. Explicar a los involucrados en qué consiste la metodología, para que la metodología pueda tener éxito en la productividad de los proyectos de negocio, es importante involucrar a los interesados desde el comienzo para que tomen conciencia de la importancia y beneficios de la metodología.
4. Establecer quién tendrá cada rol de Crystal clear, los roles establecidos por la metodología se componen de patrocinador, usuario experto, diseñador principal, diseñador programador, experto en negocios, coordinador, especialista en pruebas, escritor, considerar los roles sugeridos previamente en este documento.
5. Realizar artefactos de Crystal clear, consiste en elaborar las plantillas de análisis y diseño que utilizaran los programadores de negocio, adaptarlas junto con el gestor de gobierno.
6. Establecer las políticas, identificar de acuerdo a los tipos de proyectos que reglas y políticas aplican para cada uno.
7. Implementar estrategias de Crystal Clear
8. Implementar técnicas de Crystal Clear

9. Establecer procedimiento cierre de proyecto, establecer las condiciones de cierre de proyecto considerando que el usuario puede agregar requerimientos durante toda la ejecución del proyecto o solicitar modificaciones.
10. Capacitar a involucrados, elaborar manuales que hablen sobre la metodología Crystal clear enfatizando las funciones que debe realizar cada rol.
11. Seguir metodología en proyecto piloto, aplicar la metodología a un proyecto de distinto tipo con todos los roles que apliquen para el mismo.
12. Observar los resultados, realizar mejoras correspondientes luego de haber aplicado la metodología a proyectos reales del banco, considerar si existen pasos de la misma que deban cambiarse para obtener mejores resultados en tiempo y calidad de las aplicaciones.
13. Aplicar actividades de la sección de consolidar administración de hardware y software al gobierno de TI.

6.4 ACOPLAR METODOLOGÍA ÁGIL CON GOBIERNO DE TI UTILIZANDO COBIT 5

6.4.1 PRINCIPIOS Y POLITICAS

6.4.1.1 CONTROL DE SOFTWARE

Gobierno de Diseño:

1. Toda la documentación relacionada con la fase de requerimientos, entrevistas de proyecto y mapa del proyecto de la metodología Crystal Clear deberá realizarse en el comienzo del proyecto y debe ser cargada a la herramienta online establecida como historial de proyectos.
2. Todos los programadores de negocio y también los de TI tendrán acceso a la herramienta de historial de proyecto de negocio para efectos de consultas y retroalimentación de software.
3. El programador deberá llenar el artefacto de Objetos modificados en el proyecto, deberá llenarlo al comienzo del proyecto y al final deberá modificarlo con los objetos agregados luego del transcurso de todo el proyecto para seguir la técnica de

Arquitectura Incremental de Crystal clear, en este artefacto se debe definir el objetivo de modificar el objeto de software para que sirva al administrador de configuración para la gestión del visto bueno de experto de diseño del banco.

4. Cuando se necesite modificar o crear una nueva tabla de base de datos, el diseñador-programador deberá ingresar una solicitud explicando los objetivos y requerimientos de la tabla al administrador de configuración del área de negocios para que este se encargue de solicitar visto bueno al DBA regional del banco, el visto bueno deberá ser relacionado con el nombre de tablas, campos, tipo de los campos y considerar trazabilidad de registros de la tabla, por lo que el programador deberá seguir los acuerdos de diseño entre DBA y administrador de la configuración cuando este programando la tabla o modificándola.
5. Cuando el programador necesite crear o modificar una aplicación nueva deberá ingresar una solicitud con la explicación breve del objetivo y especificaciones para modificar una aplicación al administrador de configuración para que este se encargue de obtener el visto bueno de la modificación a realizar con el comité de revisión de proyectos regional del banco, para los proyectos de negocio un experto regional será el encargado de revisar, gestionar y aprobar con los demás expertos del comité los proyectos de negocio.
6. Los diseñadores-programadores deberán seguir las recomendaciones del DBA y experto regional del banco respecto a las mejoras que se puedan realizar al diseño, lo anterior con el objetivo de identificar cuando se puede reutilizar un proyecto realizado por otro país y mantener el control y estándar de los objetos de software.

Gobierno de Programación:

1. Los diseñadores-programadores deberán seguir el manual de normas de programación de TI regional, en el manual se encuentran especificaciones para realizar rutinas o módulos que sean claros y entendibles, dejar paramétricos y

modulares los cambios para que otros países u otros programadores puedan reutilizar código, dejar comentario del cambio realizado en los programas, así como otras normas para lograr estandarizar la forma de programar.

2. Los roles involucrados dentro del proyecto deberán seguir los ciclos que plantea la metodología Crystal Clear explicadas previamente incluyendo el planeamiento de la iteración para poder darle seguimiento a las actividades que surjan y para seguir la estrategia de esqueleto ambulante de Crystal que contribuya a que el usuario pueda visualizar avances funcionales de software, cada iteración de Crystal se compone de reuniones diarias de quince minutos, programación, controlar, integración y taller de reflexión para poder realizar una entrega al usuario.
3. En caso de que un proyecto de software sea grande, se componga de muchos objetos de software y su diseño sea complejo el programador deberá esperar antes de comenzar la programación a que el administrador de la configuración le entregue la gestión realizada con comité de expertos regionales respecto a vistos buenos de tablas y modificaciones de programas para adaptarse a programar según las sugerencias de dichos expertos de software del banco con la idea de mantener el gobierno de software del banco, así mismo deberá considerar la ayuda de TI para la configuración de un ambiente de pruebas alterno en caso de que el ambiente de pruebas asignado del programador no cuenta con lo necesario.
4. En caso de que el proyecto de software sea mediano o pequeño el programador podrá comenzar la programación en su ambiente de pruebas asignado y luego aplicar posibles sugerencias que se le hayan otorgado al administrador de configuración para el proyecto, podrá renombrar tablas o aplicaciones y seguir sugerencias realizadas una vez que estén listas, lo anterior con la idea de agilizar el desarrollo de este tipo de proyectos, el programador siempre deberá seguir las normas de programación y estándares para lograr que cuando las recomendaciones de expertos para el proyecto estén listas no se necesite cambiar lo que se hizo.

Gobierno de Pruebas:

1. Los diseñadores-programadores deberán utilizar el ambiente de pruebas que TI ha proporcionado al área de negocios, mismo que debe reflejar en gran porcentaje la configuración, registros y objetos que se encuentran en producción para poder eliminar riesgos que se producen cuando las pruebas se hacen en un ambiente muy diferente al de producción, cada programador tendrá asignada su espacio de ambiente con librería para utilizarlo en los diferentes proyectos.
2. El diseñador-programador podrá solicitar un ambiente de pruebas específico para un proyecto en caso de que este sea muy grande y complejo, el programador hace la solicitud online al administrador de configuración para que este gestione el ambiente de pruebas que tendrá especificaciones diferentes al que utiliza diariamente el programador, cabe mencionar que para cada ambiente de pruebas el programador tendrá la clave de acceso proporcionada por TI, otros programadores no podrán utilizar el mismo ambiente.
3. El diseñador-programador no podrá cambiar objetos del ambiente de pruebas que no estén en su librería para mantener la línea base de objetos de software que fue proporcionada por el área de TI.

6.4.1.2 CONTROL DE HARDWARE

Para acoplar un adecuado gobierno de TI se definen diferentes principios o políticas necesarias para el cumplimiento de métricas de los programadores de negocio en base al uso del hardware requerido para realizar sus proyectos de desarrollo de software entre las cuales podemos definir:

Políticas de uso de Servidores:

1. Los programadores de negocio y programadores subcontractados tiene que definir los requerimientos de hardware con los cuales deben de cumplir los servidores sobre los

cuales realizaran sus desarrollos, definiendo recursos como la capacidad que debe tener el procesador y motor de base de datos a utilizar, licenciamiento a instalar para el lenguaje de programación del que se hará uso, capacidad de la memoria RAM del dispositivo físico, y herramientas de software necesarias para su configuración y manipulación.

Tabla 39 Ejemplo formato de especificación de requerimientos

No	Ítem	Cantidad	Descripción requerimiento
1	Servidor Cisco UCS M200	1	Servidor el cual albergará maquinas virtuales
2	Procesador doble núcleo	1	Procesador Corel Duo con capacidad de 16 GB
3	Memoria RAM SQL server 2003	3	Se requiere una memoria RAM de 8 GB para un servidor Windows Server 2003
4	Licencias SQL	10	Licencias necesarias para realizar el desarrollo en el servidor Windows 2003 SQL
5	Computadoras Para agentes	10	Computadoras para el uso de agentes del call center

2. Los ejecutivos de negocio en conjunto con la gerencia de TI tienen que garantizar que los servidores de las áreas de negocio cuenten con un ambiente físico adecuado con todos los requerimientos mínimos de infraestructura necesarios para que los servidores estén seguros y funcionales en todo momento, contando con instalaciones de energía y red de respaldo para evitar que la falta de estos componentes produzcan errores o fallas en los sistemas.
3. Las áreas de negocio debe coordinar con el departamento de TI el contar con servidores de respaldo que estén correctamente configurados y disponibles para levantar los servicios de los servidores de negocio principales cuando estos fallen o se dañen quedando fuera de servicio.
4. Los servidores instalados en las áreas de negocio no podrán ser manipulados, reemplazados o cambiar de posición física sin la aprobación o el ingreso de un RFC por parte de un administrador de TI para llevar adelante el cambio solicitado.

Tabla 40 Ejemplo formato de solicitud de cambio (RFC)

Formulario de Solicitud de Cambio	
FECHA DE SOLICITUD ____ [Fecha] _____	RAZÓN DEL CAMBIO _____
[Justificación del requerimiento]	

DESCRIPCIONES DE CAMBIO _____ [Naturaleza del requerimiento]	

EVALUACIÓN DE RIESGOS _____)	
(USUARIO ¿Criterios de aceptación definidos y acuerdo? SÍ ____ NO ____)	
Cambio Solicitado por _____ [Enviado por:] _____	
CAMBIO AUTORIZADO POR _____ [Aprobado por:] _____	
RESPONSABLE INDIVIDUAL PARA IMPLEMENTACIÓN _____	
[Asignado a:] _____	
(RESPONSABLE INDIVIDUAL DE PRUEBA DE ACEPTACIÓN DEL USUARIO _____)	
(RÉPLICA / PROCEDIMIENTO DE ROLLBACK _____)	
(_____)	
(FECHA DE IMPLEMENTACIÓN - _____)	
TIEMPO DE IMPLEMENTACIÓN- _____)	
(CAMBIO IMPLEMENTADO POR - _____)	
ACEPTADO POR _____)	

Gestor de base de datos

1. Los programadores de negocio están obligados a solicitar los permisos y accesos de ingreso a las diferentes plataformas de software mediante solicitudes que serán enviadas al departamento de tecnología para que el administrador de bases de datos de la gerencia de TI ponga a disposición de ellos las aplicaciones solicitadas en el tiempo que los programadores necesiten hacer uso de ellas.
2. La gerencia de TI definirá cuales serán los gestores de bases de datos autorizados a utilizar por los programadores de negocio con el objetivo de verificar que se cumplan con los requerimientos de hardware necesarios para que las aplicaciones funcionen correctamente y que sean compatibles con los sistemas a los cuales debe conectarse.
3. La gerencia de TI debe contar con un administrador de bases de datos que ayude a proporcionar los permisos y accesos que necesiten los programadores de negocio y

que le oriente cuando se presenten problema o fallas para levantar los servicios y satisfacer las necesidades de negocio.

6.4.2 ESTRUCTURAS ORGANIZATIVAS

En esta parte se definen todas las entidades que se encargaran de tomar las decisiones importantes al momento de iniciar los proyectos de desarrollo de software, para este caso se considera que son tres grupos los que deben apoyar la toma de decisiones:

- Gerencia de TI: Los ejecutivos de la gerencia de Sistemas deben apoyar todas las decisiones necesarias en temas de tecnología que den soporte y apoyo a los proyectos que emprendan las áreas de negocio para que estos cumplan con las exigencias y los objetivos de los usuarios, tomando el control de los proyectos realizados fuera de la gerencia de TI para que estos se realicen bajo las especificaciones requeridas y bajo las políticas de gobierno TI especificadas.
- CEO: La alta gerencia es un ente importante y que siempre se debe tomar en cuenta ya que ellos son la estructura principal del banco, quienes autorizan y dan el apoyo económico para que los proyectos puedan ser iniciados y realizados por completo, también son las personas que ayudan con indicaciones sobre mejora e implementación de estrategias que den valor agregado al negocio.
- Gerencias de Negocios: Estas son las personas que tiene más interés en el cumplimiento de los proyectos de software que darán apoyo al cumplimiento de sus metas y objetivos, por lo cual es importante que forme parte de los grupos de toma de decisión ya que su opinión y sus iniciativas son las que cuentan en detalle para desarrollar las aplicaciones acorde a sus necesidades.

6.4.3 CULTURA, ÉTICA Y COMPORTAMIENTO

Se deberá llenar el siguiente formato cuando no se cumplan las políticas, principios o procesos definidos previamente en este plan de aplicabilidad para todo proyecto de

software que se inicie por los desarrolladores de negocio y que no cumplan con los estándares definidos por la organización, servirán para validar o aplicar las acciones correctivas necesarias que aseguren que estas aplicaciones están siguiendo el estándar.

Cuando los programadores de negocio desarrollen sin la aprobación tanto de las áreas de negocio como de la gerencia de TI se hará ingreso de un Plan de acción (PAC) el cual es un formato donde los programadores deberán de ingresar las acciones correctivas, especificando las actividades que realizarán para que sus proyectos puedan seguir siendo ejecutados, definiendo las actividades a realizar, fechas y responsables de cada actividad con el objetivo de garantizar que el proyecto esté aprobado por las entidades que se definieron en la estructura organizacional.

Tabla 41 Ejemplo de un formato Plan de acción

Formato PLAN DE ACCIÓN					
Responsable seguimiento de cumplimiento _____ Responsable plan de acción Negocio _____		Fecha de Revisión : 1 Abril 2014 Documento Completo <input type="checkbox"/> SI <input type="checkbox"/> NOO Pendientes: _____			
Incumplimiento		Código	001		
Fecha de hallazgo		Acción requerida	XXXXXX		
Fecha de finalización PNC		Gerencia Responsable	XXXXXX		
Fecha de Inicio Acción correctiva	1 Abril 2014	Fecha de Cierre	20 de Junio 2014	Duración en Días	59
Acciones	Responsable	Descripción Actividad	Estado de Aprobación		
Aprobación accesos a base de datos SQL	XXXXX	Se ingresará una solicitud de cambio cada vez que se necesite modificar datos en producción y se realizará solamente aprobado el acceso.	Aprobado		

Los programadores de negocio tienen que solicitar los permisos y accesos necesarios a la gerencia de TI cuando necesiten iniciar o modificar alguna aplicación en la cual están involucrados, en el momento que ellos ingresen o modifiquen un componente a nivel de hardware o software de algunas de las aplicaciones de negocio sin pasar por el proceso de aprobación definido por la gerencia de TI deberán llenar el formato anterior en donde se harán las respectivas observaciones para lograr erradicar el no cumplimiento del mismo.

Capacitaciones de la Metodología Crystal Clear:

Las capacitaciones de la metodología se realizarán por cada programador nuevo de negocio, se contará con un video explicativo de los ciclos, estrategias y técnicas de la metodología Crystal Clear, una vez que el programador haya revisado el video uno de los diseñadores-programadores más experimentados puede explicarle el llenado de artefactos de la metodología y los pasos más importantes aplicar en ella, adicionalmente es necesario que las áreas de negocio cuenten con un manual donde se describa todo lo relacionado a la metodología que se aplica en el banco para que en los primeros días del programador en el área pueda conocer la forma de trabajo, se sugiere realizar reuniones periódicas con programadores para enfatizar los pasos de la metodología y comunicar mejoras realizadas a la misma conforme transcurra el tiempo.

6.4.4 INFORMACIÓN

La información relacionada con el diseño, objetos, cambios y gestiones de vistas de los proyectos de software de las áreas de negocio puede ser consultada en la herramienta online general proporcionada para el seguimiento de proyecto, cada proyecto podrá ser consultado por todos los programadores que pertenezcan a las áreas de negocio para efectos de identificar si algún objeto puede servir para reutilización, para retroalimentación cuando se desea conocer que cambios se realizaron para determinado proceso operativo de negocio, para consulta técnica del programador e investigaciones que ayuden a otros proyectos.

6.4.5 SERVICIOS, INFRAESTRUCUTRA, Y APLICACIONES

Servicios: Los servicios que la gerencia de TI debe ofrecer a las gerencias de negocio y que deben estar regidas bajo en control de gobierno TI para que puedan realizar los proyectos de desarrollo de software con las herramientas necesarias son:

Accesos y permisos a los servidores: El especialista de la gerencia de tecnología debe otorgar los permisos y accesos necesarios a los servidores, aplicaciones, bases de datos mediante una solicitud de atención de servicio que el administrador de la configuración de negocio debe ingresar mediante el sistema de atención de solicitudes manejada dentro del banco, una vez ingrese esta solicitud definiendo el motivo y el tiempo que necesita los permisos, el especialista debe dar solución a dicha petición siempre y cuando esta traiga adjuntas los vistos buenos de las entidades competentes (gerencia de TI y gerencia de Negocios).

Ambientes de pruebas: El área de configuración regional estará encargada de configurar los ambientes de pruebas que sean solicitados por el administrador de configuración asignado del área de negocio previo la obtención de artefactos requeridos para el entendimiento de requerimientos del ambiente por parte del diseñador-programador, estos ambientes se resuelven en el banco mediante la herramienta de atención de servicio del banco, cabe mencionar que solo aplican para proyectos específicos que sean grandes y requieren otro tipo de registros en tablas o configuraciones que el programador necesite para probar el proyecto de lo contrario el programador utilizará su ambiente de pruebas asignado diario en el cual podrá probar la mayoría de proyectos ya que se asemeja al ambiente de producción más próximo.

Instalación y configuración de las herramientas de seguridad: que ayuden a mitigar los riesgos de pérdidas de información en cada uno de los equipos de negocio, tanto en las maquinas usadas por los programadores como en los servidores donde funcionan las aplicaciones que son desarrolladas por las áreas de negocio, en este caso la gerencia de TI debe de mantener el control de los accesos lógicos a las aplicaciones y

monitorear para evitar ataques cibernéticos que causen pérdidas o fraude por manipulación de la información almacenada en estos componentes.

Infraestructura: Se especifican los requerimientos de la infraestructura con las cuales debe contar las áreas de negocio para poder ejecutar sus propias aplicaciones contando con el apoyo de la gerencia de TI, en donde está debe garantizar que se tenga de forma disponible todos los elementos solicitados por negocio para que sus proyectos se puedan iniciar, contando con las edificaciones, red de comunicación, aires acondicionados, Electricidad y todos los componentes tecnológicos físicos que ayuden a mantener los servicios siempre funcionales.

- **Servidores:** Se debe disponer del ambiente y espacio físico donde se instalarán los servidores en las cuales se montarán las aplicaciones de negocio, de igual forma la gerencia de TI debe de brindar soporte y apoyo cuando se configuren dichos componentes en conjunto con los programadores del área de negocio a la cual pertenece dicho servidor, la gerencia de TI tiene que brindar su apoyo para que exista servidores de respaldo para levantar los servicios de las aplicaciones que están soportadas por estos componentes, garantizando su disponibilidad en todo momento
- **Redes:** La gerencia de TI debe brindar soporte técnico cuando existen fallas o problemas con la red de comunicación para que las aplicaciones que son desarrolladas por las áreas de negocio funcionen y evitar detener la operativa de estas áreas para disminuir las pérdidas por falta de disponibilidad de la red, retrasos en la operativa de negocio, insatisfacción del cliente externo por no poder brindar los servicios que la institución ofrece y continuidad del negocio.
- **Edificaciones:** En esta parte la gerencia de TI debe de realizar inspecciones a nivel de edificaciones tomando en cuenta componentes eléctricos, de aire acondicionado, requerimientos de seguridad física como extintores, alarmas detectoras de humo, generadores eléctricos, montajes de cableado de red para prevenir deterioros por falta de cambios o por daño que a futuro causen problemas o catástrofes que detengan la operativa de toda la institución o de componentes tecnológicos críticos.

- Base de datos: El gobierno de TI se encargará de proporcionar los estándares para las tablas, su nomenclatura, campos, restricciones que deben tener las tablas, trazabilidad y todo lo relacionado que afecta el funcionamiento de las bases de datos, el área de administración de configuración de TI debe proporcionar registros que se asemejen a los de producción cambiando información crítica actualizaciones masivas para no revelar información confidencial de los clientes en los ambientes de prueba. La gerencia de negocio deberá decidir que gestor de base de datos utilizará o si debe utilizar varios y coordinar con TI el mantenimiento de este gestor periódicamente, así mismo se debe otorgar acceso a los programadores de negocio al diccionario de datos regional.

Aplicaciones:

Las áreas de negocio incluyendo patrocinadores de proyecto, usuarios expertos y programadores deberán determinar previo a la adaptación a la metodología Crystal Clear cuales son los lenguajes de programación que los programadores pueden utilizar para desarrollar los proyectos para ejercer organización de los objetos, los proyectos que fueron desarrollados en un lenguaje que no sea parte del acuerdo de negocio deberán migrarse a los lenguajes de programación que queden para el área de negocio, se recomienda solamente manejar dos lenguajes de programación considerando el tipo de proyectos que manejan las áreas de negocio y considerando que algunas solicitudes de programación serán atendidos por TI para las áreas de negocio.

6.4.6 EVALUAR, ORIENTAR Y SUPERVISAR DE COBIT

6.4.6.1 ASEGURAR EL ESTABLECIMIENTO Y MANTENIMIENTO DEL MARCO DE GOBIERNO

Se debe realizar una evaluación y monitoreo continuo sobre el cumplimiento y mantenimiento de que se cumplan todos los lineamientos, actividades, roles, políticas, buenas prácticas y acciones que ayuden a gestionar Gobierno TI en base a las referencias impuestas mediante COBIT 5, este proceso se puede realizar mediante la

revisión periódicas de autoridades como ser las gerencias de auditoría interna, Riesgo Integral y Calidad que son las gerencias del banco encargadas de garantizar que todas las unidades funcionales que pertenecen al banco incluida TI respecte y siga los procesos y políticas definidas dentro de la organización con el objetivo de que se cumplan las metas y se logren las estrategias a nivel de negocio.

6.4.6.2 ASEGURAR LA ENTREGA DE BENEFICIOS

Se definen las partes interesadas durante el proceso de gobierno TI para re-estructurar los objetivos de negocio y ligarlos con los objetivos de TI y la empresa, para esta fase del proceso de gobierno TI la gerencia de tecnología debe centrarse en la mejora y aseguramiento del valor que los proyectos de software proporcionan a negocio y brindar los servicios con la calidad y tiempo idealizado, se identifican y comunica la importancia de los facilitadores especificando los beneficios que tendrán las partes interesadas aplicando un adecuado control de TI sobre los desarrollados realizados en las áreas de negocio.

Mediante este proceso se asegura dar respuesta a las auditorías, organismos reguladores y organizamos legales sobre la gestión que la gerencia de tecnología ejerce sobre las unidades de negocio del banco, evitando hallazgos que penalicen o sean causas de sanciones para la institución financiera por incumplimiento de normas, estándares, controles SOX o re-certificaciones desde el momento que hacen uso de actividades que lleven a la aplicación de buenas prácticas mediante el marco de referencia COBIT.

Tabla 42 Asegurar los beneficios de la aplicación de metodología y gobierno de TI

<p>Partes Interesadas: Programadores de negocio, Administrador del proyecto, Área de Negocio, Auditoría, Gerencia de Sistemas, Gerencia de Riesgo Integral.</p>
<p>Cumplimiento de requerimientos o beneficio hacia las partes interesadas:</p> <ol style="list-style-type: none"> 1. Desarrollar un plan de acción donde se hará la migración de los objetos de negocio hacia el control y monitoreo de la gerencia de TI. 2. Hacer la documentación de las plataformas de negocio sobre su funcionalidad y guardar los códigos base de cada una de ellas para respaldo para realizar el proceso de brindar servicios cuando negocio lo requiera. 3. La gerencia de TI podrá administrar las tareas y actividades mostrando progresos y

<p>detalles de trabajo cuando se inicien proyectos de desarrollo que serán liderados por los programadores de negocio.</p> <ol style="list-style-type: none"> 4. Obtener una relación directa entre la gerencia de TI, unidades de negocio y los interesados internos y externos al proyecto, para mejorar la comunicación en la etapa de desarrollo del proyecto. 5. Gestionar adecuadamente la validación en los reporte de avances a mostrar a los usuarios de negocio en el proceso de ejecución de los proyectos que se están desarrollando las áreas de negocio. 6. Brindar el apoyo y colaboración que necesiten las áreas de negocio para proveer los recursos físicos y lógicos a asignar a los proyectos y que el personal que es responsable de los recursos mantenga su servicio activo en cada fase del proceso de desarrollo. 7. Las áreas de negocio deben poner a disposición de la gerencia de TI los recursos para su administración y definir cuáles son sus necesidades y metas estratégicas para contar con el apoyo de TI durante sus proyectos de desarrollo de software.
<p>Métodos o Tecnologías utilizadas para la comunicación entre las partes interesadas:</p> <ul style="list-style-type: none"> - Reuniones informativas - Minutas de Reunión - Correos Electrónicos - Chat corporativo - Llamadas telefónicas - Técnicas de la metodología Crystal
<p>Frecuencia de la comunicación durante la ejecución de los proyectos de software: Dependiendo de la complejidad del proyecto</p>
<p>Procedimientos de escalamiento para resolver los problemas:</p> <ol style="list-style-type: none"> 1. Los programadores de negocio deben dar el adecuado mantenimiento a sus aplicaciones y retro-alimentar a la gerencia de TI sobre los cambios a realizar en sus aplicaciones para obtener el servicio y apoyo de la gerencia de sistemas. 2. En el caso que negocio deje de prescindir de su propio analista programador la gerencia de TI podrá brindar soporte a las aplicaciones gobernadas mientras negocio asigna un nuevo especialista para que administre sus aplicaciones. 3. En el caso que la gerencia de TI no tenga recursos disponibles para brindar soporte a las gerencias de negocio, se podrá considerar obtener la ayuda y apoyo de un proveedor externo que administre y de soporte a las aplicaciones de negocio mientras se asigna un especialista interno que puede hacerse cargo del sistema afectado. 4. Se escala el problema de un proyecto a la gerencia general cuando la decisión a aplicar sea de gran impacto para el seguimiento de este proyecto y que se necesite aumentar o disminuir fechas de entrega del proyecto, presupuesto, recursos humanos o físicos, etc.

6.4.6.3 ASEGURAR LA OPTIMIZACIÓN DEL RIESGO

En esta parte de la gestión de gobernabilidad la gerencia de TI definirá un plan sobre el manejo de los riesgos que se identifiquen al momento de realizar las actividades necesarias para controlar los componentes físicos y lógicos de las áreas de negocio, manejando un adecuado plan de manejo de riesgos y mitigación de eventualidades que

puedan producir incidentes con impactos perjudiciales para la continuidad del negocio y las áreas de negocio que se puedan ver afectadas.

- ✓ Reuniones para elaborar el plan de gestión de riesgos: En estas reuniones estarán involucrados todas las partes interesadas, de forma que las acciones de contingencia se puedan incluir en el costo y cronograma durante el proceso de migración de los objetos a la gerencia de TI, así mismo se asignen las responsabilidades. En la primera reunión se aplicará la técnica Delphi para la identificación de riesgos y en reuniones consecuentes se aplicará la lluvia de ideas para la identificación de respuestas a riesgo, de esta forma también se hará la respectiva identificación de riesgos sobre el proceso de desarrollo de software y sus actividades relacionadas.

Tabla 43 Periodicidad de la gestión de riesgos

PROCESO	MOMENTO DE EJECUCIÓN	ENTREGABLE	PERIODICIDAD DE EJECUCIÓN
Planificación de Gestión de los Riesgos	Inicio del proyecto de migración y control de TI sobre las aplicaciones de negocio	Plan del proyecto	Una vez
Identificación de los riesgos	En cada reunión del equipo del proyecto de migración de objetos	Plan del proyecto actualizado	Mensualmente
Análisis Cualitativo de Riesgos	En cada reunión del equipo del proyecto de migración de objetos	Plan del proyecto actualizado	Mensualmente
Análisis Cuantitativo de Riesgos	En cada reunión del equipo del proyecto de migración de objetos	Plan del proyecto actualizado	Mensualmente
Planificación de Respuesta a los Riesgos	En cada reunión del equipo del proyecto de migración de objetos	Plan del proyecto actualizado	Mensualmente
Seguimiento y Control del Riesgos	En cada fase del proyecto de migración e inicio de proyectos de desarrollo por programadores de negocio	Documento de evaluación actualizado	Mensualmente

Tabla 44 Identificación, Evaluación Cualitativa y Respuesta a riesgos

ID	Descripción del riesgo	Probabilidad	Impacto	Categoría	Evento/Indicador	Respuesta al riesgo	Responsable	Periodicidad de revisión
1	Que el proyecto de migración de objetos no se termine a tiempo	6	2	Recursos Humano	Alguna actividad de la ruta crítica del cronograma esté atrasada	Mitigar. Dar seguimiento continuo a la ruta crítica y aplicar técnicas de optimización de cronograma como ser crashing en caso de ser requerido.	Gestor de gobernabilidad	Semanalmente
2	Que el programador de negocio no involucre al usuario en el proceso de desarrollo del sistema	2	4	Recursos Humano	Las reuniones no se dan en los plazos establecidos, no se brinda la información o documentación requerida por el programador	Mitigar. Solicitar reuniones en con los usuarios interesados y ponerlas en agenda.	Programador de negocio	Semanalmente
3	Que los programadores de negocio no comuniquen a TI nuevos proyectos de desarrollo	8	7	Recursos Humano	Las asignaciones de trabajo se siguen haciendo de la forma antigua y no se da participación activa a la gerencia de TI.	Mitigar. Brindar una capacitación y documentación clara en el uso del sistema, incluir a la gerencia de TI en el monitoreo y control de los proyectos de software. Utilizar interfaces amigables para todos los usuarios.	Ejecutivos de negocio	Avances mensuales
4	Que no se migren las aplicaciones de negocio dentro de gobierno TI	4	10	Recursos Humano	Que se dejen fuera de gobierno las aplicaciones de negocio causando problemas de fallas o indisponibilidad de las herramientas por falta de soporte por parte de TI	Mitigar Establecer la importancia de las aplicaciones y realizar plan de acción con las actividades adecuadas para migrar los objetos bajo gobierno TI.	Gestor de Gobernabilidad	Semanalmente

6.4.6.4 ASEGURAR LA OPTIMIZACIÓN DE LOS RECURSOS

Se deberá evaluar periódicamente la satisfacción de los usuarios finales que utilizan los desarrollos realizados por negocio para asegurarse que el proceso implementado compuesto por la metodología ágil Crystal Clear en conjunto con el gobierno de TI brinda el mayor de los beneficios a la empresa, esta evaluación se puede realizar mediante la aplicación de encuestas y entrevistas con los involucrados que son diseñador-programador, expertos de negocio, patrocinador, administrador de la configuración, , el objetivo de este proceso de Cobit es aprovechar al máximo todo tipo de recursos, eliminar desperdicios de tiempo y de dinero; los puntos de enfoque que se sugieren estudiar periódicamente en las áreas de negocio para determinar que se aplica la optimización de los recursos son los siguientes:

- ✓ Evaluación física de servidores y máquinas de negocio para asegurarse de que no existan pérdidas de tiempo por hardware en mal estado, gestionar con TI el soporte del hardware.
- ✓ Medir el nivel de satisfacción de los desarrollos de software de negocio mensualmente a través de encuestas para identificar oportunidades de mejora en la comunicación con los usuarios y forma de trabajo con los usuarios, de esta forma se puede modificar pasos de la metodología Crystal clear conforme se identifiquen mejoras por el necesidades diferentes de los usuarios.
- ✓ Identificar la cantidad de proyectos que fallan al mes, identificar las causas para buscar soluciones ya sea en fases de la metodología Crystal clear o en oportunidades de mejora de las políticas del gobierno de TI.
- ✓ Realizar evaluaciones a los diseñadores-programadores para determinar aquellos que tengan un nivel de experiencia y habilidades superiores para poder utilizarlos en capacitaciones a nuevos programadores y considerar asignar proyectos más complejos a este tipo de programadores.

- ✓ Identificar cuellos de botella del proceso de desarrollo y de los pasos de acoplar gobierno de TI con la metodología Crystal Clear, identificar pasos que puedan agilizarse por ejemplo se debe evaluar si un administrador de la configuración será suficiente para gestionar las solicitudes que hagan los diseñadores-programadores respecto a vistos buenos para retener objeto de software y solicitudes de cambios de objetos en producción, en plan de presupuesto de este plan de aplicabilidad se sugieren 3 administradores de configuración para las áreas de negocio.
- ✓ Identificar el porcentaje de reutilización de modelos de software que ejercen las áreas de negocio para asegurarse que se optimiza el uso de la herramienta de administración de proyectos a la que tienen acceso los programadores de negocio.

6.4.6.5 ASEGURAR LA TRANSPARENCIA HACIA LAS PARTES INTERESADAS

En este proceso que se deberá aplicar de Cobit se debe establecer las metas de desempeño que debe tener cada área de negocio respecto a los desarrollos de software con la metodología nueva y debe brindarse a los involucrados de negocio y TI los informes de desempeño de dichas metas para asegurarse que la comunicación está funcionando de forma optima e identificar procesos críticos respecto a la gestión de desarrollo de software que se deba corregir o mejorar, los criterios a evaluar para asegurar la transparencia hacia las partes interesadas son:

- ✓ Identificar interrupciones en el proceso de desarrollo debido a fallas o esperas en los servicios de TI, se debe medir el tiempo de respuesta que recibe el administrador de configuración de parte de TI para las gestiones de retenciones y vistos buenos.
- ✓ Identificar el porcentaje de satisfacción de los involucrados que siguen el proceso de desarrollo de software de negocio que incluye la metodología el acoplamiento con gobierno de TI.

- ✓ Identificar la cantidad de fallas que genera los proyectos de desarrollo de software de negocio una vez aplicados en producción, se debe llevar bitácora de los errores para que sirvan como retroalimentación para proyectos futuros y que ayudar al seguimiento y solución de los mismos, se puede identificar los conceptos que se presentan en la siguiente tabla:

Tabla 45 Documento para Reporte de Error

Reporte de Error	
Fecha de inicio	
Descripción del caso	
Programador	
Estado del sistema actual	
Proyecto asociado al error	
Repetición del error	
Impacto	
Solución	
Fecha de cierre	

6.4.7 ACTIVIDADES PARA MIGRACIÓN DE PROYECTOS EXISTENTES DE SOFTWARE

En esta sección del plan de acción se establecen las actividades que negocio deberá realizar para almacenar las aplicaciones bajo un repositorio proporcionado por TI la cual debe de tener el control físico y lógico de todos los objetos relacionados con aplicaciones, hardware y software de los proyectos, así mismo la gerencia de tecnología debe otorgar un ambiente de pruebas fiable a cada área de negocio para cumplir con el control de gobierno de objetos y que se encuentren bajo la línea base de configuración de TI, así como el control de permisos y accesos a las herramientas proporcionadas por TI al momento que los programadores de negocio quieran hacer uso de ellas evitando de esta forma incidentes que afecten la seguridad de la información o la continuidad del negocio.

Las actividades involucradas para el control de infraestructura son las siguientes:

- ✓ Identificar los objetos existentes en las áreas de negocio y catalogarlos en base a criticidad, aplicabilidad, cantidad de usuarios que lo utilizan, manejo de información estratégica de negocio y cumplimiento regulatorio.
- ✓ Verificar en que plataforma están realizadas las aplicaciones, que tipo de base de datos usan, cual es su código fuente, y cuál es su funcionalidad dentro del área de negocio para identificar su importancia.
- ✓ Conocer el tipo de hardware, software y la infraestructura que debe tener la gerencia de tecnología disponible para que los objetos a gobernar funcionen adecuadamente una vez estos estén bajo el gobierno de TI
- ✓ Realizar un estudio sobre el espacio físico y lógico que ocuparan los servidores una vez estén bajo el gobierno de TI para montar la infraestructura correspondiente en caso de ser necesario.
- ✓ Identificar el tipo de datos e información que maneja cada objeto y conocer el objetivo para el cual fue desarrollado cada uno, para contemplar realizar una adecuada gestión de respaldo de la información antes de realizar la migración de los objetos para no perder la integridad de la información asegurarse que en caso de alguna eventualidad se tenga forma de recuperar la data integra.
- ✓ Identificar que servidores existentes de negocio se pueden virtualizar.
- ✓ Catalogar aplicaciones a la línea base de TI, tomar el control del código de los objetos gobernados para que TI los almacene.
- ✓ Montar la infraestructura necesaria para que los objetos de negocio funcionen correctamente y afinar las aplicaciones con componentes de seguridad, donde se definan roles y permisos de accesos a otorgar para ambiente de producción.
- ✓ Identificar cuáles son las conexiones que tienen los objetos con otras aplicaciones, documentación del sistema, ver las bases de datos a las que están conectadas.
- ✓ Realizar formatos por medio de los cuales los programadores gestionen la retención de fuentes para realizar un cambio o mantenimiento a los objetos que estén bajo el gobierno de TI.

- ✓ Contratar un administrador de bases de datos asignado a negocio, para controlar la información de las bases de datos que se manejen.
- ✓ Gestionar un plan de riesgo tecnológico existente al momento de acoplar los objetos bajo el gobierno de TI para evitar errores o problemas durante los cambios a realizar.
- ✓ Mantener soporte continuo sobre las peticiones de los programadores de negocio al momento de desear ingresar los objetos a la línea base y que se le brinden las herramientas que necesiten para realizar las actividades de desarrollo sin atrasos significativos.
- ✓ Realizar capacitaciones sobre la forma de trabajo y flujo a seguir al momento de solicitar permisos a TI y las responsabilidades de negocio sobre los objetos controlados y las responsabilidades de TI sobre las aplicaciones.

6.5 DIAGRAMA DE PROCESOS

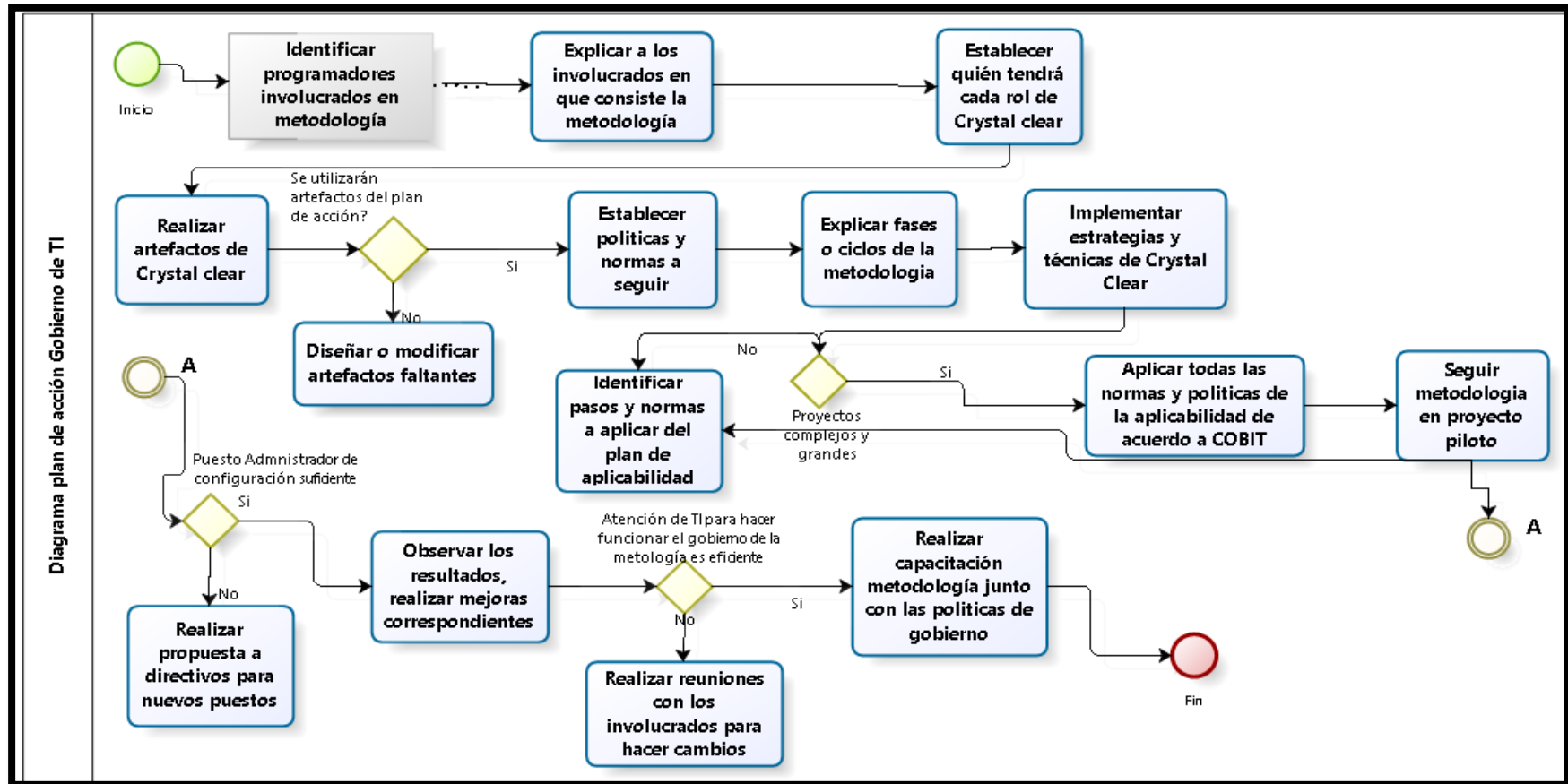


Figura 44 Diagrama Implementación Plan de acción gobierno TI

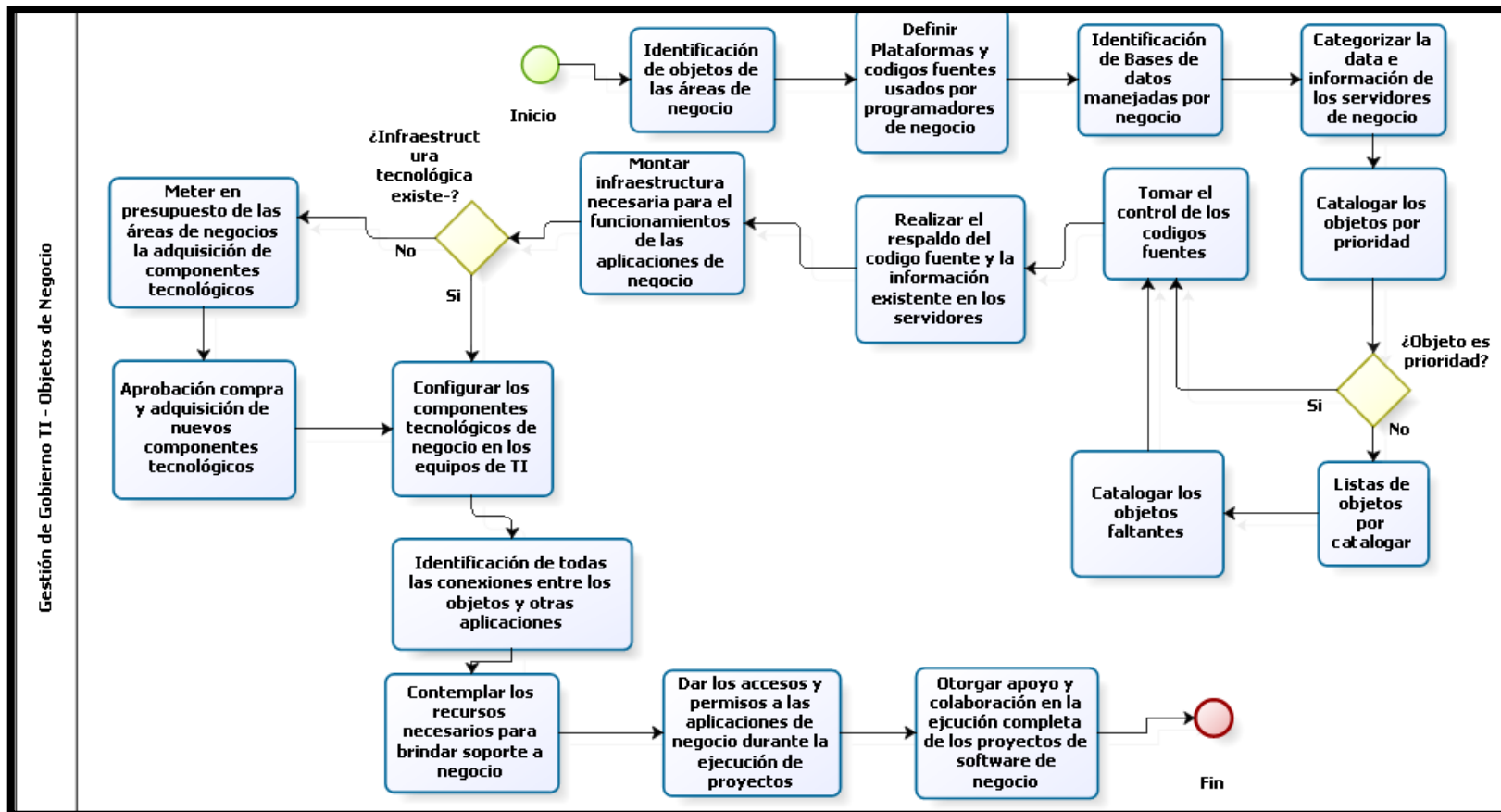


Figura 45 Diagrama Control Hardware y Software

Cada uno de los objetos de negocio que están siendo utilizados actualmente dentro de la institución financiera deben ser administrados, en este diagrama se indica cada tarea principal comenzando con la identificación de los objetos, siguiendo después con su catalogación, estudio de la infraestructura que necesita para funcionar y actividades que deben realizar los programadores como parte de la metodología para que en cada desarrollo los objetos se vayan guardando a línea base de TI.

6.6 CRONOGRAMA DE EJECUCIÓN

ID	Task Name	Duration	Start	Finish
1	Plan de Acción Metodología Ágil y acoplar Gobierno de TI	210 days	Mon 09/03/15	Fri 25/12/15
2	METODOLOGÍA ÁGIL CRYSTAL CLEAR	210 days	Mon 09/03/15	Fri 25/12/15
3	EXPLICAR CICLOS DEL PROCESO DE DESARROLLO A APLICAR	2 days	Mon 09/03/15	Tue 10/03/15
4	ESTABLECIMIENTO DE ROLES DENTRO DEL PROYECTO	3 days	Wed 11/03/15	Fri 13/03/15
5	ESTRATEGIAS Y TÉCNICAS CRYSTAL CLEAR A UTILIZAR EN PROYECTOS DE NEGOCIO DEL BA	111 days	Mon 16/03/15	Mon 17/08/15
6	EXPLICAR ARTEFACTO DE REQUERIMIENTOS	2 days	Mon 16/03/15	Tue 17/03/15
7	EXPLICAR MAPA DEL PROYECTO	1 day	Wed 18/03/15	Wed 18/03/15
8	IMPLEMENTAR ESTRATEGIAS DE ARQUITECTURA DEL PROYECTO	5 days	Thu 19/03/15	Wed 25/03/15
9	IMPLEMENTAR ESTRATEGIAS DE SEGUIMIENTO Y COMUNICACIÓN DEL PROYECTO	5 days	Thu 26/03/15	Wed 01/04/15
10	Identificar programadores involucrados en la metodología	1 day	Thu 02/04/15	Thu 02/04/15
11	Clasificar los tipos de proyectos	1 day	Fri 03/04/15	Fri 03/04/15
12	Explicar a los involucrados en qué consiste la metodología	3 days	Mon 06/04/15	Wed 08/04/15
13	Establecer quién tendrá cada rol de Crystal clear	5 days	Thu 09/04/15	Wed 15/04/15
14	Realizar artefactos de Crystal clear a utilizar	2 days	Thu 16/04/15	Fri 17/04/15
15	Establecer las políticas y normas de programación	2 days	Mon 20/04/15	Tue 21/04/15
16	Implementar estrategias de Crystal Clear	15 days	Wed 22/04/15	Tue 12/05/15
17	Implementar técnicas de Crystal Clear	15 days	Wed 13/05/15	Tue 02/06/15
18	Establecer procedimiento cierre de proyecto	1 day	Wed 03/06/15	Wed 03/06/15
19	Capacitar a involucrados para metodología Crystal Clear	5 days	Thu 04/06/15	Wed 10/06/15
20	Seguir metodología en proyecto piloto,	45 days	Thu 11/06/15	Wed 12/08/15
21	Observar los resultados de prueba piloto, realizar mejoras correspondientes	3 days	Thu 13/08/15	Mon 17/08/15
22	CONSOLIDAR HARDWARE Y SOFTWARE AL GOBIERNO DE TI	94 days	Tue 18/08/15	Fri 25/12/15
23	Verificar en que plataforma están realizadas las aplicaciones	5 days	Tue 18/08/15	Mon 24/08/15
24	Conocer el tipo de hardware, software y la infraestructura que debe tener la gerencia de tecnología dis	5 days	Tue 25/08/15	Mon 31/08/15
25	Realizar un estudio sobre el espacio físico y lógico que ocuparan los servidores	5 days	Tue 01/09/15	Mon 07/09/15
26	Catálogo aplicaciones a la línea base de TI	60 days	Tue 08/09/15	Mon 30/11/15
27	Montar la infraestructura necesaria para que los objetos de negocio funcionen correctamente	5 days	Tue 01/12/15	Mon 07/12/15
28	Realizar formatos por medio de los cuales los programadores gestionen la retención de fuentes	5 days	Tue 08/12/15	Mon 14/12/15
29	Contratar un administrador de bases de datos asignado a negocio,	2 days	Tue 15/12/15	Wed 16/12/15
30	Realizar un plan de riesgo tecnológico existente al momento de acoplar los objetos bajo el gobierno de	5 days	Thu 17/12/15	Wed 23/12/15
31	Realizar capacitaciones sobre la mecánica y flujo a seguir al momento de solicitar permisos a TI	2 days	Thu 24/12/15	Fri 25/12/15
32	Realizar los estándares de programación para los programadores de negocio	3 days	Tue 18/08/15	Thu 20/08/15
33	Capacitar y realizar manual de las normas basadas en Cobit para aplicar gobierno	3 days	Tue 18/08/15	Thu 20/08/15

Figura 46 Cronograma del Plan de acción

En el cronograma se hace la descripción de las actividades y tareas relacionadas a los procesos de gestión de gobierno TI dentro de las áreas de negocio y el acoplamiento de una metodología ágil de desarrollo para estandarizar las actividades involucradas en la programación de los proyectos de software que son desarrollados por los programadores de negocio, se especifican las actividades a ejecutar tanto por negocio como por el área de TI tanto en la migración de objetos como en el monitoreo y soporte que el área de tecnología debe brindar cuando las aplicaciones estén bajo su control.

6.7 PRESUPUESTO

Tabla 46 Presupuesto de plan de acción

Alternativas	Costos Estimados Únicos	Etapas del Proyecto
Administradores de configuración	Lps. 25,000 Mensual	Diseño
Herramienta de control de proyectos online	Lps. 35,000 (Licencia)	Seguimiento
Experto Regional para revisión de proyectos de negocio	Lps. 26,000 Mensual	Aprobación de proyectos
Capacitación Metodología ágil de desarrollo al ingresar nuevos programadores	Lps. 1,000 Anual	Seguimiento
Licenciamientos de software	Lps. 75,000	Diseño
Administrador de Bases de datos	Lps. 35,000 Mensual	Diseño
Especialista de seguridad (Brindará permisos y accesos a los programadores de negocio)	Lps. 20,000 Mensual	Seguimiento
Licencias para instalación y configuración de herramientas de seguridad	Lps. 50,000	Diseño
Licencias lenguaje de programación	Lps. 50,000	Diseño
Servidor de respaldo para las aplicaciones de negocio	Lps. 75,000	Diseño
Servidor para almacenamiento de	Lps. 5,000	Diseño

repositorios de códigos fuente		
Servidor para las Bases de Datos	Lps. 25,000	Diseño
TOTAL PRESUPUESTADO	Lps.422,000	

Se indica cuales son los costos que implica poder acoplar gobierno de TI dentro de la institución financiera BAC tomando en cuenta la migración y control de objetos de negocio, uso de la metodología ágil de desarrollo y el soporte que TI tiene que brindar a los programadores de negocio y las aplicaciones que estos desarrollen, el presupuesto está más centrado en el recurso humano que en la adquisición de componentes tecnológicos físicos o lógicos ya que la gerencia de sistemas cuenta con una infraestructura adecuada para soportar las aplicaciones de negocio y el gasto en esta parte no sería grande, solamente se presupuesta los componentes que realmente hacen falta o son requeridos para la migración y control de los objetos de negocio.

BIBLIOGRAFÍA

Carlos Fernández, & Piattini Mario. (2012). Modelo para el gobierno de las TIC basado en las normas ISO. España: AENOR - Asociación Española de Normalización y Certificación. Recuperado a partir de <http://site.ebrary.com/lib/alltitles/docDetail.action?docID=10637138>

Carlos Fernández Sánchez, & Mario Piattini Velthuis. (2012). Modelo para el gobierno de las TIC basado en las normas ISO. ISACA. (2012). COBIT 5. Recuperado a partir de http://zotero.org/support/quick_start_guide

Catherine Picazzo, Norha Villegas, & Gabriel Tamura. (2008). Análisis descriptivo del procesos de implementación del nivel 2 del modelo CMMI en una empresa regional de desarrollo de Software.

Cecilia Boggi (2014), PMP, Proyecto de mejora CMMI, un caso de éxito <http://www.pminuevocuyo.org/userfiles/file/PMIBA%20Tour%20MZA%20CB%20CMMI.pdf>

Cervantes Ojeda, Gómez Fuentes, María del Carmen, (2012) Taxonomía de los modelos y metodologías de desarrollo de software más utilizados, Red de Revistas Científicas de América Latina, el Caribe, España y Portugal, <http://www.redalyc.org/articulo.oa?id=37326902005>

COBIT 5: A Business Framework for the Governance and Management of Enterprise IT. (2014). [Tecnología de la Información]. Recuperado 1 de noviembre de 2014, a partir de <http://www.isaca.org/cobit/pages/default.aspx>

Corporación GPF. (2014). Recuperado 26 de noviembre de 2014, a partir de <https://www.corporaciongpf.com/>

Demián Arancibia Zeballos, (Abril 2008). Un nuevo proceso de implementación para software de Gobernabilidad TI en Chile http://www.tesis.uchile.cl/tesis/uchile/2008/arancibia_dz/sources/arancibia_dz.pdf

Diseño de Procesos bajo CobiT 5.0: [Source: NoticiasFinancieras]. (2014). NoticiasFinancieras. Miami, UnitedStates. Recuperado a partir de <http://search.proquest.com/docview/1601935770/46BA2FCC06D04978PQ/1?accountid=35325>

Gobierno TI - Implant Gobierno de TI.pdf. (s. f.). Recuperado a partir de <http://www.hacienda.go.cr/cifh/sidovih/spaw2/uploads/images/file/Implant%20Gobierno%20de%20TI.pdf>

Gutiérrez de Mesa, J. A., & Pagés Arévalo, C. (2009). Planificación y gestión de proyectos informáticos. España: Servicio de Publicaciones. Universidad de Alcalá. Recuperado a partir de <http://site.ebrary.com/lib/bvunitecvirtualsp/docDetail.action?docID=10280334>

INTECO. (2009). INGENIERÍA DEL SOFTWARE: METODOLOGÍAS Y CICLOS DE VIDA (Vol. 1).

International Standard ISO/IEC 12207. (2008) (Segunda Edición.). Recuperado a partir de http://www.ing.unisannio.it/cimitile/ingsw/dispense/ISO_12207.pdf

ISACA. (2012). COBIT 5: A Business Framework for the Governance and Management of Enterprise IT. ISACA.

Javier Heredia Ruiz, Lilián Álvarez Almanza, Naryana Linares Pons. (2010), Comparación y tendencias entre metodologías ágiles y formales. Metodología utilizada en el Centro de Informatización para la Gestión de Entidades

José. (2010). Pensamiento Centroamericano. Recuperado 15 de noviembre de 2014, a partir <http://elchrboy.blogspot.com/search/label/Ingenier%C3%ADa%20de%20Software>

M. A. Awad(2005), A Comparison between Agile and Traditional Software Development Methodologies.

Network Sec (2014) Gobierno_TI.doc - gobierno_ti.pdf. (s. f.). Recuperado a partir de http://www.itson.mx/publicaciones/pacioli/Documents/no59/sistemas_de_informacion/gobierno_ti.pdf

Omar Otoniel Soto Romero, Germán Harvey Salinas(2014), («Scrum, ¿Un paradigma de administración de proyectos que cumple lo que promete?», s. f.)https://www.academia.edu/528989/Scrum_Un_paradigma_de_administraci%C3%B3n_de_proyectos_que_cumple_lo_que_promete

Pantaleo, G. (2011). Calidad en el desarrollo de software. México: Alfaomega Grupo Editor. Recuperado a partir de <http://site.ebrary.com/lib/bvunitecvirtualsp/docDetail.action?docID=10757943>

Ranking 2008-2009 en Tecnologías de Información - CentralAmericaData :: Central America Data. (2009). Recuperado 21 de noviembre de 2014, a partir de http://en.centralamericadata.com/es/article/home/Ranking_global_de_aprovechamiento_de_las_TICs

Sánchez Fernández, & Carlos Manuel PiattiniVelthuis, (2012). Modelo para el gobierno de las TIC basado en las normas ISO. España: AENOR - Asociación Española de Normalización y Certificación. Recuperado a partir de <http://site.ebrary.com/lib/alltitles/docDetail.action?docID=10637138>

SEI AdministrativeAgent, (Noviembre 2010), CMMI para Desarrollo. Guía para la integración de procesos y la mejora de productos. Recuperado a partir de <http://cmmiinstitute.com/assets/Spanish%20Technical%20Report%20CMMI%20V%201%203.pdf>

United Nations Publication Unctad/IER, (2012), New York and Geneva, Information Economy Report 2012, The software industry and development countries http://unctad.org/en/PublicationsLibrary/ier2012_en.pdf

Vismar Santos Hernández (2014), Cujae, Cuba, La industria del software, estudio a nivel global y América Latina, Revista académica de economía con el Número Internacional Normalizado de Publicaciones Seriadas ISSN 1696-8352 <http://www.eumed.net/cursecon/ecolat/la/09/vsh.htm>

Programación extrema:

http://www.ecured.cu/index.php/Programaci%C3%B3n_Extrema_%28XP%29

Juan Palacio. (2008). Flexibilidad con Scrum. safeCreative.

http://www.navegapolis.net/files/Flexibilidad_con_Scrum.pdf

2008, Aplicación de la metodología ágil Crystal Clear a un caso de estudio

Ing. Rogelio Barroso Abreu, Ing. Yoanna Oliveros Guntín, Ing. Lenia García Álvarez de la Campa, Ing. Yaniel Álvarez Alfonso, Lic. Jorge Coello Mena

<https://sites.google.com/site/ingsoportelologico/home/crystal-clear>

Barrientos Oscar. (2011, noviembre). Metodología Crystal. Recuperado 7 de marzo de 2015, a partir de <http://inf162expogrupos7-22011.blogspot.com/>

COCKBURN ALISTAIR; Crystal Clear. A Human - Powered Methodology for Small Teams; Addison-Wesley. 2005

ANEXOS

Anexo 1 ENCUESTA DE SATISFACCIÓN AL USUARIO DE PROYECTOS FUERA DE GOBIERNO DE TI.

La siguiente encuesta se realiza con el objetivo de conocer el criterio de satisfacción que posee de los desarrollos de software gestionados por su área de negocio.

Encierre en un círculo la respuesta que considere:

1. ¿Cuánto tiempo dedicó el analista programador para la recolección de requerimientos?
 - a) De 3 a 8 horas
 - b) De 8 a 16 horas
 - c) Ninguna hora

2. ¿Con qué frecuencia presentó avances o entregables el analista programador a lo largo del proyecto?
 - a) Avances diarios
 - b) Entrega total (solamente un avance)
 - c) Avances periódicos (dependiendo del tipo de proyecto)

3. ¿Consideró importante la retroalimentación de avances durante un proyecto?
 - a) Sí, siempre deben existir
 - b) No, se espera la finalización del proyecto completo
 - c) Si el proyecto es grande si es necesario, si es pequeño o mediano no.

4. ¿Las pruebas realizadas previas a la puesta en producción fueron claras, el programador explicó las funcionalidades y estas fueron exitosas?
 - a) Parcialmente exitosas

- b) Totalmente exitosas
 - c) No fueron exitosas
5. ¿Se documentó un plan de proyecto donde se estipulara las fechas de cumplimiento para el cierre del proyecto?
- a) Sí
 - b) No
6. ¿El programador cumplió con las fechas de cierre o entregables de proyecto?
- a) Sí
 - b) No
 - c) No existían acuerdos de fechas
7. ¿Cuáles son los proyectos que cumplen con todos los requisitos de calidad la mayor parte del tiempo?
- a) Desarrollos de negocio
 - b) Desarrollos de TI
8. ¿Cuántas fallas o necesidades de ajustes se presentaron en producción cuando se instaló el proyecto final?
- a) De 1 a 2
 - b) De 2 a 5
 - c) Más de cinco
 - d) Ninguna
9. ¿Una vez terminado el proyecto cuánto tiempo de mantenimiento solicitó al encargado programador durante el uso activo de la herramienta?
- a) Cada dos meses

- b) Anualmente
- c) Más de dos años
- d) Nunca

10. ¿Continuaría solicitando desarrollos de software a programadores adheridos a negocio que no forman parte de TI?

- a) Dependiendo del tipo de proyecto
- b) Dependiendo de la disponibilidad de recursos
- c) Dependiendo de los recursos monetarios
- d) Dependiendo de las políticas del banco

<http://www.e-encuesta.com/answer?testId=Y/141uOI7QM=>

Validación de la Encuesta de Satisfacción al usuario de proyectos fuera de gobierno de TI

Validación de encuesta por Alfa de CRONBACH:

Escala: TODAS LAS VARIABLES

Resumen del procesamiento de los casos

		N	%
Casos	Válidos	10	100,0
	Excluidos ^a	0	,0
	Total	10	100,0

a. Eliminación por lista basada en todas las variables del procedimiento.

Estadísticos de fiabilidad

Alfa de Cronbach	N de elementos
,686	11

En la figura anterior se muestra que la encuesta para conocer la satisfacción de los usuarios es aceptable, ya que tiene 0,686 de fiabilidad.

Anexo 2 ENCUESTA PARA PROGRAMADORES DE DESARROLLO SOFTWARE QUE TRABAJAN CON LAS ÁREAS DE NEGOCIO.

1. ¿Cuántos proyectos considera puede tener asignados actualmente?
 - a) Tres proyectos asignados por diferentes niveles de complejidad.
 - b) Solamente un proyecto, se asigna el siguiente hasta terminado el anterior.
 - c) Dos proyectos en paralelo

2. ¿Qué tipo de proyectos considera resolver como prioridad?
 - a) Lo decide negocio, dependiendo de la necesidad
 - b) Regulatorios
 - c) Por elección propia

4. ¿Trabaja con alguna metodología, proceso o norma de desarrollo de software para los proyectos asignados por negocio?
 - a. Sí
 - b. No

5. ¿Qué tipo de metodologías le dan mayor valor a un proyecto, las metodologías tradicionales (Enfoque en planificación y documentación) o las metodologías ágiles (enfoque en comunicación y resultados del proyecto)?
 - a) Metodologías ágiles
 - b) Metodologías tradicionales
 - c) No estoy seguro

6. ¿Le gustaría seguir un flujo de trabajo establecido por la empresa para cual labora?
 - a) Sí
 - b) No

7. ¿En qué parte del proceso de desarrollo de software se enfoca más, da más prioridad o dedica más tiempo?
- a) Requerimientos
 - b) Análisis
 - c) Diseño
 - d) Programación
 - e) Pruebas
 - f) Seguimiento y soporte
8. ¿Cuál es el seguimiento que necesitan las áreas de negocio en el desarrollo de un proyecto de software?
- a) Avances diarios
 - b) Entrega total del proyecto
 - c) Avances periódicos (dependiendo del tipo de proyecto)

¿En qué nivel cumple con los tiempos de finalización de proyecto acordados con los usuarios?

- a) Siempre
 - b) Casi nunca
 - c) A veces
9. ¿Cuáles son las razones más recurrentes por las cuales fallan sus proyectos de software?
- a) Tiempo invertido en muchos proyectos a la vez
 - b) Falta de un ambiente de pruebas fiable
 - c) Entendimiento erróneo de los requerimientos iniciales
 - d) Disponibilidad de herramientas y accesos necesarios
 - e) Nivel de experiencia
 - d) Al finalizar el proyecto el usuario cambia requerimientos o incluye nuevos

10. ¿Qué aspectos de su forma de trabajo o la metodología adoptada en base a sus conocimientos le gustaría mejorar?
- a) Reducir el número de fases
 - b) Reducir el número de documentos a realizar
 - c) Mejorar las herramientas de colaboración con los usuarios
11. ¿Qué elementos son más relevantes para enfocarse dentro de un proyecto de software?
- a) Documentación
 - b) Comunicación con los involucrados del proyecto
 - c) Retroalimentación o seguimiento
 - d) Enfoque en programación (Prueba y error)
12. ¿Qué ambiente de pruebas utiliza para replicar los escenarios de prueba?
- a) Negocio brinda uno
 - b) Creo mi propio ambiente de pruebas
 - c) Hago las pruebas directamente en producción
13. ¿Cada cuánto el usuario adiciona requerimientos que no estaban establecidos inicialmente como parte del alcance?
- a) Siempre
 - b) Casi nunca
 - c) Muy pocas veces
 - d) Nunca sucede
14. ¿Qué tan seguido sus desarrollos de software fallan en producción o una vez finalizados se identifican muchos ajustes?
- a) Siempre
 - b) Muy pocas veces
 - c) Nunca

15. ¿Qué forma de trabajo es más importante en un proyecto, escoja las opciones que considere más importantes?
- a. Reuniones diarias, equipos organizados, respuesta al cambio, motivación del equipo
 - b. Comunicación, disminución de trabajos innecesarios, entrega frecuente, flexibilidad de trabajo
 - c. Garantizar la calidad del software, forma de trabajo a prueba y error, cambio de requerimientos iniciales durante el proyecto
16. ¿Qué características disminuyen costos en un proyecto cuando los recursos son Outsourcing?
- a. Menos tiempo en documentación
 - b. Trabajo en equipo
 - c. Adaptabilidad de acuerdo al proyecto
 - d. Mayor enfoque en programación
17. ¿Cuál es el grado de complejidad en la mayoría de los proyectos del banco?
- a) Alto
 - b) Bajo
18. ¿Cuál es el grado de interacción con el usuario final en sus proyectos?
- a) Alta
 - b) Bajo

<http://www.e-encuesta.com/answer?testId=49z+EXtGf9k=>

Validación de la Encuesta a los programadores de de Negocio

Análisis de fiabilidad utilizando Alfa de Cronbach:

Escala: TODAS LAS VARIABLES

Resumen del procesamiento de los casos

		N	%
Casos	Válidos	15	100,0
	Excluidos ^a	0	,0
	Total	15	100,0

a. Eliminación por lista basada en todas las variables del procedimiento.

Estadísticos de fiabilidad

Alfa de Cronbach	N de elementos
,729	19

Anexo 3 ENCUESTA PARA PROGRAMADORES DE DESARROLLO SOFTWARE DEL ÁREA DE TI.

1. ¿Cuántos proyectos considera puede tener asignados actualmente?
 - a) Tres proyectos asignados por diferentes niveles de complejidad.
 - b) Solamente un proyecto, se asigna el siguiente hasta terminado el anterior.
 - c) Dos proyectos en paralelo

2. ¿Qué tipo de metodologías le dan mayor valor a un proyecto, las metodologías tradicionales (Enfoque en planificación y documentación) o las metodologías ágiles (enfoque en comunicación y resultados del proyecto)?
 - a) Metodologías ágiles
 - b) Metodologías tradicionales
 - c) No estoy seguro

3. ¿En qué parte del proceso de desarrollo de software se enfoca más, da más prioridad o dedica más tiempo?
 - a) Requerimientos
 - b) Análisis

- c) Diseño
 - d) Programación
 - e) Pruebas
 - f) Seguimiento y soporte
4. ¿Cuáles son las razones más recurrentes por las cuales fallan sus proyectos de software?
- a) Tiempo invertido en muchos proyectos a la vez
 - b) Entendimiento erróneo de los requerimientos iniciales
 - c) Nivel de experiencia
 - d) Al finalizar el proyecto el usuario cambia requerimientos o incluye nuevos
5. ¿Qué elementos son más relevantes para enfocarse dentro de un proyecto de software?
- a) Documentación
 - b) Comunicación con los involucrados del proyecto
 - c) Retroalimentación o seguimiento
 - d) Enfoque en programación (Prueba y error)
6. ¿Qué tan seguido sus desarrollos de software fallan en producción o una vez finalizados se identifican muchos ajustes?
- a) Siempre
 - b) Muy pocas veces
 - c) Nunca
7. ¿Cuál es el grado de complejidad en la mayoría de los proyectos del banco?
- a) Alto
 - b) Bajo

8. ¿Cuál es el grado de interacción con el usuario final en la mayoría de los proyectos?
- a) Alta
 - b) Bajo
9. ¿Cuál es el nivel de riesgo tecnológico asociado en la mayoría de proyectos que usted desarrolla?
- a) Alto
 - b) Bajo

Validación de la Encuesta a los programadores de TI

Análisis de fiabilidad utilizando Alfa de Cronbach:

Escala: TODAS LAS VARIABLES

Resumen del procesamiento de los casos

		N	%
Casos	Válidos	15	100,0
	Excluidos ^a	0	,0
	Total	15	100,0

a. Eliminación por lista basada en todas las variables del procedimiento.

Estadísticos de fiabilidad

Alfa de Cronbach	N de elementos
,636	10

Anexo 4 ENTREVISTA PARA EJECUTIVOS DE NEGOCIOS QUE CONTRATAN PERSONAL EXTERNO A TI PARA DESARROLLO DE SOFTWARE.

1. ¿Cuál considera son las principales razones para hacer contratación de programadores en las áreas de negocio y no hacer uso de los recursos de TI?
 - a) Falta de recursos en la gerencia de TI para atender solicitudes de programación.
 - b) Proyectos finalizados que no satisfacen los requerimientos del cliente.
 - c) Por la urgencia y necesidad de tener el software en el menor tiempo posible.
 - d) Todas las anteriores.

2. ¿Qué nivel de éxito considera que tienen los proyectos que son desarrollados por los programadores contratados por su gerencia?
 - a) Altamente exitosos
 - b) Poco exitosos
 - c) Medio exitosos

3. ¿En base a la continuidad de negocio, las aplicaciones que desarrollan sus programadores permanecen siempre activas?
 - a) Si
 - b) No

4. ¿Cuáles son las fallas que más afectan el servicio brindado a los usuarios que utilizan sus propios desarrollos de software? ¿Por qué no solicitan ayuda de TI cuando estas aplicaciones fallan? Pregunta abierta

5. ¿Cuáles considera son las debilidades de TI al momento de atender solicitudes de programación? ¿En base a tiempo cuanto es el tiempo promedio que han demorado cuando ustedes han realizado una solicitud de programación? Pregunta abierta

6. ¿Cuál es su nivel de satisfacción en la atención brindada por TI al momento de atender una solicitud de programación?
- a) Muy satisfecho
 - b) Poco satisfecho, el nivel de atención es bajo
 - c) Completamente insatisfecho, el soporte es lento y el tiempo para atender un problema es muy prolongado
7. ¿Cuáles son los beneficios de contratar programadores outsourcing o programadores adheridos a la gerencia sin enlace con TI?
8. ¿Cuál es el nivel de riesgo asociado a los proyectos desarrollados en su área?
- a) Alto
 - b) Bajo
9. ¿Qué tipo de proyectos manejan más en su área?
- a) Grandes
 - b) Pequeños
 - c) Medianos
10. ¿Quién considera desarrolla una mayor cantidad de proyectos de desarrollo de software?
- a) TI
 - b) Negocio
 - c) Outsourcing
11. ¿Qué opinión tiene sobre el servicio ofrecido por los programadores outsourcing, siempre finalizan los proyectos de software con la calidad y en el tiempo esperado?
- Pregunta abierta

12. ¿Al momento de contratar programadores outsourcing que desarrollen para su gerencia, existe alguna cláusula donde se vean obligados a seguir algún proceso de desarrollo? Pregunta abierta

13. ¿Cuáles proyectos tienen un costo menor? :

- a) Proyectos de TI
- b) Proyecto de Negocio
- c) Proyectos Outsourcing

14. ¿Los proyectos más complejos e importantes para el banco a quién prefiere delegarlos?

- a) TI
- b) Negocio
- c) Outsourcing

15. ¿Qué nivel de importancia tiene la parte de documentación de la planificación, acuerdos de fechas, diseño técnico de desarrollo para usted?

- a) Altamente importante
- b) Poca importancia
- c) No estoy seguro

Anexo 5 ENTREVISTA AL GESTOR DE GOBIERNO DE DESARROLLOS DE SOFTWARE DE TI.

1. ¿Cuál considera son las principales causas para que las áreas de negocio decidan contratar sus propios recursos para desarrollar productos de software? Pregunta abierta
2. ¿En base al conocimiento adquirido cuáles son los tipos de proyectos que más desarrollan fuera del Gobierno de TI?
 - a) Regulatorios
 - b) Estratégicos de negocio
 - c) En base al orden de asignación para los programadores
3. ¿Considera que como buena práctica de Gobierno TI, sería adecuado adherir los programadores de Negocio a la gerencia de TI? ¿Por qué?
4. ¿Cuáles objetos de software de las aplicaciones desarrolladas en negocio que considera se pueden migrar a la línea base de TI?
 - a) 100% de los objetos.
 - b) Solamente objetos compatibles con herramientas existentes en TI
 - c) Solamente objetos críticos
5. ¿Cuál será la prioridad para realizar la catalogación de los objetos de software de las aplicaciones que actualmente fueron desarrolladas en las áreas de negocio sin estar bajo un gobierno de TI?
 - a) Proyectos estratégicos
 - b) Proyectos regulatorios
 - c) Por prioridad de negocio

6. ¿Cuáles roles deben estar involucrados en la migración del software de negocio al gobierno de TI?
- a) Gestor de gobierno
 - b) Programador de negocio
 - c) Ambos
7. ¿Dentro del plan de capacitaciones del banco se ha incorporado alguna referente al proceso actual de desarrollo de software de TI a los programadores de negocio? Pregunta abierta
8. ¿Considera que para implementar un adecuado Gobierno de TI, una solución podría ser adherir los programadores de negocio a la gerencia de Sistemas y dar capacitaciones sobre el proceso de desarrollo actual para que lo sigan? ¿Por qué? Pregunta abierta
9. ¿Cada cuánto se debe realizar capacitaciones para adherir a los nuevos programadores a la gerencia de TI?
- a) Mensual
 - b) Trimestral
 - c) Cuando se adhiere un nuevo programador al negocio
10. ¿Qué medios a utilizar para capacitar a los programadores de negocio sería más efectivo?
- a) Audiovisuales
 - b) Conferencias presenciales prácticas
11. ¿Quién debería dar los accesos a los programadores de negocio?
- a) Personal de Seguridad.

- b) Experto de TI a asignar para esa actividad.
- c) Gobernador de TI
- d) Otro _____

12. ¿Cuánto tiempo debe durar el acceso según el tipo de proyecto para los programadores de negocio?

- a) Dependiendo del cambio a realizar en producción
- b) Dependiendo de la experiencia del programador
- c) Según la política general de seguridad que siguen programadores de TI
- d) Política nueva de seguridad para programadores de negocio

13. ¿Cuántos riesgos pueden estar asociados al otorgar los privilegios de accesos?

- a) Riesgos por el cambio a realizar en producción.
- b) Riesgos por criticidad de la información.
- c) Otro _____

14. ¿Quién debe dar visto bueno del diseño?

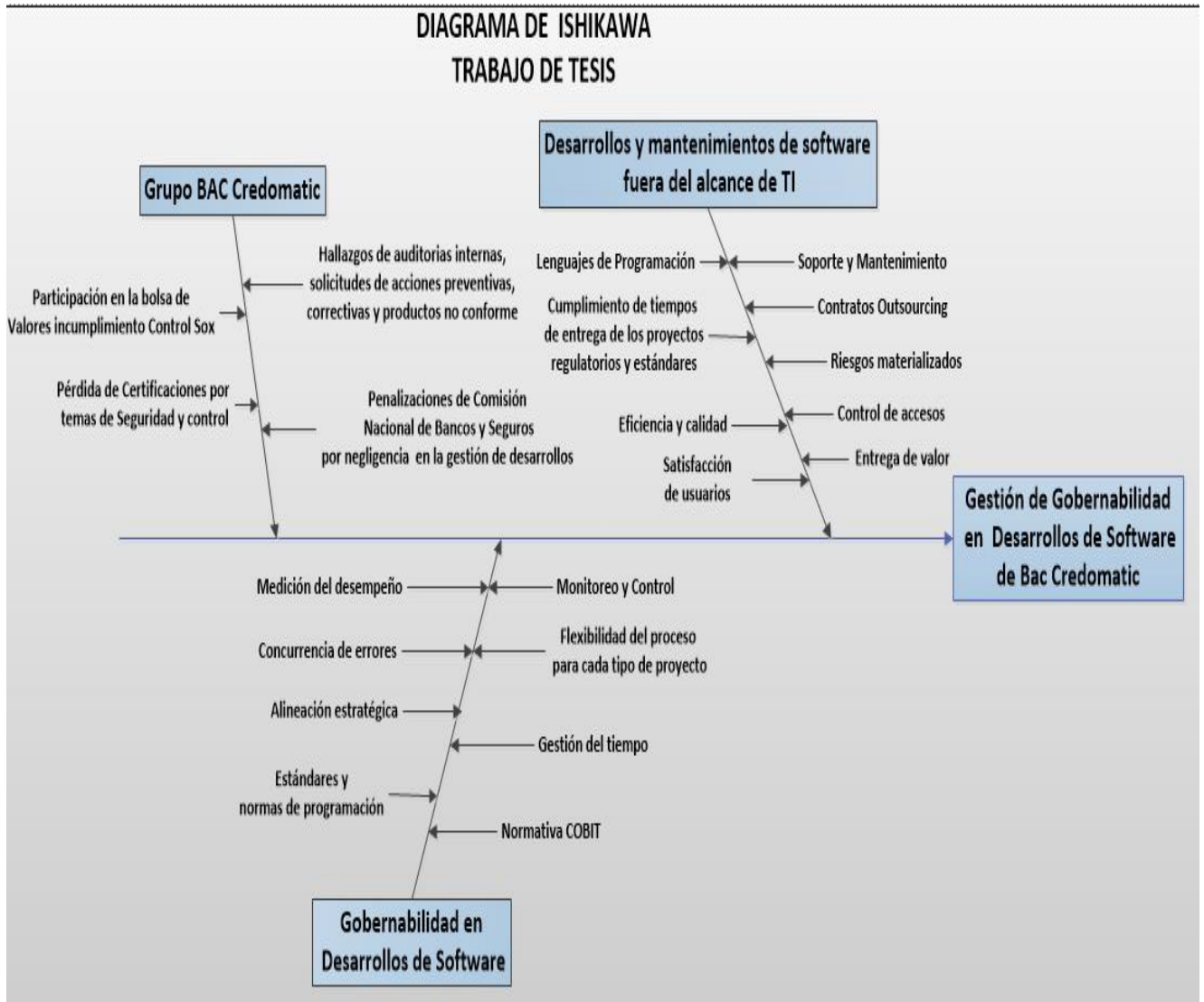
- a) Administrador de base de datos si aplica.
- b) Comité de expertos regional
- c) Experto local asignado para este fin

15. ¿Cuál considera es el mayor riesgo asociado al existir desarrollos que no son controlados por TI? Pregunta abierta

16. ¿Cuánto tiempo puede durar un acceso a un programador que no es parte de TI?

- a) Diario
- b) Según el tiempo requerido por el programador.
- c) No se dará el acceso, se asigna recurso de TI para realizar la actividad.

Anexo 6 DIAGRAMA DE ISHIKAWA DEL ENFOQUE DEL PROBLEMA



Anexo 7 MATRIZ DE ALINEACIÓN.

Preguntas de Investigación	Objetivos asociados
¿Por qué la áreas de negocio de Bac Honduras recurren a desarrollar sus propias aplicaciones sin inclusión de TI?	Identificar las causas del porqué las áreas de negocio desarrolla sin ayuda de la gerencia de TI.
¿Qué pasos se deben realizar en cada etapa del desarrollo de software para erradicar los riesgos principales de la falta de gobernabilidad de TI?	Proponer un procedimiento para adherir a los programadores de negocio en cada una de las etapas del proceso de desarrollo de software existente en la gerencia de TI.
¿Qué parte del flujo de proceso a proponer debe ser seguido por los programadores externos, que parte debe ser seguida por negocio y qué parte por la gerencia de TI?	Identificar que fases del flujo de proceso serán realizadas por programadores de negocio, programadores externos y programadores de TI dentro del flujo de la metodología ágil propuesta.
¿Cuál es la metodología de desarrollo de software ágil que podrían seguir los programadores externos?	Identificar la metodología de desarrollo ágil que mejor se adapte para cumplir con características de tiempo y calidad requeridas en un desarrollo de software para los programadores de negocio

Anexo 8 CARTA DE COMPROMISO PARA ASESORÍA TEMÁTICA.

Señores Facultad de Postgrado UNITEC.

Por este medio yo _____

Identidad No. _____

Licenciado en _____

Maestría en _____

Doctorado en _____

Hago constar que asumo la responsabilidad de asesorar técnicamente el trabajo de Tesis de Maestría denominado:

A ser desarrollado por el (los) estudiante(s):

Para lo cual me comprometo a realizar de manera oportuna las revisiones y facilitar las observaciones que considere pertinentes a fin de que se logre finalizar el trabajo de tesis en el plazo establecido por la Facultad de Postgrado.

En la ciudad de _____

Departamento _____

Nombre _____

Fecha _____ Firma: _____