



UNIVERSIDAD
DE PIURA

REPOSITORIO INSTITUCIONAL
PIRHUA

INCORPORACIÓN DE LA INTEGRACIÓN CONTINUA EN EL DESARROLLO DE SOFTWARE: CASO DE ESTUDIO: ORGANISMO SUPERVISOR DE LA INVERSIÓN EN ENERGÍA Y MINERÍA

Oscar Berta-Hinostroza

Piura, noviembre de 2011

FACULTAD DE INGENIERÍA

Maestría en Dirección Estratégica en Tecnologías de la Información

Berta, O. (2011). *Incorporación de la integración continua en el desarrollo de software: caso de estudio: Organismo Supervisor de la Inversión en Energía y Minería*. Tesis de maestría en Dirección Estratégica en Tecnologías de la Información. Universidad de Piura. Facultad de Ingeniería. Piura, Perú.



Esta obra está bajo una [licencia](#)
[Creative Commons Atribución-](#)
[NoComercial-SinDerivadas 2.5 Perú](#)

Repositorio institucional PIRHUA – Universidad de Piura

**UNIVERSIDAD DE PIURA
FACULTAD DE INGENIERIA**



**Incorporación de la Integración Continua en el Desarrollo de
Software
Caso de Estudio: Organismo Supervisor de la Inversión en
Energía y Minería**

**Tesis para optar el grado de Máster en Dirección
Estratégica en Tecnologías de
Información**

Bach. Oscar Misael Berta Hinostroza

Asesor: Dr. Christian A. Estay-Niculcar

Piura, Noviembre 2011

Dedicatoria

A mi señor Jesucristo y a mi familia que llenan mi
vida de felicidad y de energía.

Agradecimiento

Agradezco a Dios todo poderoso por brindarme las fuerzas necesarias para poder seguir adelante. A mi esposa Evelyn y a mis hijos Camila y Mauro por su apoyo constante en este proceso. A mis padres por su esfuerzo en mi desarrollo personal y profesional.

A mi hermano Paul, al Dr. Víctor Pastor Talledo y a Dr. Christian Estay, quienes con su exigencia y direccionamiento lograron materializar la tesis.

Resumen

El presente trabajo de investigación ha buscado establecer la influencia que existe entre la integración continua y el desarrollo de software en el Organismo Supervisor de la Inversión en Energía y Minería (OSINERGMIN).

La investigación se trabajó con la población total del personal de sistemas de la Oficina de Sistemas y de las diversas gerencias, la cual lo constituye un (01) jefe de Sistemas y quince (15) coordinadores de sistemas. A quienes se les aplicaron cuestionarios de forma independiente.

Los resultados de estos instrumentos fueron analizados y se encontró que existe una correlación directa y positiva entre las variables: Integración Continua y el Desarrollo de Software en el Organismo Supervisor de la Inversión en Energía y Minería (OSINERGMIN).

Índice

Introducción	23
Capítulo 1: Problemas de la investigación	25
1.1 Planteamiento del problema	25
1.2 Formulación del Problema.....	26
1.3 Justificación	26
Capítulo 2: Objetivos de la investigación	29
2.1 Objetivo General	29
2.2 Objetivos Específicos	29
Capítulo 3: Fundamentación teórica	31
3.1 Antecedentes del OSINERGMIJN.....	31
3.2 Marco Teórico	36
3.3 Glosario de términos.....	41
Capítulo 4: Hipótesis y variables.....	43
4.1 Hipótesis	43
4.2 Variables.....	43
4.3 Matriz de Consistencia	44
4.4 Matriz de Operacionalización de Variables.....	45

Capítulo 5: Metodología	47
5.1 Diseño de Investigación	47
5.2 Población y Muestra.....	47
5.3 Escala de Likert.....	47
5.4 Confi abili dad	48
5.5 Procesamiento de datos estadísticos e interpretación.....	513
Capítulo 6: Resultados	127
6.1 Cuadro Final.....	127
6.2 Hipótesis.....	128
6.3 Resultados	129
Capítulo 7: Conclusiones y sugerencias	131
7.1 Conclusiones	131
7.2 Recomendaciones.....	131
Capítulo 8: Bibliografía	133
Anexo A: Cuestionario para medir la Integración Continua.....	135
Anexo B: Cuestionario para medir el Desarrollo de Software.....	137
Anexo C: Hoja de tabulación del cuestionario para medir la Integración Continua.....	139
Anexo D: Hoja de tabulación del cuestionario para medir el Desarrollo de Software	141
Anexo E: Implementación de la Integración Continua	143

Índice de Figuras

Figura 1: Organigrama del OSINERGMIN	33
Figura 2: Tipología de los proyectos de software definidos en el MIO	34
Figura 3: Ciclo de vida del software considerados en el MIO	35
Figura 4: Tamaño del proyecto al tiempo y costo según el MIO	36
Figura 5: Ejecutar construcciones en privado	38
Figura 6: Componentes de un sistema de integración continua	38
Figura 7 : Resultado de la primera pregunta del cuestionario de la variable independiente	50
Figura 8: Resultado de la segunda pregunta del cuestionario de la variable independiente	51
Figura 9: Resultado de la tercera pregunta del cuestionario de la variable independiente ..	52
Figura 10: Resultado de la cuarta pregunta del cuestionario de la variable independiente .	53
Figura 11: Resultado de la quinta pregunta del cuestionario de la variable independiente .	54
Figura 12: Resultado de la sexta pregunta del cuestionario de la variable independiente ..	55
Figura 13: Resultado de la séptima pregunta del cuestionario de la variable independiente	56
Figura 14: Resultado de la octava pregunta del cuestionario de la variable independiente	57
Figura 15: Resultado de la novena pregunta del cuestionario de la variable independiente	58
Figura 16: Resultado de la décima pregunta del cuestionario de la variable independiente	59
Figura 17: Resultado de la décima primera pregunta del cuestionario de la variable independiente	60

Figura 18: Resultado de la décima segunda pregunta del cuestionario de la variable independiente.....	61
Figura 19: Resultado de la décima tercera pregunta del cuestionario de la variable independiente.....	62
Figura 20: Resultado de la décima cuarta pregunta del cuestionario de la variable independiente.....	63
Figura 21: Resultado de la décima quinta pregunta del cuestionario de la variable independiente.....	64
Figura 22: Resultado de la décima sexta pregunta del cuestionario de la variable independiente.....	65
Figura 23: Resultado de la décima séptima pregunta del cuestionario de la variable independiente.....	66
Figura 24: Resultado de la décima octava pregunta del cuestionario de la variable independiente.....	67
Figura 25: Resultado de la décima novena pregunta del cuestionario de la variable independiente.....	68
Figura 26: Resultado de la vigésima pregunta del cuestionario de la variable independiente	69
Figura 27: Resultado de la vigésima primera pregunta del cuestionario de la variable independiente.....	70
Figura 28: Resultado de la vigésima segunda pregunta del cuestionario de la variable independiente.....	71
Figura 29: Resultado de la vigésima tercera pregunta del cuestionario de la variable independiente.....	72
Figura 30: Resultado de la vigésima cuarta pregunta del cuestionario de la variable independiente.....	73
Figura 31: Resultado de la vigésima quinta pregunta del cuestionario de la variable independiente.....	74
Figura 32: Resultado de la vigésima sexta pregunta del cuestionario de la variable independiente.....	75
Figura 33: Resultado de la vigésima séptima pregunta del cuestionario de la variable independiente	76

Figura 34: Resultado de la vigésima octava pregunta del cuestionario de la variable independiente	77
Figura 35: Resultado del indicador Repositorio del código fuente y documentos	78
Figura 36: Resultado del indicador Documentación del código fuente	79
Figura 37: Resultado del indicador Ambientes de despliegue	80
Figura 38: Resultado del indicador Frecuencia de integración Continua	81
Figura 39: Resultado del indicador Herramientas de Integración Continua	82
Figura 40: Resultado de la Dimensión Construcción Constante.....	83
Figura 41: Resultado del indicador Calidad de Software.....	84
Figura 42: Resultado del indicador Satisfacción del Usuario	85
Figura 43: Resultado de la Dimensión Pruebas Constante	86
Figura 44: Resultado del indicador Frecuencia de Inspección.....	87
Figura 45: Resultado del indicador Auditoria de código fuente	88
Figura 46: Resultado del indicador Cobertura de código fuente.....	89
Figura 47: Resultado del indicador Duplicación de código fuente	90
Figura 48: Resultado del indicador Complejidad de código fuente.....	91
Figura 49: Resultado de la Dimensión Inspección Constante.....	92
Figura 50: Resultado del indicador Mecanismos de Retroalimentación.....	93
Figura 51: Resultado de la Dimensión Retroalimentación Constante	94
Figura 52: Resultado de la Integración Continua.....	95
Figura 53: Resultado de la primera pregunta del cuestionario de la variable dependiente .	96
Figura 54: Resultado de la segunda pregunta del cuestionario de la variable dependiente.	97
Figura 55: Resultado de la tercera pregunta del cuestionario de la variable dependiente ...	98
Figura 56: Resultado de la cuarta pregunta del cuestionario de la variable dependiente	99
Figura 57: Resultado de la quinta pregunta del cuestionario de la variable dependiente..	100

Figura 58: Resultado de la sexta pregunta del cuestionario de la variable dependiente ...	101
Figura 59: Resultado de la séptima pregunta del cuestionario de la variable dependiente	102
Figura 60: Resultado de la octava pregunta del cuestionario de la variable dependiente .	103
Figura 61: Resultado de la novena pregunta del cuestionario de la variable dependiente	104
Figura 62: Resultado de la décima pregunta del cuestionario de la variable dependiente	105
Figura 63: Resultado de la décima primera pregunta del cuestionario de la variable dependiente	106
Figura 64: Resultado de la décima segunda pregunta del cuestionario de la variable dependiente	107
Figura 65: Resultado de la décima tercera pregunta del cuestionario de la variable dependiente	108
Figura 66: Resultado de la décima cuarta pregunta del cuestionario de la variable dependiente	109
Figura 67: Resultado de la décima quinta pregunta del cuestionario de la variable dependiente	110
Figura 68: Resultado de la décima sexta pregunta del cuestionario de la variable dependiente	111
Figura 69: Resultado de la décima séptima pregunta del cuestionario de la variable dependiente	112
Figura 70: Resultado del indicador Procedimiento.....	113
Figura 71: Resultado del indicador Complejidad del código fuente.....	114
Figura 72: Resultado de la Dimensión Proceso	115
Figura 73: Resultado del indicador Documentación.....	116
Figura 74: Resultado del indicador Participación del Equipo	117
Figura 75: Resultado del indicador Aplicabilidad	118
Figura 76: Resultado del indicador Retroalimentación	119
Figura 77: Resultado de la Dimensión Metodología	120
Figura 78: Resultado del Indicador Mantenibilidad	121

Figura 79: Resultado del Indicador Aporte al Negocio.....	122
Figura 80: Resultado del Indicador Satisfacción del usuario	123
Figura 81: Resultado de la Dimensión Calidad.....	124
Figura 82: Resultado del Desarrollo de Software.....	125
Figura 83 : Resultados de las variables	127
Figura 84: Modelo de integración continua para el OSINERGMIN.....	144
Figura 85: Repositorio de control de versiones.....	148
Figura 86: Panel de principal del Hudson	149
Figura 87: Panel de control del sonar	149
Figura 88: Panel de ajustes necesarios en la aplicación	150
Figura 89: Panel de principal de Nexus búsqueda de librerías.....	150

Introducción

En la actualidad nos venimos enfrentando a un profundo cambio en todos los sectores de la industria. Un mercado global que obliga a las empresas a gestionar y controlar su negocio de la forma más competitiva posible y a pensar en distribuir sus funciones geográficamente para mantener una posición estratégica en el mercado. La ventaja competitiva se encuentra en el conocimiento, la información y la habilidad para enfrentarse a las variables del mercado. La aparición continua de nuevos productos y servicios, cuyo ciclo de vida cada vez es más corta, obliga a aumentar la productividad y reducir el tiempo de reacción, adaptarse rápidamente a las necesidades de los clientes y, en definitiva, aumentar la capacidad de competir en el mercado.

Teniendo presente que la economía mundial afecta directamente a la industria del software nos encontramos con los siguientes escenarios: trabajo distribuido, descentralización de las empresas, reducción del tiempo del mercado (time to market), avances tecnológicos, requerimientos cambiantes y otros. Bajo estos escenarios, el software tiene que adaptarse al cambio constante para que este aporte al negocio.

En relación a este contexto, la integración continua nos permite para tomar al cambio como una oportunidad para desarrollar y/o mejorar los sistemas informáticos, incrementar la satisfacción del cliente y agregar valor al negocio. La integración continua no solamente se basa en herramientas y procesos, sino también en las personas que con su compromiso y actitud permiten que el desarrollo de software logre sus objetivos planteados.

En esta tesis se desarrolló la investigación de incorporar la integración continua en el desarrollo de software en el OSINERGMIN (Organismo Supervisor de la Inversión en Energía y Minería), donde el desarrollo de software es constante y debe responder adecuadamente a los cambios a las disposiciones legales y técnicas relacionadas con las actividades de los subsectores de electricidad, hidrocarburos y minería.

En esta investigación se especifica con claridad las variables, sus dimensiones y sus indicadores, lo que permitirá obtener información cuantitativa y cualitativa a partir de la cual se formularán conclusiones y sugerencias con la intención de brindar aportes significativos que permitan mejorar el desarrollo en software en la organización.

La presente investigación está organizada de manera secuencial y lógica con la intención de tener un mejor entendimiento. Es por ello, que se encuentra dividido en VIII capítulos, además de los anexos (de acuerdo a lo normado) y que describiremos brevemente a continuación:

En el primer capítulo denominado Problemas de la Investigación, se ha formulado el planteamiento del problema de investigación, el problema general, los cuatro problemas específicos y la justificación del trabajo de investigación.

El segundo capítulo denominado Objetivos de la Investigación presenta el objetivo general como los objetivos específicos, relacionados tanto con el problema general como con los problemas específicos planteados en el capítulo anterior.

El tercer capítulo denominado Fundamentación Teórica, contiene un marco teórico sobre el tema de la investigación.

En el cuarto capítulo denominado Hipótesis y Variables se presentan las hipótesis, las variables, la matriz de consistencia y la matriz de operacionalización de las variables, los cuales han permitido elaborar los cuestionarios aplicados.

En el quinto capítulo denominado Metodología se describe la metodología utilizada, describiendo el diseño de la investigación, la población estudiada, los instrumentos aplicados, el procesamiento estadístico y la interpretación de la información obtenida.

El sexto capítulo denominado Resultados contiene la presentación de datos, la contrastación de hipótesis y la interpretación y discusión de los resultados.

En el séptimo capítulo denominado Conclusiones y Sugerencias se presenta las conclusiones y sugerencias sobre la influencia de la integración continua en el desarrollo de software. Mientras que en el octavo capítulo se detallan las referencias bibliográficas.

Existen luego de los capítulos descritos, los anexos que nos han ayudado en la investigación.

Capítulo 1

Problemas de la Investigación

1.1 Planteamiento del problema

El presente proyecto de investigación se realizó porque en la actualidad los softwares juegan un papel de vital importancia en las empresas, debido a que permite recopilar, elaborar y distribuir la información para la toma de decisiones en los diferentes niveles de una organización.

Lastimosamente, los proyectos de desarrollo de software han presentado en los últimos años una serie de deficiencias que no permiten lograr su objetivo, sumándose a esto, la existencia de diversas presiones (negocio, regulaciones y otras) para que el desarrollo de software se realice en el menor tiempo posible y con alta calidad.

Las organizaciones requieren del desarrollo de software para lograr sus objetivos estratégicos, ya que el mundo cambia y las demandas del mercado cada vez más requieren de software capaces de afrontar y adecuarse al cambio, así como también muchas veces son impulsores de las transformaciones que la organización requiere. Por lo tanto, se requiere de un adecuado desarrollo de software que permita responder al dinamismo del negocio y de la organización.

Es por eso, que para lograr estos objetivos han ido apareciendo metodologías y técnicas que en la teoría nos permiten superar las deficiencias que presenta el desarrollo de software.

Una de las técnicas del desarrollo de software que ha aparecido hace pocos años es la integración continua, por lo que fue necesario investigar qué puede aportar al desarrollo de software.

La organización que se eligió para la investigación fue el Organismo Supervisor de la Inversión en Energía y Minería (OSINERGMIN), quienes desarrollan y mantienen constantemente sus softwares de acuerdo al dinamismo de su gestión como organismo del estado, pero, que presentan una serie de deficiencias en el desarrollo del software que afecta directamente a los usuarios finales y por índole a la Institución.

La investigación a desarrollarse busca constituirse en un importante aporte teórico-práctico referencial, en función al análisis sobre la influencia del ejercicio de la integración continua en el desarrollo de software de una organización.

1.2 Formulación del Problema

1.2.1 Problema General

- ¿De qué manera influye la incorporación de la integración continua en el desarrollo de software del OSINERGMIN?

1.2.2 Problemas Específicos

- ¿De qué manera influye la construcción constante de la integración continua en el desarrollo de software del OSINERGMIN?
- ¿De qué manera influye las pruebas constantes de la integración continua en el desarrollo de software del OSINERGMIN?
- ¿De qué manera influye la inspección constante de la integración continua en el desarrollo de software del OSINERGMIN?
- ¿De qué manera influye la retroalimentación constante de la integración continua en el desarrollo de software del OSINERGMIN?

1.3 Justificación

El presente trabajo de investigación es muy importante ya que permite constatar como las técnicas de la integración continua conducen y orientan a mejorar el desarrollo de software, de tal manera, que todo el equipo de desarrollo se proponga entregar un producto software de buena calidad y que cumpla con los requisitos de los usuarios y del negocio.

La investigación desarrollada ayuda a tomar conciencia sobre la importancia de lograr un buen manejo del desarrollo de software a través de la integración del código fuente, las pruebas continuas, la inspección del código fuente y la retroalimentación continua. Por supuesto, que esto debe estar acompañado de la interiorización de la integración continua por parte de todo el equipo de desarrollo del software.

Asimismo, se hace necesario investigaciones como la desarrollada, ya que el desarrollo de software cada día afronta diversos problemas que conllevan al final a un software que no cumple con las especificaciones, que no cumple con la calidad esperada, que no se realiza a tiempo o en los costos previstos, que no se genera a tiempo para el negocio y otros problemas más. Es por ello, que la integración continua aparece como una opción, que trata de reducir los problemas de desarrollo de software identificando constantemente los problemas o errores en base a las pruebas, la inspección y la retroalimentación de la salud del software.

Es por lo referido, que el presente estudio quiere servir como un aporte referencial, en función al análisis sobre el ejercicio de la integración continua y su influencia en la mejora del desarrollo de software.

El estudio es de importancia para todo profesional inmerso en el desarrollo de software, debido a que tendrán a su disposición un material de trabajo que muestra con seriedad, validez y confiabilidad la utilidad de la integración continua.

Capítulo 2

Objetivos de la Investigación

2.1 Objetivo General

- Determinar de qué manera influye la incorporación de la Integración Continua en el desarrollo de software del OSINERGMIN.

2.2 Objetivos Específicos

- Determinar de qué manera influye la construcción constante de la integración continua en el desarrollo de software del OSINERGMIN.
- Determinar de qué manera influye las pruebas constantes de la integración continua en el desarrollo de software del OSINERGMIN.
- Determinar de qué manera influye la inspección constante de la integración continua en el desarrollo de software del OSINERGMIN.
- Determinar de qué manera influye la retroalimentación constante de la integración continua en el desarrollo de software del OSINERGMIN.

Capítulo 3

Fundamentación Teórica

3.1 Antecedentes del OSINERGMIN

3.1.1 Reseña Histórica

El 31 de diciembre de 1996, mediante la Ley N° 26734 se creó un Organismo Público, entonces bajo el nombre anterior de OSINERG, encargado de supervisar y fiscalizar el cumplimiento de las disposiciones legales y técnicas de las actividades que desarrollan las empresas en los subsectores de electricidad e hidrocarburos, así como el cumplimiento de las normas legales y técnicas referidas a la conservación y protección del medio ambiente. El OSINERG inició efectivamente el ejercicio de sus funciones el 15 de octubre de 1997. Mediante Ley N° 27332, publicada el 29 de julio del 2000, se promulga la Ley Marco de los Organismos Reguladores de la Inversión Privada en los Servicios Públicos, por la cual se asignan a los organismos reguladores las funciones de supervisión, regulación, fiscalización y sanción, normativa, solución de controversias y de solución de reclamos. Según esta Ley, el OSINERG asumía las funciones de regulación que hasta esa fecha venía desarrollando la Ex Comisión de Tarifas de Energía. OSINERG asume, además, por fusión, a este ex Comisión. El 16 de abril del 2002 se promulgó la Ley Complementaria de Fortalecimiento Institucional del OSINERG, N° 27699, que amplía las facultades del Organismo Regulador, entre las cuales destaca el control de calidad y cantidad de combustibles y mayores prerrogativas dentro de su facultad sancionadora.

Finalmente, el 24 de enero del 2007, conforme los Artículos 1°, 2° y 18 de la Ley 28964, se creó el actual Organismo Supervisor de la Inversión en Energía y Minería (OSINERGMIN), como organismo regulador, supervisor y fiscalizador de las actividades que desarrollan las personas jurídicas de derecho público interno o privado y las personas naturales, en los subsectores de electricidad, hidrocarburos y minería, siendo integrante del Sistema Supervisor de la Inversión en Energía compuesto por el Instituto Nacional de Defensa de la Competencia y Protección de la Propiedad Intelectual y el Organismo Supervisor de la Inversión en Energía. Tiene personería jurídica de derecho público interno y goza de autonomía funcional, técnica, administrativa, económica y financiera. La misión del OSINERGMIN es regular, supervisar y fiscalizar, en el ámbito nacional, el cumplimiento de las disposiciones legales y técnicas relacionadas con las actividades de los subsectores de electricidad, hidrocarburos y minería, así como el cumplimiento de las

normas legales y técnicas referidas a la conservación y protección del medio ambiente en el desarrollo de dichas actividades.

3.1.2 Misión

Regular y supervisar los sectores de energía y minería con autonomía y transparencia para generar confianza a la inversión y proteger a la población.

3.1.3 Visión

Que la sociedad reciba un adecuado abastecimiento de energía y que las actividades supervisadas por OSINERGMIN se realicen en forma segura y con cuidado del medio ambiente.

3.1.4 Valores

- **Compromiso**
Actuar identificados con el Organismo y sus funciones de manera proactiva.
- **Excelencia**
Actuar con eficacia y eficiencia.
- **Servicio**
Tener la disposición para atender a los clientes y grupos de interés en los sectores minero - energéticos.
- **Integridad**
Actuar con profesionalismo y honestidad.

3.1.5 Factores críticos de éxito

Factores Críticos de Éxito	Descripción
Capital humano íntegro y de alto desempeño.	Cuenta con capital humano, profesional, competitivo, independiente, motivado e identificado con los objetivos institucionales en todos los niveles.
Marco legal para un adecuado funcionamiento del sistema.	Desarrolla y propone normas que permiten el funcionamiento adecuado del Sistema.
Participación y satisfacción de los actores del sistema.	Propicia la interacción y el involucramiento de los grupos de interés incluyendo a los usuarios e inversionistas.
Crecimiento sostenido del sistema.	Promueve el crecimiento y expansión de las inversiones en la red energética.
Desarrollo de los Modelos de Regulación y Supervisión.	Cuenta con procedimientos efectivos de regulación y supervisión, y herramientas tecnológicas que los apoyen.

Tabla 1: Factores críticos de éxito del OSINERGMIN

3.1.6 Organigrama

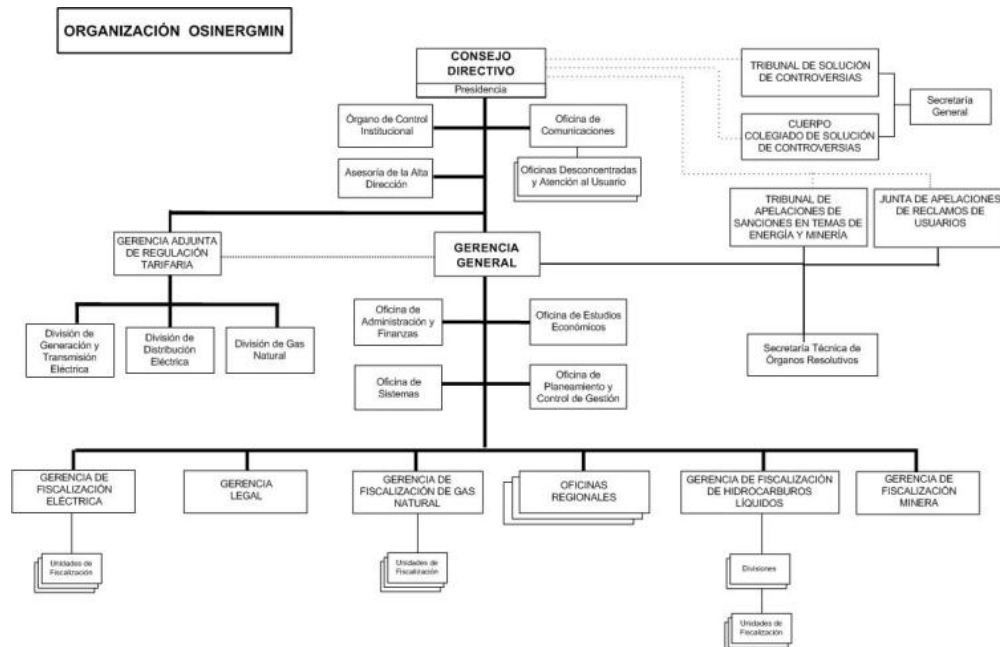


Figura 1: Organigrama del OSINERGMIN

3.1.7 Metodología Integrada del OSINERGMIN (MIO)

La Metodología Integrada del OSINERGMIN (MIO) contiene las mejores prácticas del Modelo CMMI (Capability Maturity Model Integration) y del PMI (Project Management Institute). Los estándares que contiene el MIO son modelos mundialmente aceptados como la Norma Técnica Peruana NTP-ISO/IEC 12207, COBIT, RUP y SCRUM; los cuales han sido adaptados y revisados para gestionar proveedores y proyectos específicos de software caracterizados por los siguientes criterios:

1. Tipo de Servicio:
 - Desarrollo
 - Mantenimiento
 - Implantación de producto
 - Soporte a producción
2. Ciclo de vida:
 - Iterativo
 - Incremental ágil
 - Personalización
 - Atención de requerimientos

3. Tamaño de proyecto:

- XSmall
- Small
- Medium
- Large

4. Técnica de programación:

- Orientado a objetos
- Diseño estructurado

La metodología integrada del OSINERGMIN contiene los procesos que se deben llevar a cabo para desarrollar, mantener, implantar o soportar un software cubriendo desde la modelamiento de requerimientos hasta la implantación del software.

La tipología de proyectos definidos en el MIO consiste en la definición de varios tipos de proyectos en base a características comunes, considerando 4 criterios identificados: tipo de servicio, ciclo de vida, tamaño del proyecto y técnica de programación:



Figura 2: Tipología de los proyectos de software definidos en el MIO

La conjugación de los valores de cada uno de estos criterios nos define un tipo de proyecto donde el MIO soporta tan sólo un grupo de ellos, que son los siguientes:

Tipo de Servicio	Ciclo de Vida	Tamaño	Técnica de Programación
Desarrollo	Iterativo	Large	Orientado a objetos
		Medium	
		Small	
Mantenimiento	Iterativo	Large	Orientado a objetos
	Incremental Ágil	Medium	
		Medium	
		Small	
Implantación de Producto	Personalización	Large	Orientado a objetos
		Medium	
		Small	

Para los proyectos de tamaño Small sólo es aplicable cuando el esfuerzo está en el rango de 1 a 8 meses hombre y un costo estimado de 20K a 30K.

Tabla 2: Matriz de la tipología de los proyectos de software definidos en el MIO

Los ciclos de vida que han sido considerados en el MIO son los siguientes:



Figura 3: Ciclo de vida del software considerados en el MIO

El tamaño de los proyectos que licita y ejecuta el OSINERGMIN se ha definido en el MIO por dos factores: el Costo estimado del Proyecto medido en miles de soles y el Esfuerzo estimado medido en Meses Hombre. Los proyectos se podrían ubicar en cuatro grupos: Muy Pequeño, Pequeño, Mediano y Grande:

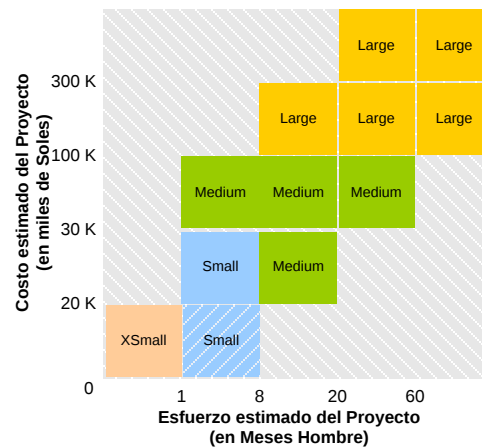


Figura 4: Tamaño del proyecto al tiempo y costo según el MIO

3.2 Marco Teórico

3.2.1 Integración continua

Martin Flower¹ describe la Integración Continua así:

“La integración continua es una práctica de desarrollo de software donde los miembros de un equipo integran su trabajo con frecuencia, por lo general cada persona integra por lo menos una vez al día – que conduce a múltiples integraciones por día. Cada integración es verificada por una construcción automatizada (incluidas pruebas) para detectar errores de integración lo más rápido posible. Muchos equipos encuentran que este enfoque conduce a reducir significativamente los problemas de integración y permite desarrollar software cohesivo más rápidamente”

2.2.2 Beneficios de la integración continua

- **Reducir los riesgos**

La integración continua (varias veces al día) nos ayuda a mitigar los riesgos de un proyecto de software como son: pérdida de cohesión, detección tardía de los defectos, baja calidad del software y la pérdida de visibilidad del proyecto.

- **Reducir los procesos repetitivos**

La integración continua automatiza la compilación de código fuente, pruebas, integración de base de datos, inspección, implementación y retroalimentación. Reduciendo así los costos, tiempo y esfuerzo.

¹ Flower, Martin. (2006). *Continuous Integration [en línea]*.

<<http://www.martinfowler.com/articles/continuousIntegration.html>>

- **Generar software de despliegue**
Este beneficio es el más tangible a los clientes como al equipo de desarrollo, ya que la integración continua nos permite generar el software en cualquier punto de tiempo con alto niveles de calidad.
- **Permitir una mejor visibilidad del proyecto**
La integración al ejecutarse continuamente proporciona la habilidad de percibir los problemas o errores rápidamente, lo cual nos ayuda a tomar decisiones basadas en información real.
- **Establecer una mayor confianza del producto**
La aplicación práctica y efectiva de la integración continua puede proporcionar una mayor confianza en la construcción de un producto de software, debido a que por cada construcción, el equipo conoce las pruebas que son ejecutadas contra el software y verifica su comportamiento, conoce los estándares de diseño y código que reúnen y conoce el resultado funcional del producto.

3.2.3 Prácticas de la integración continua

- **Sincroniza y sube código frecuentemente**
Debemos sincronizar y subir el código fuente al repositorio de control de versiones por lo menos una vez al día.
- **No subas código que haga fallar la construcción**
Debemos tener cuidado de no subir código fuente al repositorio de control de versiones cuando este no se puede compilar con otro código fuente o posea fallas en las pruebas.
- **Reparar las construcciones rotas**
Cuando un equipo opera en un entorno de integración continua, los problemas que se presenten a lo largo del desarrollo deben de corregirse de inmediato y debe involucrar a todo el equipo de desarrollo si fuera necesario.
- **Escribir pruebas de desarrollo automatizadas**
Verificar que el funcionamiento correcto del software con pruebas automatizadas y ejecutadas frecuentemente.
- **Todas las pruebas e inspecciones deben superarse**
El 100% de las pruebas automatizadas deben ser satisfactorias para que la construcción pueda subir al repositorio de control de versiones, ya que sino se cumple esta práctica se abaticina que el software construido será de mala calidad.
- **Ejecutar construcciones en privado**
Para evitar las construcciones rotas, el desarrollador debería obtener los cambios recientes del equipo de desarrollo desde el repositorio de control de versiones y ejecutar localmente la integración del código.

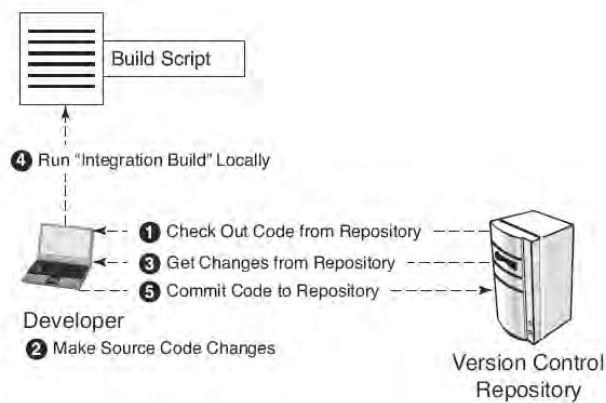


Figura 5: Ejecutar construcciones en privado

- **Evitar obtener código roto**

Cuando la construcción se ha roto, debemos gastar tiempo en encontrar la solución del error, comprometiendo a todo el equipo de desarrollo, para luego recién obtener el último código en el repositorio de control de versiones para continuar con el desarrollo.

3.2.4 Componentes de un sistema de integración continua

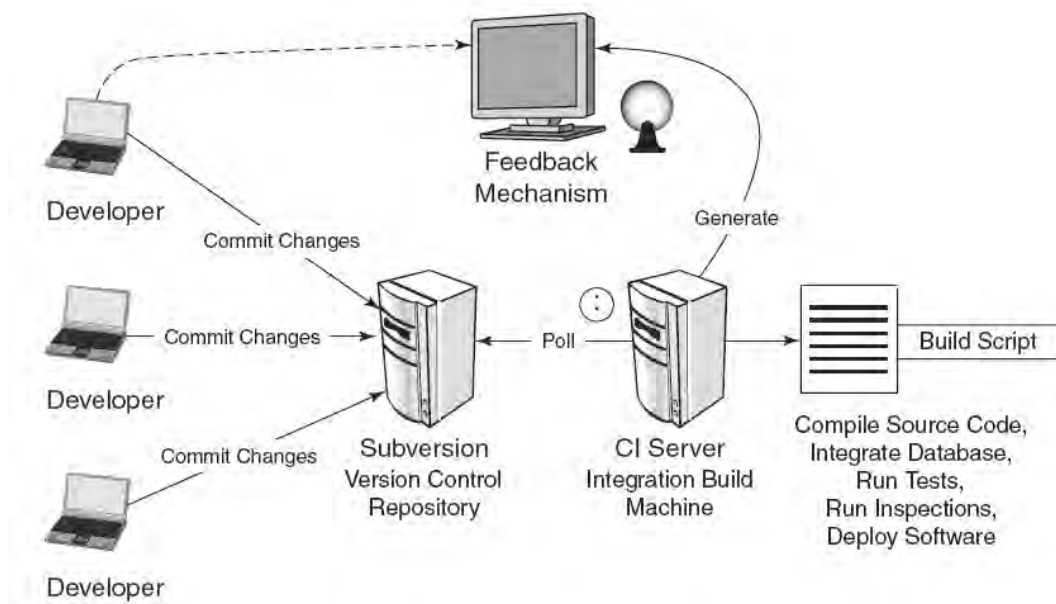


Figura 6: Componentes de un sistema de integración continua

- **Desarrolladores**

La Integración Continua requiere de una disciplina por parte de los desarrolladores para lograr software de calidad, por lo tanto, la disciplina que tienen que cumplir está basada en las prácticas de la Integración Continua anteriormente expuestas.

- **Repositorio de control de versiones**

El repositorio de control de versiones permite al equipo de desarrollo trabajar en forma colaborativa, basándose en mantener todos los archivos del proyecto en un lugar centralizado, normalmente un único servidor, aunque también hay sistemas distribuidos, donde los desarrolladores se conectan y descargan una copia local del proyecto. Con ella, envían periódicamente los cambios que realizan al servidor y van actualizando su directorio de trabajo que otros usuarios a su vez han ido modificando.

Los repositorios de control de versiones suelen aportar una amplia gama de funcionalidades, de las cuales se extraen continuación las más relevantes:

- a. **Control de usuarios.** Establecer qué usuarios tienen acceso a los archivos del proyecto de desarrollo de software y qué tipo de acceso tienen asignado.
- b. **Mantener un control detallado de los cambios realizados en cada archivo.** Disponer de un control de los cambios efectuados en cada archivo del proyecto de desarrollo de software, indicando: fecha, causa del cambio y quién los realizó. Además, permite mantener las diferentes versiones de un solo archivo.
- c. **Resolver conflictos de actualización de archivos.** En caso de que dos usuarios modifiquen un archivo, el repositorio proporciona herramientas para gestionar este tipo de situaciones.
- d. **Bifurcación de proyectos.** A partir de un punto concreto, crear ramas del proyecto para que puedan evolucionar por separado. Es habitual utilizar esta técnica cuando se ha liberado una versión de un producto y se requiere que siga evolucionando.

- **Servidor de integración continua**

El servidor de integración continua es el orquestador de todo el proceso y debe estar configurado para comprobar los cambios en el repositorio de control de versiones cada pocos minutos o menos. Por cada construcción el servidor comúnmente cuenta con reportes y un panel de control para mostrarnos la salud del software.

- **Script de construcción**

Muchos de los pasos para la construcción de un software son definidos a través de scripts que constan de una o varias secuencias para compilar, probar, inspeccionar y desplegar el software.

- **Mecanismos de retroalimentación**

Una vez finalizada la construcción, los resultados deben ser accesibles tan pronto como sea posible, brindando información respecto a la salud del software desarrollado a todo el equipo de desarrollo para la toma de decisiones más adecuada.

- **Gestión de dependencias**

El desarrollo de software utiliza artefactos propios o de terceros para complementar sus funcionalidades. Es por eso que se necesita la gestión de las dependencias del software para poder conocer que artefactos se están utilizando y que versiones.

3.2.5 Características de la integración continua

Las características de la Integración continua son las siguientes:

- **Compilación de código fuente**

Compilación implica la creación del código ejecutable de su fuente legible. En la integración continua la compilación de código fuente es una de las características más básicas y comunes de un sistema de integración continua.

- **Integración de base de datos**

Mediante el uso de un sistema de integración continua se puede garantizar la integración de la base de datos a través de una sola fuente (repositorio de control de versiones), es decir, se puede ejecutar sentencias DDL y DML, procedimientos almacenados y otros.

- **Pruebas**

Las pruebas son pieza fundamental de la integración continua para lograr un desarrollo del software de calidad.

- **Inspección**

La inspección se basa en las normas o mejores prácticas del desarrollo de software para poder validar la calidad de la programación.

- **Despliegue**

El sistema de integración continua tiene como objetivo clave la generación de los artefactos de software (con los últimos cambios) y la disposición de los artefactos en el entorno adecuado (desarrollo, pruebas o producción).

- **Documentación**

El sistema de integración continua se basa en el código fuente del repositorio de control de versiones para crear o actualizar la documentación del software.

- **Retroalimentación continua**

La retroalimentación proporciona información de la salud del software para la toma de decisiones.

3.3 Glosario de términos

- **Metodología**
Es la aplicación real de un conjunto de procesos, estructuras de roles, políticas, métricas y herramientas conceptuales a determinada problemática de manera ordenada y coherente.
- **Construcción**
Un conjunto de actividades llevadas a cabo para generar, probar, inspeccionar e implementar el software.
- **Ambiente de desarrollo**
Es el entorno en cual el software es escrito. Esto puede incluir el entorno integrado de desarrollo, los scripts de construcción, las herramientas, las librerías, los servidores y los archivos de configuración.
- **Inspección**
Es el análisis de código fuente con respecto a la calidad interna del software.
- **Calidad** ²
Es un atributo esencial y distinto de algo o alguien.
- **Integración continua**
La integración continua es una práctica de desarrollo de software donde los miembros de un equipo integran su trabajo con frecuencia, por lo general cada persona integra por lo menos una vez al día – que conduce a múltiples integraciones por día. Cada integración es verificada por una construcción automatizada (incluidas pruebas) para detectar errores de integración lo más rápido posible. Muchos equipos encuentran que este enfoque conduce a reducir significativamente los problemas de integración y permite desarrollar software cohesivo más rápidamente.
- **Pruebas**
Es el proceso de verificar y revelar la calidad de un software.
- **Proceso**
Secuencia de pasos ejecutados con un propósito específico.
- **Proceso de software**
Es un conjunto de actividades, métodos, prácticas y transformaciones que el personal usa para desarrollar y mantener el software, y los productos asociados (planificación del proyecto, diseño de documentos, código fuente, casos de prueba, manuales y otros). (SEI, 1995)

² Quality [en línea]. <<http://www.thefreedictionary.com/quality>>

- **Repositorio de control de versiones**
Es un sistema que permite mantener distintas versiones de un archivo, generalmente código fuente, librerías y documentación.
- **Construcción en privado**
Realizar la construcción del software en forma local (en su computadora) antes de ejecutar los cambios en el repositorio de control de versiones.
- **Servidor de integración continua**
Nos permite orquestar todo el proceso de la Integración Continua, ya que nos permite la construcción y pruebas del software en forma continua y además nos brinda la información de los resultados obtenidos de la construcción y pruebas.
- **Software libre**
Es aquel tipo de software donde usuario una vez obtenido puede ser usado, modificado y redistribuido libremente.
- **Plugins**
Es un complemento que permite una aplicación se relaciona con otra con la finalidad de agregarle valor.
- **Compilación**
Permite la traducción del código de un lenguaje de programación a un código que la maquina es capaz de interpretar.
- **Artefactos**
Son los productos resultantes del proceso de desarrollo de software.

Capítulo 4

Hipótesis y Variables

4.1 Hipótesis

4.1.1 Hipótesis General

- La incorporación de la integración continua influye en el desarrollo de software del OSINERGMIN.

4.1.2 Hipótesis Específicas

- La construcción constante de la integración continua influye en el desarrollo de software del OSINERGMIN.
- Las pruebas constantes de la integración continua influye en el desarrollo de software del OSINERGMIN.
- La inspección constante de la integración continua influye en el desarrollo de software del OSINERGMIN.
- La retroalimentación constante de la integración continua influye en el desarrollo de software del OSINERGMIN.

4.2 Variables

4.2.1 Variable Independiente

- Integración continua.

4.2.2 Variable Dependiente

- Desarrollo de software.

4.3 Matriz de Consistencia

	Problema	Objetivo	Hipótesis	Variables
General	¿De qué manera influye la incorporación de la integración continua en el desarrollo de software del OSINERGMIN?	Determinar de qué manera influye la incorporación de la Integración Continua en el desarrollo de software del OSINERGMIN.	La incorporación de la integración continua influye en el desarrollo de software del OSINERGMIN.	Variable Independiente Integración continua
Específicos	¿De qué manera influye la construcción constante de la integración continua en el desarrollo de software del OSINERGMIN?	Determinar de qué manera influye la construcción constante de la integración continua en el desarrollo de software del OSINERGMIN.	La construcción constante de la integración continua influye en el desarrollo de software del OSINERGMIN.	Variable Dependiente Desarrollo de software
	¿De qué manera influyen las pruebas constantes de la integración continua en el desarrollo de software del OSINERGMIN?	Determinar de qué manera influye las pruebas constantes de la integración continua en el desarrollo de software del OSINERGMIN.	Las pruebas constantes de la integración continua influyen en el desarrollo de software del OSINERGMIN.	
	¿De qué manera influye la inspección constante de la integración continua en el desarrollo de software del OSINERGMIN?	Determinar de qué manera influye la inspección constante de la integración continua en el desarrollo de software del OSINERGMIN.	La inspección constante de la integración continua influye en el desarrollo de software del OSINERGMIN.	
	¿De qué manera influye la retroalimentación constante de la integración continua en el desarrollo de software del OSINERGMIN?	Determinar de qué manera influye la retroalimentación constante de la integración continua en el desarrollo de software del OSINERGMIN.	La retroalimentación constante de la integración continua influye en el desarrollo de software del OSINERGMIN.	

Tabla 1: Matriz de Consistencia

4.4 Matriz de Operacionalización de Variables

Variables (tipo)	Definición	Dimensiones	Definición	Indicadores	Ítems
Variable Independiente Integración Continua	La integración continua es una práctica de desarrollo de software donde los miembros de un equipo integran su trabajo con frecuencia, por lo general cada persona integra por lo menos una vez al día – que conduce a múltiples integraciones por día. Cada integración es verificada por una construcción automatizada (incluidas pruebas) para detectar errores de integración lo más rápido posible.	Construcción Constante	Hace referencia a la construcción continua del software.	Repositorio del código fuente y documentos Documentación del código fuente Ambientes del despliegue Frecuencia de Integración Continua Herramientas de Integración Continua	1,2,3 4,5 6,7 8,9,10 11,12
		Pruebas Constante	Proceso que permite verificar y revelar la calidad del software.	Calidad del software Satisfacción del usuario	13,14 15
		Inspección Constante	Revisión del código fuente respecto a las normas o reglas (mejores prácticas) que deben cumplir.	Frecuencia de la Inspección Auditoria de código fuente Cobertura de código fuente Duplicación de código fuente Complejidad de código fuente	16,17 18,19 20,21 22,23 24,25
		Retroalimentación Constante	Brindar la información adecuada a las personas adecuadas en el momento adecuado y de manera adecuada y en línea.	Mecanismos de la retroalimentación	26,27,28

Variable Dependiente Desarrollo de Software	Es el proceso que transforma los requerimientos de los usuarios en productos de software.	Proceso Metodología Calidad del Producto software	Hace referencia al proceso de desarrollo de software. Hace referencia a la metodología para el desarrollo de software. El grado de cumplimiento del producto software respecto a los requerimientos del usuario.	Procedimiento Complejidad Documentación Participación del equipo Aplicabilidad Retroalimentación Mantenibilidad Aporte al Negocio Satisfacción del usuario	1,2,3,4 5,6 7,8 9,10 11 12,13 14,15 16 17

Tabla 2: Matriz de Operacionalización de Variables

Capítulo 5

Metodología

5.1 Diseño de Investigación

El diseño de la investigación es no experimental transversal, la cual tiene el propósito de recoger datos en un tiempo determinado sin intervenir en los sujetos y ambientes del estudio, por lo tanto, no habrá manipulación de variables.

El tipo de estudio es descriptivo y correlacional ya que se busca analizar la relación que existe entre dos o más conceptos, categorías o variables (como es en nuestra investigación).

5.2 Población y Muestra

La población del estudio está conformada por la totalidad del personal de sistemas de la Oficina de Sistemas y de las diversas gerencias del Organismo Supervisor de la Inversión en Energía y Minería (OSINERGMIN), la cual lo constituye un (01) jefe de Sistemas y quince (15) coordinadores de sistemas.

5.3 Escala de Likert

Se elaboró dos cuestionarios de 28 y 17 preguntas respectivamente, que fueron respondidas por el personal de sistemas utilizando la escala de Likert, cuyas alternativas fueron las siguientes:

- a. Totalmente en desacuerdo.
- b. En desacuerdo.
- c. Ni de acuerdo ni en desacuerdo.
- d. De acuerdo.
- e. Totalmente de acuerdo.

5.4 Confiabilidad

Es el grado en que un instrumento mide con precisión, sin error. Indica la condición del instrumento de ser fiable, es decir, de ser capaz de ofrecer en su empleo repetido resultados veraces y constantes en condiciones similares de medición.

La fiabilidad de un instrumento de medida se valora a través de la consistencia, la estabilidad temporal y la concordancia inter observadores.

- **Consistencia:** Se refiere al nivel en que los diferentes ítems o preguntas de una escala están relacionados entre sí. Esta homogeneidad entre los ítems nos indica el grado de acuerdo entre los mismos y, por tanto, lo que determinará que éstos se puedan acumular y dar una puntuación global. La consistencia se puede comprobar a través de diferentes métodos estadísticos.

El coeficiente alfa de Cronbach es un método estadístico muy utilizado. Sus valores oscilan entre 0 y 1. Se considera que existe una buena consistencia interna cuando el valor de alfa es superior a 0,7.

- **Estabilidad temporal:** Es la concordancia obtenida entre los resultados del test al ser evaluada la misma muestra por el mismo evaluador en dos situaciones distintas (fiabilidad test-retest).

La fiabilidad (normalmente calculada con el coeficiente de correlación intraclase [CCI], para variables continuas y evaluaciones temporales distantes) nos indica que el resultado de la medida tiene estabilidad temporal. Una correlación del 70% indicaría una fiabilidad aceptable.

- **Concordancia íter observadores:** En el análisis del nivel de acuerdo obtenido al ser evaluada la misma muestra en las mismas condiciones por dos evaluadores distintos, o en diferente tiempo, se obtienen iguales resultados-fiabilidad íter observadores.

El Alfa de Cronbach

Se utiliza la fórmula:

$$\alpha = \left[\frac{K}{K - 1} \right] \left[1 - \frac{\sum_{i=1}^k S_i^2}{S_t^2} \right]$$

Siendo

S_i^2	La suma de varianzas de cada ítem.
S_t^2	La varianza del total de filas (puntaje total de los jueces)
K	El número de preguntas o ítems.

Para el cuestionario de la variable independiente “Integración Continua” se obtuvo un alfa de Cronbach de 91%, siendo este un porcentaje aceptable.

E	Preguntas																												Total
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	
1	5	5	5	4	5	4	4	5	4	5	5	5	5	5	5	5	5	5	5	5	5	5	4	5	5	5	5	5	135
2	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	140
3	5	5	4	5	5	4	4	4	4	5	5	5	5	5	4	4	5	4	4	4	5	4	4	4	5	5	4	4	125
4	4	5	5	4	5	4	5	4	4	5	5	5	5	5	5	5	4	4	5	5	4	5	4	5	5	5	5	4	130
5	5	4	5	5	4	4	4	3	5	4	5	4	4	4	5	5	5	4	4	4	5	4	4	5	5	5	5	4	124
6	4	5	5	4	5	4	5	4	4	5	5	5	5	5	4	5	5	4	5	5	5	5	5	5	5	5	5	5	133
7	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	4	4	4	5	4	4	4	5	5	4	4	132
8	5	5	5	5	4	5	5	4	5	5	5	5	5	5	5	5	5	4	5	5	4	5	4	5	4	5	5	5	134
9	5	5	4	4	5	4	4	3	4	4	4	4	5	4	5	5	4	4	4	4	5	4	4	4	4	4	5	4	119
10	4	5	5	5	5	5	5	5	5	5	5	5	5	5	4	5	5	5	4	5	5	5	5	5	5	5	5	5	137
11	5	5	5	5	4	5	5	4	5	5	5	5	5	5	5	5	5	5	5	5	4	4	4	4	4	4	4	4	130
12	5	5	5	4	5	4	4	4	4	5	5	4	4	4	5	4	4	4	4	4	5	4	4	4	5	5	5	4	123
13	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	140
14	4	5	5	4	4	4	4	4	5	5	5	5	5	5	4	5	5	4	4	4	4	4	4	4	5	5	5	4	126
15	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	140
16	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	140
	0.20	0.06	0.12	0.25	0.20	0.27	0.25	0.50	0.25	0.12	0.06	0.16	0.12	0.16	0.20	0.12	0.16	0.26	0.26	0.25	0.20	0.26	0.25	0.23	0.16	0.12	0.16	0.27	

Tabla 1: Matriz del cálculo del alfa de Cronbach del Cuestionario de la Variable Independiente

Para el cuestionario de la variable dependiente “Desarrollo de Software” se obtuvo un alfa de Cronbach de 82%, siendo este un porcentaje aceptable.

E	Preguntas																	Total
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	
1	5	5	5	5	4	4	5	5	4	5	5	1	1	4	5	5	5	73
2	5	5	5	5	4	4	5	5	4	5	5	1	1	4	5	5	5	73
3	4	4	5	4	2	2	5	5	4	5	5	1	1	1	4	5	4	61
4	4	4	4	2	1	1	2	4	2	5	5	1	1	1	4	5	4	50
5	4	4	5	4	2	2	4	4	2	5	5	1	1	1	1	5	4	54
6	4	4	5	4	5	2	5	5	5	4	4	2	2	1	4	4	5	65
7	5	5	5	5	2	2	5	5	2	5	4	2	2	1	1	4	2	57
8	4	4	4	4	1	1	5	5	1	5	5	3	3	2	1	5	4	57
9	4	4	5	4	2	1	5	5	1	5	4	2	2	2	1	4	2	53
10	5	5	5	5	4	2	4	4	2	4	4	2	2	2	2	4	2	58
11	5	5	5	5	4	2	4	4	2	4	4	2	2	2	2	4	1	57
12	4	4	4	4	2	1	5	5	1	5	4	3	3	2	2	4	1	54
13	5	5	5	5	2	1	4	5	1	5	4	2	2	2	2	4	1	55
14	4	4	4	2	2	1	4	5	2	5	4	2	2	1	5	4	4	55
15	5	5	5	5	5	5	5	5	5	5	5	1	1	4	4	5	5	75
16	5	5	5	5	5	5	5	5	5	5	5	1	1	5	5	5	5	77
	0.27	0.27	0.20	1.00	2.06	2.07	0.67	0.20	2.36	0.16	0.27	0.50	0.50	1.76	2.67	0.27	2.52	

Tabla 2: Matriz del cálculo del alfa de Cronbach del Cuestionario de la Variable Dependiente

5.5 Procesamiento de datos estadísticos e interpretación

Cuestionario aplicado para la Variable Independiente Integración Continua

1. El repositorio del código fuente y documentos ha permitido centralizar la información del software.

Categorías	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.0%
b. En desacuerdo	0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0%
d. De acuerdo	4	25.0%
e. Totalmente de acuerdo	12	75.0%
<i>Total</i>	16	100%

Tabla 3: Resultado de la primera pregunta del cuestionario de la variable independiente

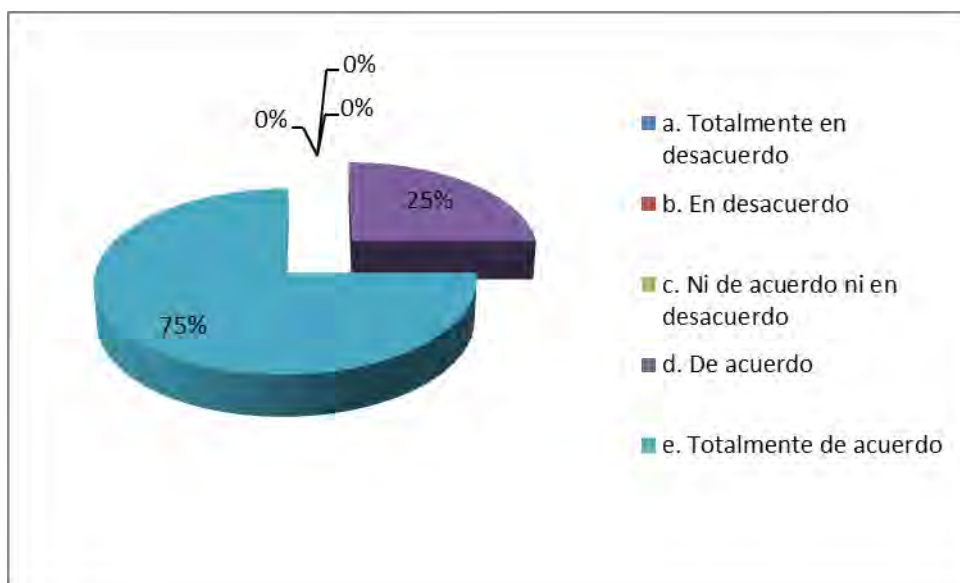


Figura 1 : Resultado de la primera pregunta del cuestionario de la variable independiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que el repositorio del código fuente y documentos le ha permitido centralizar la información del software.

2. El versionar el código fuente y documentos ha permitido tener un control de la historia de la información del software.

Categorías	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.0%
b. En desacuerdo	0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0%
d. De acuerdo	1	6.25%
e. Totalmente de acuerdo	15	93.75%
<i>Total</i>	16	100%

Tabla 4: Resultado de la segunda pregunta del cuestionario de la variable independiente

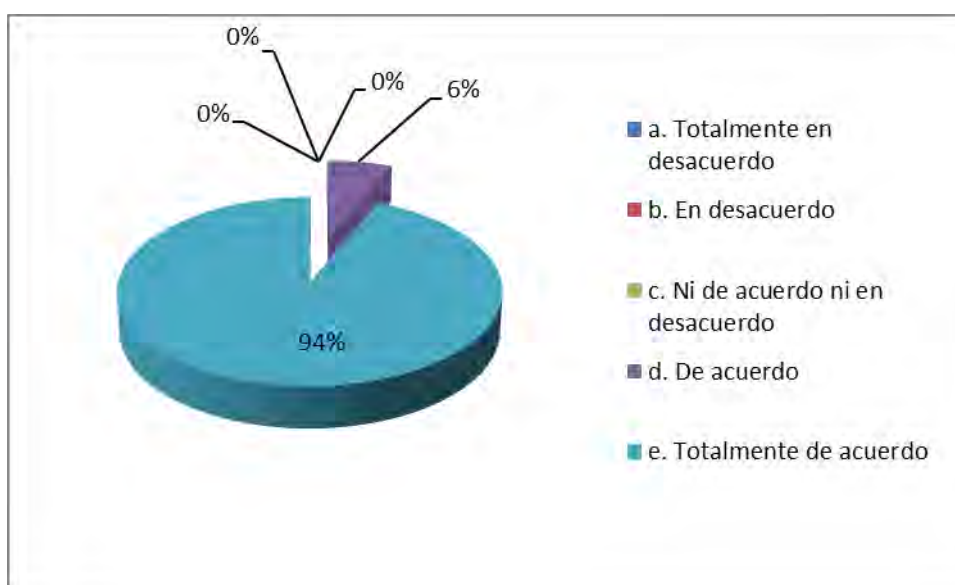


Figura 2: Resultado de la segunda pregunta del cuestionario de la variable independiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que el versionamiento del código fuente y documentos le ha permitido tener un control de la trazabilidad de la información del software.

3. El repositorio del código fuente y documentos ha permitido potenciar el trabajo colaborativo del equipo de desarrollo.

Categorías	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.0%
b. En desacuerdo	0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0%
d. De acuerdo	2	12.5%
e. Totalmente de acuerdo	14	87.5%
<i>Total</i>	16	100%

Tabla 5: Resultado de la tercera pregunta del cuestionario de la variable independiente

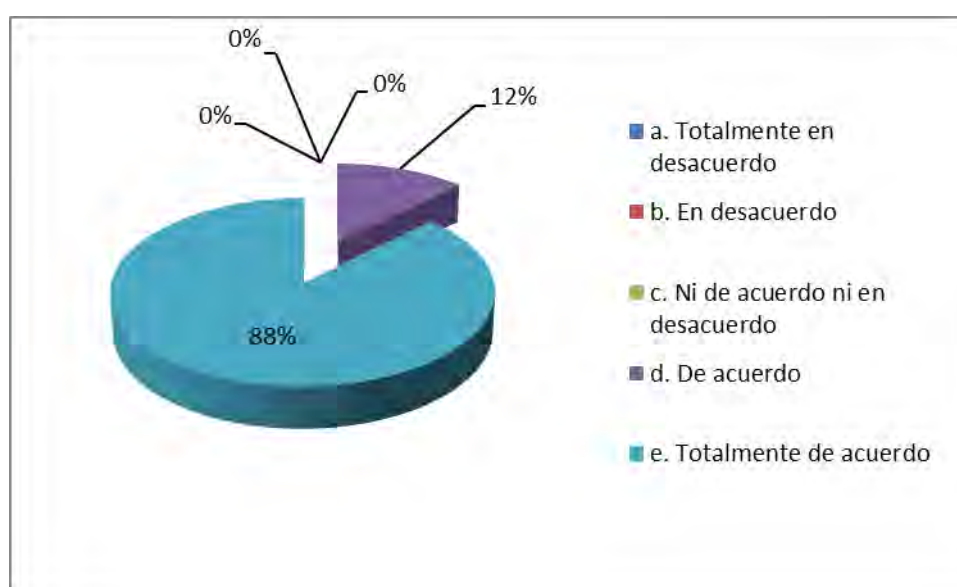


Figura 3: Resultado de la tercera pregunta del cuestionario de la variable independiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que el repositorio del código fuente y documentos le ha permitido potenciar el trabajo colaborativo del equipo de desarrollo.

4. La documentación del código fuente agrega valor al desarrollo de software.

Categorías	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.0%
b. En desacuerdo	0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0%
d. De acuerdo	6	37.5%
e. Totalmente de acuerdo	10	62.5%
<i>Total</i>	16	100%

Tabla 6: Resultado de la cuarta pregunta del cuestionario de la variable independiente

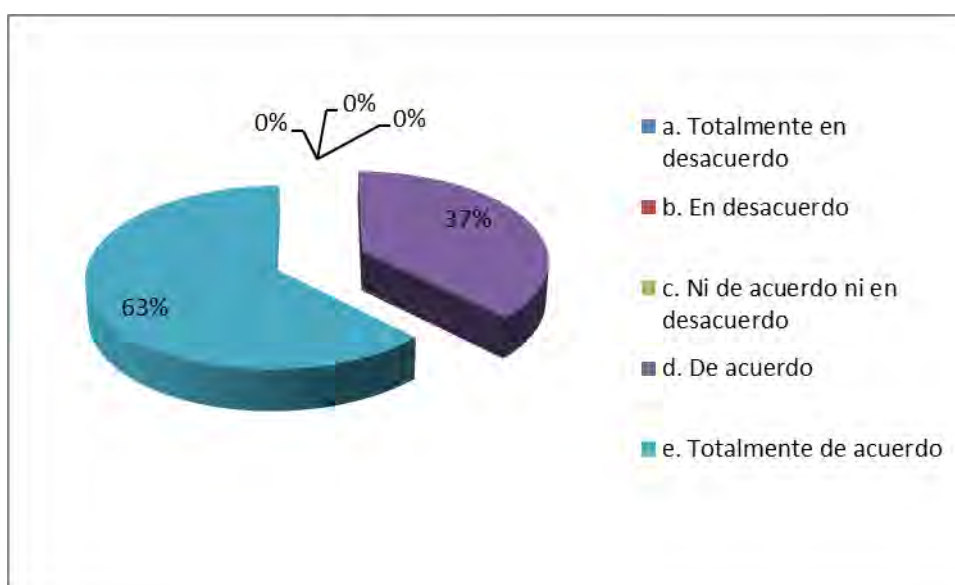


Figura 4: Resultado de la cuarta pregunta del cuestionario de la variable independiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que la documentación en tiempo real del código fuente agrega valor al desarrollo de software.

5. La documentación del código fuente debe generarse constantemente.

Categorías	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.0%
b. En desacuerdo	0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0%
d. De acuerdo	4	25.0%
e. Totalmente de acuerdo	12	75.0%
<i>Total</i>	16	100%

Tabla 7: Resultado de la quinta pregunta del cuestionario de la variable independiente

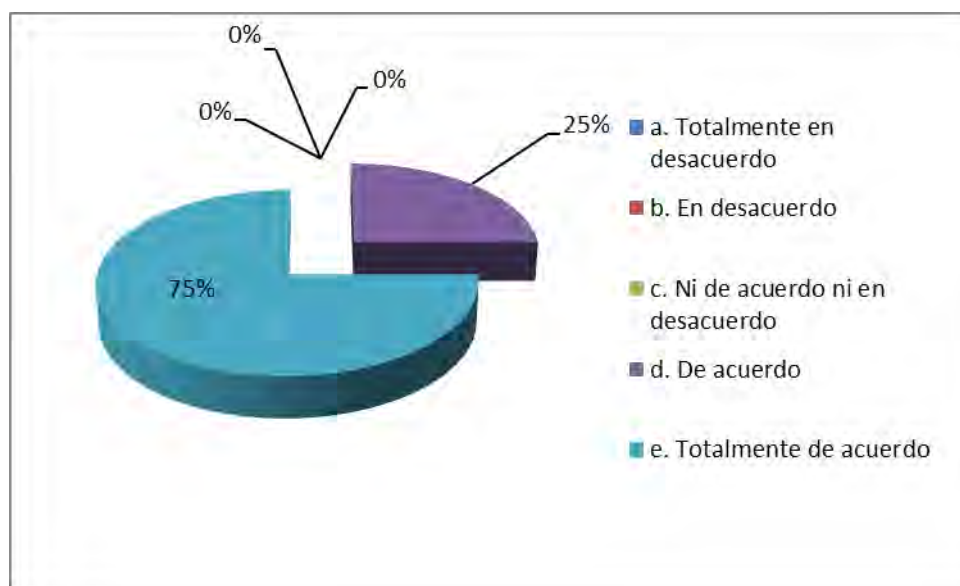


Figura 5: Resultado de la quinta pregunta del cuestionario de la variable independiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que la documentación debe generarse constantemente.

6. La integración continua permite desplegar el software rápidamente en los ambientes de desarrollo, pre producción y producción.		
Categorías	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.0%
b. En desacuerdo	0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0%
d. De acuerdo	8	50.0%
e. Totalmente de acuerdo	8	50.0%
<i>Total</i>	16	100%

Tabla 8: Resultado de la sexta pregunta del cuestionario de la variable independiente

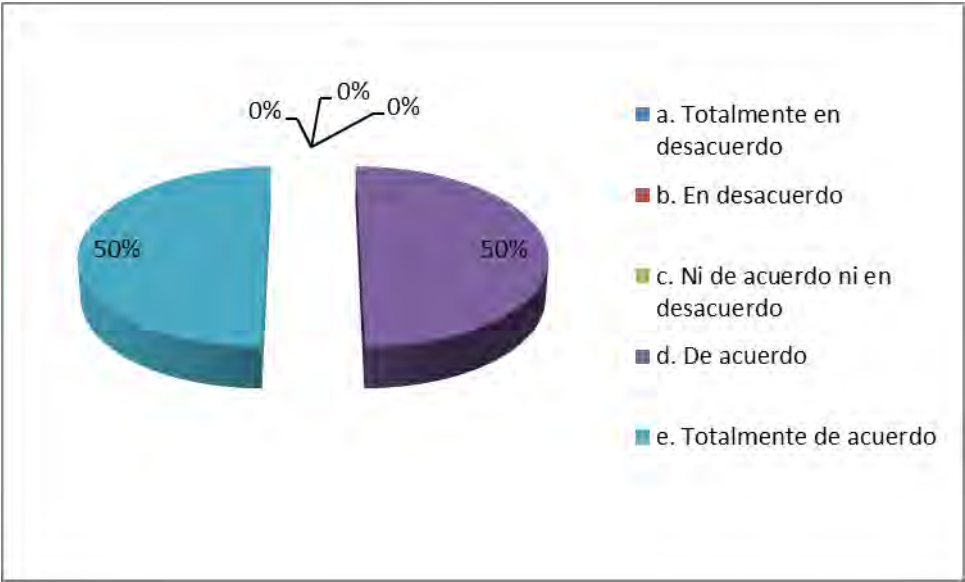


Figura 6: Resultado de la sexta pregunta del cuestionario de la variable independiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que los ambientes de desarrollo, pre producción y producción son rápidamente desplegados por medio de la integración continua.

7. La integración continua permite construir reportes de retroalimentación que permite conocer el estado del software desplegado en los ambientes de desarrollo, pre producción y producción.

Categorías	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.0%
b. En desacuerdo	0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0%
d. De acuerdo	6	37.5%
e. Totalmente de acuerdo	10	62.5%
<i>Total</i>	16	100%

Tabla 9: Resultado de la séptima pregunta del cuestionario de la variable independiente

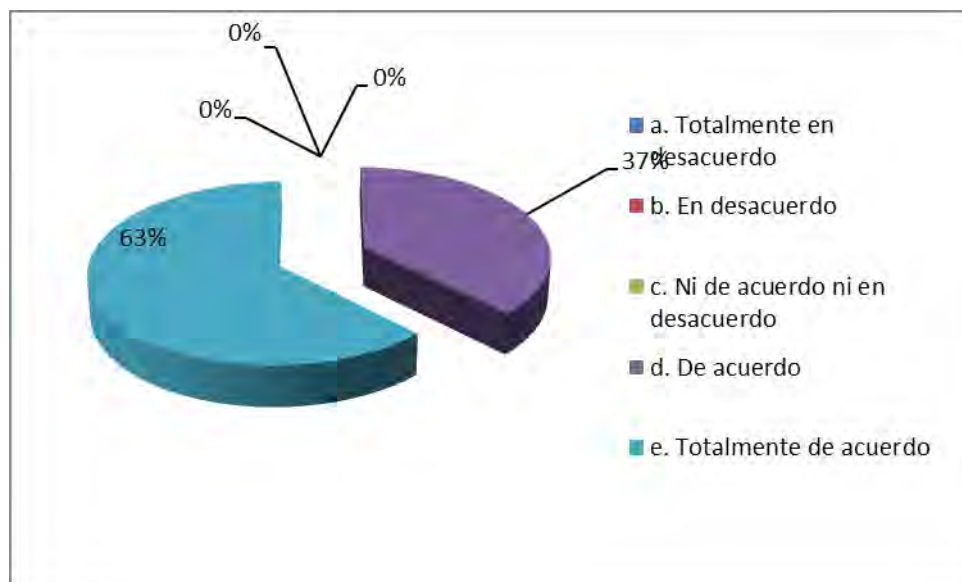


Figura 7: Resultado de la séptima pregunta del cuestionario de la variable independiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que la integración continua permite construir reportes de retroalimentación que permiten conocer el estado del software desplegado en los ambientes de desarrollo, pre producción y producción.

8.Cuál es tu posición respecto a que la integración se realice al menos una vez al día.

Categorías	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.0%
b. En desacuerdo	0	0.0%
c. Ni de acuerdo ni en desacuerdo	2	12.5%
d. De acuerdo	7	43.75%
e. Totalmente de acuerdo	7	43.75%
<i>Total</i>	16	100%

Tabla 10: Resultado de la octava pregunta del cuestionario de la variable independiente

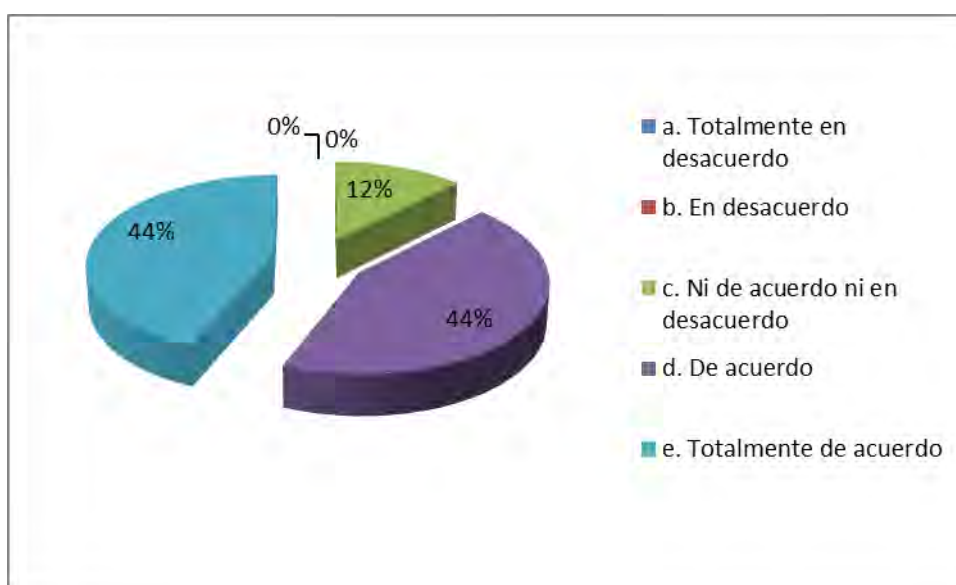


Figura 8: Resultado de la octava pregunta del cuestionario de la variable independiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que la integración del código fuente se realice al menos una vez al día.

9. La frecuencia de la integración continua ha permitido arreglar construcciones rotas rápidamente.

Categorías	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.0%
b. En desacuerdo	0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0%
d. De acuerdo	6	37.5%
e. Totalmente de acuerdo	10	62.5%
<i>Total</i>	16	100%

Tabla 11: Resultado de la novena pregunta del cuestionario de la variable independiente

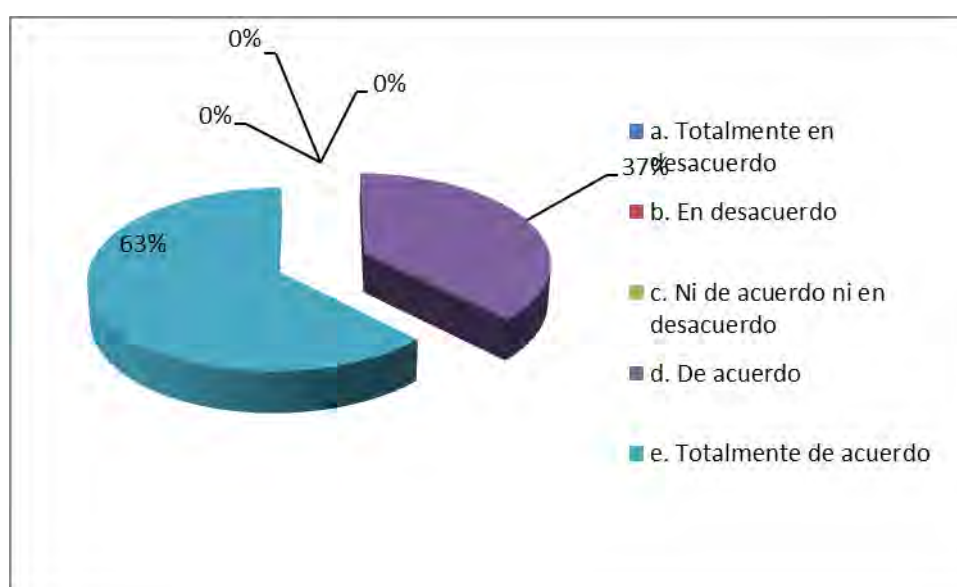


Figura 9: Resultado de la novena pregunta del cuestionario de la variable independiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que la frecuencia de la integración del código fuente ha permitido corregir construcciones rotas rápidamente.

10. La frecuencia de la integración continua ha permitido potenciar el trabajo colaborativo del equipo de desarrollo.		
Categorías	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.0%
b. En desacuerdo	0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0%
d. De acuerdo	2	12.5%
e. Totalmente de acuerdo	14	87.5%
<i>Total</i>	16	100%

Tabla 12: Resultado de la décima pregunta del cuestionario de la variable independiente

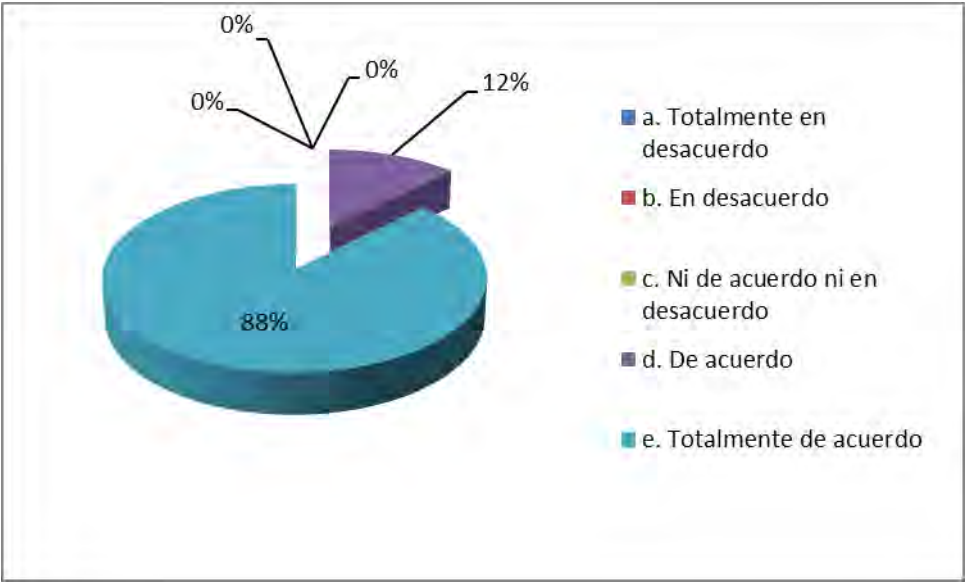


Figura 10: Resultado de la décima pregunta del cuestionario de la variable independiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que la frecuencia de la integración del código fuente ha permitido potenciar el trabajo colaborativo del equipo de desarrollo.

11. Las herramientas de integración continua le facilitan el trabajo de monitoreo y control del desarrollo de software.

Categorías	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.0%
b. En desacuerdo	0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0%
d. De acuerdo	1	6.25%
e. Totalmente de acuerdo	15	93.75%
<i>Total</i>	16	100%

Tabla 13: Resultado de la décima primera pregunta del cuestionario de la variable independiente

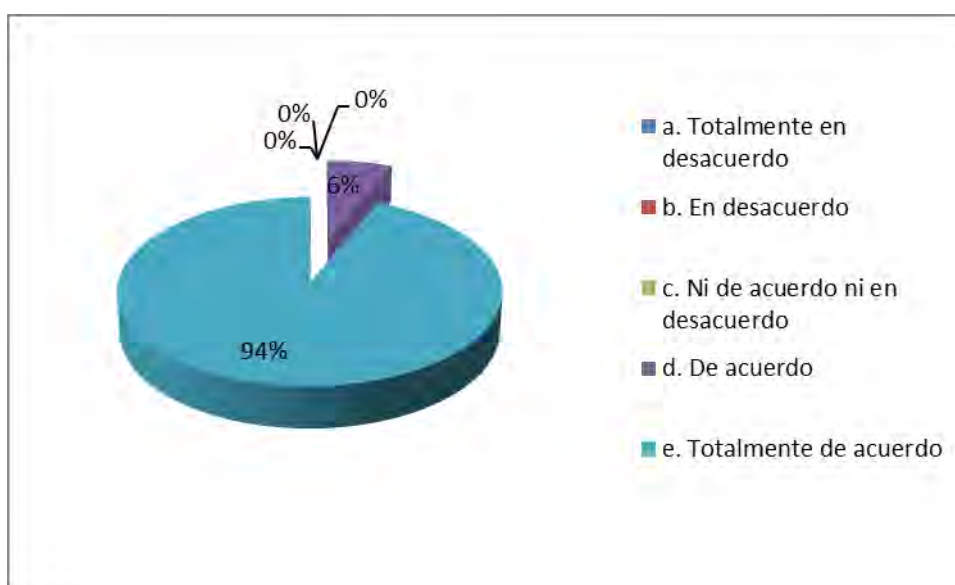


Figura 11: Resultado de la décima primera pregunta del cuestionario de la variable independiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que las herramientas de integración continua le facilitan el trabajo de monitoreo y control del desarrollo de software.

12. Las herramientas de integración continua han permitido la adaptación de la integración continua rápidamente en el equipo de desarrollo.		
Categorías	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.0%
b. En desacuerdo	0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0%
d. De acuerdo	3	18.75%
e. Totalmente de acuerdo	13	81.25%
<i>Total</i>	16	100%

Tabla 14: Resultado de la décima segunda pregunta del cuestionario de la variable independiente

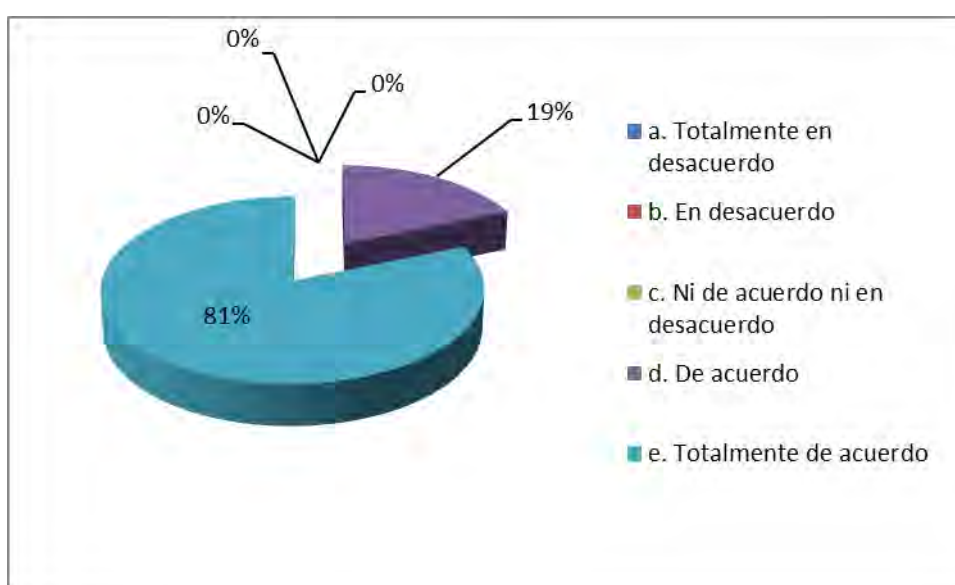


Figura 12: Resultado de la décima segunda pregunta del cuestionario de la variable independiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que las herramientas de integración continua han permitido la adaptación de la integración continua rápidamente en el equipo de desarrollo.

13. Las pruebas continuas le ha permitido mejorar la calidad del software.

Categorías	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.0%
b. En desacuerdo	0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0%
d. De acuerdo	2	12.5%
e. Totalmente de acuerdo	14	87.5%
<i>Total</i>	16	100%

Tabla 15: Resultado de la décima tercera pregunta del cuestionario de la variable independiente

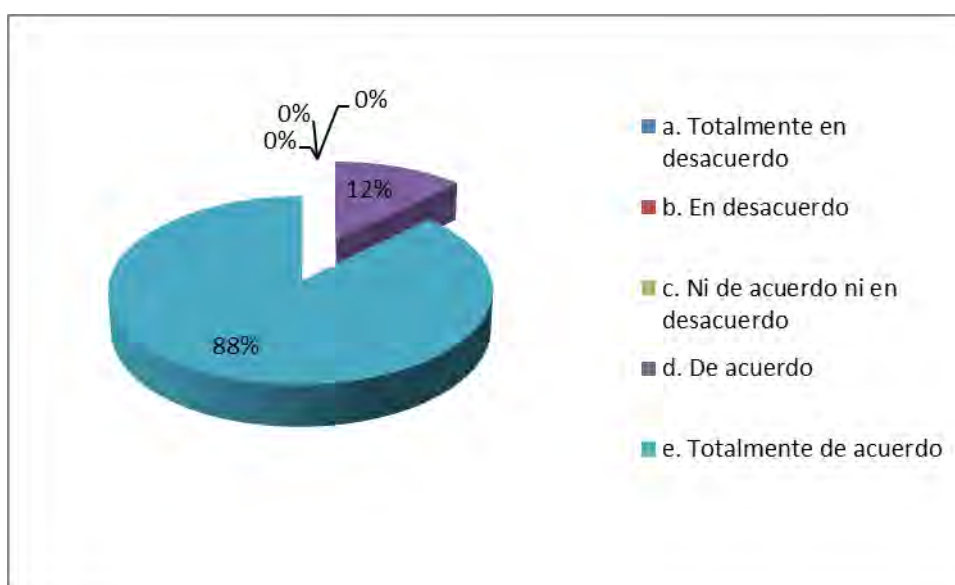


Figura 13: Resultado de la décima tercera pregunta del cuestionario de la variable independiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que las pruebas continuas le permiten mejorar la calidad del software.

14. La ejecución constante de las pruebas hace posible mejorar el desarrollo del software.		
Categorías	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.0%
b. En desacuerdo	0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0%
d. De acuerdo	3	18.75%
e. Totalmente de acuerdo	13	81.25%
<i>Total</i>	16	100%

Tabla 16: Resultado de la décima cuarta pregunta del cuestionario de la variable independiente

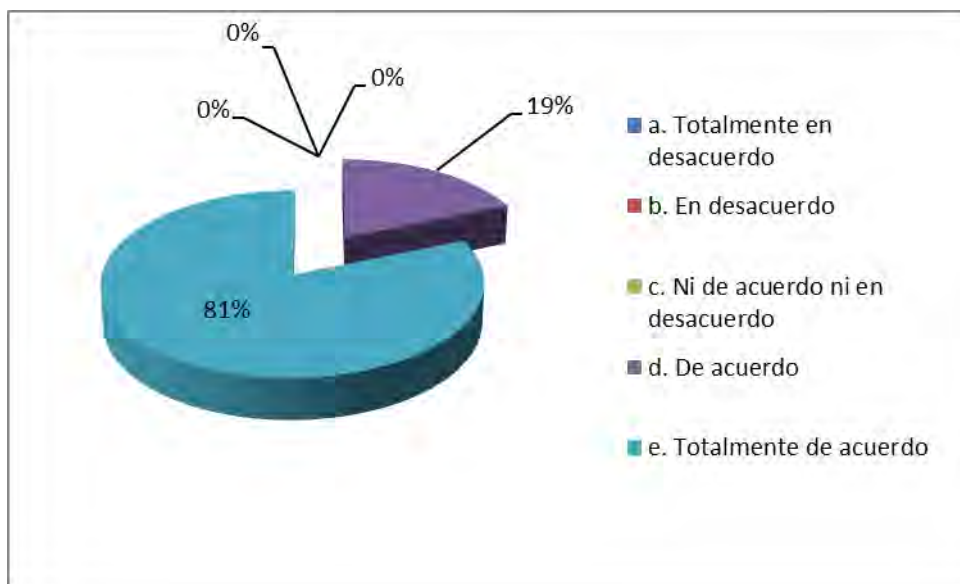


Figura 14: Resultado de la décima cuarta pregunta del cuestionario de la variable independiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que las pruebas continuas hacen posible mejorar el desarrollo de software.

15. Los usuarios se encuentran satisfechos de los softwares entregados a través de la integración continua.

Categorías	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.0%
b. En desacuerdo	0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0%
d. De acuerdo	4	25.0%
e. Totalmente de acuerdo	12	75.0%
<i>Total</i>	16	100%

Tabla 17: Resultado de la décima quinta pregunta del cuestionario de la variable independiente

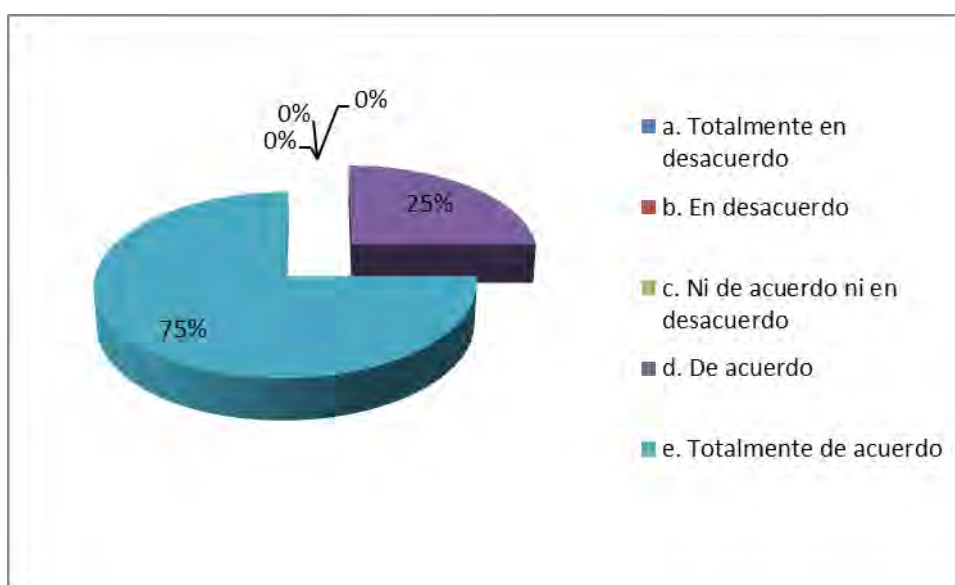


Figura 15: Resultado de la décima quinta pregunta del cuestionario de la variable independiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que los usuarios se encuentran satisfechos de los softwares desarrollados a través de la integración continua.

16. La frecuencia continua de la inspección le agrega valor al desarrollo de software.

Categorías	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.0%
b. En desacuerdo	0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0%
d. De acuerdo	2	12.5%
e. Totalmente de acuerdo	14	87.5%
<i>Total</i>	16	100%

Tabla 18: Resultado de la décima sexta pregunta del cuestionario de la variable independiente

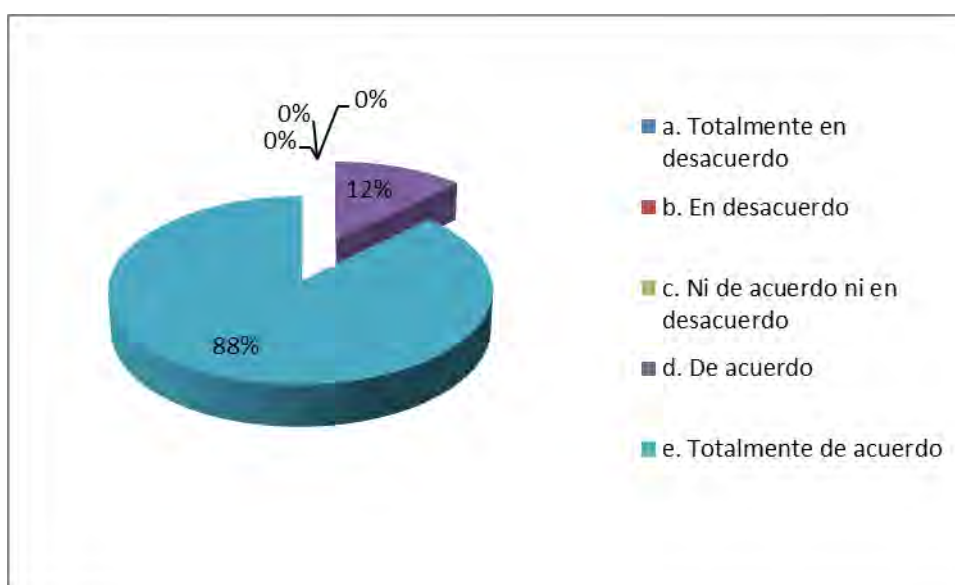


Figura 16: Resultado de la décima sexta pregunta del cuestionario de la variable independiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que la frecuencia de la inspección le agrega valor al desarrollo de software.

17. La frecuencia continua de la inspección le ha permitido mejorar la calidad de la programación del desarrollo de software.

Categorías	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.0%
b. En desacuerdo	0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0%
d. De acuerdo	3	18.75%
e. Totalmente de acuerdo	13	81.25%
<i>Total</i>	16	100%

Tabla 19: Resultado de la décima séptima pregunta del cuestionario de la variable independiente

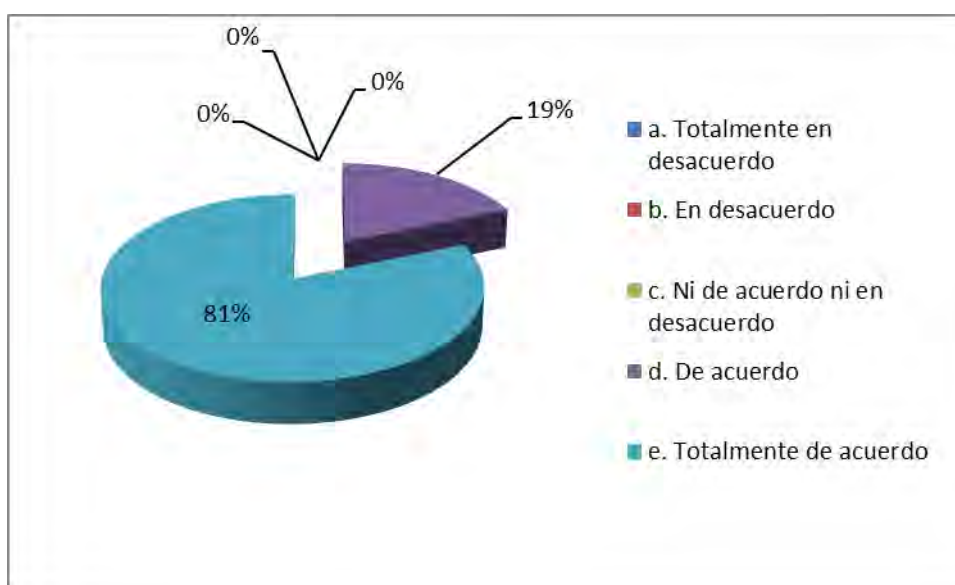


Figura 17: Resultado de la décima séptima pregunta del cuestionario de la variable independiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que la frecuencia de la inspección le ha permitido mejorar la calidad de la programación del desarrollo de software.

18. La auditoría de código fuente le permite mejorar la calidad del software.

Categorías	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.0%
b. En desacuerdo	0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0%
d. De acuerdo	9	56.25%
e. Totalmente de acuerdo	7	43.75%
<i>Total</i>	16	100%

Tabla 20: Resultado de la décima octava pregunta del cuestionario de la variable independiente

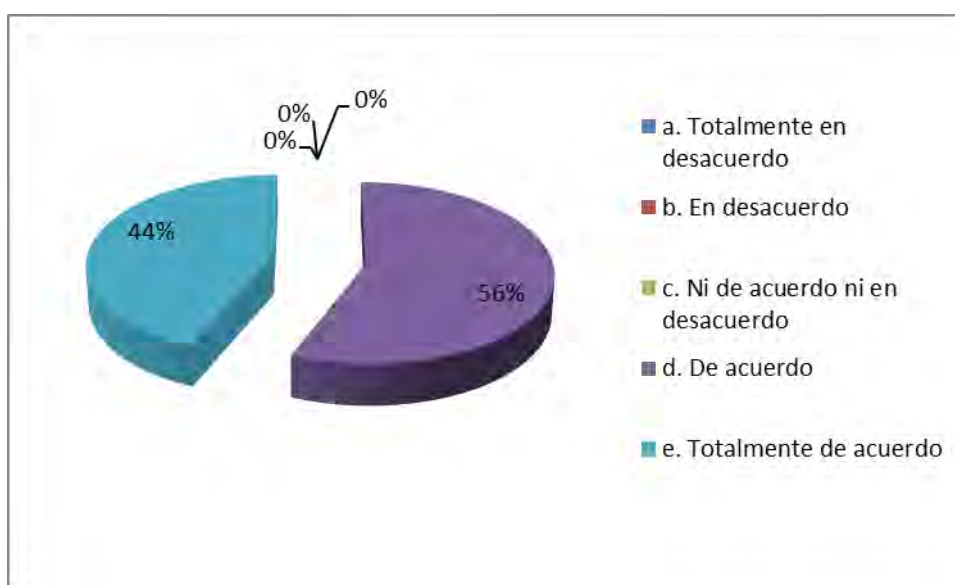


Figura 18: Resultado de la décima octava pregunta del cuestionario de la variable independiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que la auditoría de código fuente le permite mejorar la calidad del software.

19. La codificación estándar le permite facilitar un conocimiento común de la programación a los diversos equipos de desarrollo.		
Categorías	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.0%
b. En desacuerdo	0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0%
d. De acuerdo	7	43.75%
e. Totalmente de acuerdo	9	56.25%
<i>Total</i>	16	100%

Tabla 21: Resultado de la décima novena pregunta del cuestionario de la variable independiente

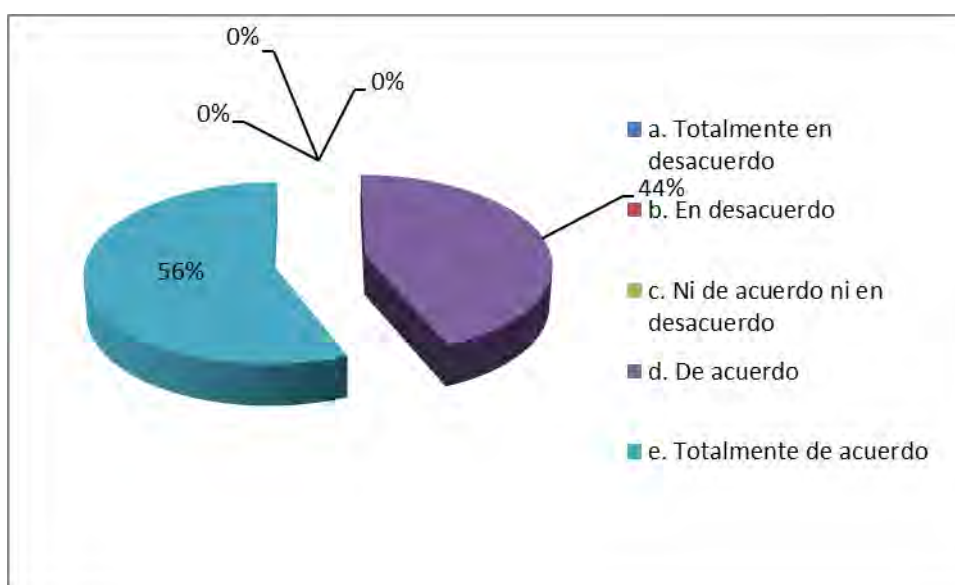


Figura 19: Resultado de la décima novena pregunta del cuestionario de la variable independiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que la codificación estándar le permite facilitar un conocimiento común de la programación a los diversos equipos de desarrollo.

20. El porcentaje de cobertura de código fuente le permite mejorar la calidad del software.		
Categorías	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.0%
b. En desacuerdo	0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0%
d. De acuerdo	6	37.5%
e. Totalmente de acuerdo	10	62.5%
<i>Total</i>	16	100%

Tabla 22: Resultado de la vigésima pregunta del cuestionario de la variable independiente

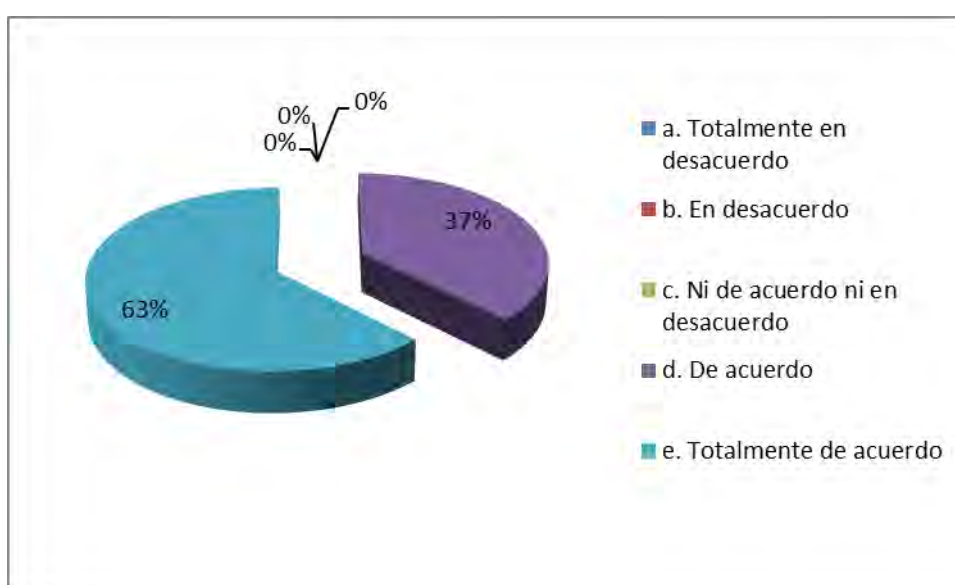


Figura 20: Resultado de la vigésima pregunta del cuestionario de la variable independiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que el porcentaje de cobertura de código fuente le permite mejorar la calidad del software.

21. La cobertura de código fuente ha permitido mejorar el trabajo colaborativo en el equipo de desarrollo.

Categorías	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.0%
b. En desacuerdo	0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0%
d. De acuerdo	4	25.0%
e. Totalmente de acuerdo	12	75.0%
<i>Total</i>	16	100%

Tabla 23: Resultado de la vigésima primera pregunta del cuestionario de la variable independiente

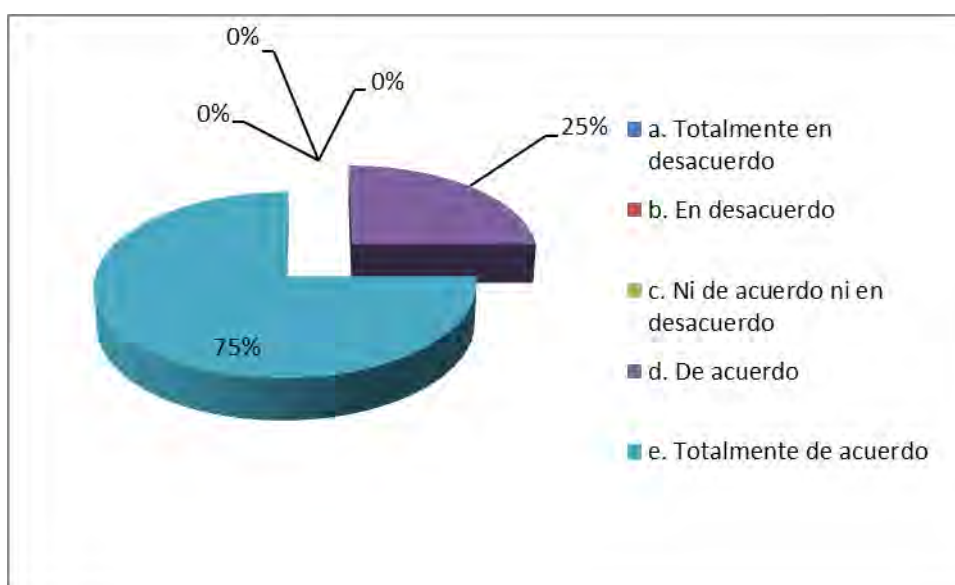


Figura 21: Resultado de la vigésima primera pregunta del cuestionario de la variable independiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que la cobertura de código fuente ha permitido mejorar el trabajo colaborativo en el equipo de desarrollo.

22. La información de duplicación de código fuente le permite mejorar la calidad del software.

Categorías	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.0%
b. En desacuerdo	0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0%
d. De acuerdo	7	43.75%
e. Totalmente de acuerdo	9	56.25%
<i>Total</i>	16	100%

Tabla 24: Resultado de la vigésima segunda pregunta del cuestionario de la variable independiente

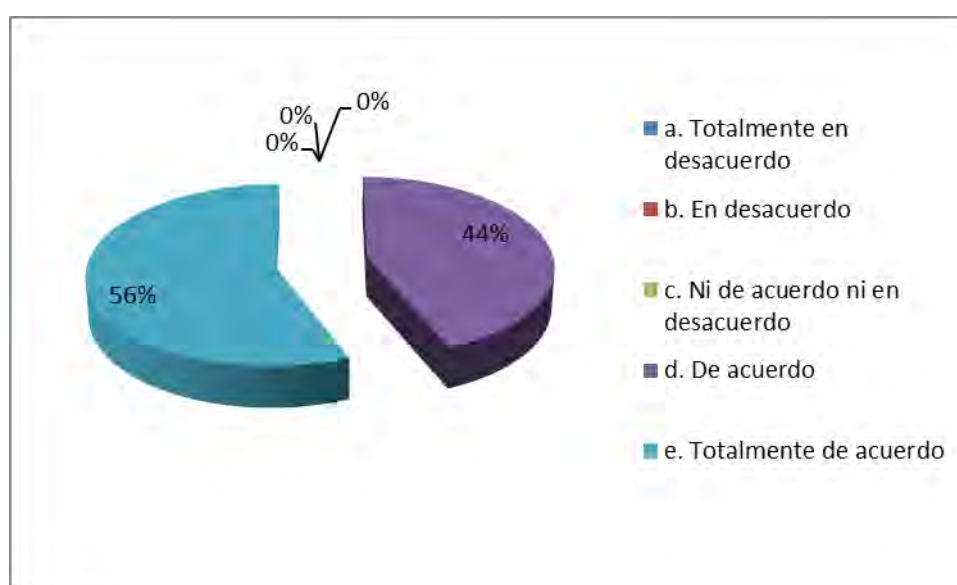


Figura 22: Resultado de la vigésima segunda pregunta del cuestionario de la variable independiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que la información de duplicación de código fuente le permite mejorar la calidad del software.

23. La duplicidad de código fuente incrementa los costos del proyecto de desarrollo de software, por lo que la integración continua lo ayuda a manejar los costos		
Categorías	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.0%
b. En desacuerdo	0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0%
d. De acuerdo	10	62.5%
e. Totalmente de acuerdo	6	37.5%
<i>Total</i>	16	100%

Tabla 25: Resultado de la vigésima tercera pregunta del cuestionario de la variable independiente

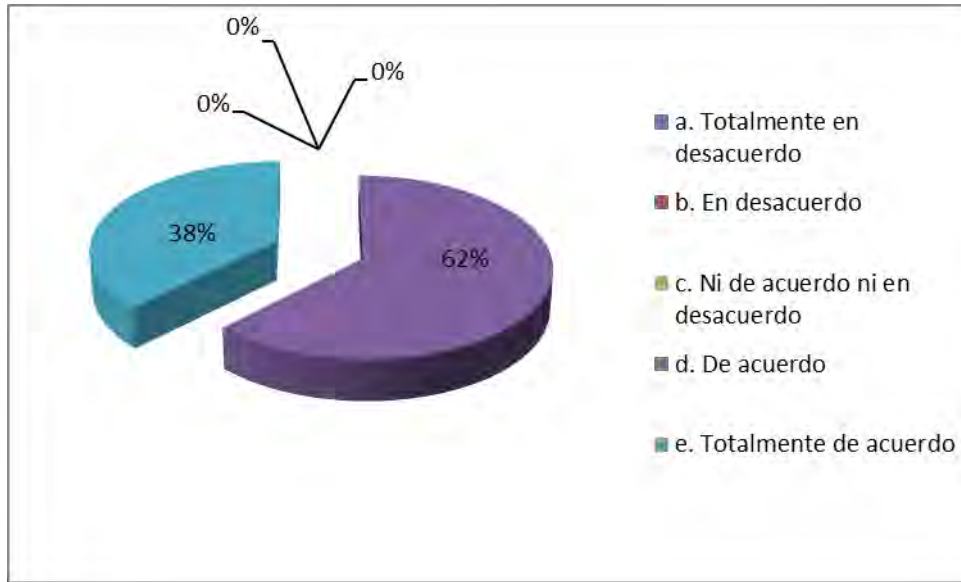


Figura 23: Resultado de la vigésima tercera pregunta del cuestionario de la variable independiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que la integración continua le ayuda a mitigar el incremento de costos por la duplicidad de código fuente.

24. El control de la complejidad del código fuente le ha permitido mejorar la mantenibilidad del software.

Categorías	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.0%
b. En desacuerdo	0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0%
d. De acuerdo	5	31.25%
e. Totalmente de acuerdo	11	68.75%
<i>Total</i>	16	100%

Tabla 26: Resultado de la vigésima cuarta pregunta del cuestionario de la variable independiente

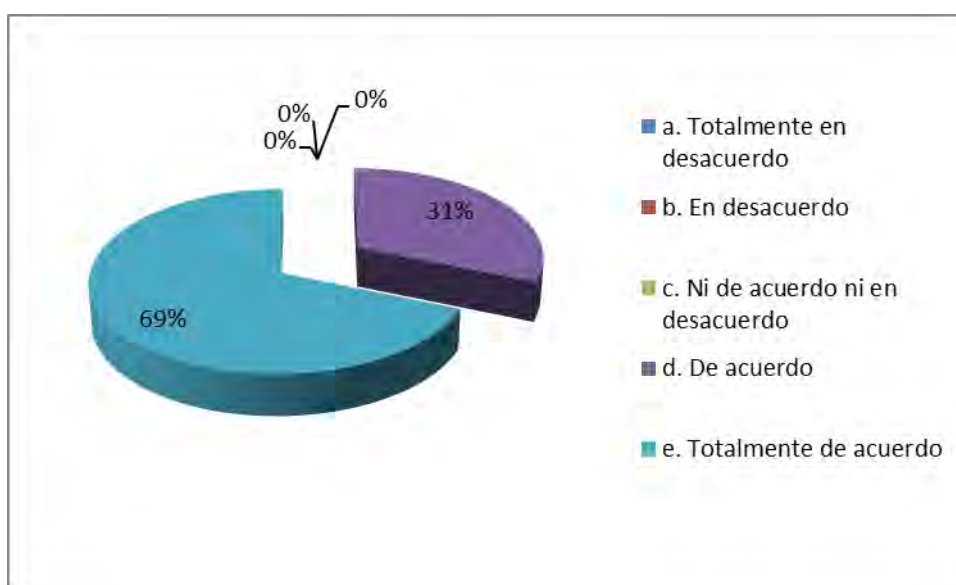


Figura 24: Resultado de la vigésima cuarta pregunta del cuestionario de la variable independiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que el control de la complejidad del código fuente le ha permitido mejorar la mantenibilidad del software.

25. El control de la complejidad de código fuente le ha permitido reducir los riesgos del desarrollo de software.

Categorías	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.0%
b. En desacuerdo	0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0%
d. De acuerdo	3	18.75%
e. Totalmente de acuerdo	13	81.25%
<i>Total</i>	16	100%

Tabla 27: Resultado de la vigésima quinta pregunta del cuestionario de la variable independiente

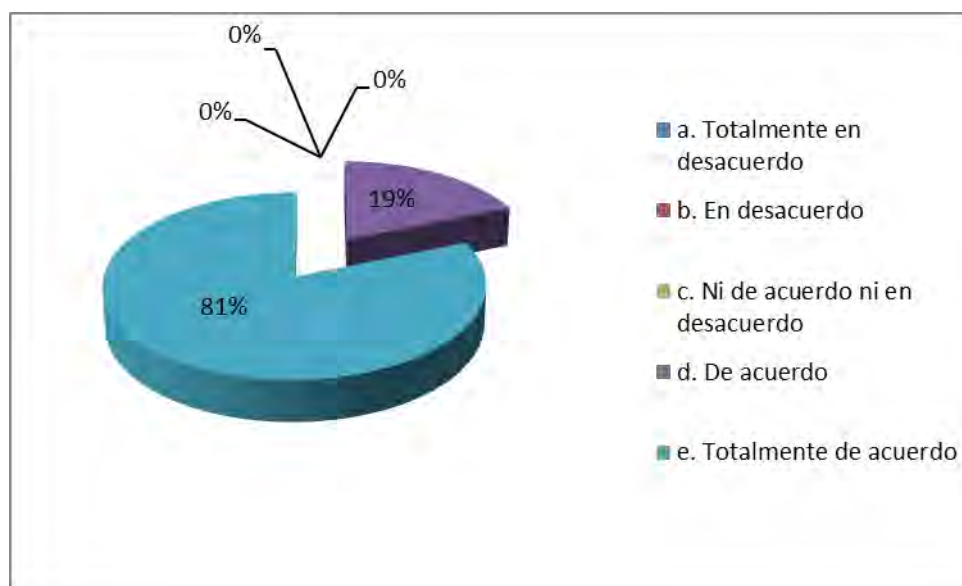


Figura 25: Resultado de la vigésima quinta pregunta del cuestionario de la variable independiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que el control de la complejidad del código fuente le ha permitido reducir los riesgos del desarrollo de software.

26. Los mecanismos de retroalimentación le han permitido controlar el proceso de desarrollo de software.

Categorías	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.0%
b. En desacuerdo	0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0%
d. De acuerdo	2	12.5%
e. Totalmente de acuerdo	14	87.5%
<i>Total</i>	16	100%

Tabla 28: Resultado de la vigésima sexta pregunta del cuestionario de la variable independiente

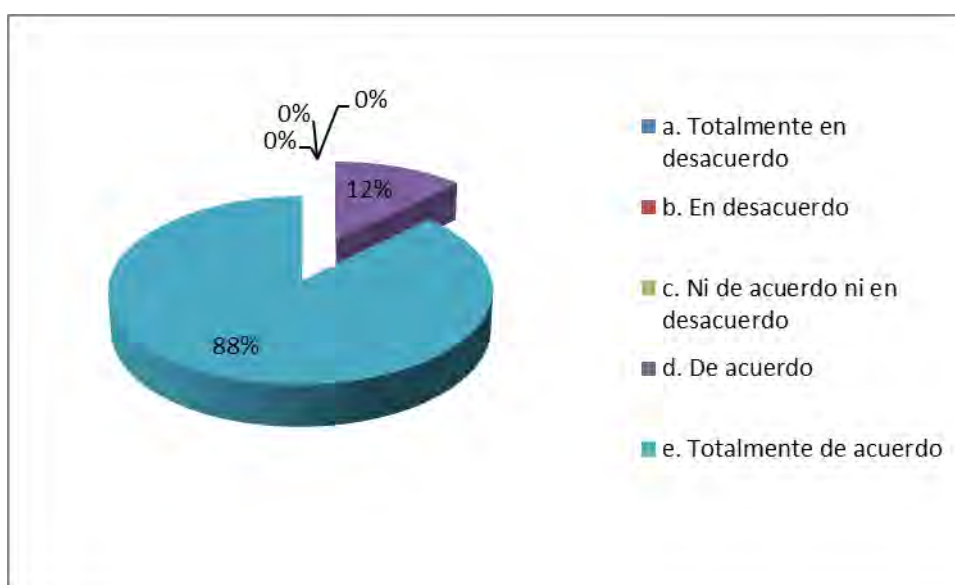


Figura 26: Resultado de la vigésima sexta pregunta del cuestionario de la variable independiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que los mecanismos de retroalimentación le han permitido controlar el proceso de desarrollo de software.

27. La frecuencia de la retroalimentación continua le agrega valor al desarrollo de software.		
Categorías	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.0%
b. En desacuerdo	0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0%
d. De acuerdo	3	18.75%
e. Totalmente de acuerdo	13	81.25%
<i>Total</i>	16	100%

Tabla 29: Resultado de la vigésima séptima pregunta del cuestionario de la variable independiente

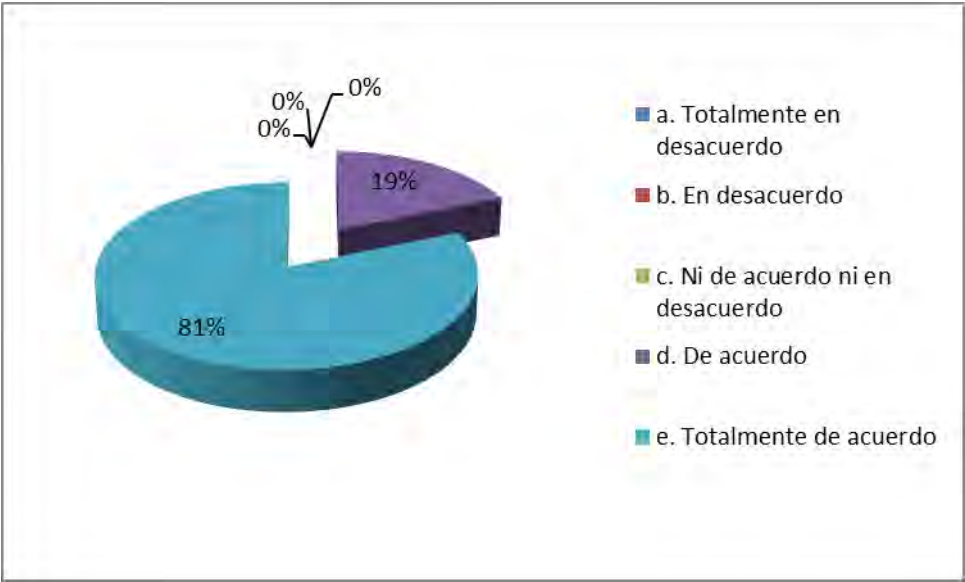


Figura 27: Resultado de la vigésima séptima pregunta del cuestionario de la variable independiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que la frecuencia de la retroalimentación continua le agrega valor al desarrollo de software.

28. La retroalimentación continua ha generado una cultura en el equipo de desarrollo que le permite mejorar el proceso de desarrollo de software y la calidad del software.

Categorías	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.0%
b. En desacuerdo	0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0%
d. De acuerdo	8	50.0%
e. Totalmente de acuerdo	8	50.0%
<i>Total</i>	16	100%

Tabla 30: Resultado de la vigésima octava pregunta del cuestionario de la variable independiente

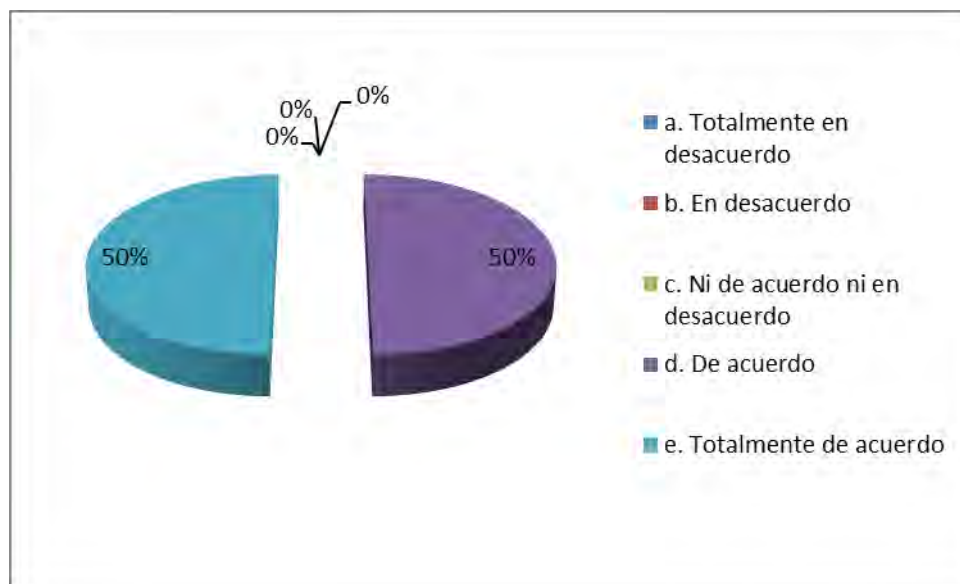


Figura 28: Resultado de la vigésima octava pregunta del cuestionario de la variable independiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que la retroalimentación continua ha generado una cultura de trabajo en equipo de desarrollo que permite mejorar el proceso de desarrollo de software y la calidad del software.

Análisis de los Indicadores de la Variable Independiente
Integración Continua
Dimensión: Construcción Constante

Indicador: Repositorio del código fuente y documentos					
Categorías	Preguntas			$\bar{X}$	%
	No 1	No 2	No 3		
a. Totalmente en desacuerdo	0	0	0	0.0	0.0%
b. En desacuerdo	0	0	0	0.0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0	0	0.0	0.0%
d. De acuerdo	4	1	2	2.3	14.6%
e. Totalmente de acuerdo	12	15	14	13.7	85.4%
<i>Total</i>	<i>16</i>	<i>16</i>	<i>16</i>	<i>16</i>	<i>100.0%</i>

Tabla 31: Resultado del indicador Repositorio del código fuente y documentos

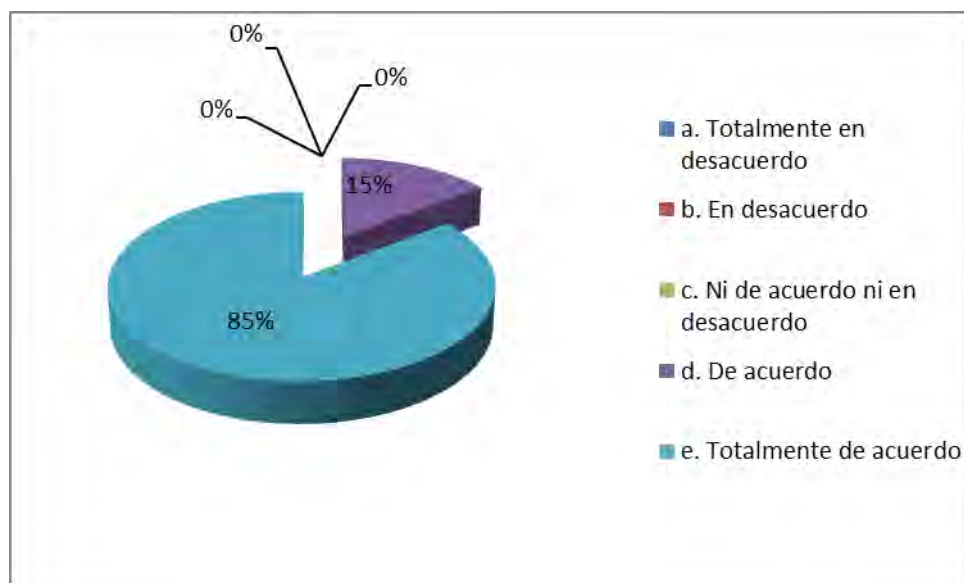


Figura 29: Resultado del indicador Repositorio del código fuente y documentos

Interpretación:

El personal de Sistemas refiere que está de acuerdo con los beneficios del repositorio del código fuente y documentos.

Indicador: Documentación del código fuente				
Categorías	Preguntas		$\bar{X}$	%
	No 4	No 5		
a. Totalmente en desacuerdo	0	0	0.0	0.0%
b. En desacuerdo	0	0	0.0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0	0.0	0.0%
d. De acuerdo	6	4	5.0	31.3%
e. Totalmente de acuerdo	10	12	11.0	68.8%
<i>Total</i>	16	16	16	100.0%

Tabla 32: Resultado del indicador Documentación del código fuente

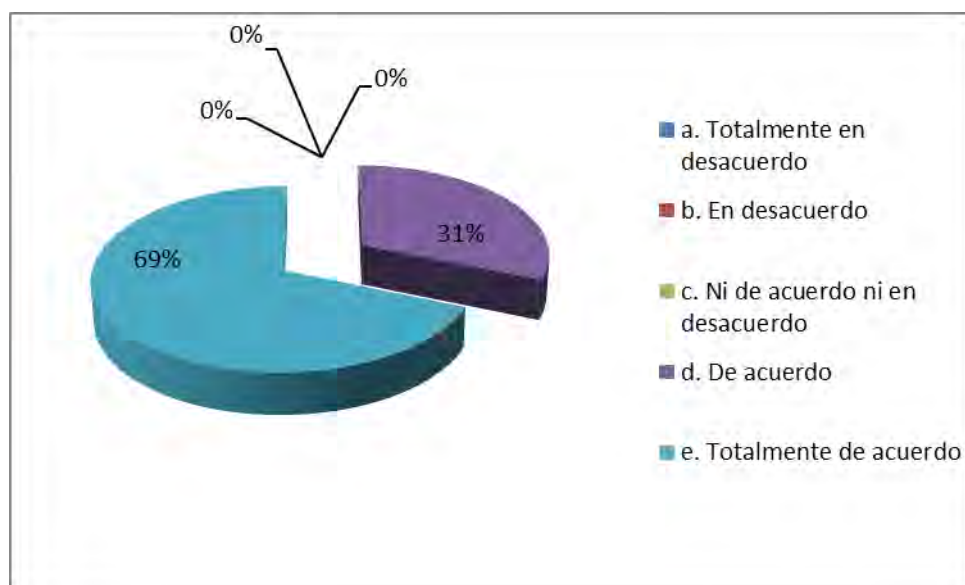


Figura 30: Resultado del indicador Documentación del código fuente

Interpretación:

El personal de Sistemas refiere que está de acuerdo con los beneficios de la documentación constante del código fuente.

Indicador: Ambientes del despliegue				
Categorías	Preguntas		$\bar{X}$	%
	No 6	No 7		
a. Totalmente en desacuerdo	0	0	0.0	0.0%
b. En desacuerdo	0	0	0.0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0	0.0	0.0%
d. De acuerdo	8	6	7.0	43.8%
e. Totalmente de acuerdo	8	10	9.0	56.3%
<i>Total</i>	16	16	16	100.0%

Tabla 33: Resultado del indicador Ambientes de despliegue

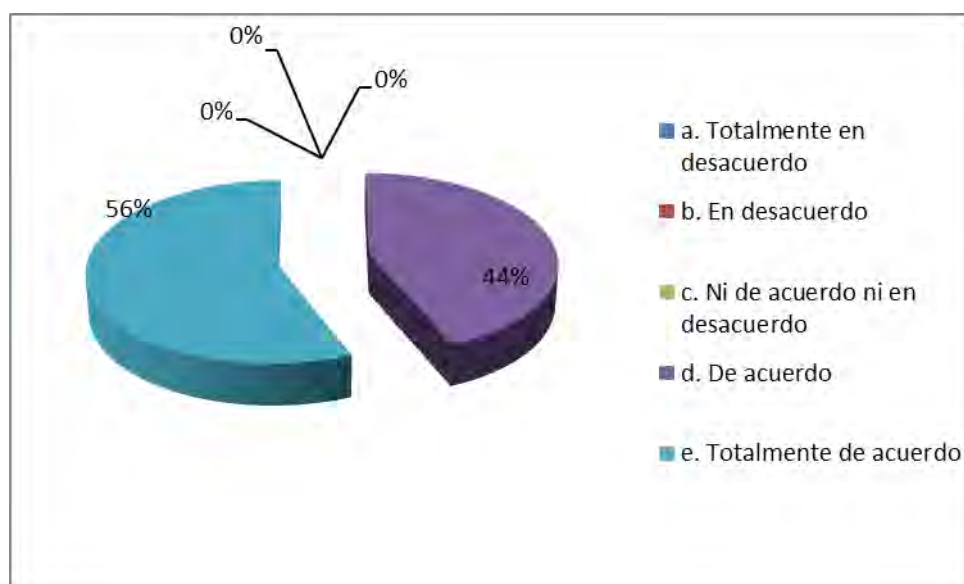


Figura 31: Resultado del indicador Ambientes de despliegue

Interpretación:

El personal de Sistemas refiere que está de acuerdo con los beneficios del manejo de los ambientes de despliegue.

Indicador: Frecuencia de Integración Continua					
Categorías	Preguntas			$\bar{X}$	%
	No 8	No 9	No 10		
a. Totalmente en desacuerdo	0	0	0	0.0	0.0%
b. En desacuerdo	0	0	0	0.0	0.0%
c. Ni de acuerdo ni en desacuerdo	2	0	0	0.7	4.2%
d. De acuerdo	7	6	2	5.0	31.3%
e. Totalmente de acuerdo	7	10	14	10.3	64.6%
<i>Total</i>	16	16	16	16	100.0%

Tabla 34: Resultado del indicador Frecuencia de Integración Continua

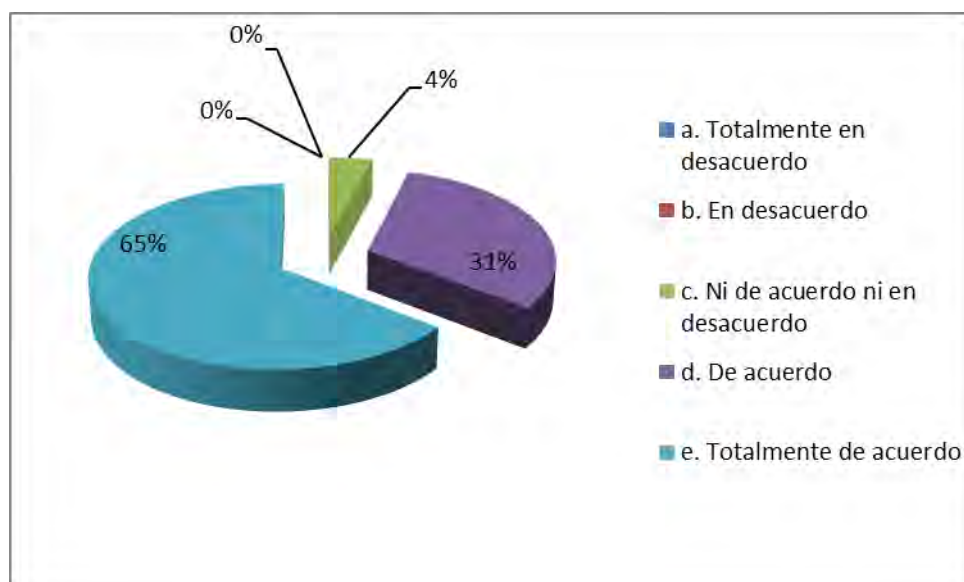


Figura 32: Resultado del indicador Frecuencia de Integración Continua

Interpretación:

El personal de Sistemas refiere que está de acuerdo con los beneficios de la frecuencia continua de la integración del código fuente.

Indicador: Herramientas de Integración Continua				
Categorías	Preguntas		$\bar{X}$	%
	No 11	No 12		
a. Totalmente en desacuerdo	0	0	0.0	0.0%
b. En desacuerdo	0	0	0.0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0	0.0	0.0%
d. De acuerdo	1	3	2.0	12.5%
e. Totalmente de acuerdo	15	13	14.0	87.5%
<i>Total</i>	16	16	16	100.0%

Tabla 35: Resultado del indicador Herramientas de Integración Continua

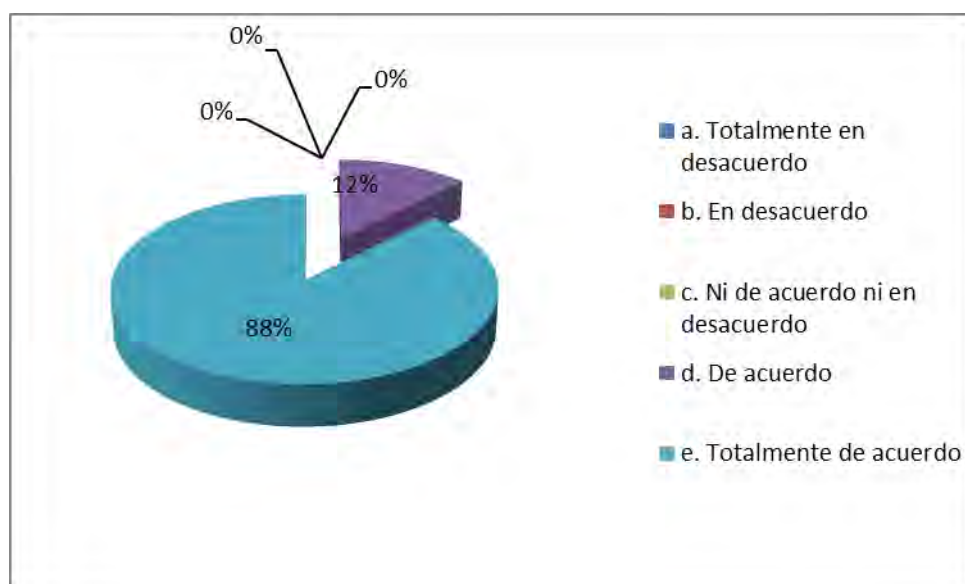


Figura 33: Resultado del indicador Herramientas de Integración Continua

Interpretación:

El personal de Sistemas refiere que está de acuerdo con los beneficios de las herramientas de la integración continua.

Dimensión: Construcción Constante							
Categorías	Indicadores					$\bar{X}$	%
	No 1	No 2	No 3	No 4	No 5		
a. Totalmente en desacuerdo	0.0	0.0	0.0	0.0	0.0	0.0	0.0%
b. En desacuerdo	0.0	0.0	0.0	0.0	0.0	0.0	0.0%
c. Ni de acuerdo ni en desacuerdo	0.0	0.0	0.0	0.7	0.0	0.1	0.9%
d. De acuerdo	2.3	5.0	7.0	5.0	2.0	4.3	26.6%
e. Totalmente de acuerdo	13.7	11.0	9.0	10.3	14.0	11.6	72.5%
<i>Total</i>	16	16	16	16	16	16	100.0%

Tabla 36: Resultado de la Dimensión Construcción Constante

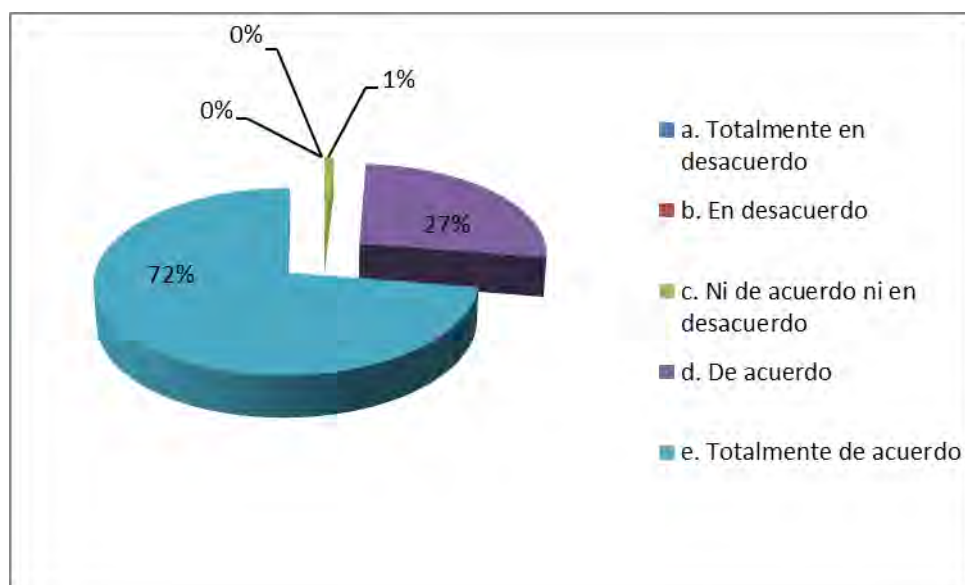


Figura 34: Resultado de la Dimensión Construcción Constante

Interpretación:

El personal de Sistemas refiere que está de acuerdo con la construcción constante de la Integración Continua.

Análisis de los Indicadores de la Variable Independiente
Integración Continua
Dimensión: Pruebas Constante

Indicador: Calidad de Software				
Categorías	Preguntas		$\bar{X}$	%
	No 13	No 14		
a. Totalmente en desacuerdo	0	0	0.0	0.0%
b. En desacuerdo	0	0	0.0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0	0.0	0.0%
d. De acuerdo	2	3	2.5	15.6%
e. Totalmente de acuerdo	14	13	13.5	84.4%
<i>Total</i>	16	16	16	100.0%

Tabla 37: Resultado del indicador Calidad de Software

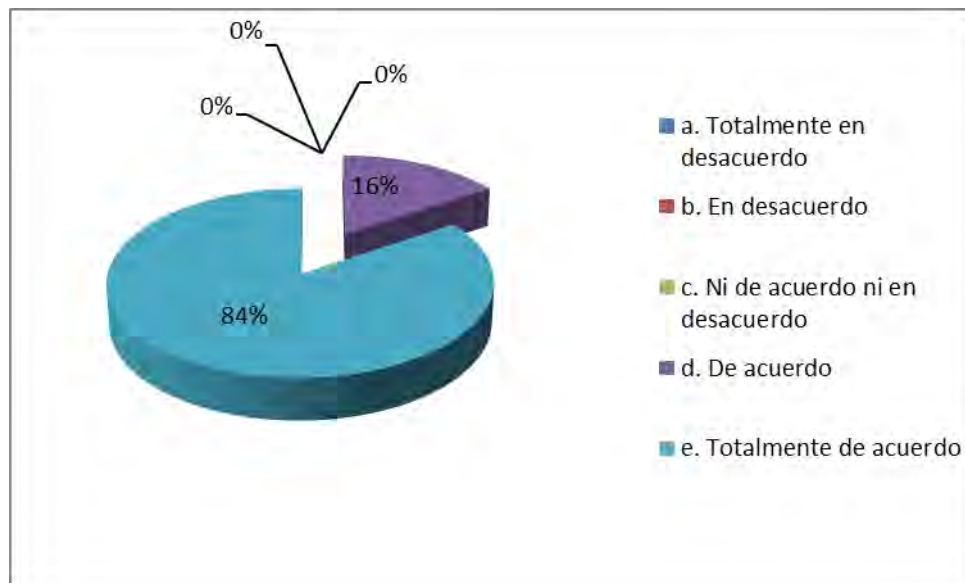


Figura 35: Resultado del indicador Calidad de Software

Interpretación:

El personal de Sistemas refiere que está de acuerdo con los beneficios respecto a la calidad del software.

Indicador: Satisfacción del usuario			
Categorías	Preguntas	$\bar{X}$	%
	No 15		
a. Totalmente en desacuerdo	0	0.0	0.0%
b. En desacuerdo	0	0.0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0	0.0%
d. De acuerdo	4	4.0	25.0%
e. Totalmente de acuerdo	12	12.0	75.0%
<i>Total</i>	16	16	100.0%

Tabla 38: Resultado del indicador Satisfacción del Usuario

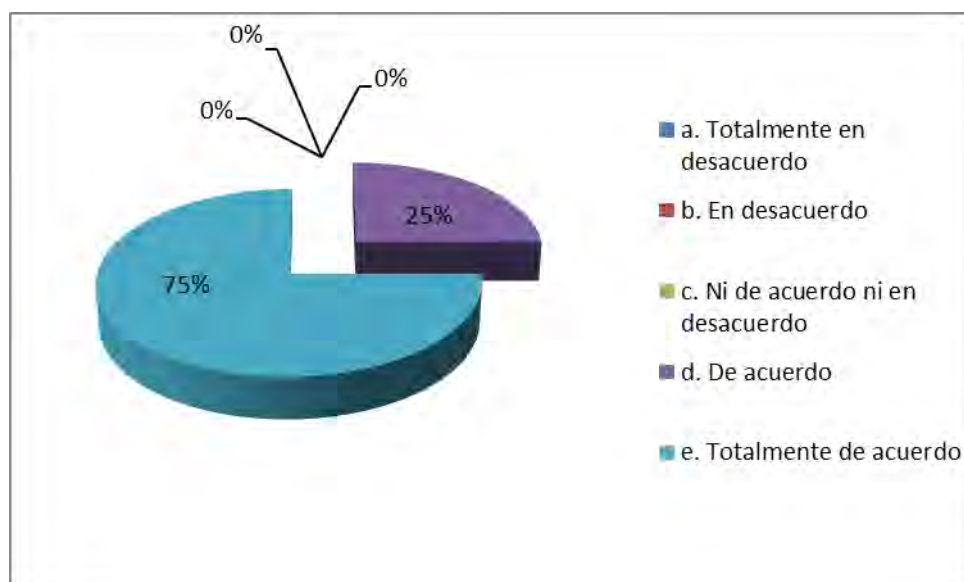


Figura 36: Resultado del indicador Satisfacción del Usuario

Interpretación:

El personal de Sistemas refiere que está de acuerdo al respecto a la satisfacción de los usuarios.

Dimensión: Pruebas Constante				
Categorías	Indicadores		$\bar{X}$	%
	No 1	No 2		
a. Totalmente en desacuerdo	0	0	0.0	0.0%
b. En desacuerdo	0	0	0.0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0	0.0	0.0%
d. De acuerdo	2.5	4	3.3	20.3%
e. Totalmente de acuerdo	13.5	12	12.8	79.7%
<i>Total</i>	16	16	16	100.0%

Tabla 39: Resultado de la Dimensión Pruebas Constante

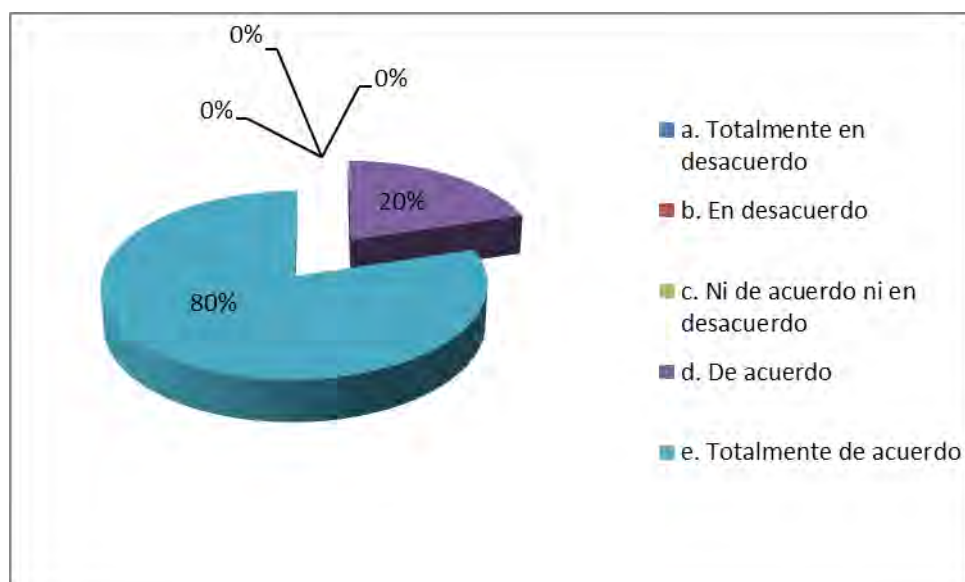


Figura 37: Resultado de la Dimensión Pruebas Constante

Interpretación:

El personal de Sistemas refiere que está de acuerdo respecto a las pruebas constantes.

Análisis de los Indicadores de la Variable Independiente
Integración Continua
Dimensión: Inspección Constante

Indicador: Frecuencia de la Inspección				
Categorías	Preguntas		$\bar{X}$	%
	No 16	No 17		
a. Totalmente en desacuerdo	0	0	0.0	0.0%
b. En desacuerdo	0	0	0.0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0	0.0	0.0%
d. De acuerdo	2	3	2.5	15.6%
e. Totalmente de acuerdo	14	13	13.5	84.4%
<i>Total</i>	<i>16</i>	<i>16</i>	<i>16</i>	<i>100.0%</i>

Tabla 40: Resultado del indicador Frecuencia de Inspección

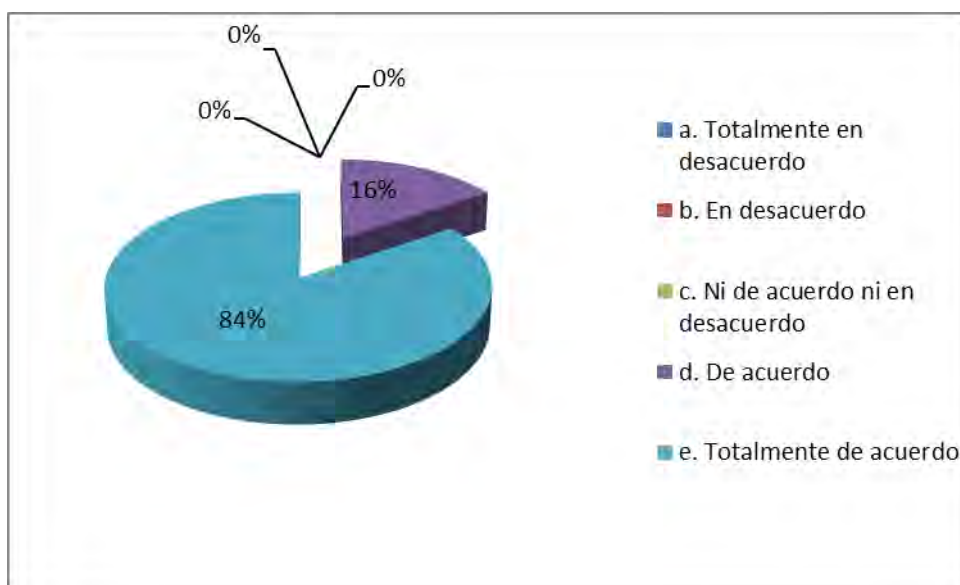


Figura 38: Resultado del indicador Frecuencia de Inspección

Interpretación:

El personal de Sistemas refiere que está de acuerdo con los beneficios de la frecuencia de la inspección.

Indicador: Auditoria de código fuente				
Categorías	Preguntas		$\bar{X}$	%
	No 18	No 19		
a. Totalmente en desacuerdo	0	0	0.0	0.0%
b. En desacuerdo	0	0	0.0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0	0.0	0.0%
d. De acuerdo	9	7	8.0	50.0%
e. Totalmente de acuerdo	7	9	8.0	50.0%
<i>Total</i>	<i>16</i>	<i>16</i>	<i>16</i>	<i>100.0%</i>

Tabla 41: Resultado del indicador Auditoria de código fuente

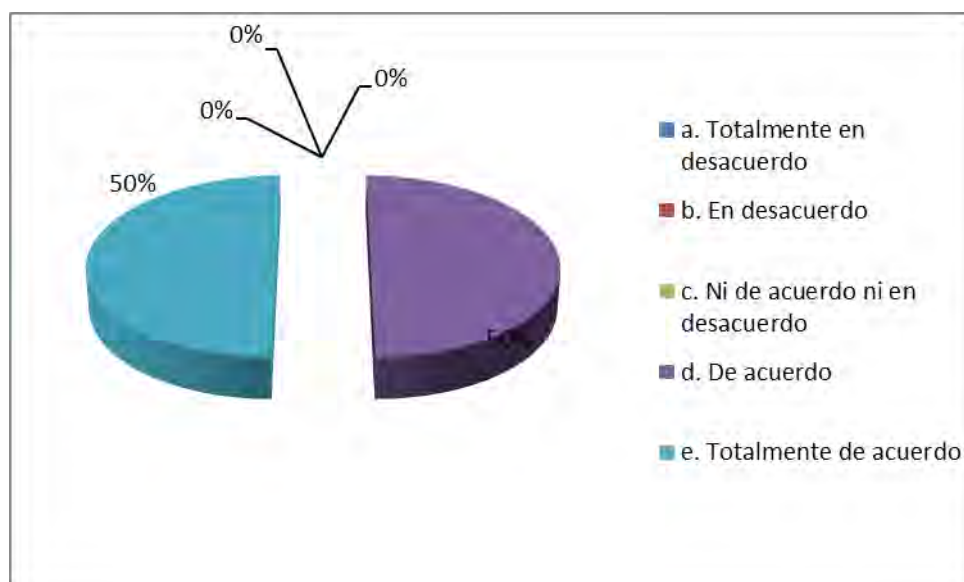


Figura 39: Resultado del indicador Auditoria de código fuente

Interpretación:

El personal de Sistemas refiere que está de acuerdo con los beneficios de la auditoria de código fuente.

Indicador: Cobertura de código fuente				
Categorías	Preguntas		$\bar{X}$	%
	No 20	No 21		
a. Totalmente en desacuerdo	0	0	0.0	0.0%
b. En desacuerdo	0	0	0.0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0	0.0	0.0%
d. De acuerdo	6	4	5.0	31.3%
e. Totalmente de acuerdo	10	12	11.0	68.8%
<i>Total</i>	16	16	16	100.0%

Tabla 42: Resultado del indicador Cobertura de código fuente

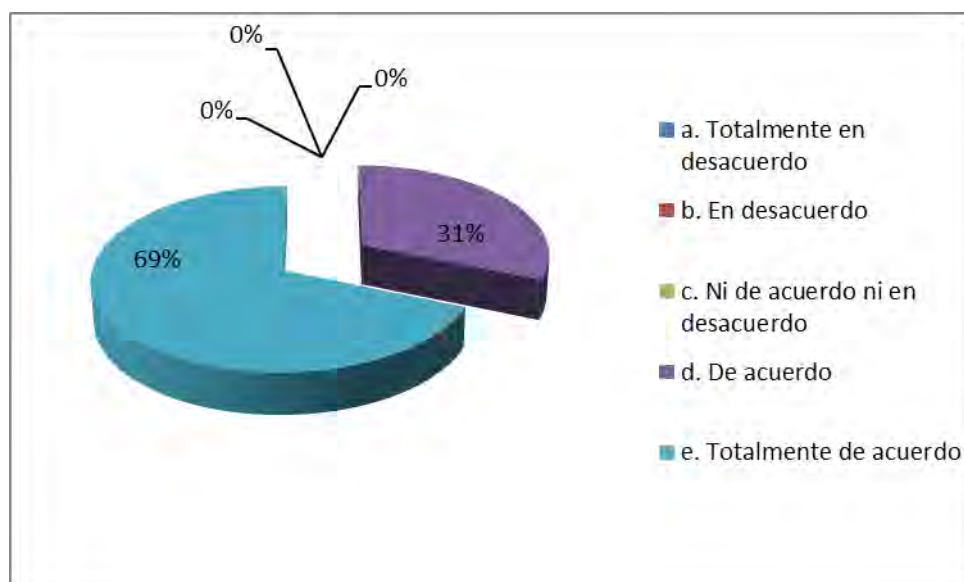


Figura 40: Resultado del indicador Cobertura de código fuente

Interpretación:

El personal de Sistemas refiere que está de acuerdo con los beneficios de la cobertura de código fuente.

Indicador: Duplicación de código fuente				
Categorías	Preguntas		$\bar{X}$	%
	No 22	No 23		
a. Totalmente en desacuerdo	0	0	0.0	0.0%
b. En desacuerdo	0	0	0.0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0	0.0	0.0%
d. De acuerdo	7	10	8.5	53.1%
e. Totalmente de acuerdo	9	6	7.5	46.9%
<i>Total</i>	16	16	16	100.0%

Tabla 43: Resultado del indicador Duplicación de código fuente

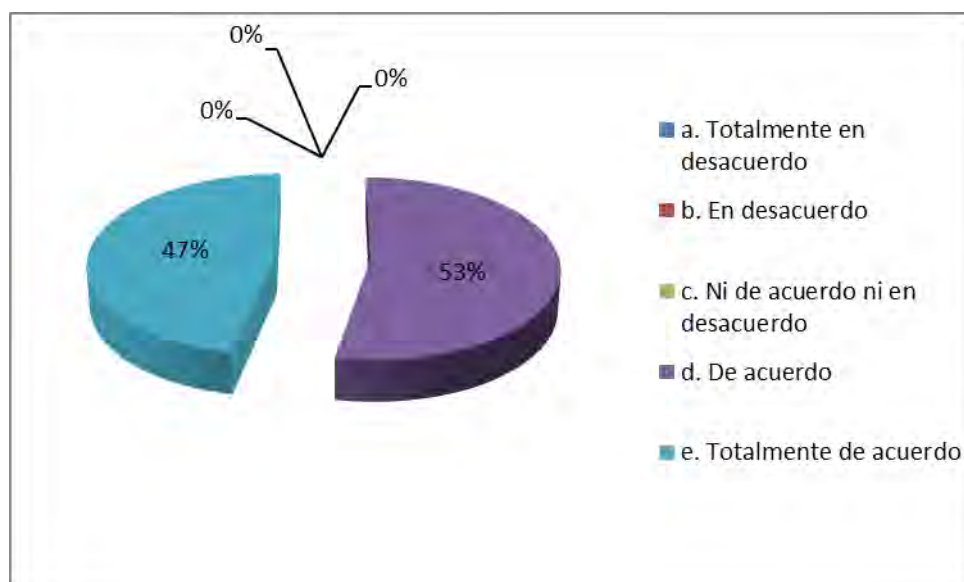


Figura 41: Resultado del indicador Duplicación de código fuente

Interpretación:

El personal de Sistemas refiere que está de acuerdo con los beneficios de identificar la duplicación del código fuente.

Indicador: Complejidad de código fuente				
Categorías	Preguntas		$\bar{X}$	%
	No 24	No 25		
a. Totalmente en desacuerdo	0	0	0.0	0.0%
b. En desacuerdo	0	0	0.0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0	0.0	0.0%
d. De acuerdo	5	3	4.0	25.0%
e. Totalmente de acuerdo	11	13	12.0	75.0%
<i>Total</i>	16	16	16	100.0%

Tabla 44: Resultado del indicador Complejidad de código fuente

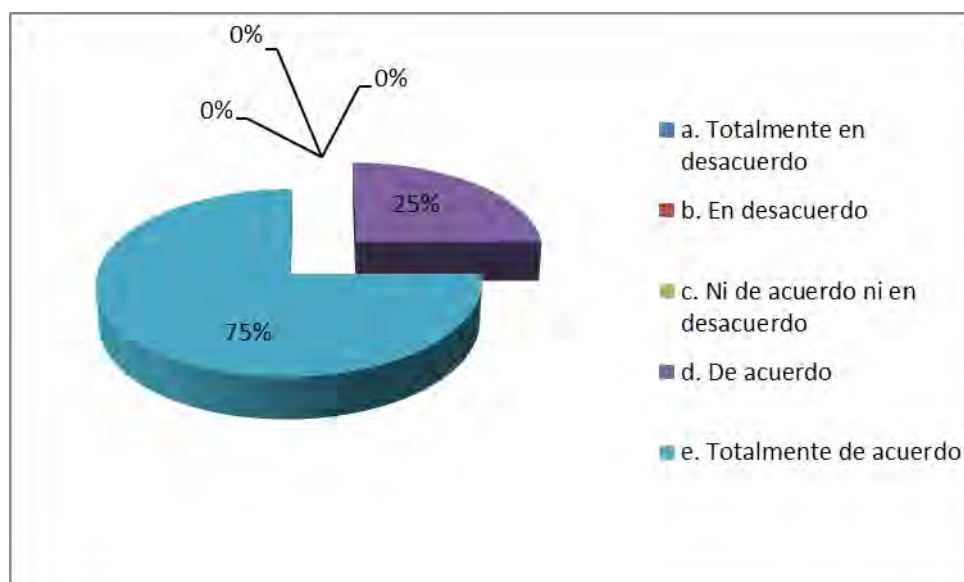


Figura 42: Resultado del indicador Complejidad de código fuente

Interpretación:

El personal de Sistemas refiere que está de acuerdo con los beneficios de identificar la complejidad del código fuente.

Dimensión: Inspección Constante							
Categorías	Indicadores					$\bar{X}$	%
	No 1	No 2	No 3	No 4	No 5		
a. Totalmente en desacuerdo	0.0	0.0	0.0	0.0	0.0	0.0	0.0%
b. En desacuerdo	0.0	0.0	0.0	0.0	0.0	0.0	0.0%
c. Ni de acuerdo ni en desacuerdo	0.0	0.0	0.0	0.0	0.0	0.0	0.0%
d. De acuerdo	2.5	8.0	5.0	8.5	4.0	5.6	35.0%
e. Totalmente de acuerdo	13.5	8.0	11.0	7.5	12.0	10.4	65.0%
<i>Total</i>	16	16	16	16	16	16	100.0%

Tabla 45: Resultado de la Dimensión Inspección Constante

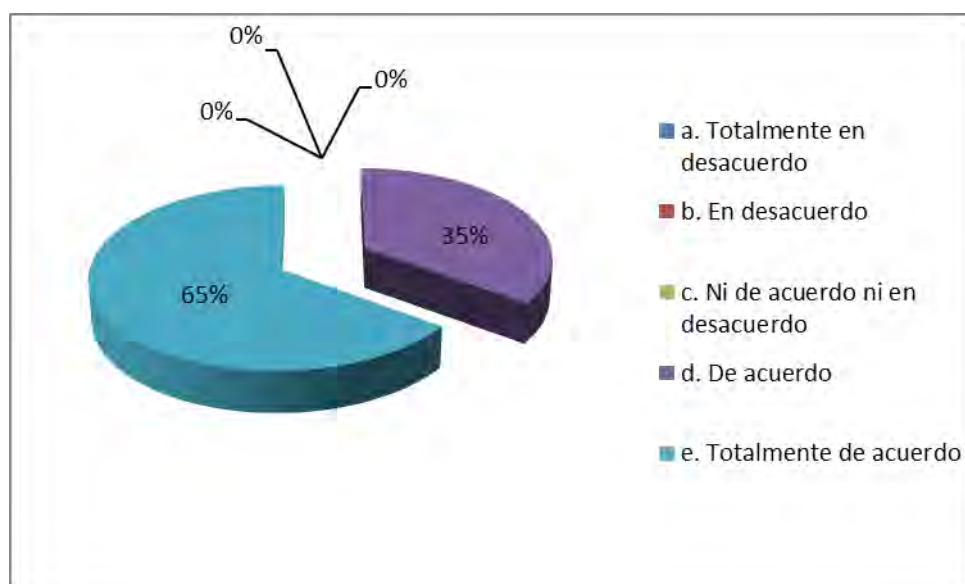


Figura 43: Resultado de la Dimensión Inspección Constante

Interpretación:

El personal de Sistemas refiere que está de acuerdo con la inspección constante de la Integración Continua.

Análisis de los Indicadores de la Variable Independiente
Integración Continua
Dimensión: Retroalimentación Constante

Indicador: Mecanismos de Retroalimentación					
Categorías	Preguntas			$\bar{X}$	%
	No 26	No 27	No 28		
a. Totalmente en desacuerdo	0	0	0	0.0	0.0%
b. En desacuerdo	0	0	0	0.0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0	0	0.0	0.0%
d. De acuerdo	2	3	8	4.3	27.1%
e. Totalmente de acuerdo	14	13	8	11.7	72.9%
<i>Total</i>	<i>16</i>	<i>16</i>	<i>16</i>	<i>16</i>	<i>100.0%</i>

Tabla 46: Resultado del indicador Mecanismos de Retroalimentación

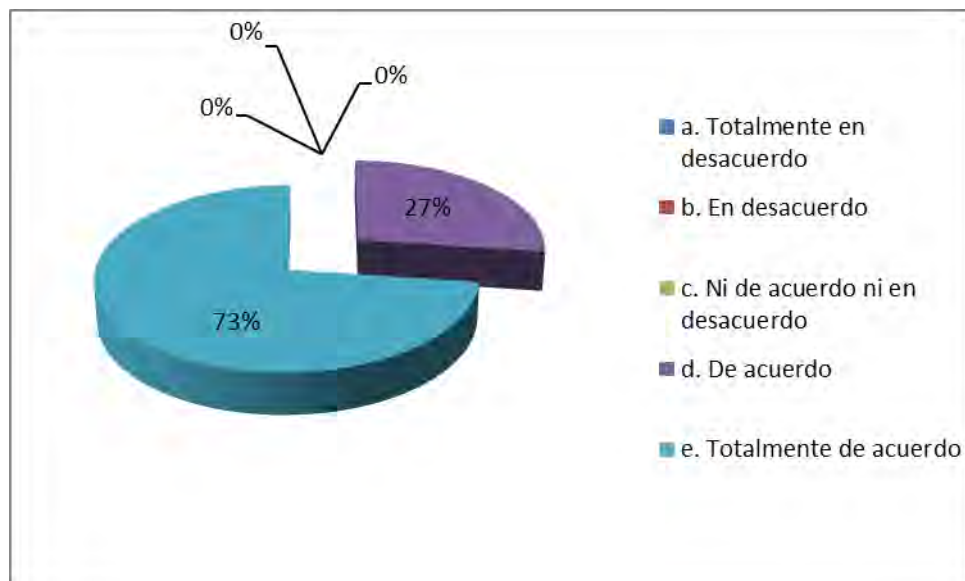


Figura 44: Resultado del indicador Mecanismos de Retroalimentación

Interpretación:

El personal de Sistemas refiere que está de acuerdo con los beneficios de los mecanismos de retroalimentación.

Dimensión: Retroalimentación Constante			
Categorías	Indicador	$\bar{X}$	%
	No 1		
a. Totalmente en desacuerdo	0.0	0.0	0.0%
b. En desacuerdo	0.0	0.0	0.0%
c. Ni de acuerdo ni en desacuerdo	0.0	0.0	0.0%
d. De acuerdo	4.3	4.3	26.9%
e. Totalmente de acuerdo	11.7	11.7	73.1%
<i>Total</i>	16	16	100.0%

Tabla 47: Resultado de la Dimensión Retroalimentación Constante

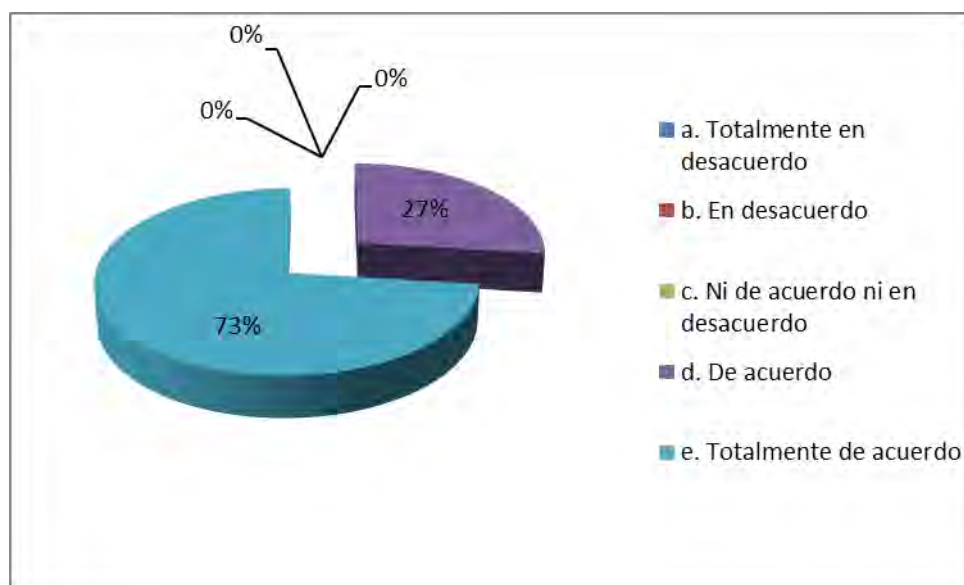


Figura 45: Resultado de la Dimensión Retroalimentación Constante

Interpretación:

El personal de Sistemas refiere que está de acuerdo respecto a la retroalimentación constante.

Análisis de los Indicadores de la Variable Independiente Integración Continua

Variable Independiente: Integración Continua						
Categorías	Dimensiones				$\bar{X}$	%
	No 1	No 2	No 3	No 4		
a. Totalmente en desacuerdo	0.0	0.0	0.0	0.0	0.0	0.0%
b. En desacuerdo	0.0	0.0	0.0	0.0	0.0	0.0%
c. Ni de acuerdo ni en desacuerdo	0.1	0.0	0.0	0.0	0.0	0.2%
d. De acuerdo	4.3	3.3	5.6	4.3	4.4	27.3%
e. Totalmente de acuerdo	11.6	12.7	10.4	11.7	11.6	72.5%
<i>Total</i>	16	16	16	16	16	100.0%

Tabla 48: Resultado de la Integración Continua

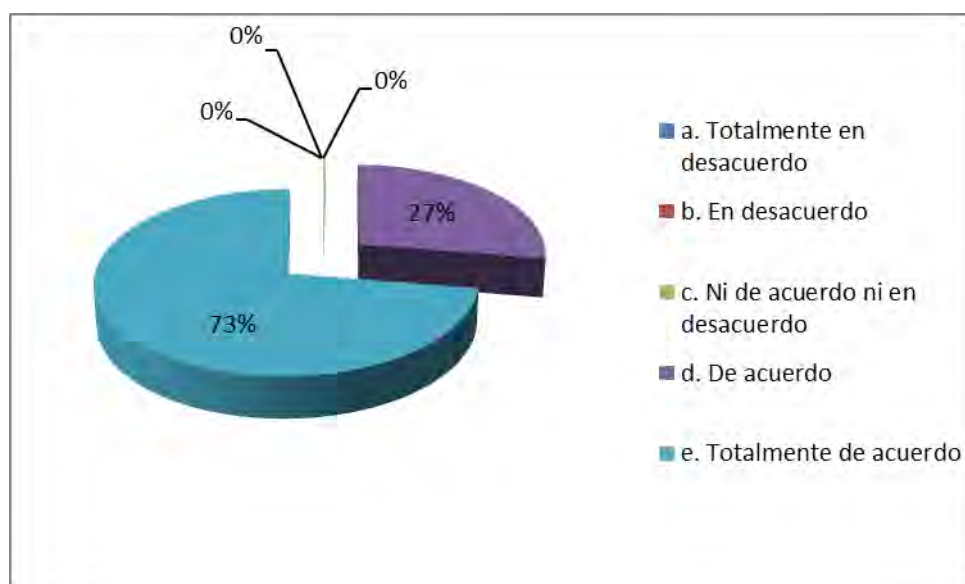


Figura 46: Resultado de la Integración Continua

Interpretación:

El personal de Sistemas refiere que está de acuerdo con la integración continua en todas sus dimensiones.

Cuestionario aplicado para la Variable Dependiente Desarrollo de Software

1. El proceso de desarrollo de software se encuentra en constante revisión.

Alternativas	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.00%
b. En desacuerdo	0	0.00%
c. Ni de acuerdo ni en desacuerdo	0	0.00%
d. De acuerdo	8	50.00%
e. Totalmente de acuerdo	8	50.00%
<i>Total</i>	16	100%

Tabla 49: Resultado de la primera pregunta del cuestionario de la variable dependiente

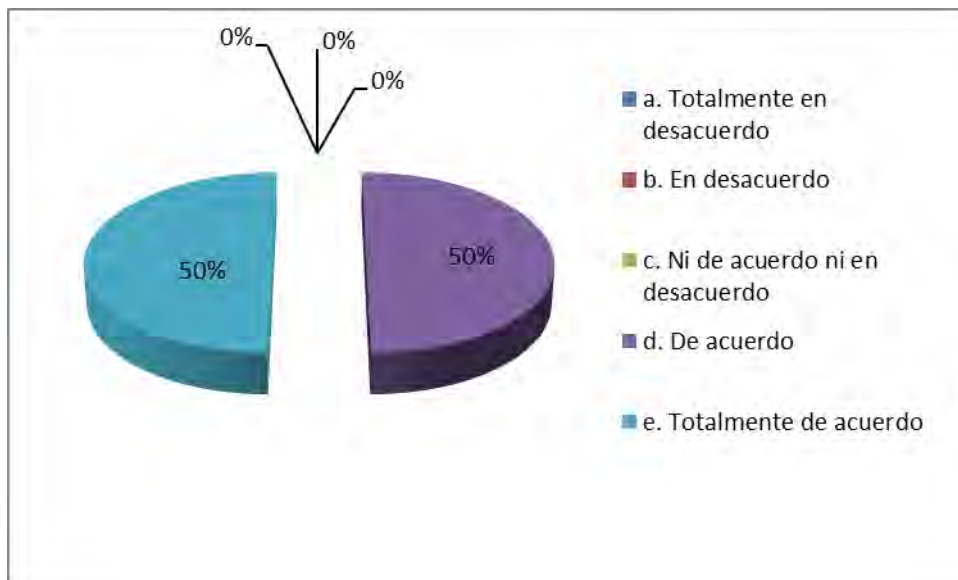


Figura 47: Resultado de la primera pregunta del cuestionario de la variable dependiente

Interpretación:

El personal de Sistemas refiere que se encuentra de acuerdo de que el proceso de desarrollo de software está en constante revisión

2. El proceso de desarrollo de software está bien definido en la metodología.

Alternativas	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.00%
b. En desacuerdo	0	0.00%
c. Ni de acuerdo ni en desacuerdo	0	0.00%
d. De acuerdo	8	50.00%
e. Totalmente de acuerdo	8	50.00%
<i>Total</i>	16	100%

Tabla 50: Resultado de la segunda pregunta del cuestionario de la variable dependiente

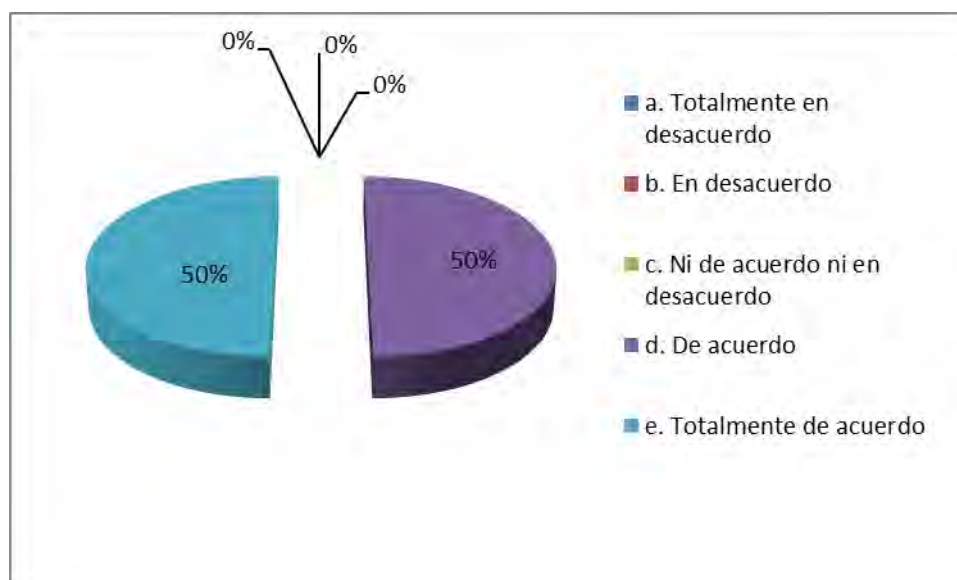


Figura 48: Resultado de la segunda pregunta del cuestionario de la variable dependiente

Interpretación:

El personal de Sistemas refiere que se encuentra de acuerdo de que el proceso de software se encuentra definido en la metodología de la Organización.

3. Existen estándares para el desarrollo de software.		
Alternativas	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.00%
b. En desacuerdo	0	0.00%
c. Ni de acuerdo ni en desacuerdo	0	0.00%
d. De acuerdo	4	25.00%
e. Totalmente de acuerdo	12	75.00%
<i>Total</i>	16	100%

Tabla 51: Resultado de la tercera pregunta del cuestionario de la variable dependiente

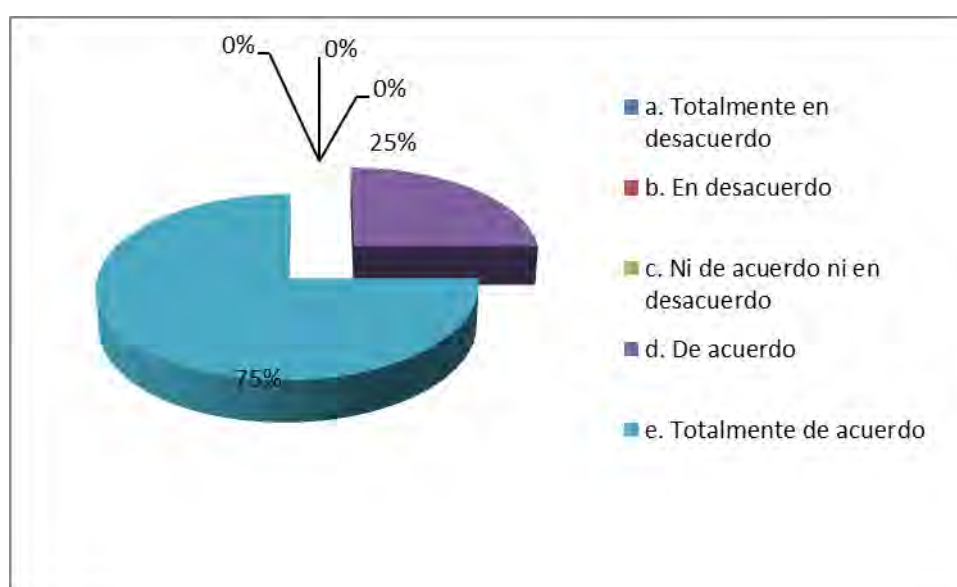


Figura 49: Resultado de la tercera pregunta del cuestionario de la variable dependiente

Interpretación:

El personal de Sistemas refiere que se encuentra de acuerdo de la existencia de estándares para el desarrollo de software.

4. Existen ambientes de despliegue de software de acuerdo al proceso de desarrollo definido.

Alternativas	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.00%
b. En desacuerdo	2	12.50%
c. Ni de acuerdo ni en desacuerdo	0	0.00%
d. De acuerdo	6	37.50%
e. Totalmente de acuerdo	8	50.00%
<i>Total</i>	16	100%

Tabla 52: Resultado de la cuarta pregunta del cuestionario de la variable dependiente

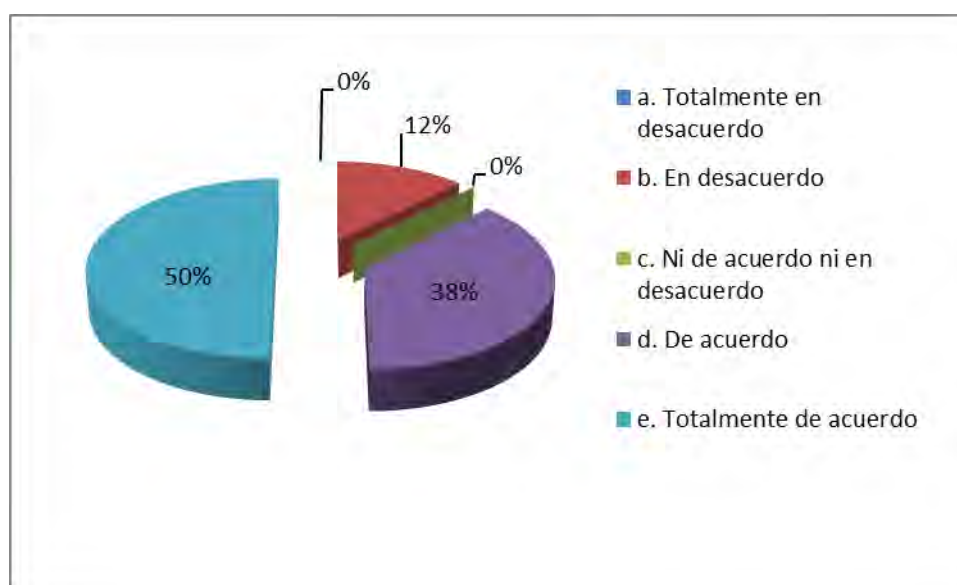


Figura 50: Resultado de la cuarta pregunta del cuestionario de la variable dependiente

Interpretación:

El personal de Sistemas refiere en más del 87% está de acuerdo de la existencia de ambientes de despliegue de software de acuerdo al proceso de desarrollo definido.

5. El proceso de desarrollo de software le permite controlar y manejar la complejidad del software.

Alternativas	Respuestas	Porcentaje
a. Totalmente en desacuerdo	2	12.50%
b. En desacuerdo	7	43.75%
c. Ni de acuerdo ni en desacuerdo	0	0.00%
d. De acuerdo	4	25.00%
e. Totalmente de acuerdo	3	18.75%
<i>Total</i>	16	100%

Tabla 53: Resultado de la quinta pregunta del cuestionario de la variable dependiente

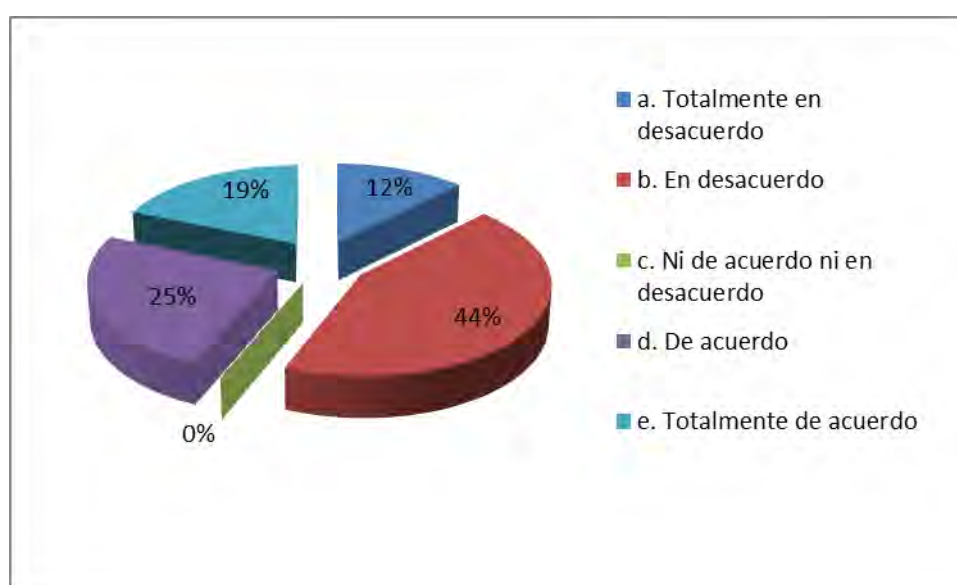


Figura 51: Resultado de la quinta pregunta del cuestionario de la variable dependiente

Interpretación:

El personal de Sistemas refiere en más del 55% está en desacuerdo del proceso de desarrollo de software para controlar y manejar la complejidad del software.

6. El manejo y control de la complejidad del software se respalda en el uso de herramientas definidas.		
Alternativas	Respuestas	Porcentaje
a. Totalmente en desacuerdo	6	37.50%
b. En desacuerdo	6	37.50%
c. Ni de acuerdo ni en desacuerdo	0	0.00%
d. De acuerdo	2	12.50%
e. Totalmente de acuerdo	2	12.50%
<i>Total</i>	16	100%

Tabla 54: Resultado de la sexta pregunta del cuestionario de la variable dependiente



Figura 52: Resultado de la sexta pregunta del cuestionario de la variable dependiente

Interpretación:

El personal de Sistemas refiere en más del 74% está en desacuerdo sobre el manejo y control del software se respalda en el uso de herramientas establecidas.

7. La metodología define la documentación técnica y del proyecto de desarrollo de software.		
Alternativas	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.00%
b. En desacuerdo	1	6.25%
c. Ni de acuerdo ni en desacuerdo	0	0.00%
d. De acuerdo	5	31.25%
e. Totalmente de acuerdo	10	62.50%
<i>Total</i>	16	100%

Tabla 55: Resultado de la séptima pregunta del cuestionario de la variable dependiente

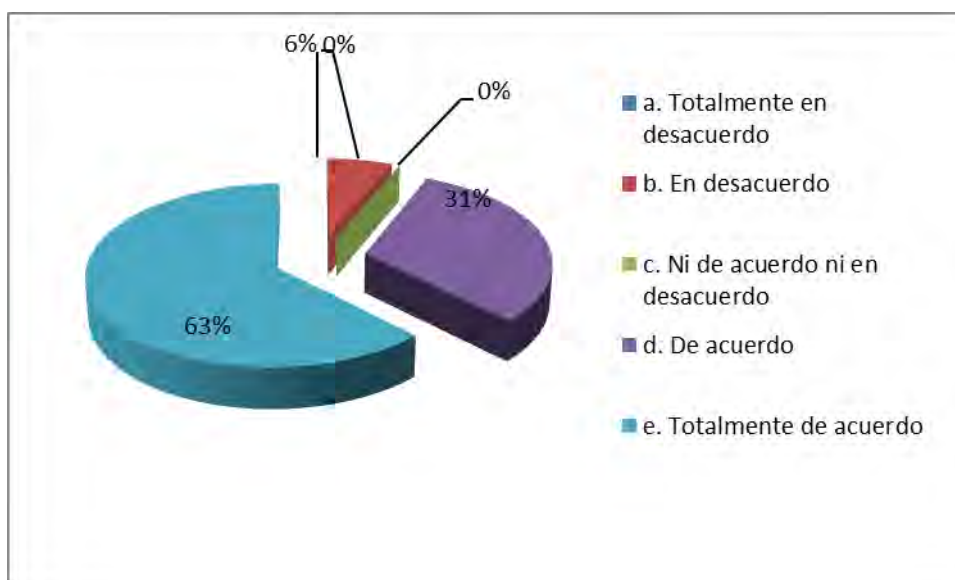


Figura 53: Resultado de la séptima pregunta del cuestionario de la variable dependiente

Interpretación:

El personal de Sistemas refiere en más del 90% está de acuerdo de que la metodología define la documentación técnica y del proyecto de desarrollo de software.

8. La metodología define los formatos, planillas, actas y otros a utilizar en los proyectos de desarrollo de software.

Alternativas	Respuestas	Porcentajes
a. Totalmente en desacuerdo	0	0.00%
b. En desacuerdo	0	0.00%
c. Ni de acuerdo ni en desacuerdo	0	0.00%
d. De acuerdo	4	25.00%
e. Totalmente de acuerdo	12	75.00%
<i>Total</i>	16	100%

Tabla 56: Resultado de la octava pregunta del cuestionario de la variable dependiente

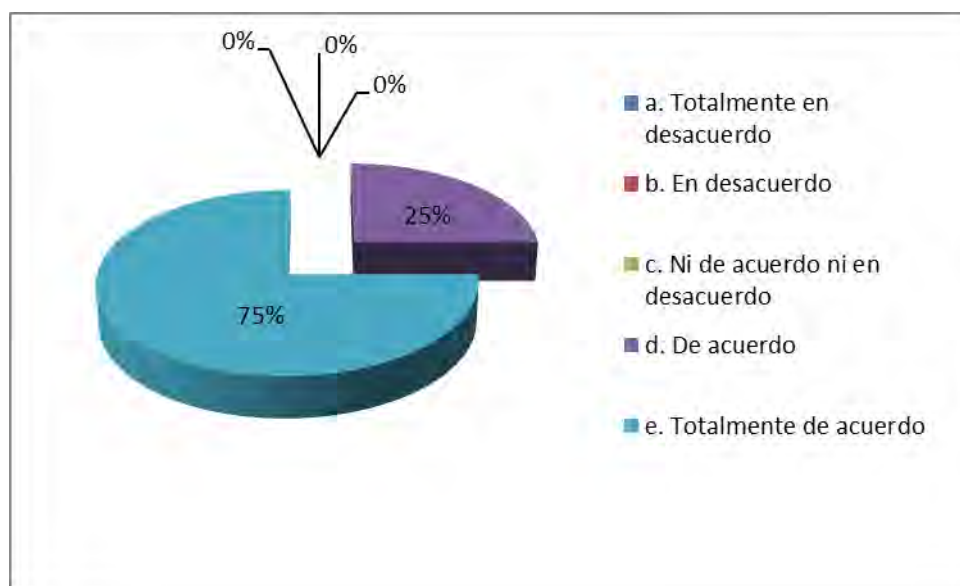


Figura 54: Resultado de la octava pregunta del cuestionario de la variable dependiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que la metodología define los formatos, planillas, actas y otros a utilizar en los proyectos de desarrollo de software

9. La metodología permite que los grupos de desarrollo de software participen como un equipo de trabajo.		
Alternativas	Respuestas	Porcentaje
a. Totalmente en desacuerdo	4	25.00%
b. En desacuerdo	6	37.50%
c. Ni de acuerdo ni en desacuerdo	0	0.00%
d. De acuerdo	3	18.75%
e. Totalmente de acuerdo	3	18.75%
<i>Total</i>	16	100%

Tabla 57: Resultado de la novena pregunta del cuestionario de la variable dependiente

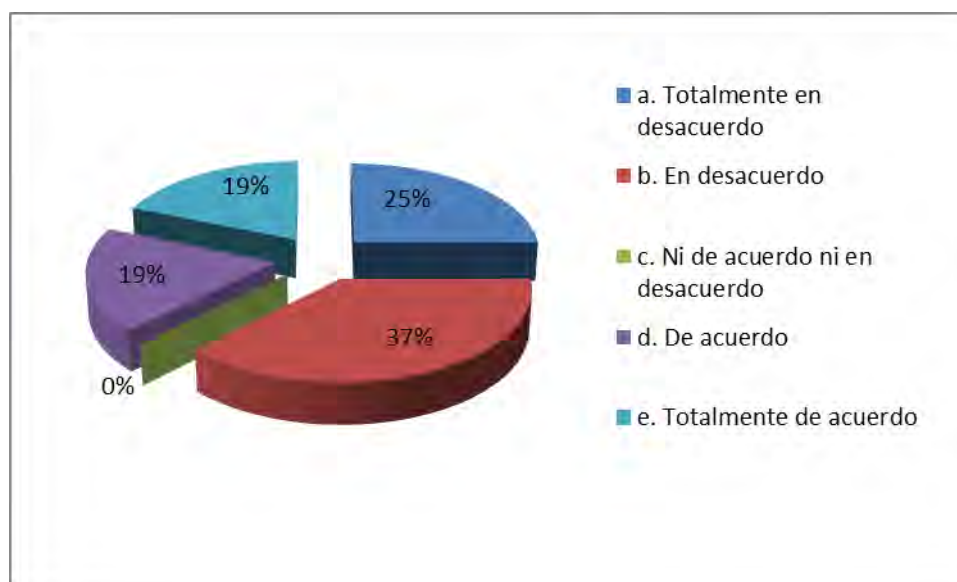


Figura 55: Resultado de la novena pregunta del cuestionario de la variable dependiente

Interpretación:

El personal de Sistemas refiere en más del 62% está en desacuerdo a que la metodología permite que los grupos de desarrollo de software participen como un equipo de trabajo.

10. Los equipos de desarrollo conocen y aplican la metodología.		
Alternativas	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.00%
b. En desacuerdo	0	0.00%
c. Ni de acuerdo ni en desacuerdo	0	0.00%
d. De acuerdo	3	18.75%
e. Totalmente de acuerdo	13	81.25%
<i>Total</i>	16	100%

Tabla 58: Resultado de la décima pregunta del cuestionario de la variable dependiente

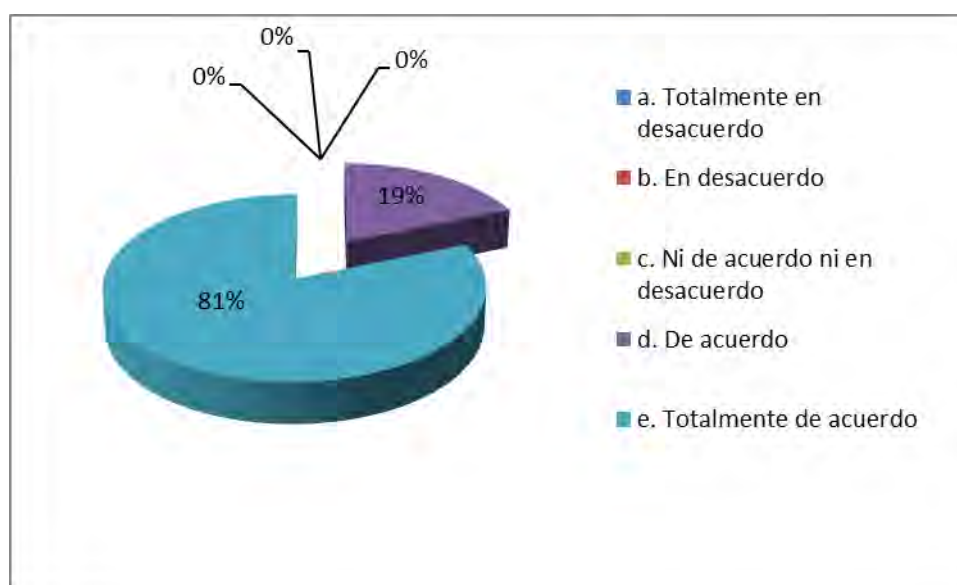


Figura 56: Resultado de la décima pregunta del cuestionario de la variable dependiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que los equipos de desarrollo conocen y aplican la metodología.

11. La metodología es utilizado en la Organización para todos los desarrollos de software.		
Alternativas	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.0%
b. En desacuerdo	0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0%
d. De acuerdo	8	50.00%
e. Totalmente de acuerdo	8	50.00%
<i>Total</i>	16	100%

Tabla 59: Resultado de la décima primera pregunta del cuestionario de la variable dependiente

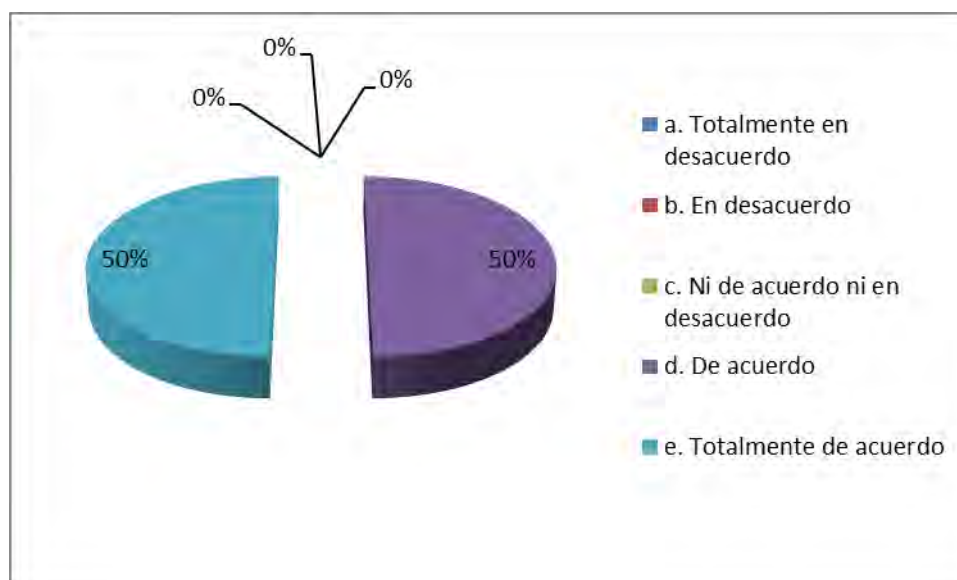


Figura 57: Resultado de la décima primera pregunta del cuestionario de la variable dependiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo con el uso de la metodología para todos los desarrollos de software.

12. La metodología define los medios de retroalimentación de la salud del software a la gestión del desarrollo de software.

Alternativas	Respuestas	Porcentaje
a. Totalmente en desacuerdo	7	43.75%
b. En desacuerdo	7	43.75%
c. Ni de acuerdo ni en desacuerdo	2	12.50%
d. De acuerdo	0	0.00%
e. Totalmente de acuerdo	0	0.00%
<i>Total</i>	16	100%

Tabla 60: Resultado de la décima segunda pregunta del cuestionario de la variable dependiente

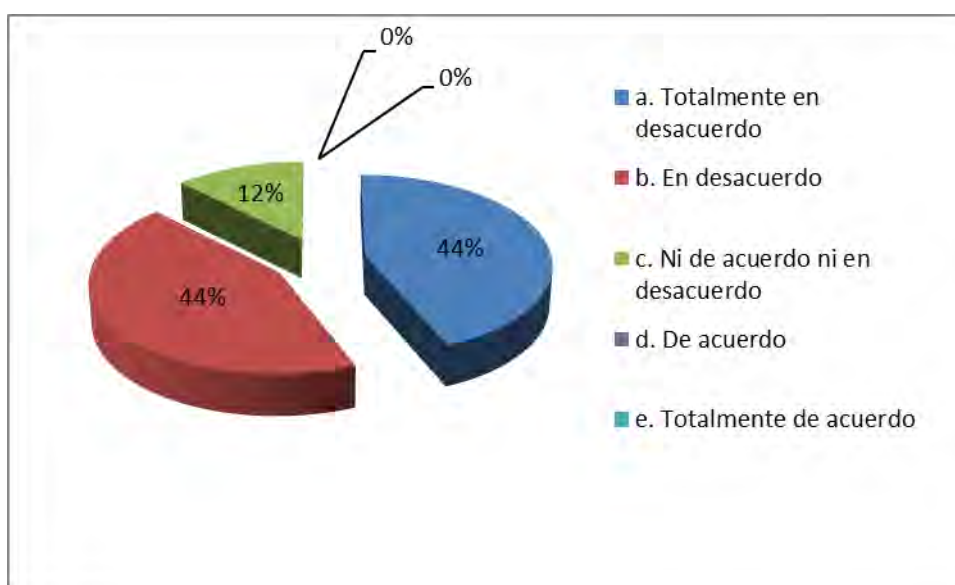


Figura 58: Resultado de la décima segunda pregunta del cuestionario de la variable dependiente

Interpretación:

El personal de Sistemas refiere que más del 86% está en desacuerdo con la retroalimentación de la salud del software que ofrece la metodología.

13. La metodología define los medios de retroalimentación adecuada al equipo de desarrollo.

Alternativas	Respuestas	Porcentaje
a. Totalmente en desacuerdo	7	43.75%
b. En desacuerdo	7	43.75%
c. Ni de acuerdo ni en desacuerdo	2	12.50%
d. De acuerdo	0	0.00%
e. Totalmente de acuerdo	0	0.00%
<i>Total</i>	16	100%

Tabla 61: Resultado de la décima tercera pregunta del cuestionario de la variable dependiente

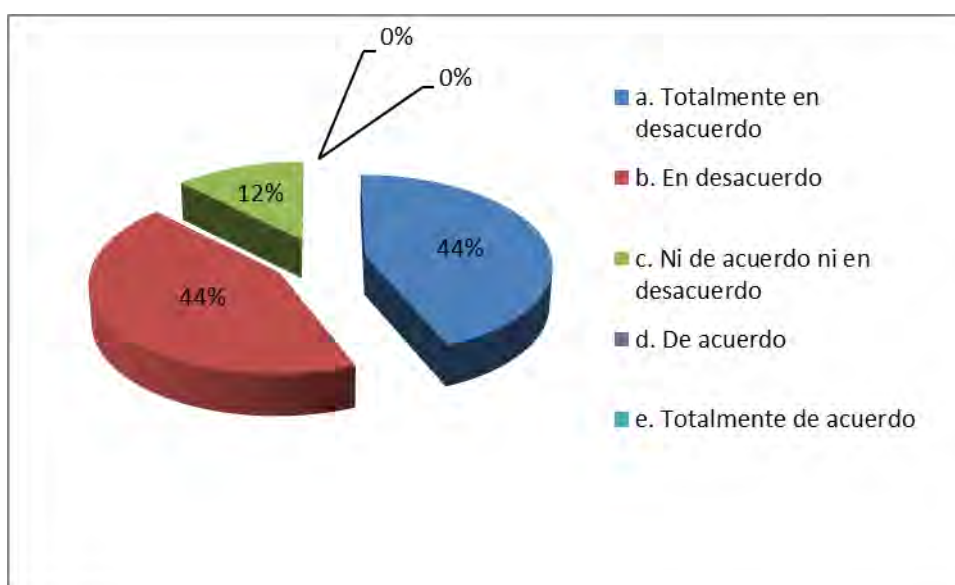


Figura 59: Resultado de la décima tercera pregunta del cuestionario de la variable dependiente

Interpretación:

El personal de Sistemas refiere que más del 86% está en desacuerdo en que la metodología define los medios de retroalimentación adecuada al equipo de desarrollo.

14. La calidad del software desarrollado le permite asegurar una adecuada mantenibilidad del software.		
Alternativas	Respuestas	Porcentaje
a. Totalmente en desacuerdo	4	25.00%
b. En desacuerdo	6	37.50%
c. Ni de acuerdo ni en desacuerdo	0	0.00%
d. De acuerdo	3	18.75%
e. Totalmente de acuerdo	3	18.75%
<i>Total</i>	16	100%

Tabla 62: Resultado de la décima cuarta pregunta del cuestionario de la variable dependiente

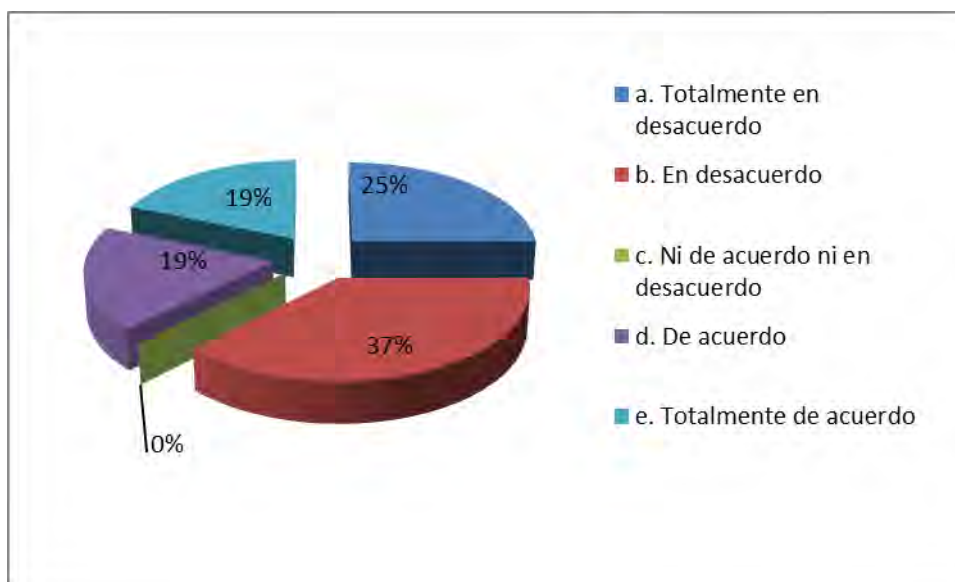


Figura 60: Resultado de la décima cuarta pregunta del cuestionario de la variable dependiente

Interpretación:

El personal de Sistemas refiere en más del 62% está en desacuerdo en que la calidad del software desarrollado le permite asegurar una adecuada mantenibilidad del software.

15. Existe estándares de desarrollo de software referente a la calidad de programación del software.		
Alternativas	Respuestas	Porcentaje
a. Totalmente en desacuerdo	4	25.00%
b. En desacuerdo	4	25.00%
c. Ni de acuerdo ni en desacuerdo	0	0.00%
d. De acuerdo	4	25.00%
e. Totalmente de acuerdo	4	25.00%
<i>Total</i>	16	100%

Tabla 63: Resultado de la décima quinta pregunta del cuestionario de la variable dependiente

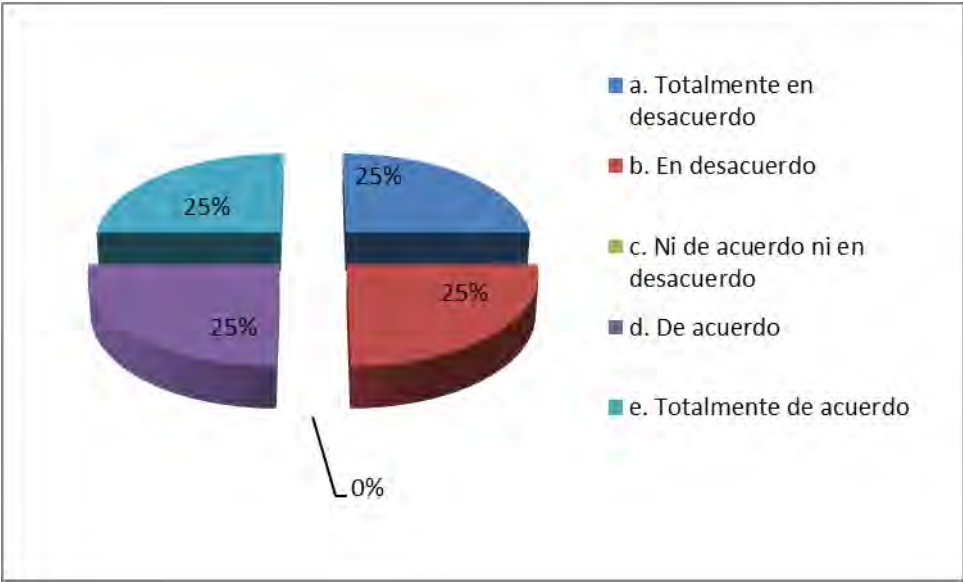


Figura 61: Resultado de la décima quinta pregunta del cuestionario de la variable dependiente

Interpretación:

El personal de Sistemas refiere que el 50% está en desacuerdo y el 50% está de acuerdo respecto a la existencia de estándares de desarrollo de software referente a la calidad de programación del software.

16. El aporte de valor al negocio de los softwares desarrollados mejora con la calidad de los mismos.

Alternativas	Respuestas	Porcentaje
a. Totalmente en desacuerdo	0	0.00%
b. En desacuerdo	0	0.00%
c. Ni de acuerdo ni en desacuerdo	0	0.00%
d. De acuerdo	8	50.00%
e. Totalmente de acuerdo	8	50.00%
<i>Total</i>	16	100%

Tabla 64: Resultado de la décima sexta pregunta del cuestionario de la variable dependiente

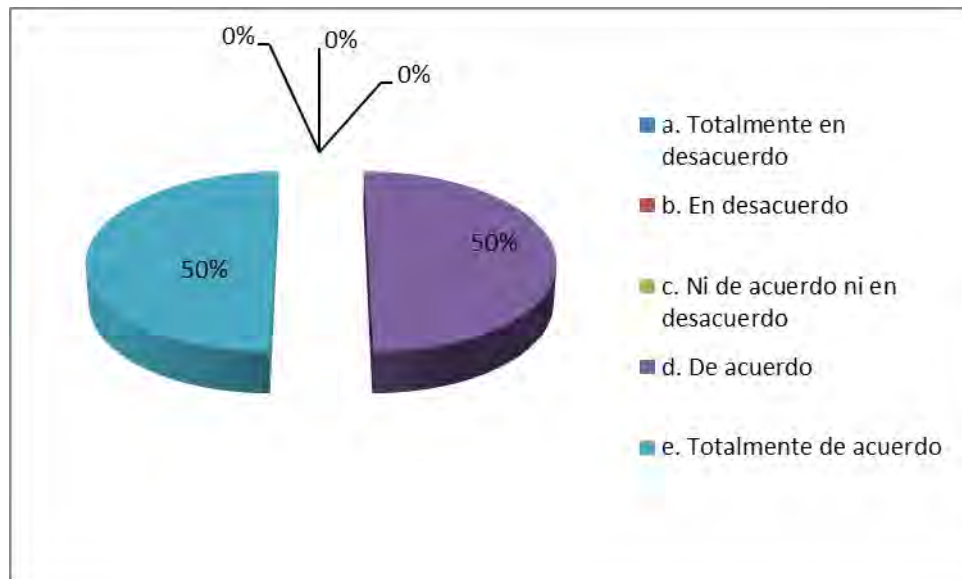


Figura 62: Resultado de la décima sexta pregunta del cuestionario de la variable dependiente

Interpretación:

El personal de Sistemas refiere que está de acuerdo que el aporte de valor al negocio de los softwares desarrollados mejora con la calidad de los mismos.

17. El usuario se encuentra satisfecho de la calidad del software desarrollado.

Alternativas	Respuestas	Porcentaje
a. Totalmente en desacuerdo	3	18.75%
b. En desacuerdo	3	18.75%
c. Ni de acuerdo ni en desacuerdo	0	0.00%
d. De acuerdo	5	31.25%
e. Totalmente de acuerdo	5	31.25%
<i>Total</i>	16	100%

Tabla 65: Resultado de la décima séptima pregunta del cuestionario de la variable dependiente

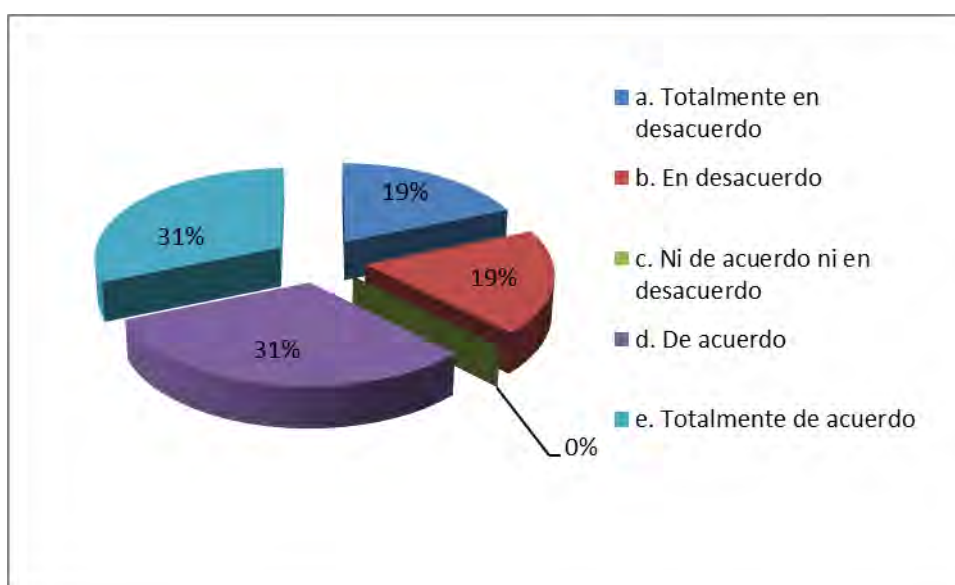


Figura 63: Resultado de la décima séptima pregunta del cuestionario de la variable dependiente

Interpretación:

El personal de Sistemas refiere que más del 62%, que el usuario está de acuerdo con la calidad del software desarrollado.

Análisis de los Indicadores de la Variable Dependiente
Desarrollo de Software
Dimensión: Proceso

Indicador: Procedimiento						
Categorías	Preguntas				$\bar{X}$	%
	No 1	No 2	No 3	No 4		
a. Totalmente en desacuerdo	0	0	0	0	0.0	0.0%
b. En desacuerdo	0	0	0	2	0.5	3.1%
c. Ni de acuerdo ni en desacuerdo	0	0	0	0	0.0	0.0%
d. De acuerdo	8	8	4	6	6.5	40.6%
e. Totalmente de acuerdo	8	8	12	8	9.0	56.3%
Total	16	16	16	16	16	100.0%

Tabla 66: Resultado del indicador Procedimiento

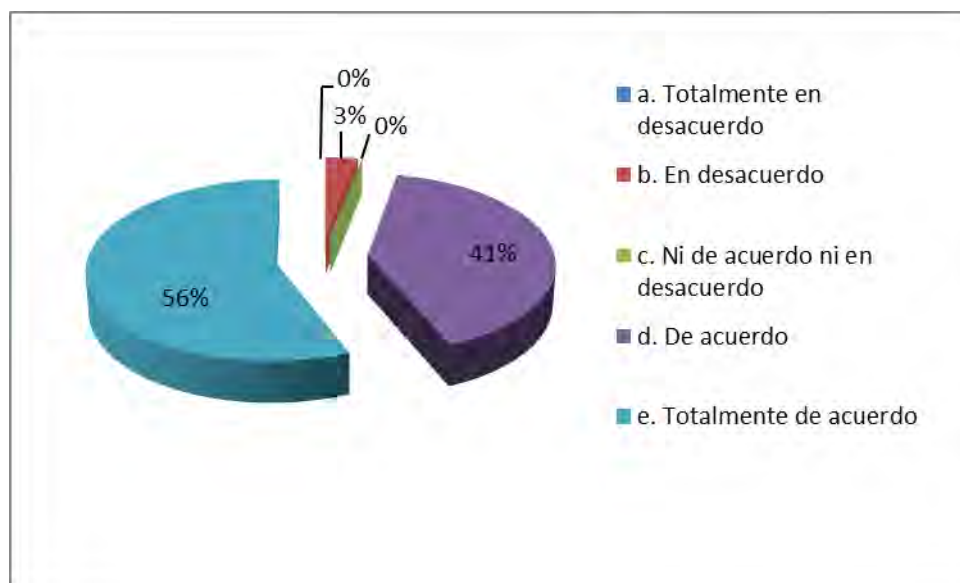


Figura 64: Resultado del indicador Procedimiento

Interpretación:

El personal de Sistemas refiere en más del 96% está de acuerdo con los beneficios del procedimiento del proceso de software.

Indicador: Complejidad de código fuente				
Categorías	Preguntas		$\bar{X}$	%
	No 5	No 6		
a. Totalmente en desacuerdo	2	6	4.0	25.0%
b. En desacuerdo	7	6	6.5	40.6%
c. Ni de acuerdo ni en desacuerdo	0	0	0.0	0.0%
d. De acuerdo	4	2	3.0	18.8%
e. Totalmente de acuerdo	3	2	2.5	15.6%
<i>Total</i>	16	16	16	100.0%

Tabla 67: Resultado del indicador Complejidad del código fuente

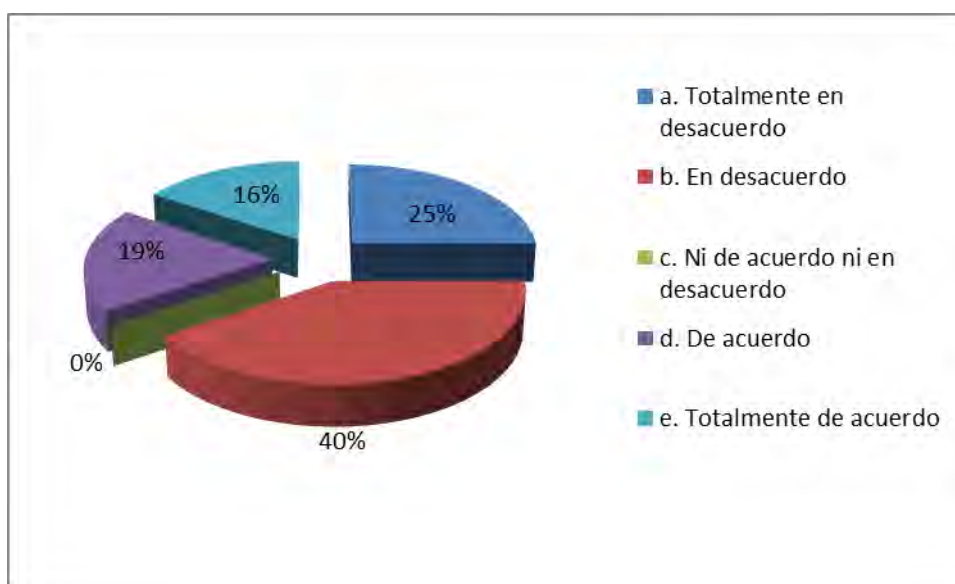


Figura 65: Resultado del indicador Complejidad del código fuente

Interpretación:

El personal de Sistemas refiere en más del 65% está en desacuerdo a como se afronta la complejidad de código fuente.

Dimensión: Proceso				
Categorías	Indicadores		$\bar{X}$	%
	No 1	No 2		
a. Totalmente en desacuerdo	0.0	4.0	2.00	12.5%
b. En desacuerdo	0.5	6.5	3.50	21.9%
c. Ni de acuerdo ni en desacuerdo	0.0	0.0	0.00	0.0%
d. De acuerdo	6.5	3.0	4.75	29.7%
e. Totalmente de acuerdo	9.0	2.5	5.75	35.9%
<i>Total</i>	16	16	16	100.0%

Tabla 68: Resultado de la Dimensión Proceso

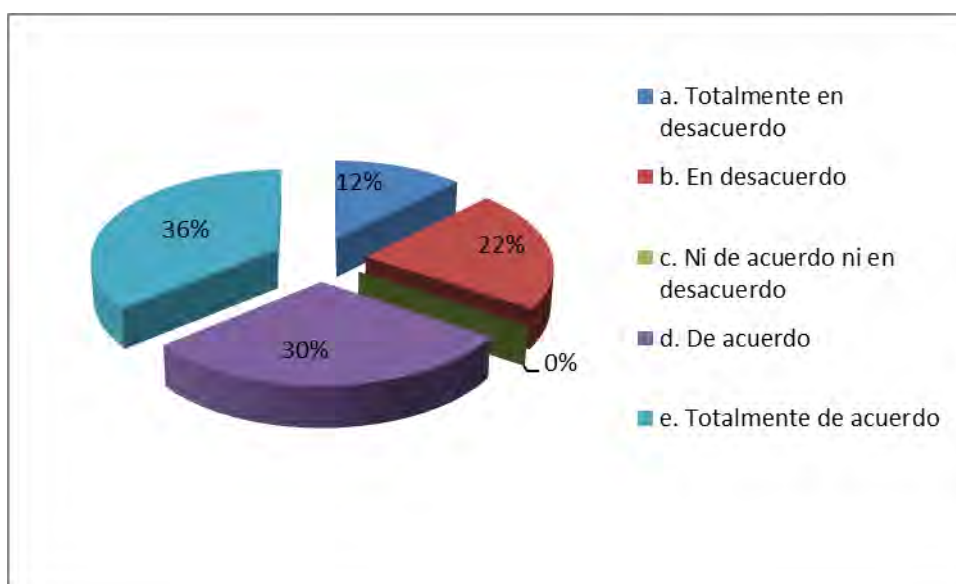


Figura 66: Resultado de la Dimensión Proceso

Interpretación:

El personal de Sistemas refiere en más del 64% está de acuerdo con el proceso del desarrollo de software.

Análisis de los Indicadores de la Variable Dependiente
Desarrollo de Software
Dimensión: Metodología

Indicador: Documentación				
Categorías	Preguntas		$\bar{X}$	%
	No 7	No 8		
a. Totalmente en desacuerdo	0	0	0.0	0.0%
b. En desacuerdo	1	0	0.5	3.1%
c. Ni de acuerdo ni en desacuerdo	0	0	0.0	0.0%
d. De acuerdo	5	4	4.5	28.1%
e. Totalmente de acuerdo	10	12	11.0	68.8%
<i>Total</i>	<i>16</i>	<i>16</i>	<i>16</i>	<i>100.0%</i>

Tabla 69: Resultado del indicador Documentación

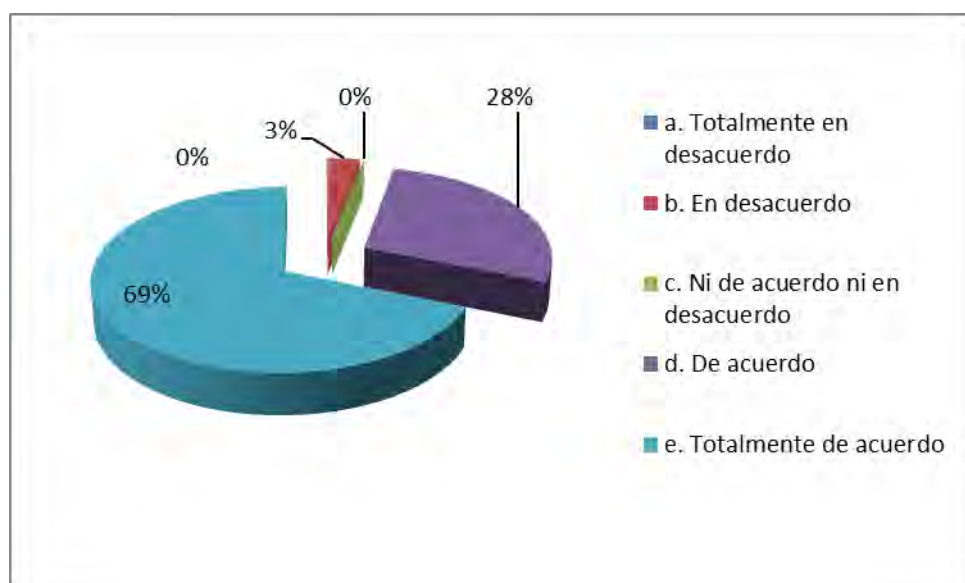


Figura 67: Resultado del indicador Documentación

Interpretación:

El personal de Sistemas refiere en más del 90% está de acuerdo de la documentación referida en la metodología del software.

Indicador: Participación del Equipo				
Categorías	Preguntas		$\bar{X}$	%
	No 9	No 10		
a. Totalmente en desacuerdo	4	0	2.0	12.5%
b. En desacuerdo	6	0	3.0	18.8%
c. Ni de acuerdo ni en desacuerdo	0	0	0.0	0.0%
d. De acuerdo	3	3	3.0	18.8%
e. Totalmente de acuerdo	3	13	8.0	50.0%
<i>Total</i>	16	16	16	100.0%

Tabla 70: Resultado del indicador Participación del Equipo

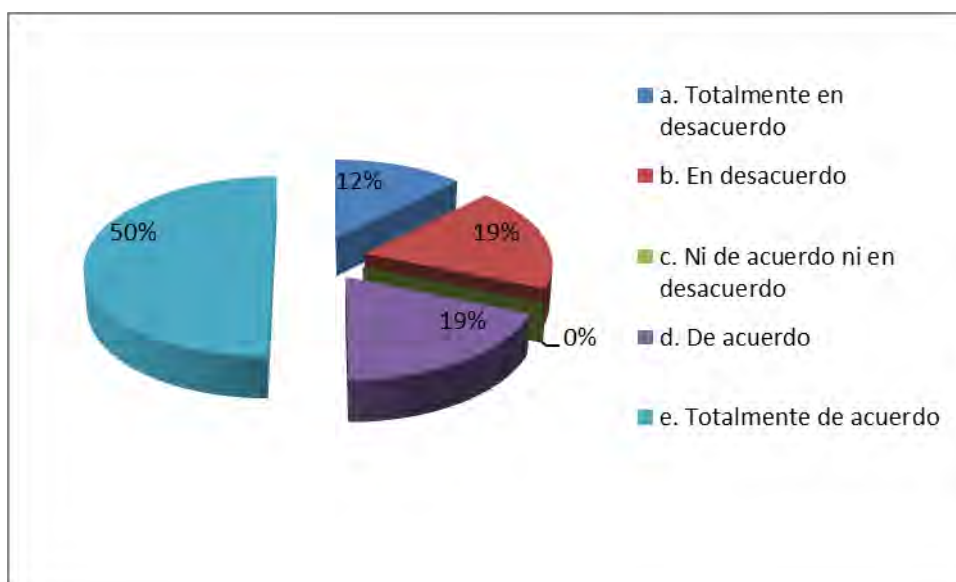


Figura 68: Resultado del indicador Participación del Equipo

Interpretación:

El personal de Sistemas refiere en más del 68% está de acuerdo al respecto de la participación del equipo de software en la metodología del software.

Indicador: Aplicabilidad			
Categorías	Pregunta	$\bar{X}$	%
	No 11		
a. Totalmente en desacuerdo	0	0.0	0.0%
b. En desacuerdo	0	0.0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0	0.0%
d. De acuerdo	8	8.0	50.0%
e. Totalmente de acuerdo	8	8.0	50.0%
<i>Total</i>	16	16	100.0%

Tabla 71: Resultado del indicador Aplicabilidad

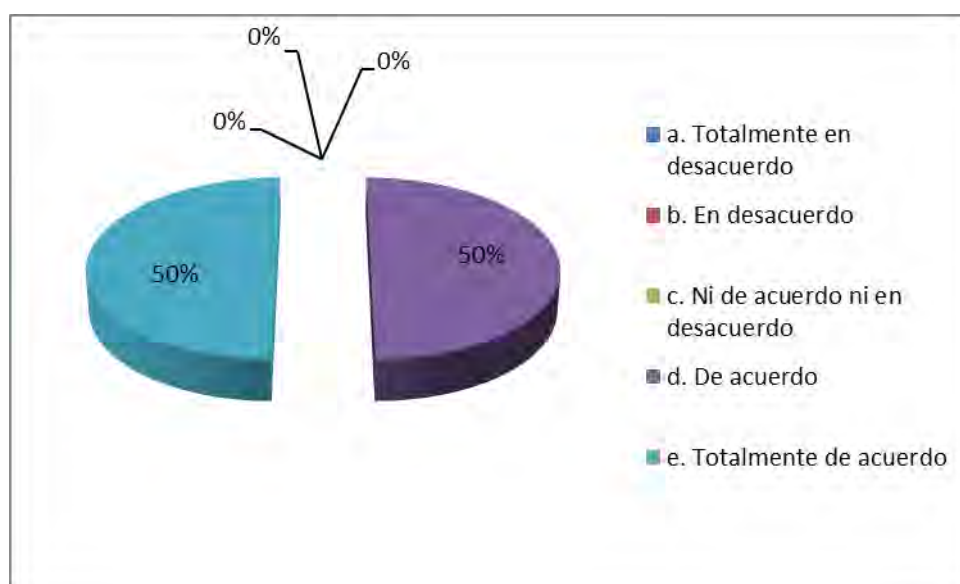


Figura 69: Resultado del indicador Aplicabilidad

Interpretación:

El personal de Sistemas refiere que está de acuerdo con la aplicabilidad de la metodología del software.

Indicador: Retroalimentación				
Categorías	Preguntas		$\bar{X}$	%
	No 12	No 13		
a. Totalmente en desacuerdo	7	7	7.0	43.8%
b. En desacuerdo	7	7	7.0	43.8%
c. Ni de acuerdo ni en desacuerdo	2	2	2.0	12.5%
d. De acuerdo	0	0	0.0	0.0%
e. Totalmente de acuerdo	0	0	0.0	0.0%
<i>Total</i>	16	16	16	100.0%

Tabla 72: Resultado del indicador Retroalimentación

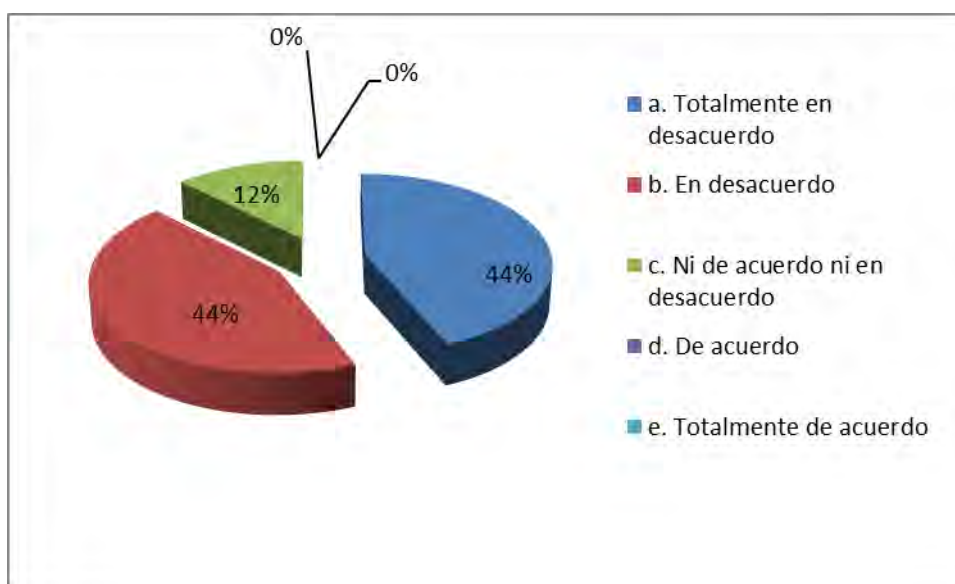


Figura 70: Resultado del indicador Retroalimentación

Interpretación:

El personal de Sistemas refiere en más del 86% está en desacuerdo a la retroalimentación de la metodología del software.

Dimensión: Metodología						
Categorías	Indicadores				$\bar{X}$	%
	No 1	No 2	No 3	No 4		
a. Totalmente en desacuerdo	0.0	2.0	0.0	7.0	2.25	14.1%
b. En desacuerdo	0.5	3.0	0.0	7.0	2.63	16.4%
c. Ni de acuerdo ni en desacuerdo	0.0	0.0	0.0	2.0	0.50	3.1%
d. De acuerdo	4.5	3.0	8.0	0.0	3.88	24.2%
e. Totalmente de acuerdo	11.0	8.0	8.0	0.0	6.75	42.2%
<i>Total</i>	16	16	16	16	16	100.0%

Tabla 73: Resultado de la Dimensión Metodología

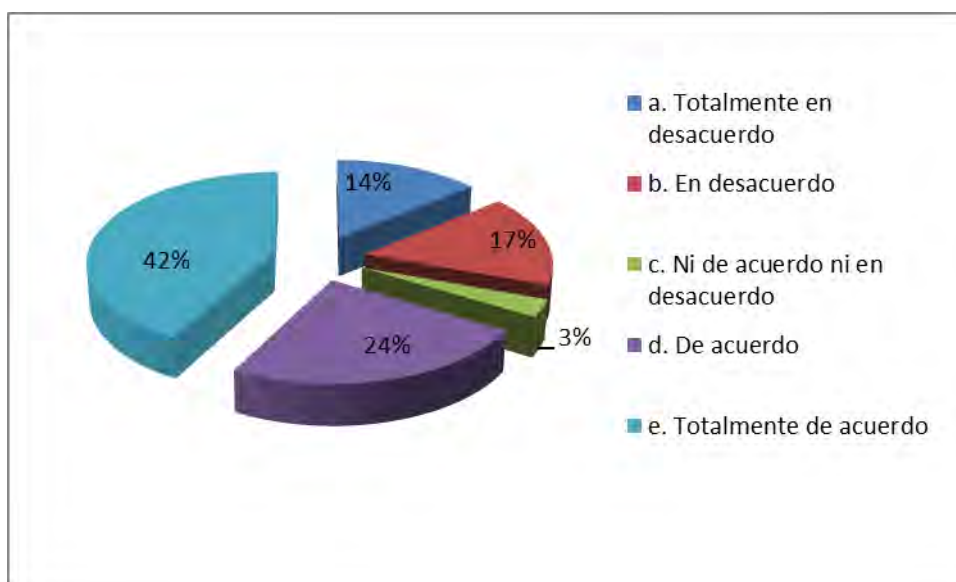


Figura 71: Resultado de la Dimensión Metodología

Interpretación:

El personal de Sistemas refiere en más del 66% que está de acuerdo con la metodología del desarrollo de software.

Análisis de los Indicadores de la Variable Dependiente
Desarrollo de Software
Dimensión: Calidad

Indicador: Mantenibilidad				
Categorías	Preguntas		$\bar{X}$	%
	No 14	No 15		
a. Totalmente en desacuerdo	4	4	4.0	25.0%
b. En desacuerdo	6	4	5.0	31.3%
c. Ni de acuerdo ni en desacuerdo	0	0	0.0	0.0%
d. De acuerdo	3	4	3.5	21.9%
e. Totalmente de acuerdo	3	4	3.5	21.9%
<i>Total</i>	<i>16</i>	<i>16</i>	<i>16</i>	<i>100.0%</i>

Tabla 74: Resultado del Indicador Mantenibilidad

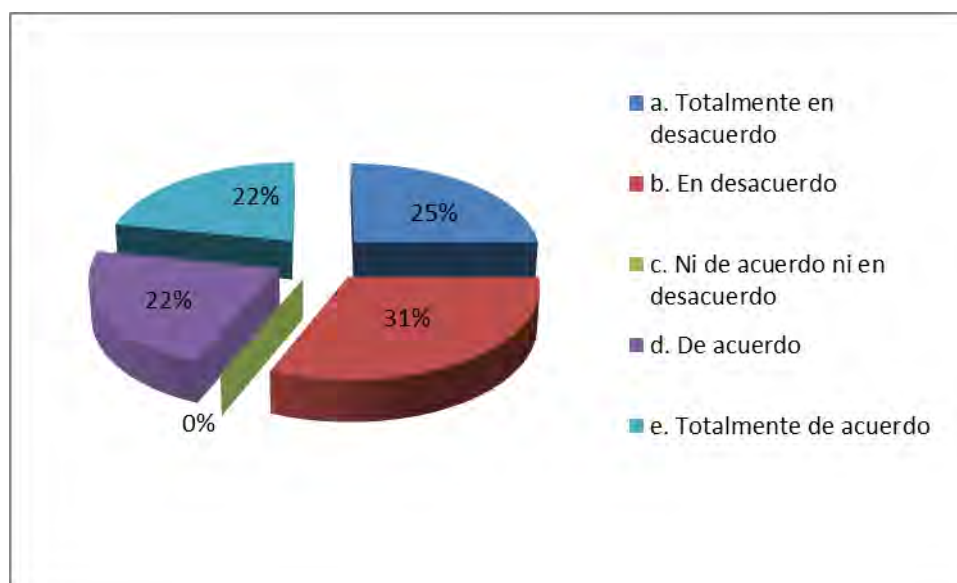


Figura 72: Resultado del Indicador Mantenibilidad

Interpretación:

El personal de Sistemas refiere en más del 56% está en desacuerdo al aseguramiento de la mantenibilidad del software.

Indicador: Aporte al negocio			
Categorías	Pregunta	$\bar{X}$	%
	No 16		
a. Totalmente en desacuerdo	0	0.0	0.0%
b. En desacuerdo	0	0.0	0.0%
c. Ni de acuerdo ni en desacuerdo	0	0.0	0.0%
d. De acuerdo	8	8.0	50.0%
e. Totalmente de acuerdo	8	8.0	50.0%
<i>Total</i>	16	16	100.0%

Tabla 75: Resultado del Indicador Aporte al Negocio

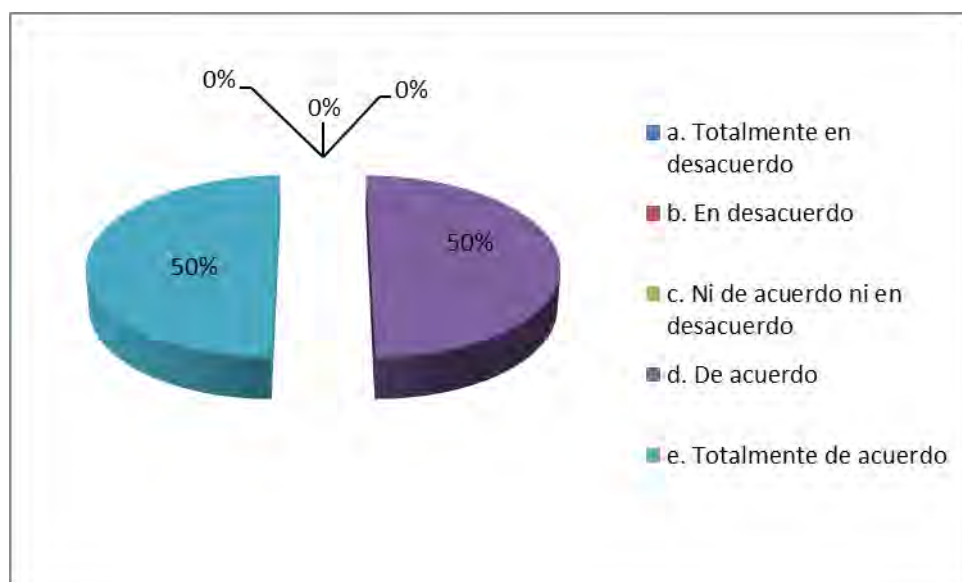


Figura 73: Resultado del Indicador Aporte al Negocio

Interpretación:

El personal de Sistemas refiere a que está de acuerdo que la calidad del software aporta al negocio.

Indicador: Satisfacción del usuario			
Categorías	Pregunta	$\bar{X}$	%
	No 17		
a. Totalmente en desacuerdo	3	3.0	18.8%
b. En desacuerdo	3	3.0	18.8%
c. Ni de acuerdo ni en desacuerdo	0	0.0	0.0%
d. De acuerdo	5	5.0	31.3%
e. Totalmente de acuerdo	5	5.0	31.3%
Total	16	16	100.0%

Tabla 76: Resultado del Indicador Satisfacción del usuario

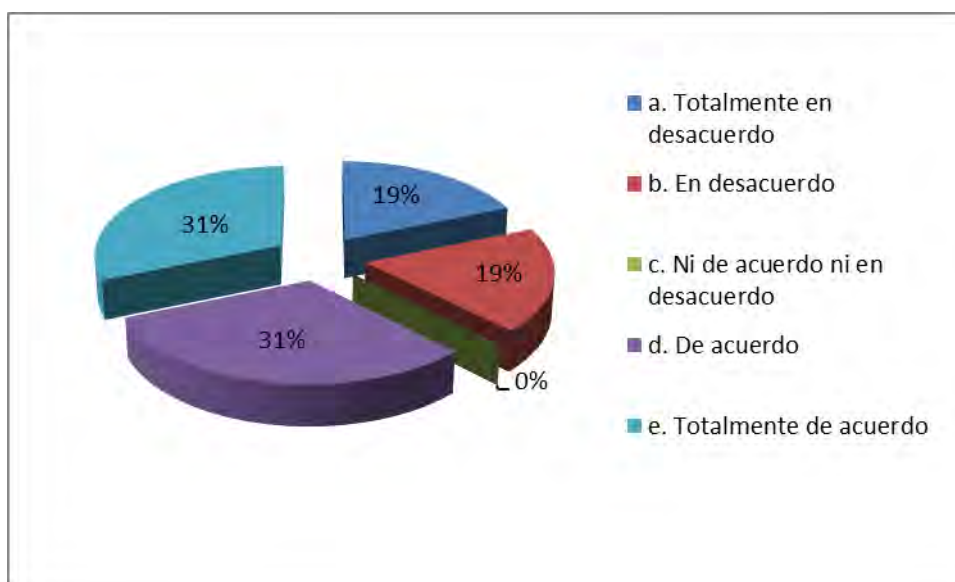


Figura 74: Resultado del Indicador Satisfacción del usuario

Interpretación:

El personal de Sistemas refiere en más del 62% que está de acuerdo con la satisfacción del usuario respecto al desarrollo de software.

Dimensión: Calidad					
Categorías	Indicadores			$\bar{X}$	%
	No 1	No 2	No 3		
a. Totalmente en desacuerdo	4.0	0.0	3.0	2.33	14.6%
b. En desacuerdo	5.0	0.0	3.0	2.67	16.7%
c. Ni de acuerdo ni en desacuerdo	0.0	0.0	0.0	0.00	0.0%
d. De acuerdo	3.5	8.0	5.0	5.50	34.4%
e. Totalmente de acuerdo	3.5	8.0	5.0	5.50	34.4%
<i>Total</i>	16	16	16	16	100.0%

Tabla 77: Resultado de la Dimensión Calidad

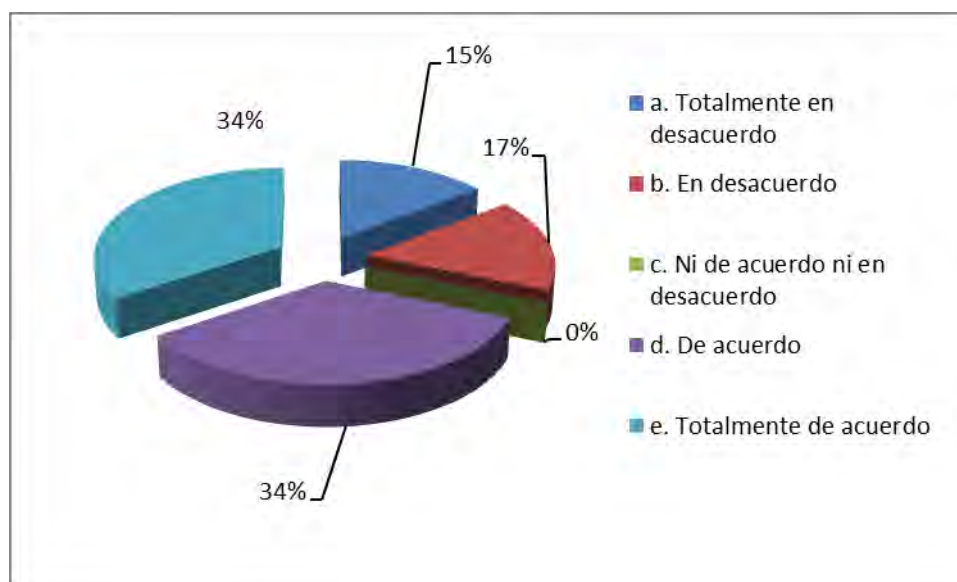


Figura 75: Resultado de la Dimensión Calidad

Interpretación:

El personal de Sistemas refiere en más del 68% que está de acuerdo con la calidad del desarrollo de software.

Análisis de los Indicadores de la Variable Dependiente Desarrollo de Software

Variable Dependiente: Desarrollo de Software					
Categorías	Dimensiones			$\bar{X}$	%
	No 1	No 2	No 3		
a. Totalmente en desacuerdo	2.00	2.25	2.33	2.2	13.7%
b. En desacuerdo	3.50	2.63	2.67	2.9	18.3%
c. Ni de acuerdo ni en desacuerdo	0.00	0.50	0.00	0.2	1.0%
d. De acuerdo	4.75	3.88	5.50	4.7	29.4%
e. Totalmente de acuerdo	5.75	6.75	5.50	6.0	37.5%
<i>Total</i>	16	16	16	16	100.0%

Tabla 78: Resultado del Desarrollo de Software

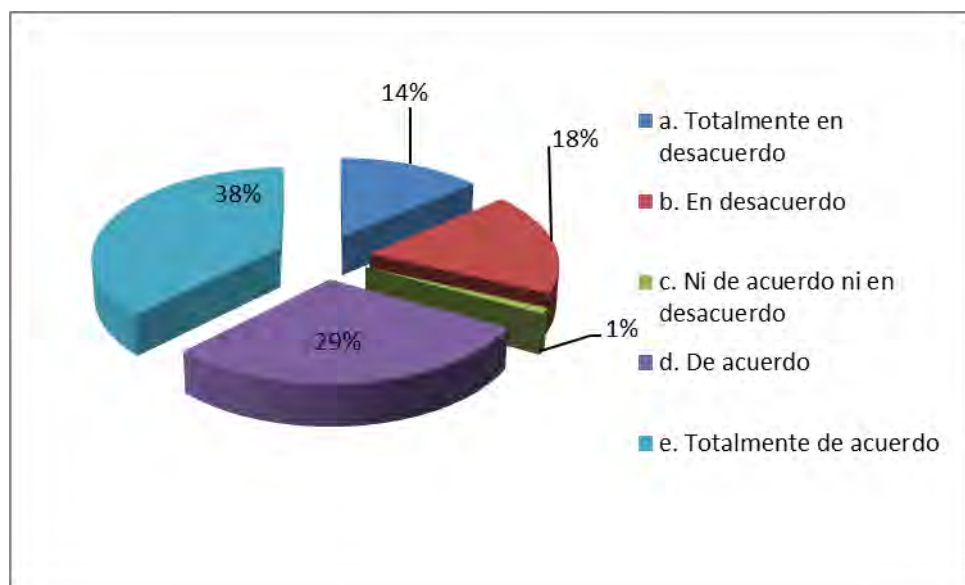


Figura 76: Resultado del Desarrollo de Software

Interpretación:

El personal de Sistemas refiere en más del 66% que está de acuerdo con el desarrollo de software en todas sus dimensiones.

Capítulo 6

Resultados

6.1 Cuadro Final

Integración Continua vs. Desarrollo de Software		
Categorías	Integración Continua	Desarrollo de Software
a. Totalmente en desacuerdo	0	2.2
b. En desacuerdo	0	2.9
c. Ni de acuerdo ni en desacuerdo	0	0.2
d. De acuerdo	4.4	4.7
e. Totalmente de acuerdo	11.6	6.0
<i>Total</i>	16	16

Tabla 1: Matriz comparativa de los resultados de las variables

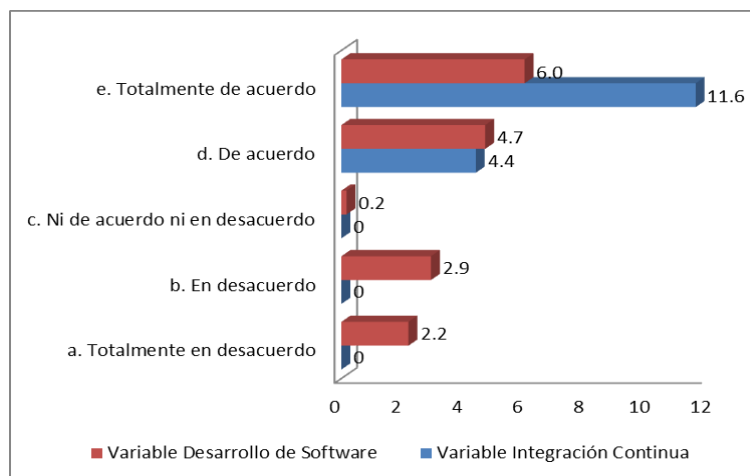


Figura 1 : Resultados de las variables

Se puede apreciar en el cuadro final que hay una relación directa entre ambas variables, es decir, a mayor integración Continua mejor desarrollo de software.

6.2 Hipótesis

Se realizará a continuación el análisis para determinar si existe correlación entre las dos variables. Siendo nuestra hipótesis general “La incorporación de la integración continua influye en el desarrollo de software del OSINERGMIN”.

Para verificar el grado de correlación se utilizó el coeficiente de Pearson, el cual se expresa con la fórmula siguiente:

$$r_{xy} = \frac{n \sum XY - \sum X \sum Y}{\sqrt{\left[n \sum X^2 - (\sum X)^2 \right] \left[n \sum Y^2 - (\sum Y)^2 \right]}}$$

Para ello se utilizó el Microsoft Excel y obteniéndose como resultado: $r_{xy} = 0.8545$. Debido a que n es menor de 20 (solamente tenemos cinco categorías) se utiliza un estimador menos sesgado que es el de Olkin y Pratt, cuya fórmula es la siguiente:

$$r' = r_{xy} \left[1 + \frac{1 - r_{xy}^2}{2(n-3)} \right]$$

El resultado es $r' = 0.9122$.

Luego probamos la hipótesis siguiente: $H_0: r' = 0$, $H_a: r' > 0$, $H_a: r' < 0$

Para ello calculamos t_0 cuya fórmula es:

$$t_{obs} = \frac{r' \sqrt{n-2}}{\sqrt{1-r'^2}}$$

El resultado es $t_0 = 3.8548$

Luego en la tabla de distribución de t student encontramos para 3 grados de libertad un valor de $t_{0.05,3gl} = 3.1824$

Luego: $3.8548 > 3.1824$

Entonces decidimos que existe una relación positiva y directa entre la variable Independiente y Dependiente porque $r' > 0$ y además significativa.

Luego se completa la relación de las Hipótesis Especificas, obteniéndose el siguiente cuadro:

Relación de las Hipótesis					
Hipótesis	r_{xy}	r'	t_0	$t_{0.05,3gl}$	Decisión
a. Hipótesis General	0.8545	0.9122	3.8548	3.1824	$r' > 0$
b. Hipótesis Especifica 1	0.8455	0.9058	3.7029	3.1824	$r' > 0$
c. Hipótesis Especifica 2	0.8141	0.8828	3.2542	3.1824	$r' > 0$
d. Hipótesis Especifica 3	0.8850	0.9330	4.4898	3.1824	$r' > 0$
e. Hipótesis Especifica 4	0.8513	0.9099	3.7990	3.1824	$r' > 0$

Tabla 2: Resultados Finales

Aquí podemos observar que todas las relaciones influyen en el desarrollo de software.

6.3 Resultados

A través de los resultados podemos decir que existe una relación directa significativa entre la integración continua y el desarrollo de software en el OSINERGMIN.

Capítulo 7

Conclusiones y Sugerencias

7.1 Conclusiones

Luego de realizar el análisis y procesamiento de los resultados obtenidos, se llegó a las siguientes conclusiones.

- Se ha determinado que la integración continua influye en el desarrollo de software del OSINERGMIN.
- Se ha determinado que la construcción constante de la integración continua influye en el desarrollo de software del OSINERGMIN.
- Se ha determinado que las pruebas constantes de la integración continua influye en el desarrollo de software del OSINERGMIN.
- Se ha determinado que la inspección constante de la integración continua influye en el desarrollo de software del OSINERGMIN.
- Se ha determinado que la retroalimentación constante de la integración continua influye en el desarrollo de software del OSINERGMIN.

7.2 Recomendaciones

Las recomendaciones de acuerdo a la investigación realizada son las siguientes:

- Organizar reuniones que permitan que fluya el intercambio de experiencias de la integración continua como parte de la mejora continua.
- Nutrir las pruebas constantes con los diversos tipos de pruebas con la finalidad de poder mejorar la calidad del software.
- Se puede enriquecer el desarrollo de software también a través de la integración continua de la base de datos.

Capítulo 8

Bibliografía

Duvall, Paul, Matyas, Steve y Glover, Andrew. (2007). Continuous Integration: Improving Software Quality and Reducing Risks. Addison-Wesley Professional. 283 pp.

Kajko-Mattsson, M., Jonson, M., Koroorian, S. y Westin, F. (2004). Lesson learned from attempts to implement daily build. Proceedings of the Eighth Euromicro Working Conference on Software Maintenance and Reengineering, pages 137-146.

Koroorian, S. y Kajko-Mattsson, M. (2008). A Tale of Two Daily Build Projects. Proceedings of the 2008 The Third International Conference on Software Engineering Advances, pages 245-251.

Karlsson, E., Andersson, L. y Leion, P. (2000). Daily build and feature development in large distributed projects. Proceedings of the 22nd international conference on Software engineering, ACM, pages 652- 654.

Fowler, Martin. (2006). Continuous Integration. En línea Internet. 01 de Junio del 2010. Accesible en <http://www.martinfowler.com/articles/continuousIntegration.html>

Hernández Sampieri, Roberto. (2006). Metodología de la Investigación. 4ta. Ed. McGraw-Hill. 850 pp.

Thoughtworks. (2011). CI Feature Matrix. En línea Internet. 01 de Junio del 2010. Accesible en <http://confluence.public.thoughtworks.org/display/CC/CI+Feature+Matrix>

Fleischer, Georg. (2009). Continuous Integration What companies expect and solutions provide. En línea Internet. 01 de Junio del 2010. Accesible en <http://appl.fontysvenlo.org/results/2008/GF/index.html>

Holck, Jesper y Jorgensen, Niels. (2004). Continuous Integration and quality assurance: A case study two open source projects. Australian Journal of Information Systems.

Fredrick, Jeffrey. (2004). Featured Articles: Continuous Integration. Better Software Magazine. En línea Internet. 01 de Junio del 2010. Accesible en <http://www.stickyminds.com/BetterSoftware/magazine.asp?fn=cifea&id=58>

Shore, James. (2005). Continuous Integration is an Attitude. James Shore. En línea Internet. 01 de Junio del 2010. Accesible en <http://jamesshore.com/Blog/Continuous-Integration-is-an-Attitude.html>

Shore, James. (2006). Continuous Integration on a Dollar a Day. James Shore. En línea Internet. 01 de Junio del 2010. Accesible en <http://jamesshore.com/Blog/Continuous-Integration-on-a-Dollar-a-Day.html>

Hudson. (2011). Hudson. En línea Internet. 01 de Junio del 2010. Accesible en <http://hudson-ci.org>

Sonar. (2011). Sonar. En línea Internet. 01 de Junio del 2010. Accesible en <http://www.sonarsource.org>

Subversion. (2011). Subversion. En línea Internet. 01 de Junio del 2010. Accesible en <http://subversion.apache.org>

Nexus. (2011). Nexus. En línea Internet. 01 de Junio del 2010. Accesible en <http://nexus.sonatype.org>

Junit. (2011). JUnit. En línea Internet. 01 de Junio del 2010. Accesible en <http://www.junit.org>

Anexo A

Cuestionario para medir la Integración Continua

I. Introducción

Este cuestionario tiene como objetivo conocer su opinión sobre la Integración Continua en el OSINERGMIN. Dicha información es completamente anónima, por lo que le solicito responda todas las preguntas con sinceridad y de acuerdo a su experiencia.

II. Indicaciones

A continuación se le presenta una serie de preguntas, las cuales deberá responder marcado con una (X) en la respuesta que considere correcta:

1. Totalmente en desacuerdo 2. En desacuerdo 3. Ni de acuerdo ni en desacuerdo
4. De acuerdo 5. Totalmente de acuerdo

ITEMS	1	2	3	4	5
1. El repositorio del código fuente y documentos ha permitido centralizar la información del software.					
2. El versionar el código fuente y documentos ha permitido tener un control de la historia de la información del software.					
3. El repositorio del código fuente y documentos ha permitido potenciar el trabajo colaborativo del equipo de desarrollo.					
4. La documentación del código fuente agrega valor al desarrollo de software.					
5. La documentación del código fuente debe generarse constantemente.					
6. La integración continua permite desplegar el software rápidamente en los ambientes de desarrollo, pre producción y producción.					
7. La integración continua permite construir reportes de retroalimentación que permite conocer el estado del software desplegado en los ambientes de desarrollo, pre producción y producción.					
8. Cuál es tu posición respecto a que la integración se realice al menos una vez al día.					
9. La frecuencia de la integración continua ha permitido arreglar construcciones rotas rápidamente.					

10. La frecuencia de la integración continua ha permitido potenciar el trabajo colaborativo del equipo de desarrollo.					
11. Las herramientas de integración continua le facilitan el trabajo de monitoreo y control del desarrollo de software.					
12. Las herramientas de integración continua han permitido la adaptación de la integración continua rápidamente en el equipo de desarrollo.					
13. Las pruebas del código de fuente le ha permitido mejorar la calidad del software.					
14. La ejecución constante de las pruebas hace posible mejorar el desarrollo del software.					
15. Los usuarios se encuentran satisfechos de los softwares entregados a través de la integración continua.					
16. La frecuencia continua de la inspección le agrega valor al desarrollo de software.					
17. La frecuencia continua de la inspección le ha permitido mejorar la calidad de la programación del desarrollo de software.					
18. La auditoría de código fuente le permite mejorar la calidad del software.					
19. La codificación estándar le permite facilitar un conocimiento común de la programación a los diversos equipos de desarrollo.					
20. El porcentaje de cobertura de código fuente le permite mejorar la calidad del software.					
21. La cobertura de código fuente ha permitido mejorar el trabajo colaborativo en el equipo de desarrollo.					
22. La información de duplicación de código fuente le permite mejorar la calidad del software.					
23. La duplicidad de código fuente incrementa los costos del proyecto de desarrollo de software, por lo que la integración continua lo ayuda a manejar los costos.					
24. El control de la complejidad del código fuente le ha permitido mejorar la mantenibilidad del software.					
25. El control de la complejidad del código fuente le ha permitido reducir los riesgos del desarrollo de software.					
26. Los mecanismos de retroalimentación le han permitido controlar el proceso de desarrollo de software.					
27. La frecuencia de la retroalimentación continua le agrega valor al desarrollo de software.					
28. La retroalimentación continua ha generado una cultura en el equipo de desarrollo que le permite mejorar el proceso de desarrollo de software y la calidad del software.					

Anexo B

Cuestionario para medir el Desarrollo de Software

I. Introducción

Este cuestionario tiene como objetivo conocer su opinión sobre el Desarrollo de Software en el OSINERGMIN. Dicha información es completamente anónima, por lo que le solicito responda todas las preguntas con sinceridad y de acuerdo a su experiencia.

II. Indicaciones

A continuación se le presenta una serie de preguntas, las cuales deberá responder marcado con una (X) en la respuesta que considere correcta:

ITEMS	1	2	3	4	5
1. El proceso de desarrollo de software se encuentra en constante revisión.					
2. El proceso de desarrollo de software está bien definido en la metodología.					
3. Existen estándares para el desarrollo de software.					
4. Existen ambientes de despliegue de software de acuerdo al proceso de desarrollo definido.					
5. El proceso de desarrollo de software le permite controlar y manejar la complejidad del software.					
6. El manejo y control de la complejidad del software se respalda en el uso de herramientas definidas.					
7. La metodología define la documentación técnica y del proyecto de desarrollo de software.					
8. La metodología define los formatos, planillas, actas y otros a utilizar en los proyectos de desarrollo de software.					
9. La metodología permite que los grupos de desarrollo de software participen como un equipo de trabajo.					
10. Los equipos de desarrollo conocen y aplican la metodología.					
11. La metodología es utilizado para todos los desarrollos de software.					
12. La metodología define los medios de retroalimentación de la salud del software a la gestión del desarrollo de software.					

13. La metodología define los medios de retroalimentación adecuada al equipo de desarrollo.					
14. La calidad del software desarrollado le permite asegurar una adecuada mantenibilidad del software.					
15. Existe estándares de desarrollo de software referente a la calidad de programación del software.					
16. El aporte de valor al negocio de los softwares desarrollados mejora con la calidad de los mismos.					
17. El usuario se encuentra satisfecho de la calidad del software desarrollado.					

Anexo C

Hoja de Tabulación del Cuestionario para medir la Integración Continua

ITEMS	1 (R.1)	2 (R.2)	3 (R.3)	4 (R.4)	5 (R.5)
1. El repositorio del código fuente y documentos ha permitido centralizar la información del software.	0	0	0	4	12
2. El versionar el código fuente y documentos ha permitido tener un control de la historia de la información del software.	0	0	0	1	15
3. El repositorio del código fuente y documentos ha permitido potenciar el trabajo colaborativo del equipo de desarrollo.	0	0	0	2	14
4. La documentación del código fuente agrega valor al desarrollo de software.	0	0	0	6	10
5. La documentación del código fuente debe generarse constantemente.	0	0	0	4	12
6. La integración continua permite desplegar el software rápidamente en los ambientes de desarrollo, pre producción y producción.	0	0	0	8	8
7. La integración continua permite construir reportes de retroalimentación que permite conocer el estado del software desplegado en los ambientes de desarrollo, pre producción y producción.	0	0	0	6	10
8. Cuál es tu posición respecto a que la integración se realice al menos una vez al día.	0	0	2	7	7
9. La frecuencia de la integración continua ha permitido arreglar construcciones rotas rápidamente.	0	0	0	6	10
10. La frecuencia de la integración continua ha permitido potenciar el trabajo colaborativo del equipo de desarrollo.	0	0	0	2	14
11. Las herramientas de integración continua le facilitan el trabajo de monitoreo y control del desarrollo de software.	0	0	0	1	15
12. Las herramientas de integración continua han permitido la adaptación de la integración continua rápidamente en el equipo de desarrollo.	0	0	0	3	13
13. Las pruebas del código de fuente le ha permitido mejorar la calidad del software.	0	0	0	2	14

14. La ejecución constante de las pruebas hace posible mejorar el desarrollo del software.	0	0	0	3	13
15. Los usuarios se encuentran satisfechos de los softwares entregados a través de la integración continua.	0	0	0	4	12
16. La frecuencia continua de la inspección le agrega valor al desarrollo de software.	0	0	0	2	14
17. La frecuencia continua de la inspección le ha permitido mejorar la calidad de la programación del desarrollo de software.	0	0	0	3	13
18. La auditoría de código fuente le permite mejorar la calidad del software.	0	0	0	9	7
19. La codificación estándar le permite facilitar un conocimiento común de la programación a los diversos equipos de desarrollo.	0	0	0	7	9
20. El porcentaje de cobertura de código fuente le permite mejorar la calidad del software.	0	0	0	6	10
21. La cobertura de código fuente ha permitido mejorar el trabajo colaborativo en el equipo de desarrollo.	0	0	0	4	12
22. La información de duplicación de código fuente le permite mejorar la calidad del software.	0	0	0	7	9
23. La duplicidad de código fuente incrementa los costos del proyecto de desarrollo de software, por lo que la integración continua lo ayuda a manejar los costos.	0	0	0	10	6
24. El control de la complejidad del código fuente le ha permitido mejorar la mantenibilidad del software.	0	0	0	5	11
25. El control de la complejidad del código fuente le ha permitido reducir los riesgos del desarrollo de software.	0	0	0	3	13
26. Los mecanismos de retroalimentación le han permitido controlar el proceso de desarrollo de software.	0	0	0	2	14
27. La frecuencia de la retroalimentación continua le agrega valor al desarrollo de software.	0	0	0	3	13
28. La retroalimentación continua ha generado una cultura en el equipo de desarrollo que le permite mejorar el proceso de desarrollo de software y la calidad del software.	0	0	0	8	8

Anexo D

Hoja de Tabulación del Cuestionario para medir el Desarrollo de Software

ITEMS	1 (R.1)	2 (R.2)	3 (R.3)	4 (R.4)	5 (R.5)
1. El proceso de desarrollo de software se encuentra en constante revisión.	0	0	0	8	8
2. El proceso de desarrollo de software está bien definido en la metodología.	0	0	0	8	8
3. Existen estándares para el desarrollo de software.	0	0	0	4	12
4. Existen ambientes de despliegue de software de acuerdo al proceso de desarrollo definido.	0	2	0	6	8
5. El proceso de desarrollo de software le permite controlar y manejar la complejidad del software.	2	7	0	4	3
6. El manejo y control de la complejidad del software se respalda en el uso de herramientas definidas.	6	6	0	2	2
7. La metodología define la documentación técnica y del proyecto de desarrollo de software.	0	1	0	5	10
8. La metodología define los formatos, planillas, actas y otros a utilizar en los proyectos de desarrollo de software.	0	0	0	4	12
9. La metodología permite que los grupos de desarrollo de software participen como un equipo de trabajo.	4	6	0	3	3
10. Los equipos de desarrollo conocen y aplican la metodología.	0	0	0	3	13
11. La metodología es utilizado para todos los desarrollos de software.	0	0	0	8	8
12. La metodología define los medios de retroalimentación de la salud del software a la gestión del desarrollo de software.	7	7	2	0	0
13. La metodología define los medios de retroalimentación adecuada al equipo de desarrollo.	7	7	2	0	0
14. La calidad del software desarrollado le permite asegurar una adecuada mantenibilidad del software.	4	6	0	3	3
15. Existe estándares de desarrollo de software referente a la calidad de programación del software.	4	4	0	4	4

16. El aporte de valor al negocio de los softwares desarrollados mejora con la calidad de los mismos.	0	0	0	8	8
17. El usuario se encuentra satisfecho de la calidad del software desarrollado.	3	3	0	5	5

Anexo E

Implementación de la Integración Continua

I. Modelo de integración continua

Basados en las prácticas de la Integración Continua se elaboró un modelo a implementar en el OSINERGMIN.

El modelo propuesto en la Figura No 84 contempla:

- El desarrollador debe asegurarse de la calidad del código que va integrar al repositorio de control de versiones, por lo tanto, es quien debe desarrollar las pruebas unitarias, ejecutar las pruebas unitarias y verificar el resultado de las pruebas unitarias. Además deberá verificar la calidad del código a través de métricas del código.
- Antes de subir los cambios al repositorio de control de versiones, de acuerdo a las buenas prácticas encontrados en la bibliografía investigada es necesario realizar la integración localmente, por lo que el desarrollador debe integrar en su equipo su código fuente contra el código fuente que se encuentra en el repositorio de control de versiones, si existe conflictos de código debe solucionarlos y luego realizar las pruebas descritas en el punto anterior.
- Luego de asegurarse el desarrollador que su código no presenta conflictos, debe subir los cambios al repositorio de control de versiones.
- Debido a que el servidor de Integración Continua detecta cambios en el repositorio de control de versiones, disparará los scripts de construcción.
- El servidor de integración continua identifica los artefactos o librerías necesarios para la compilación, por lo que el repositorio de artefactos provee las dependencias solicitadas.
- El servidor de integración continua ejecuta las pruebas, realiza la integración de la base de datos (si es que hubiese) y construye el software.
- Luego comunica al servidor de inspección de código para ejecutar los scripts de inspección y generar los reportes de métricas de la calidad del código fuente.
- Cuando se construye el software, el servidor de integración continua despliega el software en el servidor de aplicaciones de acuerdo a la programación realizada: servidor de pruebas, servidor de pruebas de aceptación de usuario o servidor de producción; y además publica los nuevos artefactos o librerías en el repositorio de artefactos.

- La retroalimentación hacia el desarrollador respecto a la integración continua se podrá realizar por mensajes de textos, mensajes por correo electrónico u otros.
- Si existen fallas en el proceso de integración continua, el desarrollador y todo el equipo (si fuera necesario) deberán corregir la falla presentada para poder continuar con el desarrollo del software.

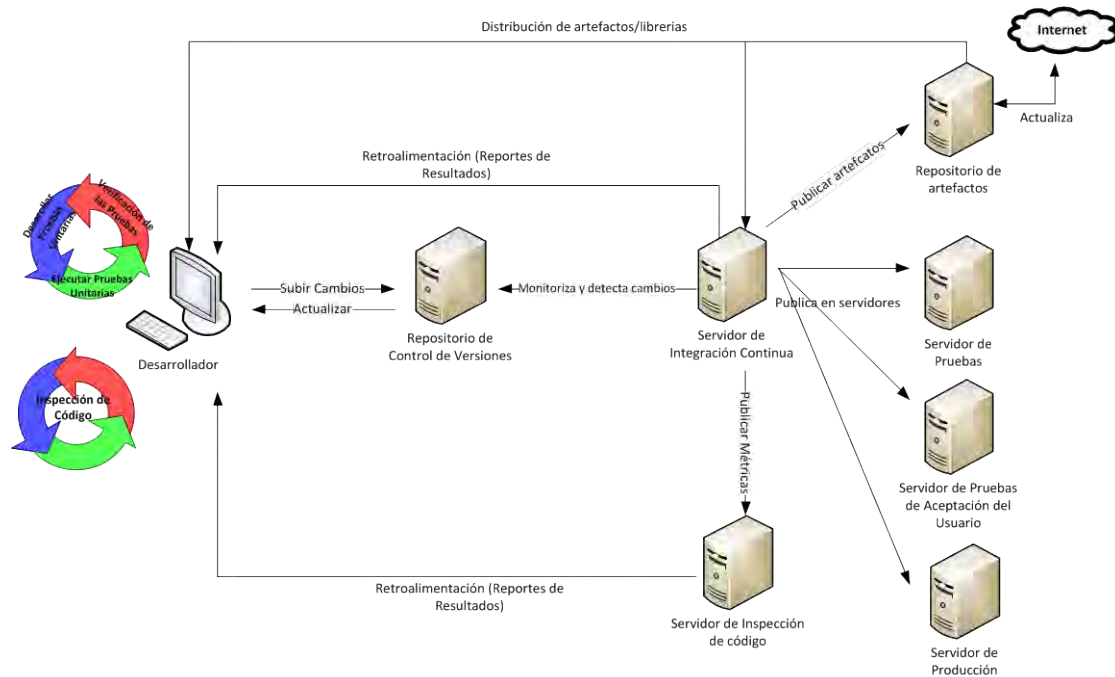


Figura 1: Modelo de integración continua para el OSINERGMIN

II. Evaluando Pre requisitos para la integración continua

Kajko-Mattson evalúa la predisposición de las organizaciones a la introducción de construcción diaria. El objetivo de sus estudios era reconocer las razones del éxito y del fracaso en la introducción del proceso de construcción diaria en una empresa. Este estudio se realizó en dos organizaciones grandes de desarrollo de software.

A continuación se detallan los requisitos mencionados en el estudio:

Una nueva construcción es creada frecuentemente

Kajko-Mattson y Koroorian en sus estudios iniciales mencionan que se debe reducir el tiempo de retroalimentación de cambios entre los desarrolladores, pero discrepan a que se tenga que realizar al menos una vez al día ya que afecta al equipo de desarrollo. Luego en sus estudios posteriores apoyan la integración diaria, pero recomiendan que la organización debería proporcionar las directrices que indique cómo y cuándo los desarrolladores deben presentar su código. Además menciona que las construcciones deben realizarse desde un repositorio central.

Los desarrolladores deben probar su código antes de entregar el software construido

Kajko-Mattson y Koroorian a través de sus estudios indican que las empresas deben desarrollar directrices para los procesos de la integración continua, donde se acentúe la responsabilidad del desarrollador por su código, la especificación de las interfaces y de la documentación.

Se debe tener en cuenta que una directriz de construcción diaria no asegura la calidad del software, si el equipo de desarrollo no realiza la revisión de su código, ya sea individualmente o en pares (para código complejo). Se debe tener en cuenta que esta revisión se realiza antes del integra al repositorio, con la finalidad de no afectar al proyecto.

Los desarrolladores están comprometidos a crear un software exitoso

Kajko-Mattson y Koroorian a través de sus estudios indican que los desarrolladores descubrieron que el proceso de construcción diaria los ayudo a encontrar los problemas rápidamente y que se encontraban comprometidos a solucionar estos problemas en el momento. Así también, el estudio pone énfasis que la empresas no deben impulsar castigos a los desarrolladores cuando se generan problemas en el código, sino que debe de impulsar la cultura de responsabilidad del código y la calidad del software a entregar.

Los desarrolladores se organizan en equipos de trabajo

Kajko-Mattson y Koroorian a través de sus estudios indican que los desarrolladores cuando utilizan el proceso de construcción diaria deben organizarse en equipos de trabajo, ya que requieren de una mayor comunicación y cooperación entre los miembros.

Un punto resaltante que menciona Kajko-Mattson y Koroorian es que los miembros del equipo, primero debe obtener los cambios existentes del repositorio, para luego integrarlo en su código y realizar las pruebas y afinamientos en su equipo informático antes de integrarlo al repositorio.

El rol del Ingeniero de Construcción está establecido

Kajko-Mattson y Koroorian mencionan que debe existir un papel responsable de la construcción del software. Esta persona permitirá ayudar a los desarrolladores en la construcción del software. Según el estudio realizado se llegó a la conclusión que la empresa necesita este rol a tiempo completo y que se encuentre comprometido con su papel.

El proceso de construcción diaria es soportado por un software de gestión de la configuración

Kajko-Mattson y Koroorian mencionan que debe existir un software de gestión de la configuración donde se apoye el proceso de desarrollo de software, siendo la información que almacena este software resguardada de acuerdo a las políticas de seguridad de la información.

El proceso de construcción automatizado

Kajko-Mattson y Koroorian mencionan que el proceso de construcción manual genera errores manuales y genera procesos en el equipo de desarrollo que no agrega valor al desarrollo de software (ya que es repetitivo y en cada construcción la horas hombre van en aumento), en cambio, al desarrollar un proceso automatizado de construcción, este se realiza por única vez y solamente se afina en el tiempo.

Prueba automatizada

Kajko-Mattson y Koroorian a través de su estudio establecen que las pruebas deben ser automatizadas para poder realizar todas las pruebas que un software debe cumplir. Además menciona el valor que tiene el tiempo de respuesta de las pruebas automatizadas en la mejora de la calidad del software.

III. Herramientas de integración continua

En la actualidad, existen una amplia gama de herramientas de integración continua que nos van ayudar a validar el modelo que se ha propuesto. Pero se debe tener presente, que las herramientas que nos dan mayor valor son aquellas más adoptadas o soportadas.

Una de las evaluaciones que encontré y me parece interesante fue la realizada por la Thoughtworks donde evaluó las características de las herramientas existentes en el mercado y su integración entre ellas, lo interesante de esta matriz es que constantemente se encuentra actualizada.

En este aspecto, se va a explicar de las características que se debe de tener en cuenta al momento de elegir implementar la integración continua:

Repositorio de control de versiones

Existe una amplia gama de repositorio de control de versiones, por lo que se debe elegir aquel que se integra fácilmente al servidor de integración continua. Se debe tener en cuenta que el repositorio de control de versiones es la base preliminar para todas las soluciones de Integración Continua.

Entre estas herramientas tenemos: Team Foundation Server, Subversion, CVS, CA Harvest y otros.

Herramientas para realizar pruebas

Las pruebas son realizadas por el servidor de integración continua y son estas herramientas las que proporcionan un soporte adicional para la evaluación de los resultados de las pruebas realizadas en la construcción del software.

Entre estas herramientas tenemos: NUnit, JUnit, MbUnit, PHPUnit y otros.

Herramientas para la generación de la documentación

Permite generar la documentación rápidamente a lo largo del desarrollo del software. Lo interesante es que siempre vamos a tener información actualizada y en tiempo real.

Entre estas herramientas tenemos: Javadoc, Doxygen y otros.

Herramientas para la publicación de artefactos (entregables)

Permite generar en forma automatizada la publicación del desarrollo realizado.

Entre estas herramientas tenemos: Maven y otros.

Herramientas para inspección del código

Permite generar evaluar la calidad del desarrollo realizado.

Entre estas herramientas tenemos: Sonar y otros.

Herramientas para ejecución e integración del código

Permite la integración continua del desarrollo de software.

Entre estas herramientas tenemos: Hudson y otros.

Notificaciones

La retroalimentación es importante para alcanzar dinamismo que brinda la integración continua al desarrollo y mantenimiento de software. Por lo tanto, esta herramienta mantiene informados a los desarrolladores sobre el resultado de la construcción.

Los servicios de notificación que comúnmente se utilizan son: Blog, Correo electrónico, mensajería instantánea, RSS, twitter y otros.

Integración a los entornos de desarrollo de integrado (IDE)

Los servidores de Integración comúnmente brindan una serie de accesorios para que el desarrollador pueda visualizar el estado de las aplicaciones a través de su herramienta de desarrollo.

Entre los IDE más conocidos tenemos: Eclipse, Visual Studio, NetBeans y otros.

Seguridad

Permite controlar quien está autorizado para cambiar la configuración del Servidor de Integración Continua y de cada uno de sus componentes.

Para ello el Servidor de Integración Continua nos debe permitir la creación de usuarios, creación de grupos y roles, configurar seguridad por usuario y otros.

IV. Implementación del modelo de integración continua

Las herramientas que se van utilizaron para el desarrollo del modelo planteado son aquellos que tienen mayor uso por la comunidad de software libre. Estas herramientas son:

- Repositorio de control de versiones: Subversion.
- Herramientas para realizar pruebas: JUnit.
- Servidor de Inspección de código: Sonar.
- Servidor de Integración Continua: Hudson.
- Servidor de Repositorio de Artefactos: Nexus.
- Herramientas para inspección de código: PDM.

Debido a que la forma en que implementar estas herramientas se encuentran en la bibliografía presentada, se procederá a mostrar lo implementado.

Subversión

Es una herramienta de control de versiones que administra el acceso a los archivos y mantiene un historial de los mismos.



Figura 2: Repositorio de control de versiones

Hudson

Es una herramienta software libre que nos ayuda a monitorizar la ejecución de la construcción constante del software.

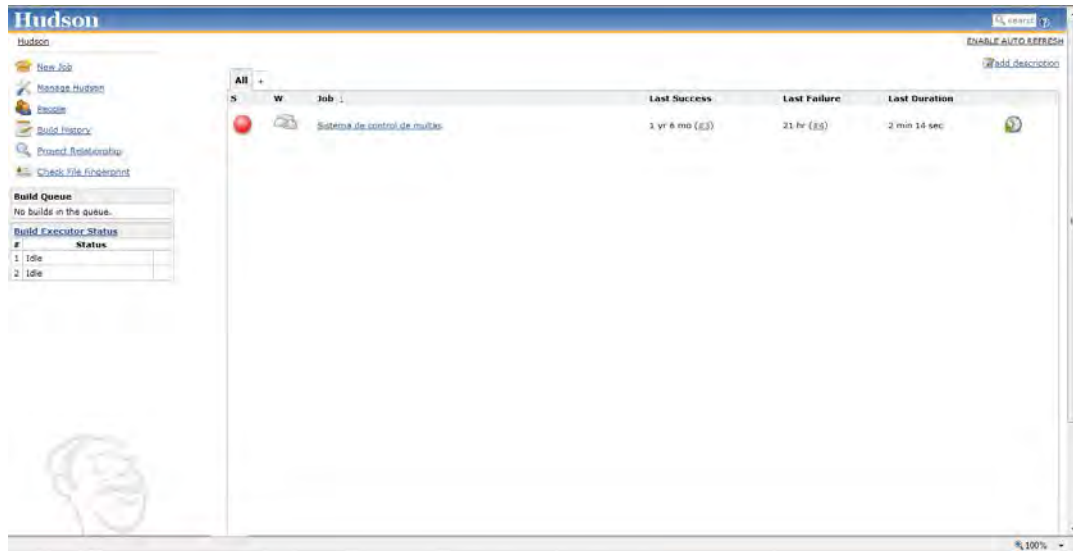


Figura 3: Panel de principal del Hudson

Sonar

Es una herramienta software libre que nos permite la verificación de la calidad del código, calculando y mostrando en formato web las métricas de nuestro proyecto de software.

La misma aplicación nos muestra información en Sonar como líneas de código, cantidad de clases, cobertura de comentarios, duplicidad, cobertura de código (pruebas) y otros.

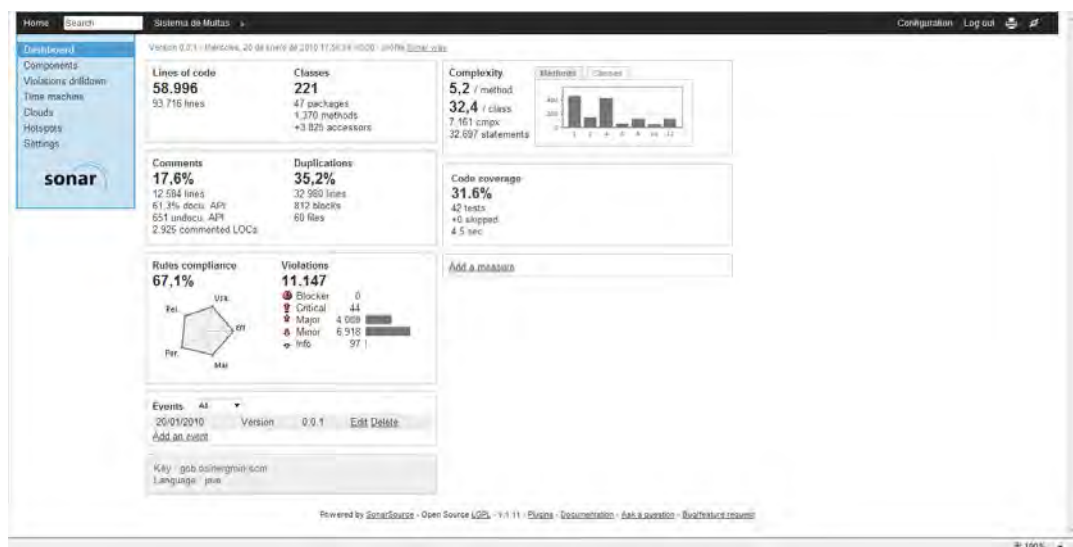


Figura 4: Panel de control del sonar

En otra viñeta Sonar nos muestra los puntos a atacar para poder mejorar la calidad del software.

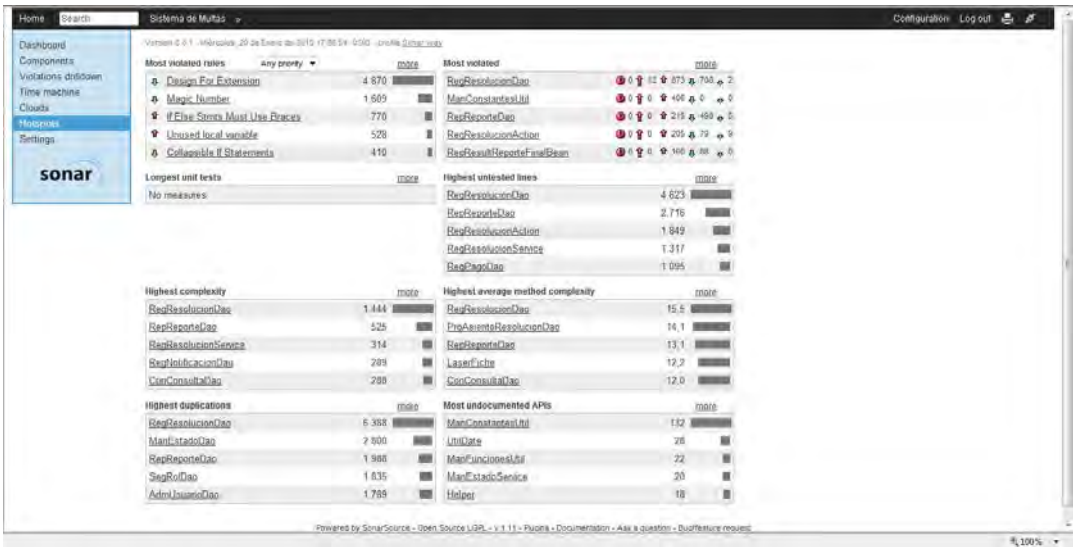


Figura 5: Panel de ajustes necesarios en la aplicación

Nexus

Es el repositorio propio de artefactos o librerías que se utilizarán en el desarrollo del software.

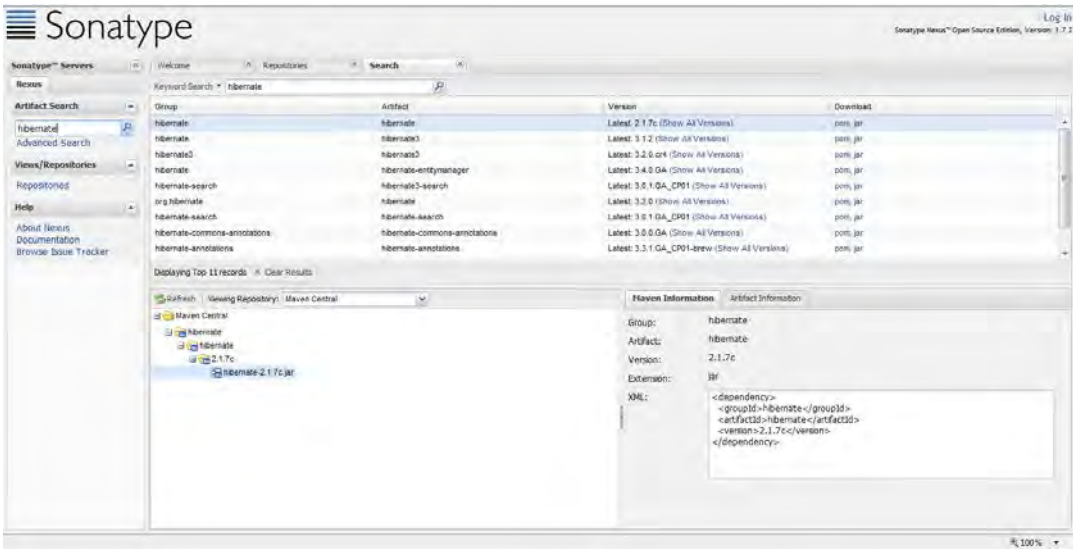


Figura 6: Panel de principal de Nexus – búsqueda de librerías