

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA EN SISTEMAS

**IMPACTO DE LA APLICACIÓN DE UN ENFOQUE
DE ENTREGA CONTINUA EN LA CALIDAD DEL SOFTWARE
BANCARIO**

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL GRADO
DE MAGISTER EN SOFTWARE, MENCIÓN CALIDAD**

ROMERO FERNÁNDEZ JOSÉ GUILLERMO

jose.romero03@epn.edu.ec

Directora: FLORES NARANJO PAMELA CATHERINE PhD.

pamela.flores@epn.edu.ec

QUITO, MAYO 2019

APROBACIÓN DEL DIRECTOR

Como directora del trabajo de titulación "IMPACTO DE LA APLICACIÓN DE UN ENFOQUE DE ENTREGA CONTINUA EN LA CALIDAD DEL SOFTWARE BANCARIO" desarrollado por José Guillermo Romero Fernández, estudiante de la Maestría en Software, mención en Calidad, habiendo supervisado la realización de este trabajo y realizado las correcciones correspondientes, doy por aprobada la redacción final del documento escrito para que prosiga con los trámites correspondientes a la sustentación de la Defensa oral.

Flores Naranjo Pamela Catherine, PhD.

DIRECTOR

DECLARACIÓN DE AUTORÍA

Yo, José Guillermo Romero Fernández, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normativa institucional vigente.

José Guillermo Romero Fernández

DEDICATORIA

A Kary, Ale, Mika y Guille.

A mi Madre y Padre +, herman@s, cuñad@s y sobrin@s.

AGRADECIMIENTO

A mi esposa e hij@s por su presión, amor y apoyo incondicional.

A mi Madre por ser un ejemplo de lucha y trabajo, y enseñarme a siempre seguir adelante.

A mi Padre que fue un ejemplo de honestidad.

A mis herman@s y cuñad@s por demostrarme su fortaleza ante cualquier adversidad.

A mis sobrin@s en especial Sebastián que es un ejemplo de dedicación y perseverancia.

A mi tutora Dra. Pame, que supo guiarme, y tuvo mucha paciencia.

A Christian por su apoyo y guía.

A mis compañeros de aulas y de trabajo.

A tod@s l@s que de alguna forma me han apoyado.

ÍNDICE DE CONTENIDO

Lista de Figuras	i
Lista de Tablas	ii
RESUMEN.....	iii
<i>ABSTRACT</i>	iv
1. INTRODUCCIÓN	1
1.1 Antecedentes de la organización.....	3
1.2 Planteamiento del Problema	3
1.3 Objetivo General.....	5
1.4 Objetivos Específicos	5
2. MARCO TEÓRICO	6
2.1 Impacto	6
2.2 Evaluación del Impacto	6
2.3 Entrega Continua	7
2.3.1 Exploración Continua.....	8
a) Actividades de exploración continua.....	8
b) Técnica para cumplir con las actividades de exploración continua	9
2.3.2 Integración Continua.....	10
a) Niveles de Integración Continua	11
2.3.3 Despliegue Continuo	14
3. REVISIÓN DE LITERATURA.....	17
3.1 Trabajos relacionados a Evaluación del Impacto.....	17
3.2 Trabajos relacionados a Entrega Continua.....	17
3.3 Trabajos relacionados a Estudio de Caso.....	18
4. METODOLOGÍA.....	20
4.1 Estudio de Caso	20
4.2 Diseño del Estudio de Caso	21
5. IMPLEMENTACIÓN DEL ESTUDIO DE CASO	26
5.1 Definición del Entorno	26
5.2 Implementar Entrega Continua	28
5.2.1 Design Thinking.....	28

5.2.2	Integración continua.....	30
5.2.3	Despliegue de funcionalidades	31
5.3	Recoger información.....	32
5.3.1	Punto de vista del Cliente	32
5.3.2	Punto de vista del Desarrollo.....	33
5.3.3	Punto de vista de la Organización	33
5.4	Análisis de Resultados	34
5.4.1	Impacto desde el punto de vista del Cliente	34
5.4.2	Impacto desde el punto de vista del Desarrollo	41
5.4.3	Impacto desde el punto de vista de la Organización.....	44
5.5	Lineamientos para la evaluación de impacto de implementar entrega continua	47
6.	DISCUSIÓN	49
7.	CONCLUSIONES.....	53
8.	REFERENCIAS BIBLIOGRÁFICAS.....	55

Lista de Figuras

Figura 1. Fases de la entrega continua [8].	8
Figura 2. Actividades de la exploración continua [9].	8
Figura 3. Etapas de “Design Thinking” [15].	9
Figura 4. Niveles “Integración Continua” [16].	10
Figura 5. Dividir las características en historias de usuario [16].	11
Figura 6. Entorno automatizado de “Integración continua” [16].	11
Figura 7. Integración de pruebas automáticas [16].	12
Figura 8. Integración del trabajo de todos los equipos [16].	13
Figura 9. Integración a nivel de solución [16].	14
Figura 10. Recomendaciones para el despliegue [17].	15
Figura 11. Proceso de despliegue automático [17].	15
Figura 12. Revisión de Literatura.	19
Figura 13. Organigrama Funcional.	27
Figura 14. Tiempo de atención de requerimientos.	35
Figura 15. Disponibilidad de nuevas características.	35
Figura 16. Probabilidad de que se repitan los problemas reportados anteriormente.	36
Figura 17. Habilitar nuevas características.	36
Figura 18. Resolución de problemas reportados.	37
Figura 19. Calidad del servicio.	37
Figura 20. Frecuencia de problemas.	38
Figura 21. Valoración de opinión.	38
Figura 22. Cumplimiento de expectativas.	39
Figura 23. Contribución al mejoramiento continuo.	39
Figura 24. Nivel de expectativas.	40
Figura 25. Puntos de historia realizados por período.	42
Figura 26. Retrabajo por período.	42
Figura 27. Tiempo promedio de entrega por período.	43
Figura 28. Comparación de errores por período y fase.	44
Figura 29. Comparación del número de créditos otorgados por mes y año.	45
Figura 30. Lineamiento para la evaluación del impacto de aplicar entrega continua.	48

Lista de Tablas

Tabla 1. Puntos de historia, retrabajo y tiempo de entrega	41
Tabla 2. Número de errores por período	43
Tabla 3. Número de créditos otorgados	45

RESUMEN

Este trabajo evalúa el impacto en la calidad del software al aplicar un enfoque de entrega continua. La entrega continua consta de tres fases que son: Exploración Continua que permite determinar las necesidades del cliente más allá de los requerimientos únicamente, Integración Continua que permite la automatización de los procesos de construcción y pruebas automáticas del software y Despliegue Continuo que permite realizar el paso de nuevas funcionalidades a los diferentes ambientes como son: Desarrollo, Pruebas e incluso el de Producción, no necesariamente liberado para que se utilice inmediatamente, sino que se lo habilite en cualquier momento. El resultado de este trabajo puede servir para proveedores de software o desarrolladores internos en entidades bancarias que deseen tiempo de reacción menores, reducción del riesgo, exposición de defectos y costos, entre otros beneficios de aplicar esta técnica. Para determinar el impacto en la calidad se realizó un estudio de caso, que mide y registra la conducta de las personas involucradas en el fenómeno estudiado, número de errores, retrabajo, entrega de funcionalidades antes y después de aplicar el enfoque de Entrega Continua. El estudio de caso permitirá observar los resultados de cambios introducidos en contextos similares bajo las mismas condiciones, aplicando diferentes enfoques.

Palabras clave: Entrega Continua, Exploración Continua, Integración Continua, Despliegue Continuo.

ABSTRACT

This work evaluates the impact on software quality by applying a continuous delivery approach. The continuous delivery consists of three phases: Continuous Exploration that allows to determine the needs of the client beyond the requirements, Continuous Integration that allows the automation of builds and automatic tests, and Continuous Deployment that allows the passage of new functionalities to different environments such as: Development, Testing and even Production, not necessarily released to be used immediately, but to be enabled at any time. The result of this work is for software suppliers or internal developers in banking entities that want a shorter reaction time, risk reduction, exposure of defects and costs, among other benefits of applying this technique. To determine the impact on quality, a case study was carried out, which measures and records the behavior of the people involved in the phenomenon studied, number of errors, rework, delivery of functionalities before and after applying the Continuous Delivery approach. The case study will allow us to observe the results of changes introduced in similar contexts under the same conditions, applying different approaches.

Keywords: Continuous Delivery, Continuous Exploration, Continuous Integration, Continuous Deployment.

1. INTRODUCCIÓN

La calidad del software es “el grado con el que un sistema, componente o proceso cumple los requerimientos especificados y las necesidades o expectativas del cliente o usuario” [1], es decir es uno de los atributos más importantes para la aceptación del sistema por parte del usuario, ya que de ésta depende la satisfacción de los clientes, al verificar que el sistema cumple con las especificaciones solicitadas. Si un sistema presenta baja calidad, este puede generar pérdidas económicas y de tiempo, problemas legales e incluso peligrar la vida de las personas [2].

En el software bancario los problemas por baja calidad como el mal funcionamiento del sistema o las caídas del sistema por ataques efectuados al introducirse brechas en la seguridad en el código o virus informáticos [3], son algunos de los problemas más críticos ya que las pérdidas económicas pueden multiplicarse, y al ser software que requiere cambios de funcionalidad o nuevas funcionalidades que permiten receptor más ingresos económicos, es imprescindible que se realicen pasos de versión que incluyan los cambios solicitados con mayor frecuencia, pero que provoquen la menor cantidad de errores posibles.

Por lo tanto, se evidencia que la satisfacción de los usuarios de estos sistemas es baja, ya que no sienten que el proveedor se preocupe por sus necesidades.

El desarrollo de software ágil es el desarrollo de software que reconoce la naturaleza empírica del software y está preparado para asumir cambios de manera rápida a partir de la retroalimentación de los usuarios [4]. El desarrollo ágil tiene su base en el manifiesto ágil en el cual se indica que se valora [5]:

- **Personas e interacciones entre ellas** sobre procesos y herramientas.
- **Software funcional** sobre documentación extensa.
- **Colaboración con el cliente** sobre negociación contractual.
- **Respuesta a cambios** sobre seguir un plan.

El desarrollo Lean es una metodología que intenta eliminar todo el desperdicio cuando se realiza algo, es decir todo lo que no genera valor [6]. En el desarrollo Lean se hace referencia a siete principios que son [7]:

1. Eliminar el desperdicio
2. Ampliar el conocimiento

3. Decidir lo más tarde posible
4. Entregar valor lo más pronto posible
5. Empoderar/Potenciar al equipo
6. Crear la integridad
7. Visualizar el todo

La “*Entrega Continua*” es el proceso de explorar continuamente el mercado y las necesidades del usuario, para definir un conjunto de pasos a seguir para satisfacer esas necesidades [8, 9].

Los beneficios de la entrega continua son: tiempo de reacción menores, reducción del riesgo, exposición de defectos y costos, entre otros [10].

La entrega continua consta de tres fases que son: Exploración Continua que permite determinar las necesidades del cliente más allá de los requerimientos únicamente, Integración Continua que permite la automatización de los procesos de construcción y pruebas automáticas del software y Despliegue Continuo que permite realizar el paso de versiones automáticamente en los diferentes ambientes incluso el de Producción, no necesariamente liberado, y que se lo habilite en cualquier momento. Juntas, las tres dimensiones Exploración Continua, Integración Continua y Despliegue Continuo bajo demanda proveen una estrategia Lean y Agile integrada para acelerar la entrega de valor al cliente [8].

La fortaleza de este estudio de caso es que a través del mismo se mida y registre la conducta de las personas involucradas en el fenómeno estudiado [11]. La comparación de datos dentro del estudio de caso nos permitirá observar los resultados de cambios introducidos en contextos similares bajo las mismas condiciones, aplicando diferentes enfoques [11, 12].

El presente trabajo contribuye al estado del arte con una solución para la mejora de la calidad de software bancario. Los resultados de este estudio pueden ser utilizados por proveedores de software o desarrolladores internos en entidades bancarias de similares características.

Este trabajo está enfocado a una institución del sector financiero del Ecuador, en este caso es una Cooperativa de Ahorro y Crédito de la ciudad de Quito.

1.1 Antecedentes de la organización

La cooperativa en la que se realiza el estudio es una institución líder en el sector cooperativo nacional, sólida, solvente y de confianza. Reconocida por sus socios, clientes y la comunidad, regulada por la Superintendencia de Economía Popular y Solidaria (SEPS), con casi medio siglo de experiencia en el país.

Administra el dinero de más de 200 mil socios para brindar productos y servicios financieros que faciliten su vida. Su gestión transparente, confiable y segura les ha otorgado la calificación de riesgo de “AA+”, la mejor del sector cooperativo.

Pertenece al segmento 1, que agrupa a las cooperativas más grandes del país y cuenta con 25 agencias a nivel nacional.

En esta cooperativa su compromiso es con la comunidad y cree firmemente en la retribución a ella, razón por la cual apoya a los sectores vulnerables de la sociedad a través programas y proyectos de responsabilidad social basados en tres líneas de acción: Educación, Salud y Bienestar.

Más allá de brindar productos y servicios financieros de calidad, creen que la razón de ser de esta institución es apoyar el desarrollo sostenible de sus socios, clientes, y de la comunidad en general.

1.2 Planteamiento del Problema

El software bancario es de misión crítica para el sector financiero y al requerir cambios continuos, es necesario que el proceso de desarrollo sea de calidad, es decir que, al liberar una nueva funcionalidad, ésta cumpla con las expectativas del cliente, dicho de otra manera, que no tenga errores, que tenga tiempo de respuestas óptimos, que cumpla con su propósito, etc. Actualmente con el proceso tradicional de cambios se evidencia que existe insatisfacción por parte del cliente, ya que actualmente si se solicita una nueva funcionalidad, ésta se demora en estar en funcionamiento y por lo general tiene defectos de calidad.

Es por esta razón que surge la pregunta, *¿Cuál es el impacto de introducir el enfoque de entrega continua en el proceso de desarrollo de software con respecto al usuario, organización y desde el punto de vista del desarrollador?*

Para responder esta pregunta se utilizará un estudio de caso que es una forma esencial de investigación en ciencias sociales, desarrollo tecnológico e investigaciones sobre

problemas sociales, el mismo que comparará el proceso actual de paso de correcciones o nuevas funcionalidades a Producción versus la aplicación del enfoque de entrega continua, y además, aplicando la técnica de la encuesta que nos permita determinar el grado de satisfacción del cliente [12].

Para mejorar la calidad del software y la percepción que el cliente tiene con respecto al producto y al proveedor se debe realizar una exploración continua de sus necesidades, integración continua del código y mejorar el paso de versiones al ambiente de producción mediante la implementación de entrega continua que es un enfoque de desarrollo de software que permite construir el software correcto de tal manera que se lo pueda liberar a Producción en cualquier momento.

Entre los problemas que se pueden mencionar al poner nuevas funcionalidades en producción, se encuentra muchos errores, el tiempo desde que se solicita la funcionalidad hasta que ésta está en funcionamiento es muy alto, si existen problemas con alguna funcionalidad se toma mucho tiempo hasta resolver el problema y pasar nuevamente la corrección, es decir, no se pueden deshabilitar funcionalidades, etc.

No existen estudios en el Ecuador que demuestren el impacto de la entrega continua en la calidad de software bancario. El presente trabajo de titulación pretende estudiar este aspecto en un entorno real.

1.3 Objetivo General

Estudiar el impacto que produce la aplicación del enfoque de entrega continua en la mejora de la calidad en el software mediante un estudio de caso en el sector bancario que nos permita determinar el grado de satisfacción del cliente.

1.4 Objetivos Específicos

- Identificar cuáles son las diferentes propuestas para implementar Exploración Continua.
- Determinar las métricas que permiten estimar la percepción del cliente acerca de la calidad del software.
- Determinar técnicas para automatizar el proceso de integración continua
- Determinar que técnicas permiten desplegar funcionalidades de manera continua sin que sean liberadas.
- Realizar un estudio de caso en una entidad del sector bancario.

2. MARCO TEÓRICO

En este capítulo se presentan conceptos sobre Impacto, Evaluación del Impacto, Estudio de Caso, Entrega Continua y las etapas que lo comprenden.

2.1 Impacto

Al impacto se lo define como un “conjunto de los efectos que un suceso o un hecho producen en su entorno físico o social” [13].

El impacto no puede ser tomado como una sola medida, sino que comprende un conjunto de dimensiones. La lista que se presenta a continuación, viene de una serie de estudios, acerca del desarrollo social y económico, realizados en Europa [14].

- Competitividad y explotación / efectos relacionados con el mercado (por ejemplo, productos, prototipos, patentes, procesos, transferencia de tecnología).
- Efectos de aprendizaje individuales y organizacionales (por ejemplo, nuevas colaboraciones, redes) y cambios de comportamiento (por ejemplo, mayores habilidades de gestión / colaboración)
- Impacto en normas / estándares
- Efectos estructurales (por ejemplo, tamaño de la empresa, estructura organizacional, nuevas empresas, confianza y cárteles)
- Efectos sociales (por ejemplo, contribución a las infraestructuras, sistemas y redes externas, efectos externos de aprendizaje, impacto en el medioambiente, salud, empleo, desarrollo regional, etc.)
- Contribución a la base de conocimiento, progreso científico, instituciones académicas y capital humano.

2.2 Evaluación del Impacto

La evaluación de impacto debe centrarse en las dimensiones más relevantes para este estudio de caso, como son los efectos de aprendizaje organizacional y efectos sociales directamente, y de una manera indirecta la competitividad y efectos relacionados con el mercado.

Existen métodos para evaluar el impacto actualmente, los mismos están categorizados en tres categorías: los que proveen un marco de trabajo para la evaluación, aquellos interesados en la recolección de datos, y aquellos interesados en el análisis de datos [14].

Los métodos basados en marco de trabajo generalmente involucran comparación, por ejemplo, comparación antes y después, y comparación con grupos fuera del esquema siendo evaluado.

Los métodos de recolección de datos incluyen entrevistas, encuestas y compilación de estadísticas.

La tercera categoría de métodos para el análisis de datos incluye estudios de caso, modelos econométricos y procesamiento de estadísticas para construir indicadores incluyendo bibliometría.

Para este estudio se realizó una combinación de los métodos basados en un marco de trabajo y los de recolección de datos. Los basados en un marco de trabajo fueron considerados para abarcar la dimensión de los efectos sociales y, por otro lado, los de recolección de datos para las dimensiones de aprendizaje organizacional y la competitividad y efectos relacionados con el mercado.

2.3 Entrega Continua

La entrega continua es un conjunto de procesos, flujos de trabajo, actividades para explorar continuamente el mercado y las necesidades del usuario, con el fin de liberar continuamente valor para cubrir dichas necesidades [8].

Los beneficios de la entrega continua son: tiempo de reacción menores, reducción del riesgo, los costos y la exposición a defectos, entre otros [10].

La entrega continua consta de tres fases, como se muestra en la Figura 1, que son: **Exploración Continua** que permite determinar las necesidades del cliente más allá de los requerimientos únicamente, **Integración Continua** que permite la automatización de los procesos de construcción y pruebas automáticas del software y **Despliegue Continuo** que permite realizar el paso de versiones automáticamente en los diferentes ambientes incluso el de Producción, no necesariamente liberado, y que se lo habilite en cualquier momento [8].

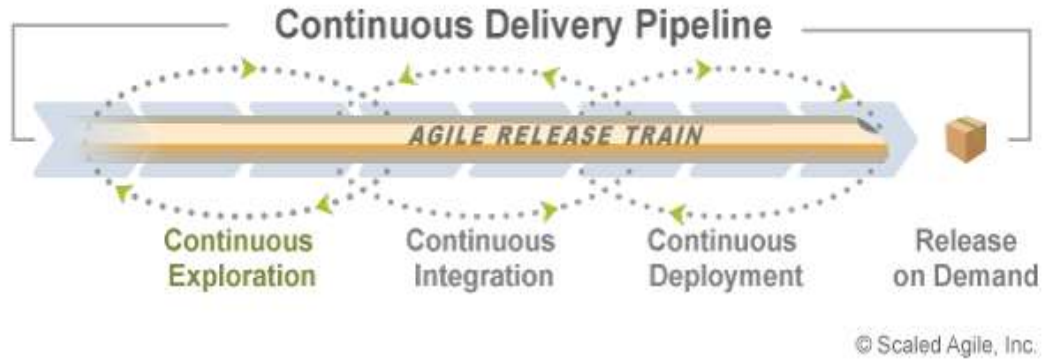


Figura 1. Fases de la entrega continua [8].

2.3.1 Exploración Continua

La exploración continua es el proceso de explorar continuamente el mercado y las necesidades del usuario, para definir un conjunto de características que permitan satisfacer esas necesidades [9].

a) Actividades de exploración continua

En el proceso de Exploración Continua se realizan tres actividades, como se muestra en la Figura 2, que son: **Colaboración**, **Búsqueda** y **Síntesis**.

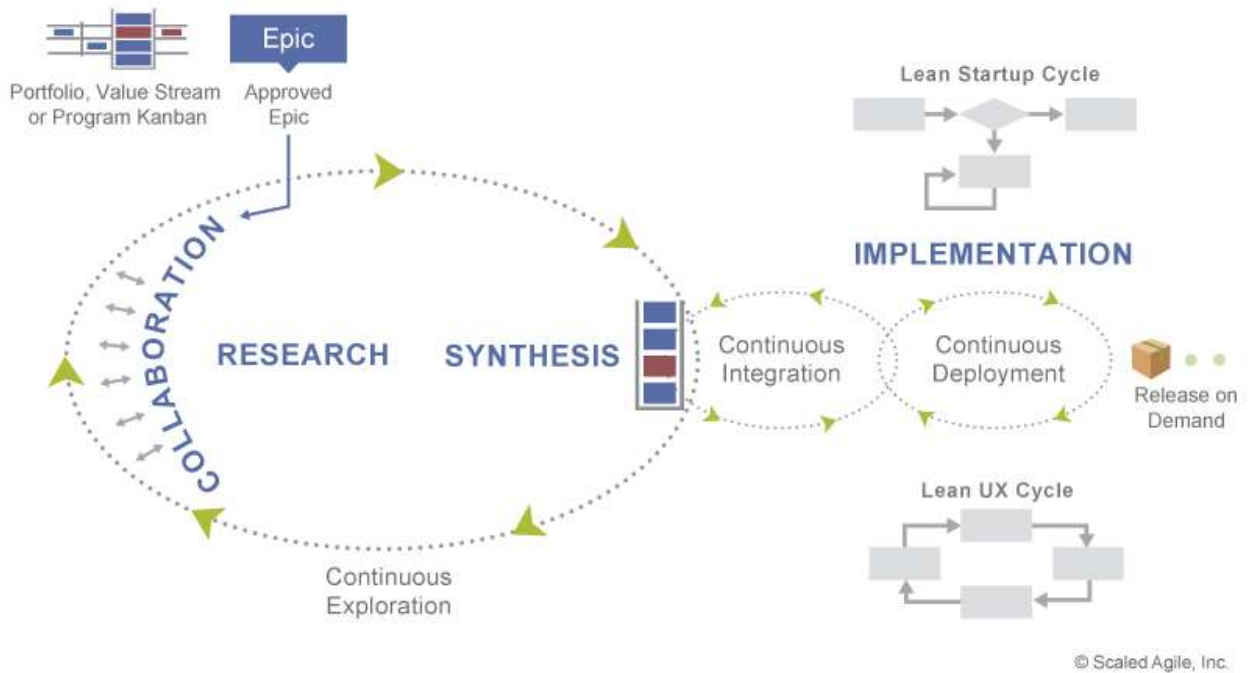


Figura 2. Actividades de la exploración continua [9].

- **Colaboración**

Procesos que involucran la participación de los interesados para crear una visión común

- **Búsqueda**

Actividades y técnicas que permiten establecer la visión.

- **Síntesis**

Definir la visión, el backlog necesario para cumplir con las necesidades identificadas.

b) Técnica para cumplir con las actividades de exploración continua

- **Design Thinking**

Para cumplir con estas actividades, se ha utilizado las técnicas de “*Design Thinking*” [15] que es una metodología para generar ideas innovadoras, que centra su eficacia en entender y dar solución a las necesidades reales de los usuarios.

Se compone de cinco etapas, como se muestra en la Figura 3, que son: **Empatizar, Definir, Idear, Prototipar, Testear.**

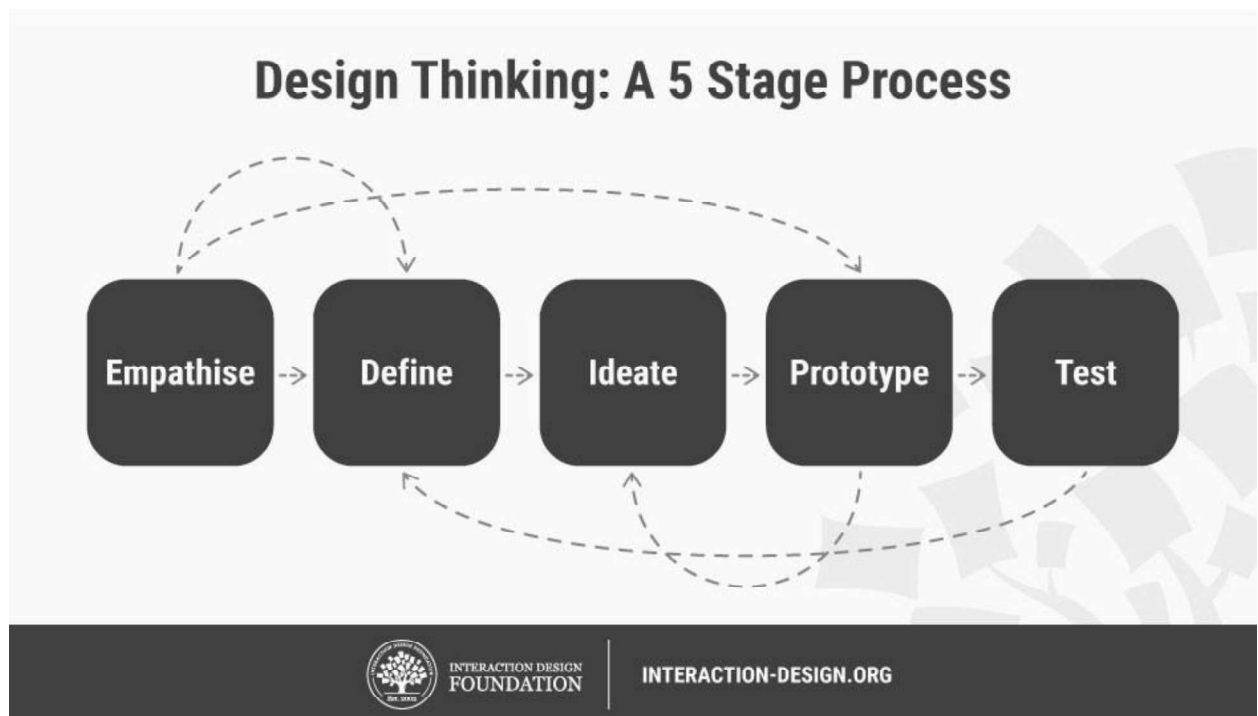


Figura 3. Etapas de “Design Thinking” [15].

- **Empatizar**

Ganar un entendimiento claro del problema que se está intentando resolver. Permite apartar las creencias propias para centrarse en el punto de vista de los interesados.

- **Definir**

Recoger la información generada en la etapa “Empatizar” para sintetizarla y así determinar que hacer.

- **Idear**

Generar ideas a partir de las necesidades de los interesados.

- **Prototipar**

Implementar soluciones a las necesidades de los usuarios.

- **Testear**

Pruebas rigurosas de las soluciones prototipadas.

2.3.2 Integración Continua

La integración continua, que es la segunda parte en la entrega continua, es el proceso de tomar características y desarrollarlas, probarlas, integrarlas y validarlas en un entorno controlado para luego permitir su despliegue [16].

Para la Integración continua se ha utilizado una estrategia de integración de soluciones de tres niveles como se muestra en la Figura 4.

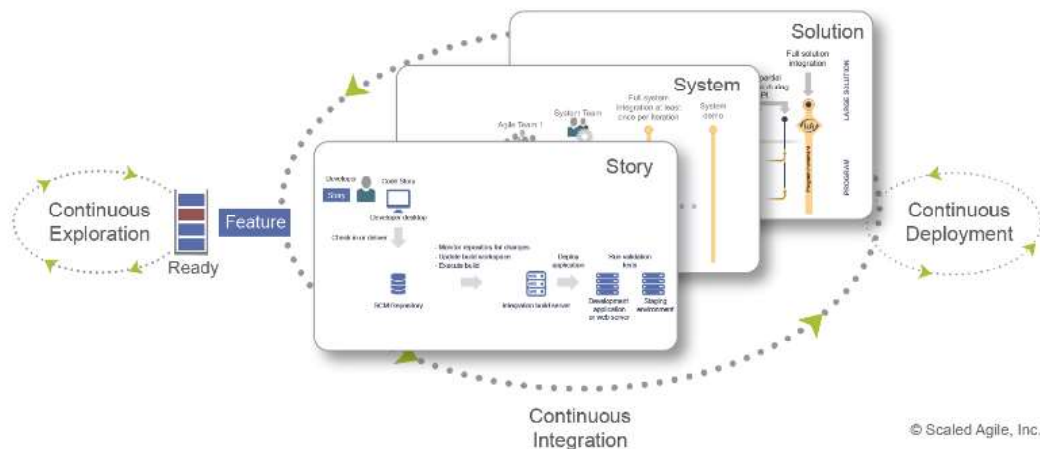


Figura 4. Niveles “Integración Continua” [16].

a) Niveles de Integración Continua

- Integración y pruebas de historias

Dividir las características en historias de usuario, como se muestra en la Figura 5, para diseñarlas, implementarlas, integrarlas, probarlas y validarlas en un entorno intermedio.

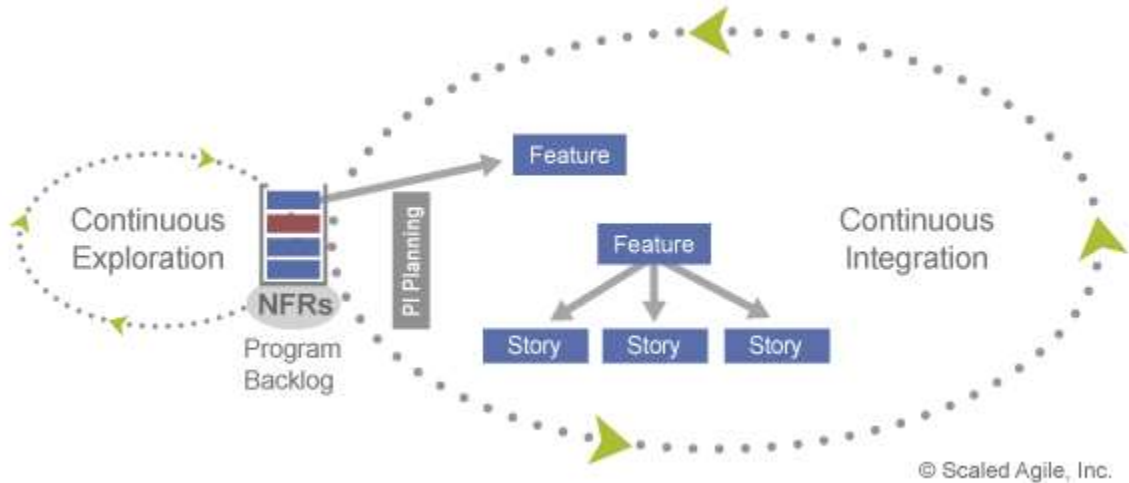


Figura 5. Dividir las características en historias de usuario [16].

Una vez que se cuenta con las historias de usuario estas se codifican, prueban e integran a la línea base. Cada miembro del equipo integra su trabajo individual frecuentemente, automatizando la integración continua, como se muestra en la Figura 6.

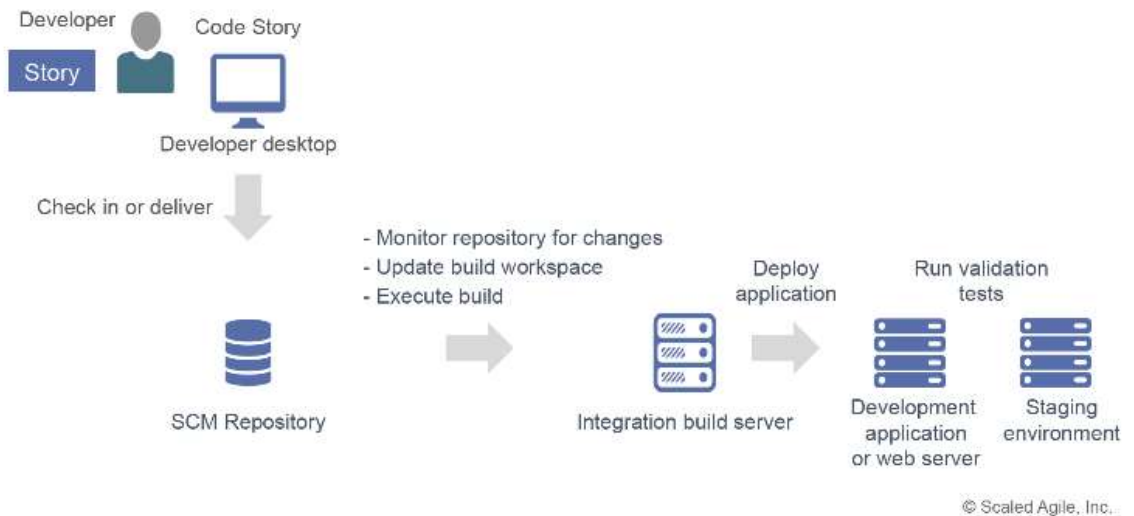


Figura 6. Entorno automatizado de "Integración continua" [16].

Para que las nuevas funcionalidades se integren de manera correcta con las anteriores es necesario que se ejecuten pruebas automáticas, al momento de la construcción del código. Las pruebas deben crearse para cada funcionalidad nueva, como se muestra en la Figura 7.

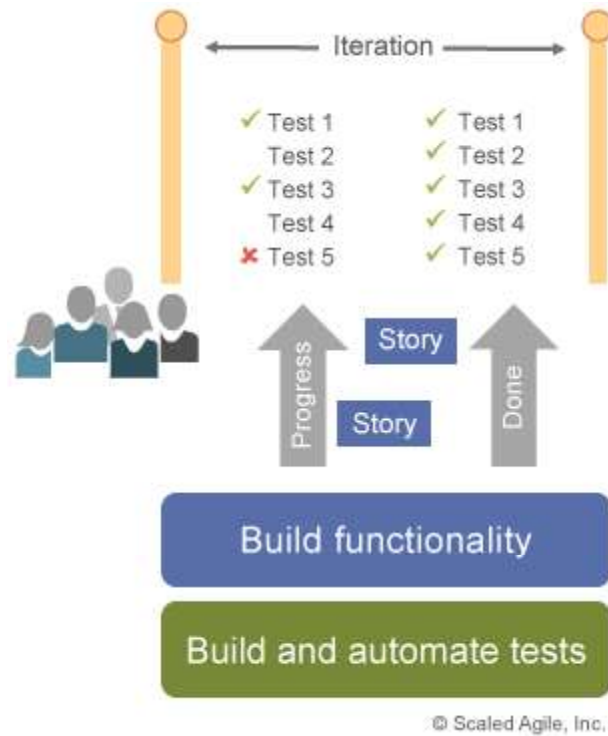


Figura 7. Integración de pruebas automáticas [16].

- Integración y pruebas de sistema

Para conseguir probar completamente las nuevas funcionalidades es necesario realizar las pruebas a nivel de sistema. Es decir, que todos los equipos integren sus nuevas funcionalidades y el sistema se comporte como fue definido, tal como se muestra en la Figura 8.

- ▶ Avoid physical branching for software
- ▶ Frequently integrate hardware branches

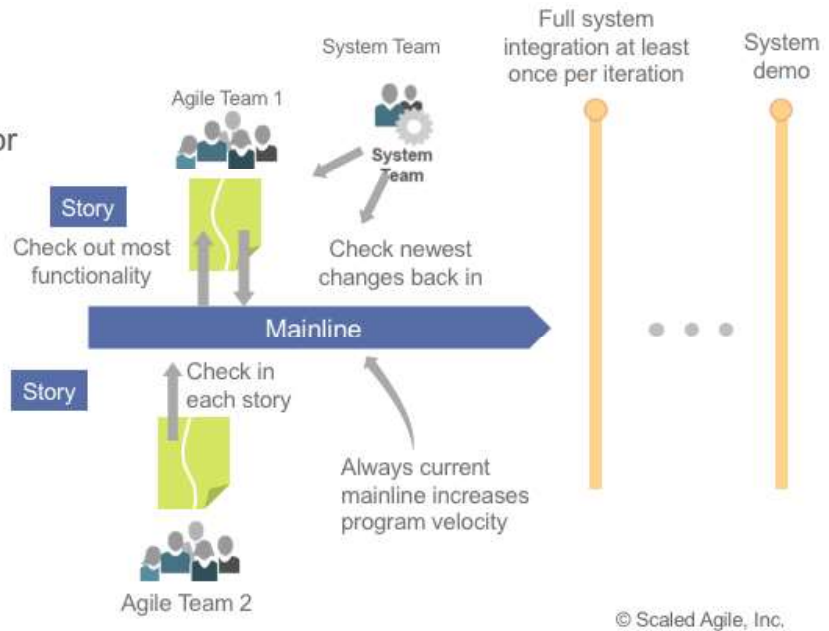


Figura 8. Integración del trabajo de todos los equipos [16].

- Integración y pruebas de la solución

Finalmente, se debe realizar las pruebas a nivel de la solución para validar todo el trabajo conjunto con las funcionalidades existentes a lo largo del Incremento del programa (PI) como se muestra en la Figura 9.

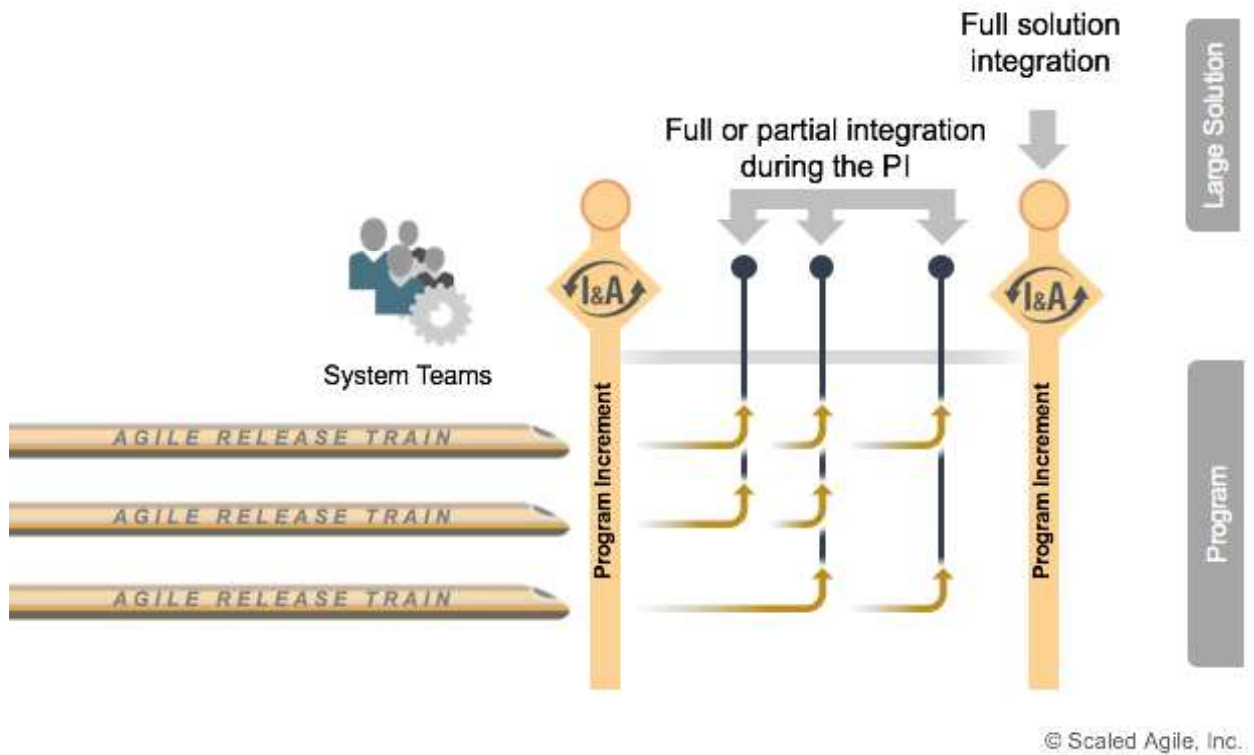


Figura 9. Integración a nivel de solución [16].

2.3.3 Despliegue Continuo

El Despliegue continuo es el proceso de tomar las nuevas características validadas desde la Integración Continua y desplegarlas en el ambiente de producción donde pueden ser liberadas y habilitadas bajo demanda [17].

El objetivo es ofrecer soluciones cada vez más valiosas a los usuarios finales con la mayor frecuencia posible. Para conseguirlo se recomienda que los equipos de desarrollo y operaciones se centren en el proceso de despliegue tomando en cuenta las recomendaciones que se muestran en la Figura 10.

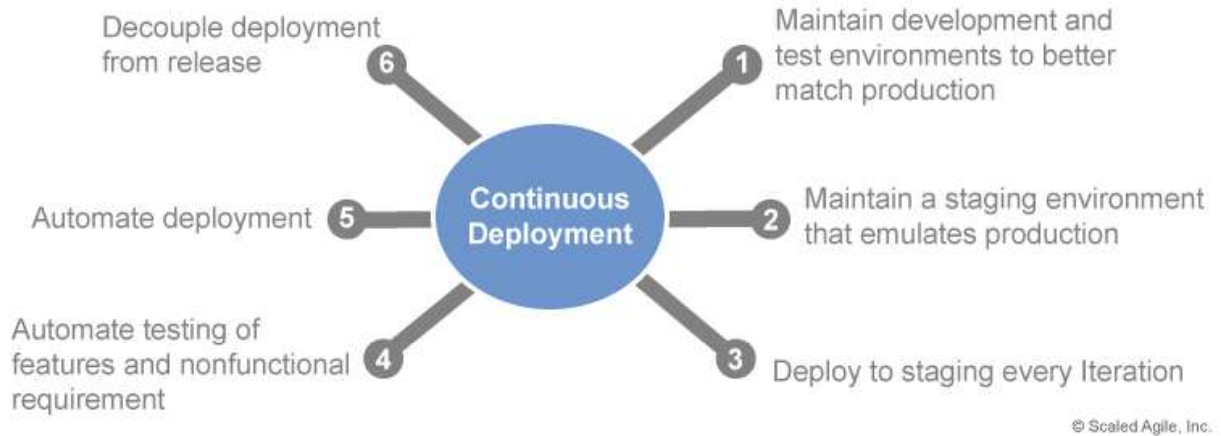


Figura 10. Recomendaciones para el despliegue [17].

- Mantener los ambientes de desarrollo y pruebas lo más parecido posible al ambiente de producción.
- Mantener un ambiente que emule al de Producción.
- Desplegar al ambiente de pruebas cada vez que sea posible.
- Automatizar las pruebas de características y requerimientos no funcionales
- Automatizar el Despliegue.

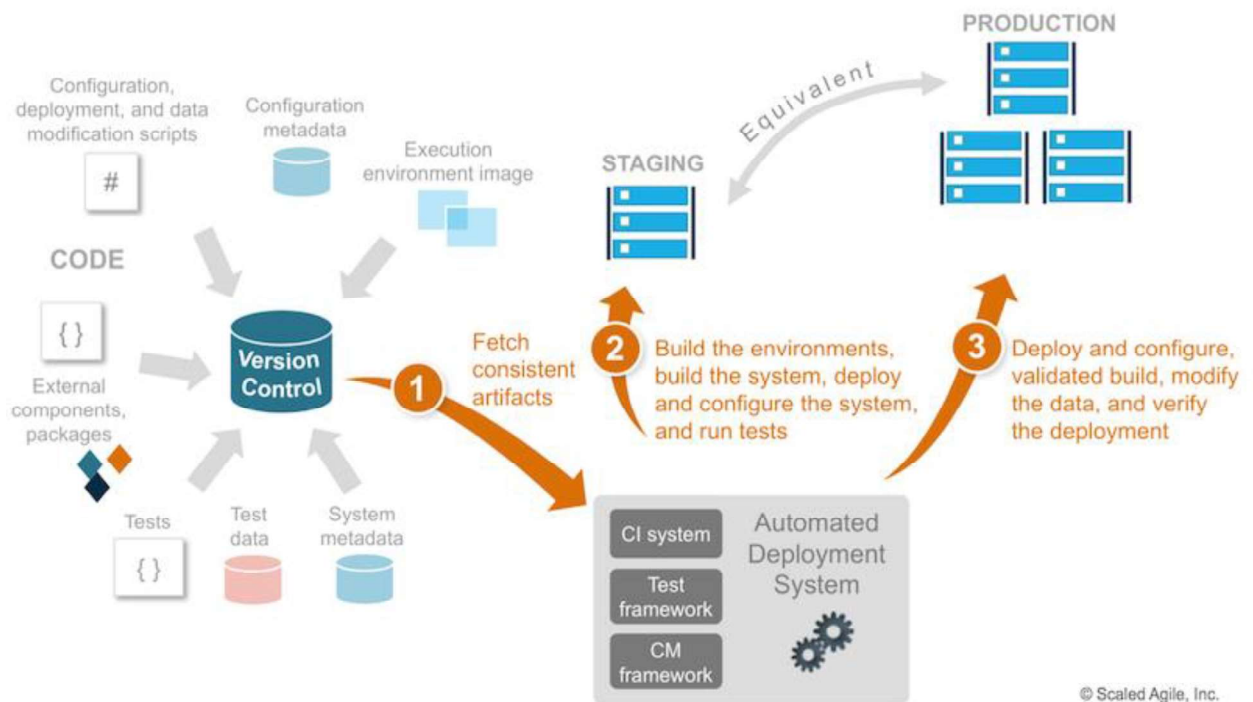


Figura 11. Proceso de despliegue automático [17].

- Desacoplar el Despliegue de la Liberación

Para el paso de nuevas características al ambiente de producción sin que estas se habiliten para ser usadas por los usuarios finales se hace uso de “Feature Toggles” o también conocidos como banderas de características, que son técnicas poderosas que permiten a los equipos modificar el comportamiento del sistema en producción sin la necesidad de modificar el código [18].

3. REVISIÓN DE LITERATURA

En este capítulo se expone la revisión de literatura, en la cual se hizo un análisis de diferentes trabajos encontrados al buscar “Entrega Continua”, “Evaluación de Impacto” y “Estudio de caso”.

Estudios sobre el impacto que produce aplicar entrega continua en el desarrollo de software no existen, por lo tanto, para realizar el análisis de literatura se divide el estudio en evaluación del impacto, entrega continua y estudio de caso.

3.1 Trabajos relacionados a Evaluación del Impacto

De la literatura revisada se puede determinar que para medir el impacto varios trabajos han realizado de tipo cuantitativo y cualitativo, tomando en cuenta indicadores que ayuden a determinar los cambios experimentados después de aplicar algún proceso. Para medir de forma cuantitativa el impacto en el desarrollo de software se pueden utilizar indicadores como el tiempo [14, 19-22], número de errores [19, 20, 22], números de pruebas pasadas [19, 21] y tamaño del equipo [21]. Los trabajos centrados en analizar el impacto de manera cualitativa se han apoyado en referenciar indicadores como ambiente de trabajo [14, 19], calidad en forma general [14, 19-22] y satisfacción o efectos sociales [14, 19, 20, 22] en donde se trata de determinar el impacto en la sociedad por medio de entrevistas o encuestas que permitan conocer de mejor manera si se consiguieron los resultados esperados, efectos estructurales o contribución a la base de conocimiento.

En el trabajo presentado se consideraron los indicadores de calidad y satisfacción del cliente para medir el impacto de forma cualitativa y los indicadores como el tiempo y número de errores para medir de forma cuantitativa.

3.2 Trabajos relacionados a Entrega Continua

Otros trabajos se han encontrado que hacen referencia a cómo adoptar la entrega continua, los beneficios de utilizar entrega continua y también los problemas y desafíos. Entrega continua es un nuevo enfoque en el desarrollo de software el mismo que promete mejorar en los tiempos para entregar valor al cliente y así mejorar la percepción que este tiene con respecto al desarrollo de software. Entre los beneficios que se han documentado al aplicar entrega continua se analiza de forma cualitativa el “Ambiente de trabajo” [23-25], la “calidad en general” [23-28], la “satisfacción del cliente” [23-25, 28]; y de forma cuantitativa el “tiempo de desarrollo” [23-28], los “errores o fallas” [24, 25, 28]

y las mejoras en las “pruebas” [24, 25]. Entre la literatura revisada se encontró que los desafíos o problemas más comunes al implementar este enfoque es el cambio en la cultura de la gente y el diseño de la aplicación [24-28].

3.3 Trabajos relacionados a Estudio de Caso

Los trabajos revisados aplican métodos como lo es un estudio de caso que permite realizar una comparación de resultados al introducir cambios en contextos similares bajo las mismas condiciones.

Para la elaboración del estudio de caso se tiene los siguientes pasos [12, 28]:

1. Diseño del estudio de caso.
2. Preparación para la recolección de información.
3. Recopilación de la información.
4. Análisis de la información.
5. Informe de resultados.

Para realizar la evaluación del impacto existen guías que se pueden seguir, como por ejemplo el “estudio de caso” [14, 19, 21, 23-26, 28], “Matriz de marco de trabajo” [14, 20, 22], “Comparación de datos” [14, 19, 21, 28] y “Revisión de literatura” [27, 28].

A continuación, se muestra la Figura 12, donde se expone una tabla que resume la literatura encontrada, diferenciándola entre aquellos estudios que se centraron en análisis cualitativo o análisis cuantitativo. Adicionalmente, se hace explícito las diferentes metodologías que se han usado para estudiar temas relacionados. Se ha decidido etiquetar al actual estudio bajo el apellido del autor “ROMERO”, con el fin de dar a conocer en qué parte del estado del arte aporta esta tesis.

		Análisis						
		Cualitativo			Cuantitativo			
		Ambiente	Calidad	Satisfacción	Tiempo	Errores	Pruebas	Tamaño equipo
Evaluación del impacto	Vafaie [19]	Vafaie [19]	Vafaie [19]	Vafaie [19]	Vafaie [19]	Vafaie [19]	Vafaie [19]	
		Rodríguez [20]	Rodríguez [20]	Rodríguez [20]	Rodríguez [20]	Rodríguez [20]		
		Mann [21]	Mann [21]	Mann [21]	Mann [21]		Mann [21]	Mann [21]
		Mendoza [22]			Mendoza [22]	Mendoza [22]		
		Ormala [14]	Ormala [14]	Ormala [14]	Ormala [14]			
		Lehtonen [23]	Lehtonen [23]	Lehtonen [23]	Lehtonen [23]			
		Chen [24]	Chen [24]	Chen [24]	Chen [24]	Chen [24]	Chen [24]	
		Chen [25]	Chen [25]	Chen [25]	Chen [25]	Chen [25]	Chen [25]	
		Laukkanen [26]			Laukkanen [26]			
		Laukkanen [27]			Laukkanen [27]			
	Claps [28]			Claps [28]				
		ROMERO	ROMERO	ROMERO	ROMERO			

Entrega continua	Vafaie [19]	Lehtonen [23]	Chen [24]	Chen [25]	Laukkanen [26]	Laukkanen [27]	Claps [28]	ROMERO
------------------	-------------	---------------	-----------	-----------	----------------	----------------	------------	--------

Metodología	Caso de estudio	Vafaie [19]	Mann [21]	Lehtonen [23]	Chen [24]	Laukkanen [26]	Claps [28]	ROMERO
	Matriz de Marco (Framework)	Rodríguez [20]	Mendoza [22]	Ormala [14]				
	Comparación de datos	Vafaie [19]	Mann [21]	Ormala [14]				
	Revisión de literatura	Laukkanen [26]	Laukkanen [27]					

Figura 12. Revisión de Literatura

El objetivo de este trabajo se centra en evaluar el impacto de aplicar un enfoque de entrega continua en el proceso de desarrollo, en la literatura se han encontrado trabajos que se enfocan en medir el impacto basados en indicadores cualitativos, como el aspecto social, y cuantitativos, como el tiempo, considerando la metodología para realizar el análisis, mientras que otros trabajos aplican métricas para medir el desempeño con entrega continua. Para este estudio se consideran tres aristas para determinar el impacto de un desarrollo de software con entrega continua, la primera arista es el impacto en la satisfacción del cliente, la segunda arista es desde el punto de vista del desarrollo, y la tercera es el impacto en la organización. Para llevar a cabo este estudio se realiza un estudio de caso para determinar el impacto según las personas que tienen que ver directa o indirectamente con el proyecto en el cual se va a implementar la entrega continua (punto de vista del Cliente), además se analizará de forma cuantitativa los efectos de aplicar este enfoque en el desarrollo de software analizando el retrabajo, entrega de funcionalidades y el número de errores (punto de vista del Desarrollo), y trabajo en equipo, créditos otorgados, costos del proyecto (punto de vista de la Organización).

4. METODOLOGÍA

En este capítulo se describe la metodología que se va a utilizar para realizar el estudio de caso, así como la forma en la que se va a obtener la información.

El presente estudio es una investigación descriptiva porque describe el impacto de aplicar el enfoque de entrega continua, consta de una parte aplicativa porque el aplicar el enfoque de entrega continua pretende generar conocimientos directamente a un entorno social como lo es el desarrollo de software bancario.

Para determinar cómo mejoró la percepción de la calidad se siguió una aproximación de un estudio de caso.

4.1 Estudio de Caso

Según [12], un estudio de caso es “una investigación empírica que estudia un fenómeno contemporáneo dentro de su contexto en la vida real”, este permitirá evaluar el efecto social ya que es una forma esencial de investigación en ciencias sociales, desarrollo tecnológico e investigaciones sobre problemas sociales, el mismo que comparará el proceso actual de paso de correcciones o nuevas funcionalidades a Producción versus la aplicación del enfoque de entrega continua, y realizando encuestas y/o entrevistas que permitan determinar el grado de satisfacción del cliente.

Para determinar cuándo utilizar el estudio de caso según [29] se debe considerar el tipo de pregunta que se requiere contestar, el control que se tiene sobre lo que se estudia y la “edad del problema”. El estudio de caso es válido generalmente cuando las preguntas son del tipo “cómo” o “por qué”, cuando se tiene poco control sobre el estudio y el tema es contemporáneo. Muchas de las preguntas “qué” son exploratorias y se contestan realizando encuestas o consultando bases de datos.

Según [30], el método de estudio de caso puede obtener los datos de diversas fuentes, tanto cualitativas como cuantitativas. Estas fuentes pueden ser documentos, archivos, entrevistas, encuestas, observación directa, entre otros.

El método de estudio de caso permite que a través de este se mida y registre la conducta de las personas involucradas en el fenómeno estudiado.

La comparación de datos nos permitirá observar los resultados de cambios introducidos en contextos similares bajo las mismas condiciones, aplicando diferentes enfoques [11, 12].

La encuesta según [31] es “un instrumento de la investigación de mercados que consiste en obtener información de las personas encuestadas mediante el uso de cuestionarios diseñados en forma previa para la obtención de información específica”.

El número de errores es la cantidad de defectos encontrados en software, el porcentaje de retrabajo es el porcentaje de tiempo que se dedica a realizar trabajo nuevamente sobre algo, que se supone ya estaba bien, y el número de funcionalidades, se refiere a la cantidad de características nuevas.

Para la elaboración del estudio de caso se van a seguir los siguientes pasos:

4.2 Diseño del Estudio de Caso

El diseño es el conjunto de pasos a seguir para realizar el estudio, aquí se especifica el entorno, propósito, alcance y las preguntas que se quieren contestar con el estudio, así como también las técnicas que se van a utilizar para la recolección de la información y cómo se va a realizar el análisis de la información.

- **Definir el Entorno.**

Determinar en donde y bajo qué condiciones se va a realizar el estudio.

- **Implementar Entrega Continua**

Implementar las 3 fases de entrega continua: Exploración Continua, Integración Continua y Despliegue Continuo.

- **Recoger Información**

En esta etapa se realiza la recolección de la información que ayudará a contestar las preguntas que se plantearon al diseñar el estudio, además se depuró la información recolectada.

- Medir la percepción de calidad antes y después de aplicar el enfoque de entrega continua.

- Realizar encuestas para determinar la satisfacción del cliente.
- Recolectar información del número de errores, porcentaje de retrabajo y número de funcionalidades puestas en producción, para medir el impacto desde el punto de vista del “Desarrollo”.
- Obtener información del trabajo en equipo, número de créditos otorgados y los costos en que se incurrieron para determinar el impacto desde el punto de vista de la organización.

Según el Modelo EFQM de excelencia [32] para medir la satisfacción del cliente se tienen varias medidas de percepción, de las cuales para elaborar el cuestionario y generar las preguntas se tomaron en cuenta las medidas que tienen que ver con la valoración temporal de entrega de servicio, tiempo que toma en tener nuevas características en funcionamiento, probabilidad de que un error vuelva a ocurrir, número de errores, funcionamiento del sistema, entre las más importantes.

Preguntas

Para la encuesta se seleccionaron 11 preguntas:

1. ¿Qué tan rápido es atendido su requerimiento de cambio?

Extremadamente Rápido	Muy rápido	Moderadamente rápido	Poco rápido	Nada rápido

Esta pregunta nos ayuda a determinar si al aplicar “Entrega Continua”, específicamente “Exploración Continua” la atención de los requerimientos de cambio es más rápida.

2. ¿Qué tan rápido una nueva característica está disponible?

Extremadamente Rápido	Muy rápido	Moderadamente rápido	Poco rápido	Nada rápido

Esta pregunta nos ayuda a determinar si al aplicar “Entrega Continua” la puesta en Producción de las nuevas características es más rápida (Entrega de valor).

3. ¿Cuál piensa que es la probabilidad de que se repitan los problemas reportados anteriormente?

Nada probable	Poco probable	Algo probable	Bastante probable	Extremadamente probable

Esta pregunta nos ayuda a determinar si al aplicar “Entrega Continua” disminuye los problemas reportados.

4. ¿Qué tan fácil es habilitar una nueva característica para su uso?

Extremadamente fácil	Muy fácil	Moderadamente fácil	Poco fácil	Nada fácil

Esta pregunta nos ayuda a determinar si al aplicar “Entrega Continua” es más fácil habilitar nuevas características.

5. ¿Qué tan rápido se resuelve su problema reportado?

Extremadamente Rápido	Muy rápido	Moderadamente rápido	Poco rápido	Nada rápido

Esta pregunta nos ayuda a determinar si al aplicar “Entrega Continua”, específicamente “Exploración Continua” se mejora en el tiempo de respuesta y corrección de problemas.

6. En general, ¿cómo calificaría la calidad de nuestro servicio?

Excelente	Buena	Ni buena ni mala	Mala	Malísima

Esta pregunta nos ayuda a determinar si al aplicar “Entrega Continua”, mejoró la percepción de calidad de nuestro servicio.

7. ¿Con qué frecuencia experimenta problemas con nuestro servicio?

Casi nunca	Rara vez	A veces	Frecuentemente	Casi siempre

Esta pregunta nos ayuda a determinar si al aplicar “Entrega Continua”, específicamente “Exploración Continua” se producen menos problemas.

8. En general, ¿qué tan valorada siente que su opinión es?

Extremadamente Valorada	Muy valorada	Moderadamente Valorada	Poco valorada	Nada valorada

Esta pregunta permite entender si la opinión de las personas está siendo tomada en cuenta y si es así en qué medida.

9. En general, ¿con qué frecuencia nuestro servicio cumple con sus expectativas?

Siempre	Frecuentemente	A veces	Rara vez	Nunca

Esta pregunta nos ayuda a determinar si al aplicar “Entrega Continua”, mejoró la percepción de calidad de nuestro servicio.

10. ¿El nivel con el que hemos contribuido al mejoramiento continuo de los procesos de negocio es?

Muy Satisfactorio	Satisfactorio	Poco Satisfactorio	Insatisfactorio	Muy Insatisfactorio

Esta pregunta nos ayuda a determinar si al aplicar “Entrega Continua”, mejoró la percepción de calidad de nuestro servicio.

11. En general, ¿el nivel con el que recibe la funcionalidad de acuerdo a sus expectativas es?

Muy Satisfactorio	Satisfactorio	Poco Satisfactorio	Insatisfactorio	Muy Insatisfactorio

Esta pregunta nos ayuda a determinar si al aplicar “Entrega Continua”, mejoró la percepción de calidad de nuestro servicio.

- **Analizar los resultados**

En esta etapa se analiza críticamente los resultados del estudio, comparando el antes y después de aplicar el enfoque de entrega continua en el desarrollo de software, de lo que se desprenden las conclusiones y lecciones en cuanto a la percepción de calidad de los involucrados.

- **Lineamiento**

Aquí se determinan lineamientos que ayuden a realizar el estudio.

5. IMPLEMENTACIÓN DEL ESTUDIO DE CASO

En este capítulo se va a realizar la implementación del estudio de caso, partiendo desde el Diseño del estudio de caso siguiendo con la Recopilación de la información hasta llegar a obtener y analizar los resultados y la diseminación de los mismos.

5.1 Definición del Entorno

El estudio se realiza en una cooperativa de ahorro y crédito del Ecuador, la misma que intenta ser líder en el sector financiero brindando servicios financieros de calidad.

Por tal motivo la cooperativa está creando nuevas formas de satisfacer a sus clientes, entre las cuales se destaca el proyecto denominado “Créditos ágiles”.

El organigrama funcional de la cooperativa es el que se muestra en la Figura 13.

ORGANIGRAMA FUNCIONAL GENERAL

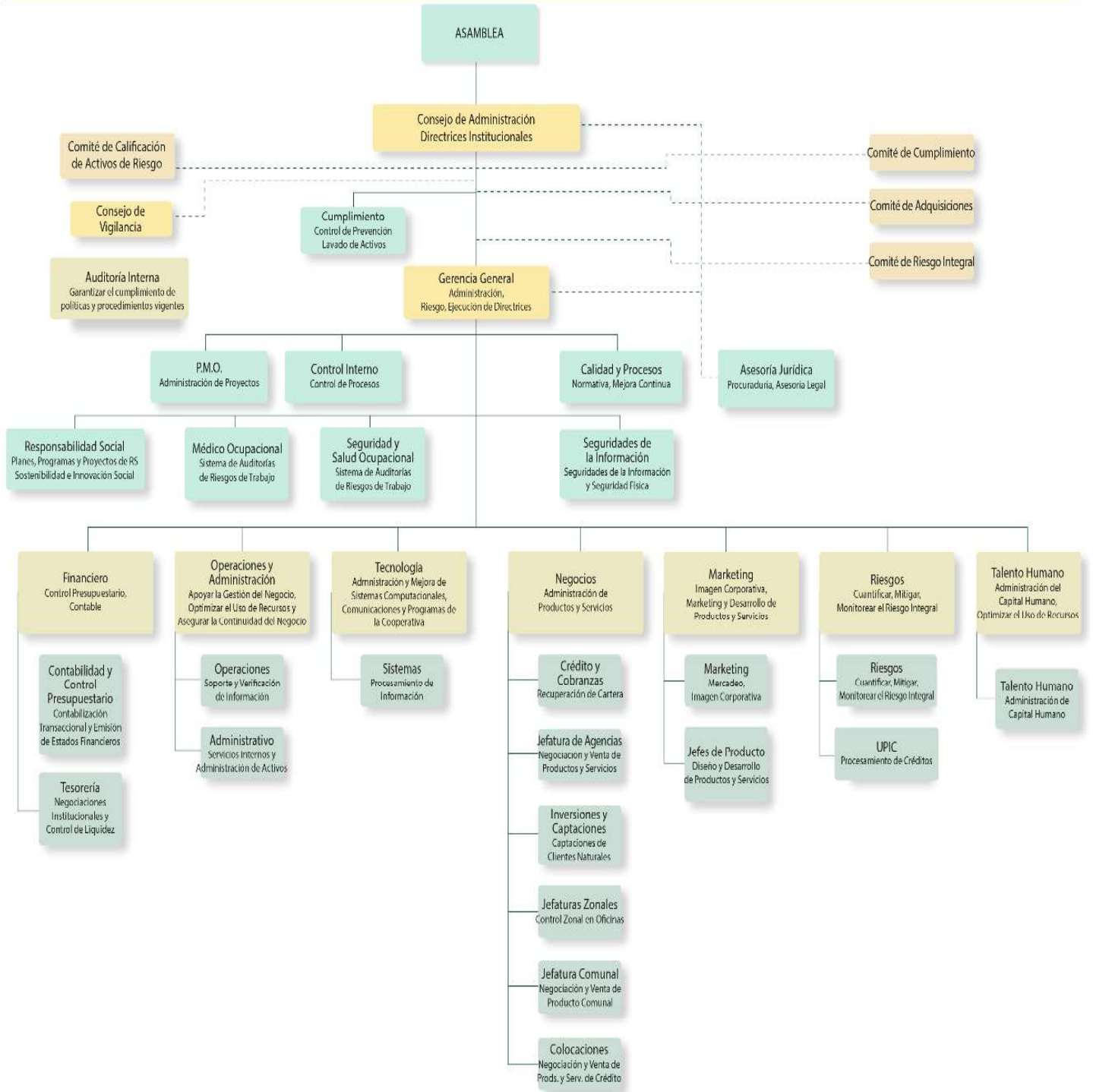


Figura 13. Organigrama Funcional

La empresa que realiza el estudio es Denarius, que es una empresa de desarrollo de software bancario, que busca innovar en su forma de trabajar para brindar un mejor servicio a sus clientes.

En el proyecto participan:

Cooperativa

Gerente de Tecnología e Innovación

Gerente de Operaciones

Gerente de Riesgos

Personal Técnico

Personal Operativo

Denarius

Gerente de Desarrollo

Gerente de Calidad

Desarrollador

El Tiempo estimado para este proyecto es de 2 meses, el mismo que se divide en 4 “sprints” de 2 semanas cada uno. Los 3 primeros “sprints” son para el Desarrollo y el 4 es para Estabilización.

5.2 Implementar Entrega Continua

Para cumplir con las actividades de la “Exploración Continua”, se ha utilizado las técnicas de “*Design Thinking*”.

5.2.1 Design Thinking

Es una metodología para generar ideas innovadoras, que centra su eficacia en entender y dar solución a las necesidades reales de los usuarios. Se compone de cinco etapas, como se muestra en la Figura 3, que son: Empatizar, Definir, Idear, Prototipar, Testear.

- **Empatizar**

Como parte de las actividades de colaboración se han involucrado desde el comienzo a personas tanto de la cooperativa como de Denarius.

Entre las técnicas empleadas, se realizan visitas a la cooperativa y reuniones, para determinar las oportunidades que tiene la cooperativa para mejorar sus

servicios frente a los clientes. Las mismas que permitieron obtener varias alternativas de mejora que luego de formalizar su ejecución entran en el proceso de entrega continua. Para este estudio de caso se llega a determinar la construcción de un producto de crédito que permita otorgar de forma ágil y en corto tiempo créditos para microempresarios.

- **Definir**

Una vez determinada la alternativa "Crédito Ágil", se procede a obtener las especificaciones necesarias con el levantamiento de historias de usuario (Requerimiento) en colaboración con los interesados.

Luego de definido el alcance del proyecto y el "Release Backlog" se procede a planificar, determinando que la construcción (desarrollo) contará con tres sprints de dos semanas cada uno, y un sprint para estabilización. Al final de cada sprint se realizará una demostración (prototipo) con el avance hasta ese momento y poder tomar acciones correctivas de forma ágil con la colaboración de los involucrados.

- **Idear**

Como parte de la etapa tres (Idear) se han realizado reuniones para determinar de forma gráfica el proceso y los contenidos por pantalla, creando diseños que ayuden a ajustar de mejor manera los requerimientos.

- **Prototipar**

Desde el final del primer sprint se tiene un prototipo creado, que permite ir conociendo la nueva funcionalidad, y en caso de ser necesario, modificarla de acuerdo a las necesidades.

- **Testear**

La etapa 5 se la cumple día a día realizando pruebas de cada una de las funcionalidades, añadidas constantemente mediante los pasos de versión al ambiente que se requiera.

5.2.2 Integración continua

Para la Integración y pruebas de historia se consideraron 6 características, las mismas que se dividieron en 15 historias de usuario como se muestra a continuación:

Feature: Parametrización Nuevo Producto

Historia: Creación Producto

Feature: Creación de Crédito

Historia: Ingreso y consulta datos cliente

Historia: Integración Fuente externa

Historia: Consulta Score

Historia: Calculo Capacidad de Pago

Historia: Validar políticas

Historia: Calcular Solicitud de crédito

Historia: Presentación Tabla Amortización

Historia: Validar Score antes de ingresar información

Feature: Digitalización de Documentos

Historia: Cargar documentos

Historia: Validar Lista de Control

Historia: Regularizar documentos

Feature: Notificaciones

Historia: Notificar aprobación y desembolso

Feature: Impresión de Documentos

Historia: Crear documentos a imprimir

Feature: Proceso de Desembolso

Historia: Desembolsar y acreditar cuenta

La integración se la realiza utilizando dos ramas (branches) de código, una es la línea base de producción y la otra es la línea base para desarrollo; es en esta donde se va integrando el código mientras se va agregando nuevas funcionalidades, una vez se validan las nuevas funcionalidades, se pasa el nuevo código a la línea base de producción.

Existen dos soluciones: una es la que contiene la lógica de negocio y el acceso a datos, y la otra contiene la capa de presentación. En cada solución se tienen varios proyectos que se organizan de acuerdo a la funcionalidad para la cual fueron requeridos.

Las pruebas unitarias se crean en un proyecto paralelo al proyecto que implementa la funcionalidad, y se construye igual que cualquier otro proyecto, es decir que es parte del build.

5.2.3 Despliegue de funcionalidades

Para este estudio de caso, se ha creado el “Feature Toggle” en el número de transacción que permite acceder al menú en donde se presentará la opción para crear este tipo de productos. De esta forma, la nueva funcionalidad se pasa en el siguiente paso de versión al ambiente de Producción, sin que esta funcionalidad este completa o sin errores, es decir sin que haya pasado por el proceso de certificación, pero no estará habilitada ya que la transacción no se ha asignado al “tenant” (cooperativa) para su uso. La transacción se habilitará en el momento en que el usuario certifique y de su aprobación para que la funcionalidad sea visible y utilizable.

5.3 Recoger información

En esta etapa se recolectó información del antes y después de aplicar el enfoque de entrega continua para que con esta información se pueda realizar un análisis comparativo desde los 3 puntos de vista que son: Cliente, Desarrollo y Organización. Para evaluar el aprendizaje organizacional y la competitividad y efectos relacionados con el mercado se recopiló información de desarrollos realizados en la cooperativa para comparar por su tamaño entre los que utilizaron entrega continua versus los de “Método tradicional de cambios” y determinar el tiempo que tomó.

5.3.1 Punto de vista del Cliente

Para realizar el análisis comparativo de la información que nos ayude a determinar el impacto desde el punto de vista del cliente se buscó diseñar una encuesta corta y de fácil entendimiento con la finalidad de que los destinatarios puedan llenarla sin problemas.

La encuesta se realizó antes y después de aplicar entrega continua y se entregó individualmente de forma personal.

Las personas se seleccionaron por su incidencia directa en el proyecto, las mismas que son:

- Gerente de operaciones. Que se encarga de especificar qué características pasan a Producción.
- Gerente de Tecnología e innovación. Que se encarga de especificar las nuevas características que se requieren.
- Gerente de Riesgos. Que es el encargado de identificar y evaluar los riesgos de la cooperativa ante cambios en su “core”.
- Jefe comercial. Que es el encargado de poner los objetivos y velar por su cumplimiento.
- Analistas de Gestión Organizacional. Que son los encargados de especificar las historias de usuario y realizar las pruebas funcionales.
- Personal de Tecnología. Que son los encargados de validar que las nuevas funcionalidades estén habilitadas.

5.3.2 Punto de vista del Desarrollo

Para realizar el análisis comparativo desde el punto de vista del Desarrollo se tomó en cuenta los ingresos que percibe Denarius, el retrabajo que genera el equipo de desarrollo, tiempo de entrega de funcionalidades y número de errores que se generan realizando proyectos donde se aplicó entrega continua vs proyectos donde no se aplicó.

La unidad de medida que Denarius usa para medir el tamaño del desarrollo de software es el punto de historia, que incluye todo el trabajo para convertir la especificación del cliente en software que funciona. (No se demandan puntos por corrección de errores ni por actividades que no contribuyen directamente con la construcción y el control de calidad del software).

Otro de los aspectos en los cuales se puede determinar el impacto de haber introducido el enfoque de entrega continua en el desarrollo de software, es el número de errores reportados.

Denarius cuenta con herramientas que permiten ingresar los errores como ítems de trabajo en los cuales se puede identificar mediante una característica, a que fase del proceso de certificación pertenece el error.

Las posibles fases en las cuales puede encontrarse el error son:

Deployment: Cuando el error se introdujo por el nuevo paso de versión.

Mantenimiento: Cuando el error se determinó por el cliente en el ambiente de producción.

Testing Q2: Cuando el error se determinó internamente por el equipo de QA en el ambiente de certificación.

Testing Q3: Cuando el error se determinó por el cliente en el ambiente de certificación.

5.3.3 Punto de vista de la Organización

Para realizar el análisis comparativo de información desde el punto de vista organizacional se tomará en cuenta el trabajo en equipo (Denarius), Número de créditos entregados (cooperativa) y costos tanto para el cliente como para Denarius.

Al ser la cooperativa una institución financiera, la fuente principal de sus ingresos viene por la otorgación de créditos a sus clientes, por tal motivo para determinar el impacto organizacional se ha tomado en cuenta el número de créditos otorgados antes y después de haber aplicado el enfoque de entrega continua.

Otra de las medidas que se deben tomar en cuenta para determinar el impacto organizacional es el costo que significa invertir en este tipo de iniciativas.

5.4 Análisis de Resultados

5.4.1 Impacto desde el punto de vista del Cliente

Se realizó la encuesta a 7 personas involucradas directamente en el proyecto y que tienen muy claro la diferencia entre el proceso “Tradicional de cambios” y el proceso de entrega continua.

Los resultados encontrados se detallan a continuación haciendo un análisis por cada pregunta.

1. En la pregunta 1 al aplicar “Exploración Continua” se puede evidenciar que la atención de los requerimientos es más rápida, ya que con el proceso “Tradicional de cambios” lo máximo era moderadamente rápido, en cambio con “Entrega Continua” se llega hasta extremadamente rápido como se puede ver en la Figura 14.

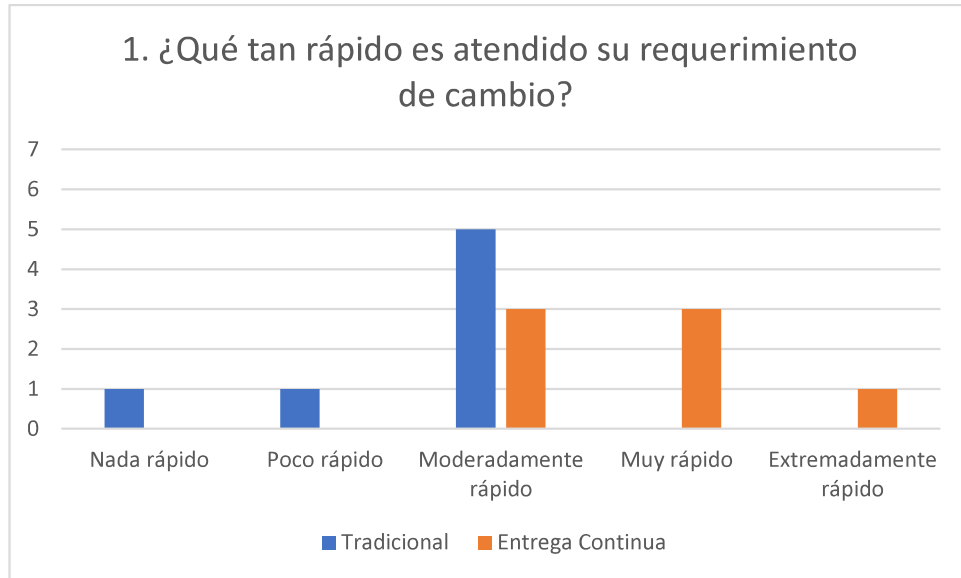


Figura 14. Tiempo de atención de requerimientos.

2. En la pregunta 2 los resultados indican que antes de aplicar “Entrega Continua” el proceso de paso de nuevas características a Producción era muy lento, en cambio ahora es muy rápido, como se puede ver en la Figura 15.

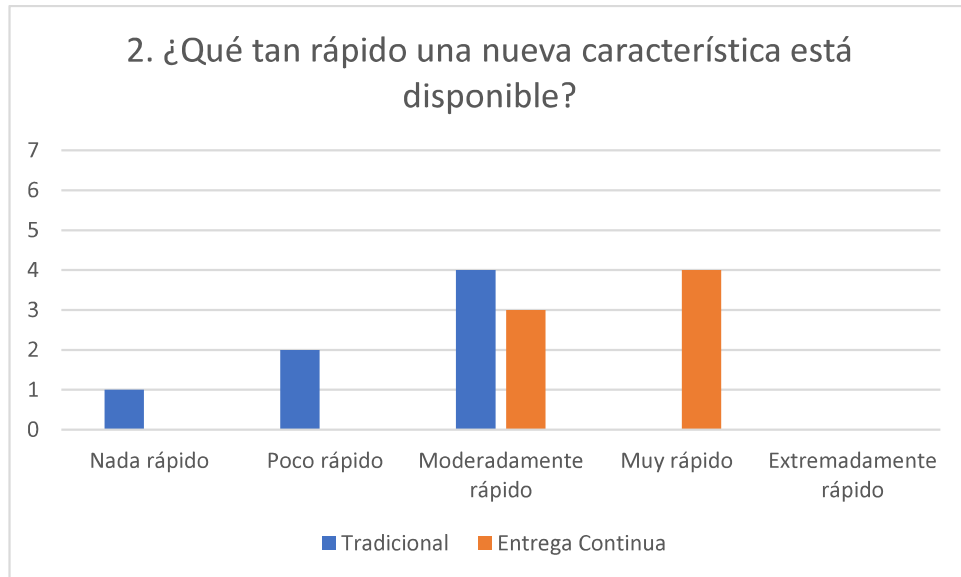


Figura 15. Disponibilidad de nuevas características.

3. En la pregunta 3 los problemas reportados anteriormente se perciben que va a ser poco probable que vuelvan a suceder, y en el proceso “Tradicional de cambios” era bastante probable, como se puede ver en la Figura 16.

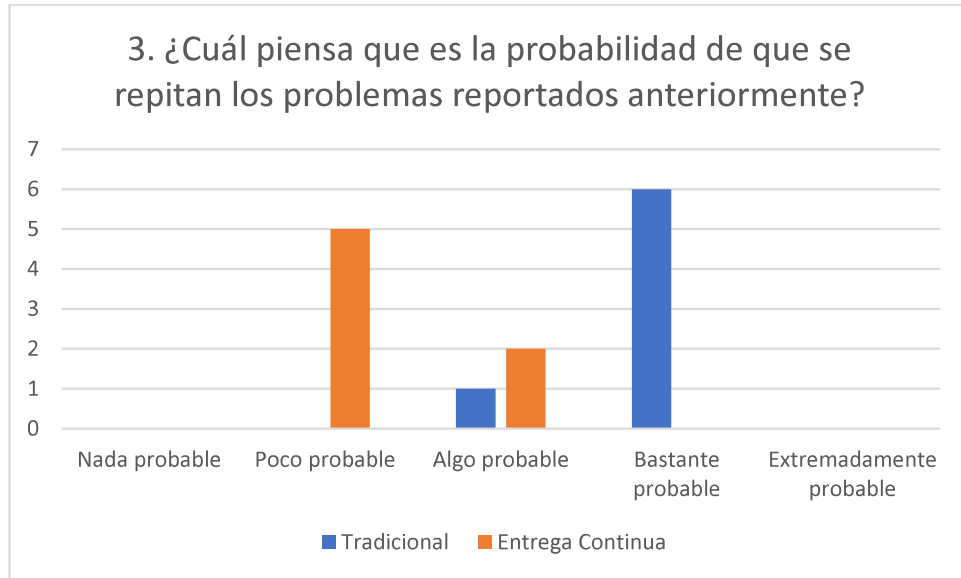


Figura 16. Probabilidad de que se repitan los problemas reportados anteriormente.

4. Con la pregunta 4 se puede evidenciar que el habilitar nuevas características para su uso es más fácil después de aplicar “Entrega Continua” que con el método “Tradicional de cambios”, como se puede ver en la Figura 17.

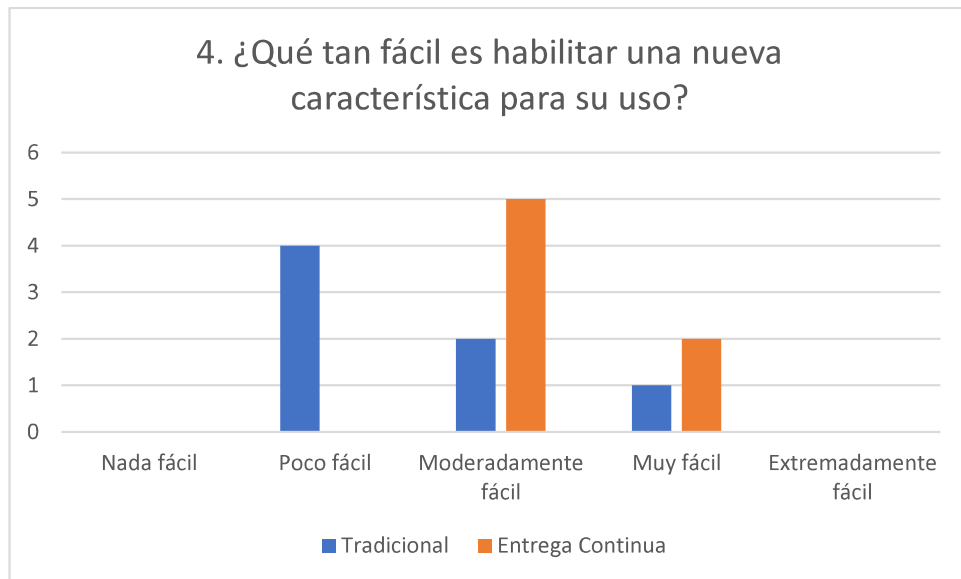


Figura 17. Habilitar nuevas características.

5. En la pregunta 5 la percepción es que el tiempo de respuesta y corrección de problemas mejora al aplicar “Entrega Continua”, como se puede ver en la Figura 18.

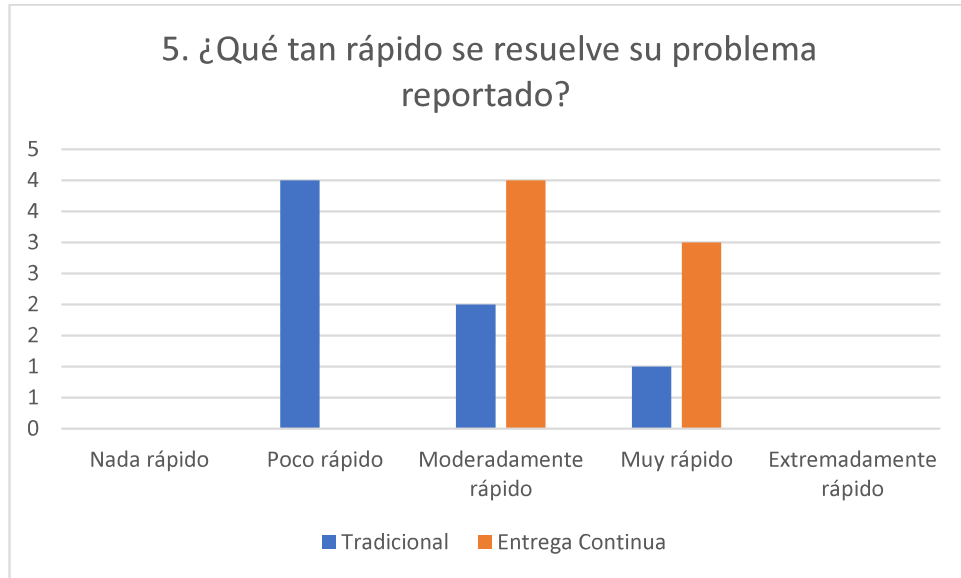


Figura 18. Resolución de problemas reportados.

6. En la pregunta 6 la percepción de calidad de nuestro servicio mejora al aplicar “Entrega Continua”, como se puede ver en la Figura 19.

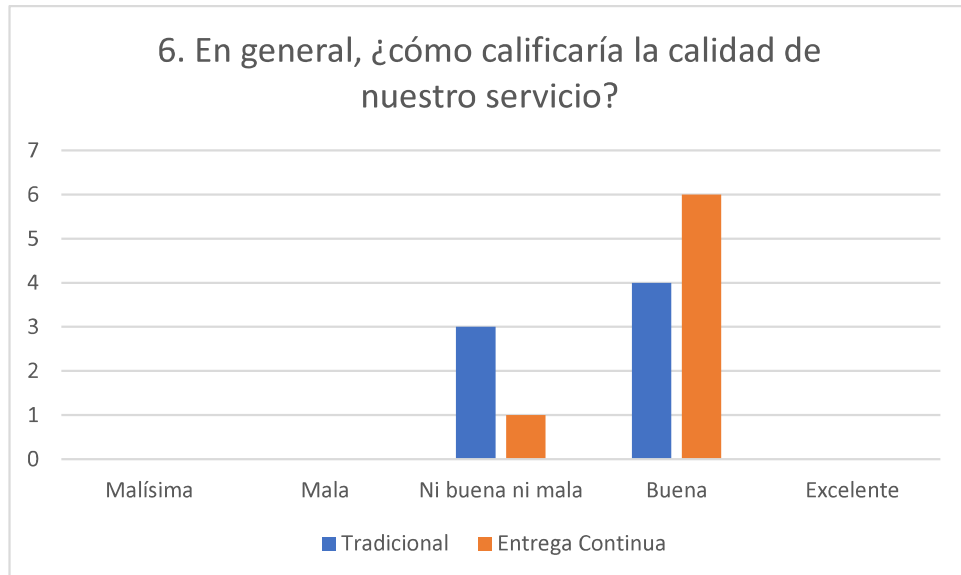


Figura 19. Calidad del servicio.

7. Con la pregunta 7 la experiencia de problemas con nuestro servicio baja al aplicar “Entrega Continua” en relación al proceso “Tradicional de cambios”, como se puede ver en la Figura 20.

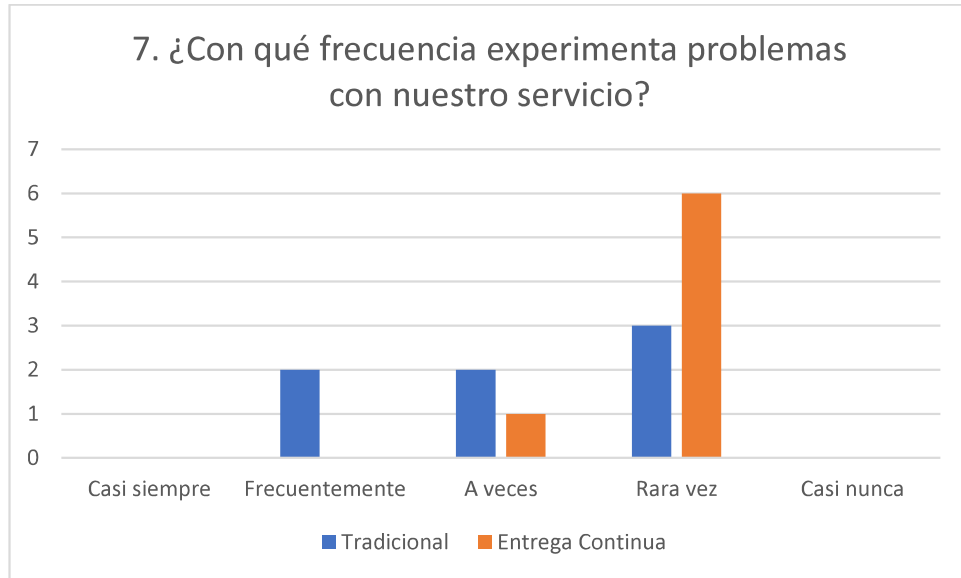


Figura 20. Frecuencia de problemas.

8. Con esta pregunta se puede evidenciar que la percepción del cliente con relación a que su opinión es valorada, mejora al aplicar “Entrega Continua”, como se puede ver en la Figura 21.

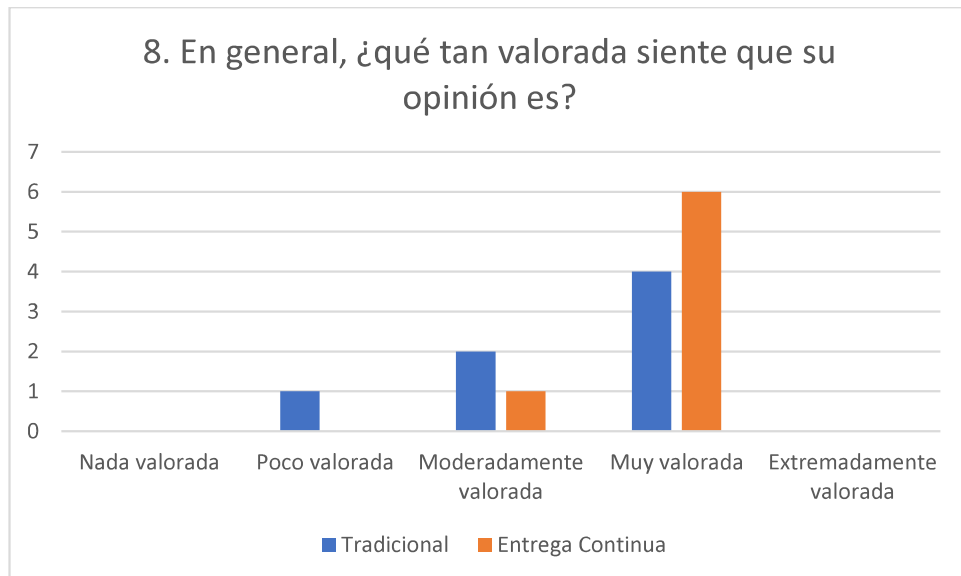


Figura 21. Valoración de opinión.

9. Aquí se puede evidenciar que la percepción de que se cumple con las expectativas del cliente mejora mucho al aplicar “Entrega Continua”, como se puede ver en la Figura 22.

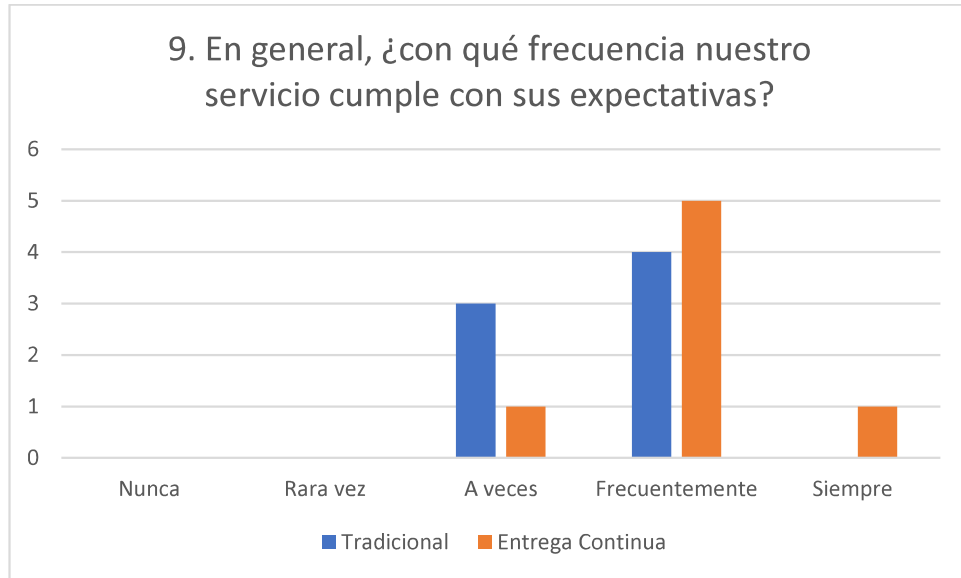


Figura 22. Cumplimiento de expectativas.

10. Con esta pregunta se puede evidenciar que el aplicar “Entrega Continua” contribuye al mejoramiento de los procesos del negocio, como se puede ver en la Figura 23.

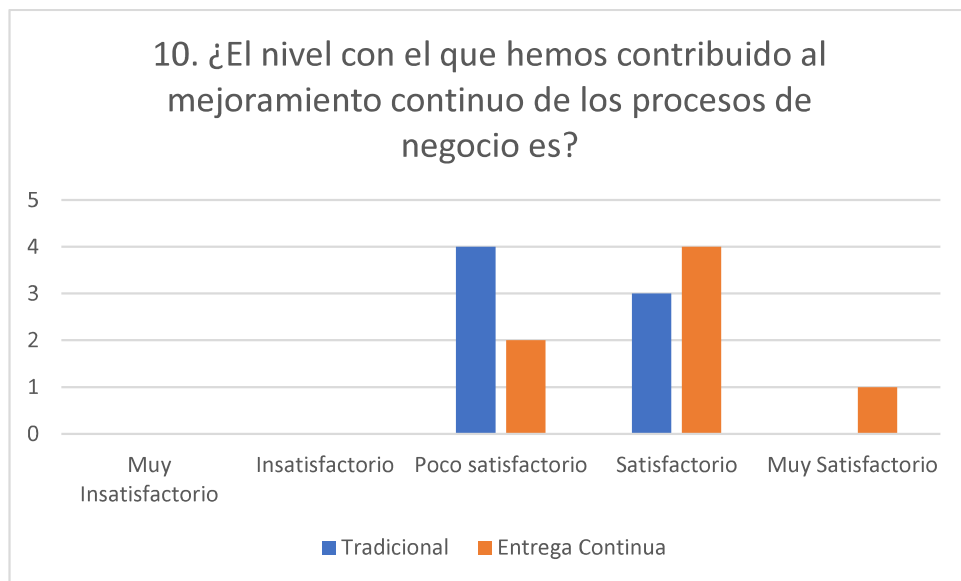


Figura 23. Contribución al mejoramiento continuo.

11. Finalmente, con la pregunta 11, se percibe que las funcionalidades satisfacen de manera satisfactoria las expectativas del cliente al aplicar “Entrega Continua”, como se puede ver en la Figura 24.

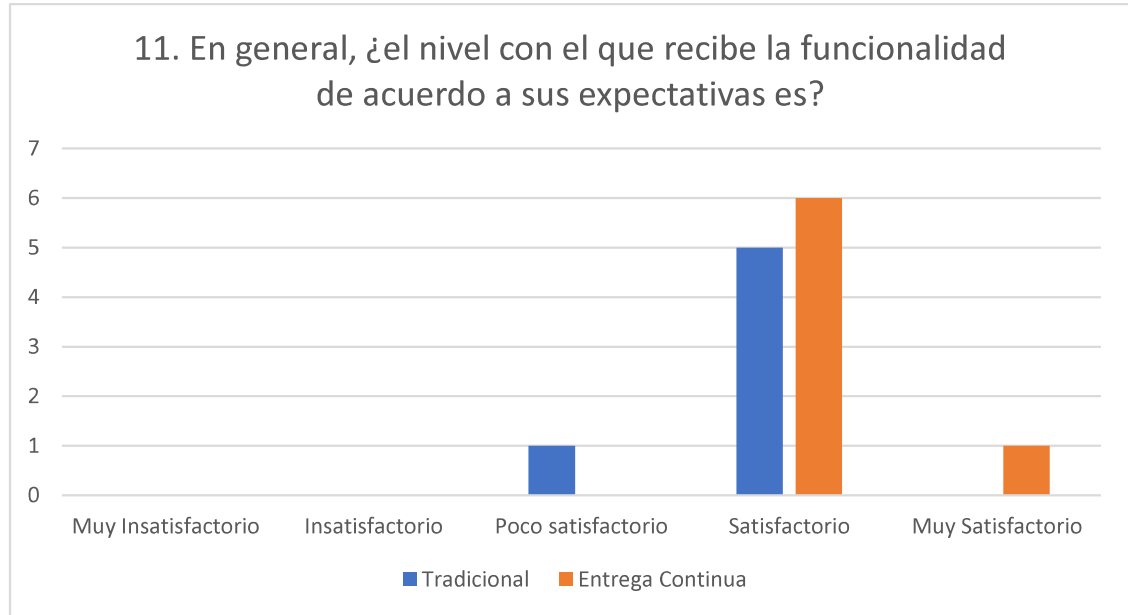


Figura 24. Nivel de expectativas.

5.4.2 Impacto desde el punto de vista del Desarrollo

En la Tabla 1 se pueden apreciar los datos de Historias de usuario, retrabajo y tiempo de entrega obtenidos de los periodos 3 y 4 (Q3, Q4) del año 2017 y los periodos 1 y 2 del año 2018 (Q1, Q2).

	Puntos de historia				Retrabajo	Desperdicio (Lead Time)
	Proceso Normal de Cambios	Entrega Continua	Total			
2017Q3	489	0	489		14%	8.3
2017Q4	246	0	246		16%	8.3
2017S2	735	0	735		15%	8.3
				% Entrega Continua		
2018Q1	462	270	732	37%	22%	5.3
2018Q2	595	179	774	23%	16%	3.6
2018S1	1057	449	1506	30%	19%	4.5
Incremento	44%	61%	105%		4%	-46%

Tabla 1. Puntos de historia, retrabajo y tiempo de entrega

Al comparar los resultados antes (Semestre 2 2017) y después (Semestre 1 2018) de la implantación del proceso de entrega continua en la cooperativa se puede apreciar que:

El incremento total del trabajo realizado para el cliente es del 105% del cual el 61% se atribuye al nuevo proceso como se muestra en la Figura 25.

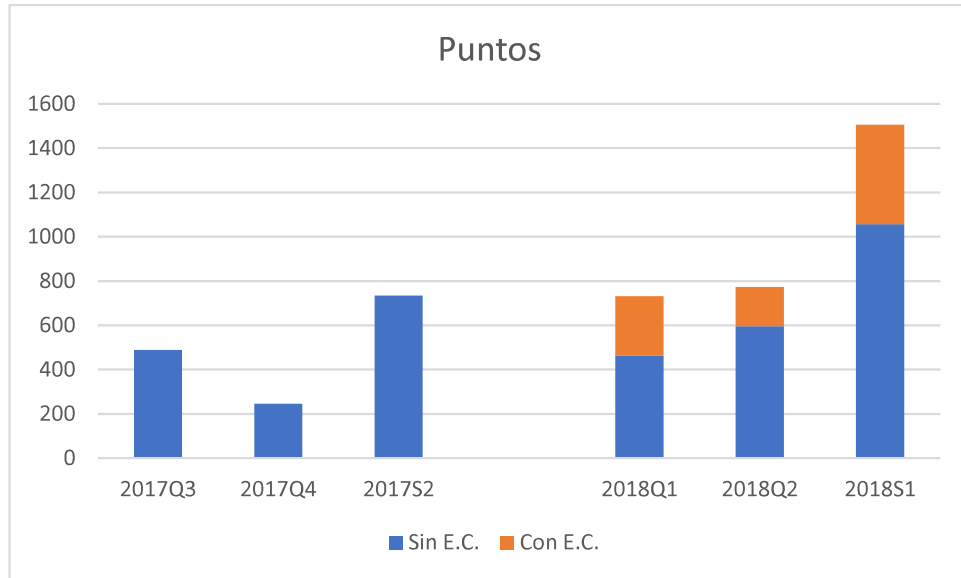


Figura 25. Puntos de historia realizados por período.

El trabajo que se realiza con el nuevo proceso representa el 30% trimestral del total para el cliente.

- Retrabajo

El retrabajo pasó del 15% al 19% como se muestra en la Figura 26. Este incremento del 4% es esperado debido a que como parte del proceso de “Entrega Continua” aplicado se registran de mejor manera los incidentes que antes quedaban sin registrar.

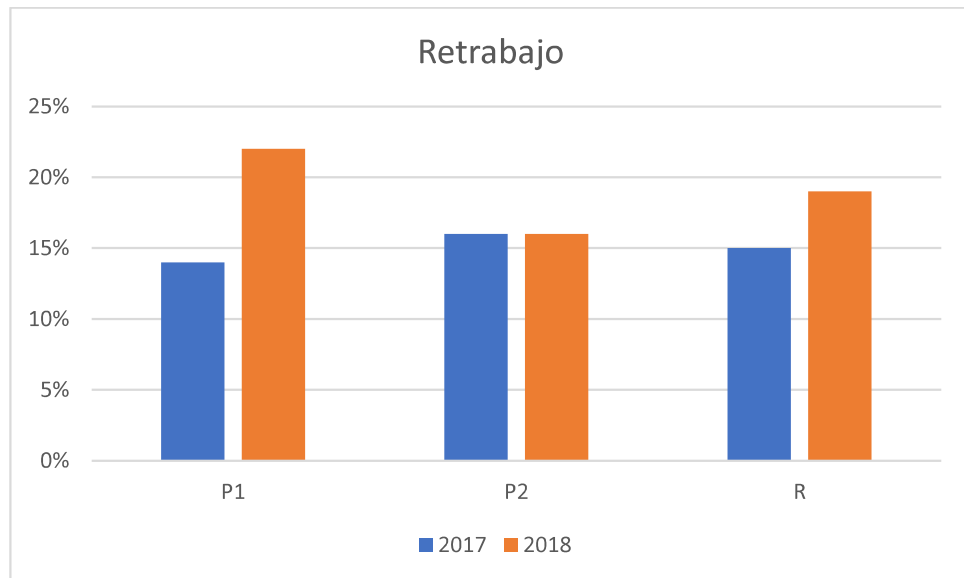


Figura 26. Retrabajo por período.

- Entrega de funcionalidades

El tiempo promedio de entrega de funcionalidades se redujo en un 46% pasando de 8.3 días a 4.5 días por cada ítem de trabajo, como se muestra en la Figura 27.

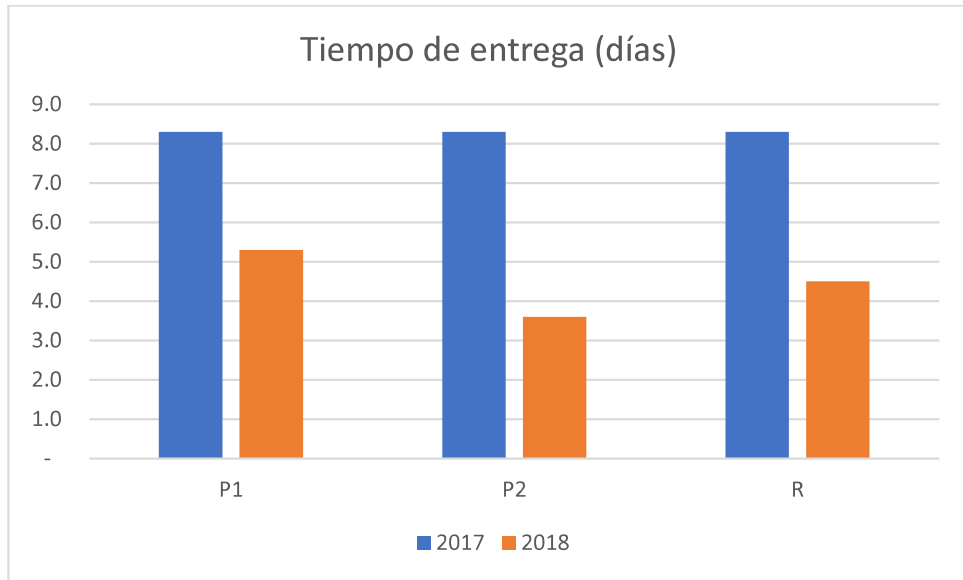


Figura 27. Tiempo promedio de entrega por período.

- Errores

Los datos que se presentan en la Tabla 2, corresponden al último cuarto del año 2017 y el primer cuarto del año 2018.

	# Errores			
	Deployment	Mantenimiento	Testing Q2	Testing Q3
2017Q4	6	55	70	15
2018Q1	5	37	60	6

Tabla 2. Número de errores por período

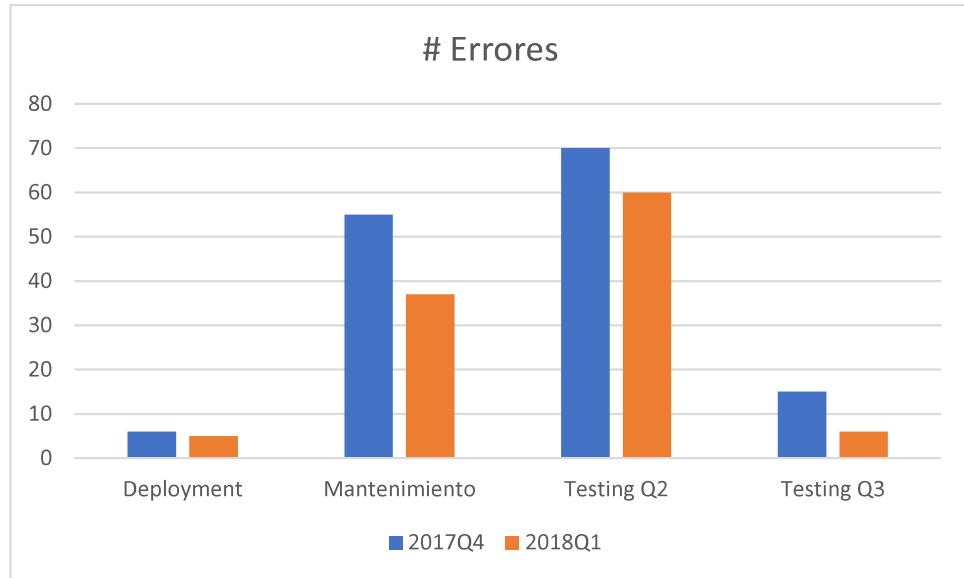


Figura 28. Comparación de errores por período y fase.

En la Figura 28, se puede ver que el número de errores relacionados al Mantenimiento, Testing Q2 y Testing Q3 disminuye cuando se aplica “Entrega Continua”.

5.4.3 Impacto desde el punto de vista de la Organización

El incremento del 105% (735 en 2017S2 a 1506 en 2018S1) en el trabajo con la cooperativa no hubiera sido posible sin la aplicación del proceso de entrega continua.

- Trabajo en equipo

- Se experimentó desgaste en las personas que participan en el proyecto debido a que las expectativas del cliente crecen al ver las mejoras que el proceso evidencia.
- A pesar de que el costo se mantuvo, el esfuerzo del equipo se incrementó para llegar a cumplir los compromisos, lo que hace que el proceso no sea sustentable a largo plazo y se tendrán que incrementar los costos.
- El incremento del retrabajo y la disminución del tiempo de entrega no está completamente explicado por la implementación del proceso de entrega continua por lo que es necesario refinar métricas para que sean más específicas.
- El proceso de entrega continua demanda de un compromiso inquebrantable con los principios y valores del desarrollo ágil, a menudo se confunde con informalidad por lo que no es aplicable en todos los clientes y en todos los proyectos aun con el mismo cliente.

- Se puede evidenciar que en el equipo de trabajo cada integrante mejora sus conocimientos ya que no solo debe hacer sus tareas, sino que también tiene que involucrarse en las tareas de los demás por lo tanto el rendimiento general del equipo mejora.

- Número de créditos

Cómo se puede ver en la Tabla 3 y Figura 29, a partir del mes de marzo del 2018, que es cuando el producto se libera en producción, el incremento en el número de créditos otorgados es evidente, evidenciando así un impacto positivo en la organización.

Mes	Creditos	
	Año	
	2017	2018
1	1469	2417
2	2211	2413
3	2367	2846
4	2052	2825
5	2579	3095
6	3036	3471
7	3424	
8	3496	
9	3180	
10	2687	
11	2597	
12	2493	

Tabla 3. Número de créditos otorgados

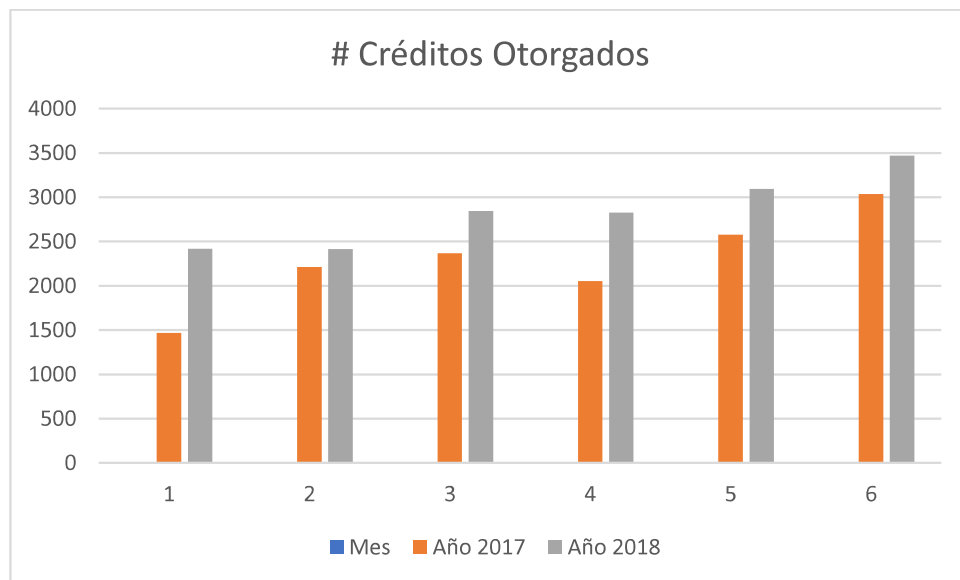


Figura 29. Comparación del número de créditos otorgados por mes y año.

- **Costos para el cliente Vs Costos de desarrollo**

Para la cooperativa cuando quería realizar un desarrollo nuevo tenía que hacerlo por el proceso tradicional en donde los costos relacionados al desarrollo podían incrementarse al tener mucha incertidumbre en la consecución del cambio, ya que las definiciones o historias de usuario por lo general no cumplían con los estándares necesarios y además los criterios de aceptación no estaban de acuerdo a la especificación del cambio. Después de implementar el enfoque de entrega continua la cooperativa incurre en mayores costos por el personal de Denarius, ya que se necesita que estén a tiempo completo dedicados al proyecto, pero se reducen costos por incertidumbre y corrección en las funcionalidades solicitadas.

Para Denarius el costo se mantiene, ya que no fue necesario contratar más personas, pero en relación a los ingresos es evidente el incremento que recibe por proyecto con entrega continua y el personal al terminar el proyecto puede dedicarse a otras tareas o proyectos de mejor manera que antes con el proceso tradicional, ya que no es necesario realizar muchos cambios o correcciones.

5.5 Lineamientos para la evaluación de impacto de implementar entrega continua

En la Figura 30 se puede observar el proceso para realizar la evaluación del impacto, el mismo que empieza con la definición del entorno, en donde se determina los involucrados y en donde se va a aplicar el estudio, luego se implementa el proceso de entrega continua, utilizando técnicas que puedan ayudar al acercamiento con el cliente y determinar de mejor manera lo que realmente quiere el Cliente. Luego se procede a realizar el Estudio de Caso, definiendo las métricas y como obtenerlas, luego se realiza la recolección y el análisis de la información, tanto para el estudio del impacto desde el punto de vista del cliente con el análisis de las encuestas, como para el Desarrollo con el análisis del número de errores, porcentaje de retrabajo y entrega de funcionalidades, y la Organización con el trabajo en equipo, número de créditos otorgados y costos relacionados.

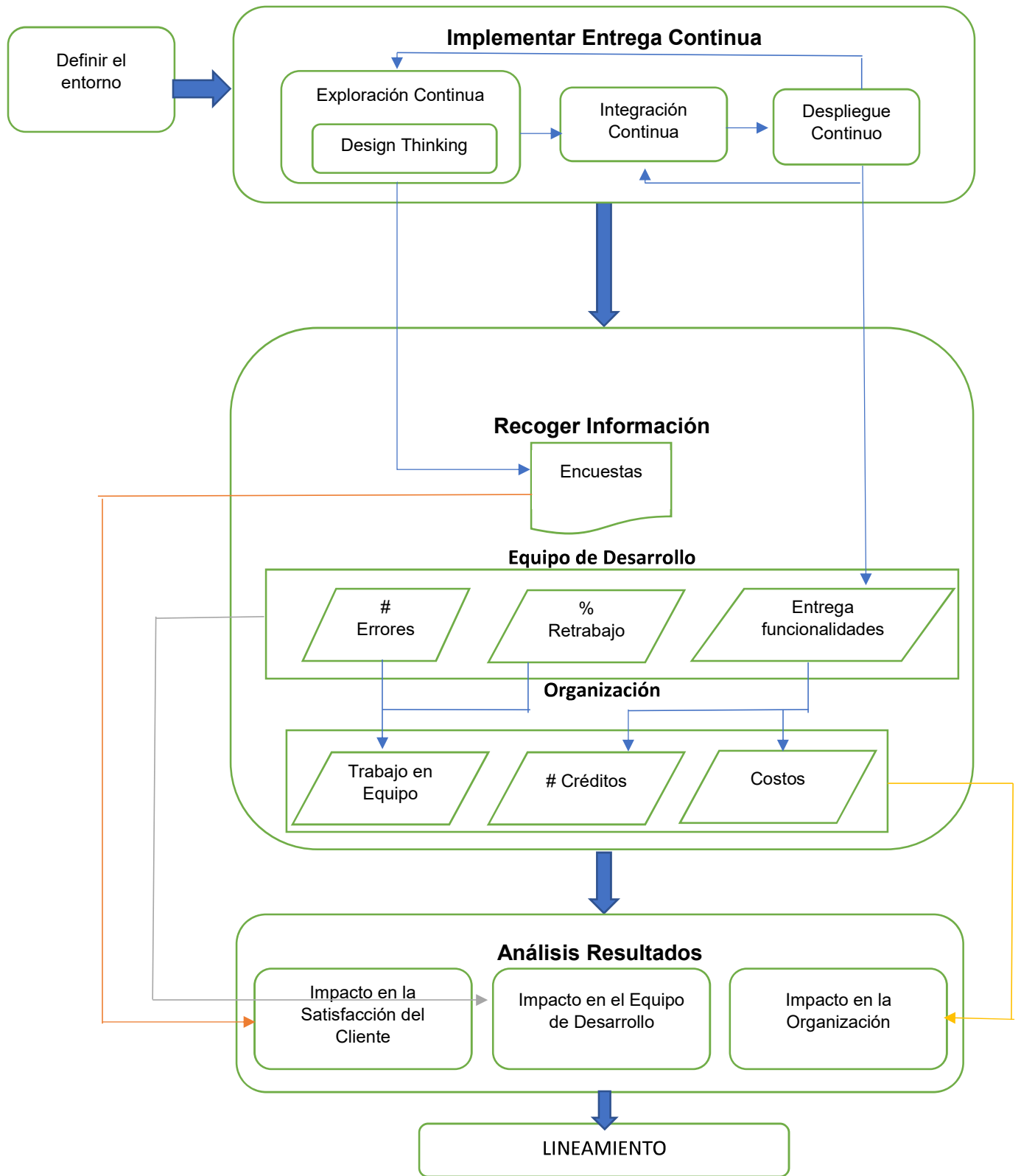


Figura 30. Lineamiento para la evaluación del impacto de aplicar entrega continua

6. DISCUSIÓN

Medir el impacto en la calidad final, de aplicar un nuevo enfoque en el desarrollo de software, puede llegar a ser complicado sino se tiene claro los que se quiere determinar. Al utilizar el método de estudio de caso se puede seguir lineamientos que ayuden en esta labor.

En el artículo [22] se habla de las métricas a tomar en cuenta para determinar el impacto de la "Integración Continua", estas métricas son: Ambiente de trabajo, calidad del código, satisfacción del cliente, tiempos, errores, pruebas. Se obtuvieron resultados negativos, por ejemplo, en lo que se refiere al ambiente de trabajo ya que se incrementó el estrés, para los desarrolladores fue más difícil aislarse para realizar su trabajo y la complejidad del trabajo se incrementó. Las mismas métricas también se utilizaron en este trabajo ya que la integración continua es parte del proceso de entrega continua y los resultados fueron parecidos ya que si se incrementó el estrés y la complejidad se incrementó, pero al tener herramientas que ayudan a la integración de código, se pudo aplacar estos inconvenientes. En el artículo [27] se describe un estudio de caso en el cual se aplicó entrega continua, indicando los beneficios y desafíos a los cuales se enfrentaron, aquí se aplicó "Code Commit" que es una configuración para que se compile la solución cada vez que se integre código, luego se realizan las pruebas de aceptación y finalmente el paso de versión al ambiente de producción. En Denarius, empresa que se dedica al desarrollo de software bancario, se realiza un proceso muy similar, porque se realizan compilaciones de código programadas o bajo demanda para que luego se pueda pasar versión a un ambiente de pruebas donde el cliente puede realizar sus pruebas, y si las pruebas son satisfactorias se realiza el paso de versión al ambiente de Producción, la diferencia principal es que en Denarius se utilizan los "Features Toogles", que permiten que si las pruebas no son satisfactorias, el paso de versión igual se da, pero con la funcionalidad deshabilitada.

En los artículos [24-27] se habla de los problemas que se pueden presentar al aplicar entrega continua, destacándose el problema en el diseño y cambio en la cultura de la gente. En este trabajo se aplacó estos problemas utilizando la técnica del "Design Thinking", la misma que permite involucrar a todos los interesados en el proceso de desarrollo de software, permitiendo que puedan ir observando de primera mano los acuerdos y definiciones que se tomaron en conjunto.

El proceso de entrega continua impacta positivamente en la percepción de calidad del usuario con respecto al software que recibe. Las encuestas realizadas nos ayudan a determinar que este impacto es positivo, ya que en la mayoría de los encuestados sus

respuestas dan a entender que la calidad general del software mejora, su opinión es valorada, se percibe reducción de errores y los tiempos de respuesta mejoran.

Según el artículo [27] que habla sobre los problemas, causas y posibles soluciones cuando se implementa entrega continua se encontraron varios problemas sintetizándolos en 7 temas que son:

- Problemas en el diseño del build

Estos problemas se presentan debido a que los scripts o procesos involucrados en el sistema de build son muy complejos o complicados. Para el presente trabajo estos problemas fueron mínimos ya que la definición de build que se realizó permite integrar de mejor manera el código que cada desarrollador genera, además se cuenta con herramientas que permiten realizar las construcciones de código de manera automática. Y, por último, pero no menos importante es el compromiso del equipo de desarrollo para la generación de código y la integración en la línea base para su construcción, y este compromiso tiene que ver mucho con la definición de listo que es una de las partes fundamentales de las metodologías ágiles. La definición de listo implica que el desarrollador que quiera integrar su código a la línea base, este primero debe obtener la última versión del código, compilar el código, validar que sus cambios no provoquen errores y por último subir sus cambios. De esta manera se reduce las rupturas del build.

- Problemas en el diseño del Sistema

Los problemas en el diseño del sistema se presentan porque el sistema consta de múltiples unidades, porque la arquitectura del sistema limita la entrega continua, dependencias internas o cambios en el esquema de base de datos. En este trabajo no se encontraron o fueron mínimos los problemas de este tipo, ya que la arquitectura (capas) con la que cuenta el sistema es robusta permitiendo que la entrega continua se adapte sin inconvenientes.

- Problemas de integración

Estos problemas según la literatura son comunes cuando existe gran cantidad de cambios a ser integrados, conflictos entre los cambios, el build es roto muy seguido o permanece roto por mucho tiempo, no permitir integrar cambios mientras el build está roto, se utilizan ramas de código que no se integran por largo tiempo, flujo de desarrollo roto o los cambios son aprobados lentamente para ser integrados a la línea base. En Denarius se cuenta con una rama de código que viene de la línea base, la misma que es utilizada por todos los desarrolladores para integrar sus cambios, Los cambios realizados no se deben aprobar, es responsabilidad de cada desarrollador hacer los cambios y correcciones necesarias, y previo

al paso de versión al ambiente de producción se hace un merge desde la rama de código a la línea base por parte del líder del equipo de desarrollo.

- Problemas de testing

Según la literatura estos problemas se presentan porque los resultados de las pruebas no son comunicados a los desarrolladores, las pruebas toman mucho tiempo o son muy complejas, el software no está disponible para ser probado, etc. Muchos de estos problemas se presentaron en el trabajo actual, debido a que el usuario no tiene la cultura de realizar pruebas continuamente y espera a que el sistema esté completo para llevar a cabo las pruebas. Esto se debe principalmente a la cultura de la gente, por eso se aplicaron técnicas como el Design thinking que permite acompañar al usuario durante todo el proceso de desarrollo y pruebas.

- Problemas de reléase

Entre los problemas relacionados al reléase se encontraron la preservación de la data, la sincronización de la documentación con las últimas versiones, errores en el deploy por releases frecuentes o porque a los usuarios no les agrada las actualizaciones. Para el paso de versión al ambiente de producción (release), Denarius no modifica de forma alguna la data del usuario, además al existir documentado los requerimientos de cambio se mantienen sincronizada la documentación con la última versión, los errores en el deploy existen, pero son pocos y es debido a daños colaterales provocados por los cambios que no fueron debidamente certificados, además el usuario siempre va a quejarse si no fue avisado de los cambios que se van a introducir.

- Problemas humanos y organizacionales

Entre los problemas encontrados relacionados a este tema, se tiene la falta de disciplina, presión alta en el equipo de desarrollo, cambio de roles y la coordinación. En el presente trabajo se evidenció de igual manera la falta de disciplina en especial al realizar las pruebas, ya que la cultura de la gente es probar una vez concluido el desarrollo, y no debería ser así, por lo que se utilizaron técnicas para acompañar al usuario en el proceso de pruebas continuas. La presión en el equipo existió durante todo el proyecto y además las personas que intervinieron debían cumplir con varios roles. La coordinación no fue un problema ya que el equipo fue pequeño.

- Problemas de recursos

Para este tema en la literatura se encontraron dificultades por el esfuerzo que demanda iniciar en el proceso de entrega continua, los recursos de hardware que son necesarios para realizar los builds y el ambiente de pruebas y las latencias de red que dificultan la integración continua. Estos problemas se presentaron, pero no fueron considerables al tener el sistema montado en la nube “Azure”, en donde los recursos están siempre disponibles.

7. CONCLUSIONES

- Las métricas que se determinaron para medir el impacto en la percepción de la calidad, por parte cliente, el equipo de desarrollo y la organización fueron el Retrabajo, Entrega de funcionalidades y el número de errores, trabajo en equipo, costo del proyecto y número de créditos otorgados.
- Para realizar “Exploración Continua” la técnica que se utilizó en este estudio es “Design Thinking”, porque proporciona un entendimiento más claro de lo que el cliente realmente necesita, ya que no siempre lo que quiere es lo que necesita o no lo puede explicar en un documento.
- El presente estudio puede servir para que las organizaciones que están involucradas en proyectos de desarrollo de software, ya sea con personal interno o con proveedores externos, puedan beneficiarse de la mejora de la calidad al aplicar un proceso de entrega continua, sin tener que recurrir a costosas consultorías.
- Con las encuestas realizadas se logró determinar que la percepción de calidad desde el punto de vista del Cliente mejora al aplicar entrega continua. En general la percepción de calidad en el software por parte del usuario después de aplicar entrega continua mejora, ya que se evidencia que su opinión es valorada, la reducción de errores es notoria y los tiempos de respuesta son más rápidos.
- Se consiguió determinar que la calidad mejora desde el punto de vista del Desarrollo, ya que el tiempo en la entrega de funcionalidades se reduce al igual que el número de errores, sin embargo, el retrabajo aumenta, ya que se registra de mejor manera los incidentes.
- El proceso de entrega continua permitió a las organizaciones involucradas en el estudio, fijar una línea base sobre la cual implementar mejoramiento continuo en función de métricas objetivas. La misma contribución se puede aplicar a otras organizaciones que utilicen la misma guía.
- Aplicar “Exploración Continua” ayuda a atender de manera más rápida los requerimientos de los usuarios ya que se interactúa directamente con los interesados y se llega a entender el problema que se quiere solucionar de una manera más objetiva y personal.

- Para realizar Implementación continua se puede utilizar estrategias de integración de soluciones basadas en historias de usuario que permitan desarrollar funcionalidades y pruebas unitarias específicas.
- Para desplegar funcionalidades de manera continua sin que sean liberadas se utilizan técnicas como los “Feature Tooggles” que permiten habilitar o deshabilitar características en caliente.
- La cultura de la gente y de las organizaciones son el principal obstáculo en la adopción de nuevas técnicas para el desarrollo de software, porque siempre existe resistencia al cambio y lo desconocido, es por esto que realizar el acercamiento directo mediante la exploración continua ayuda mucho en este aspecto.
- Entender entrega continua es fácil, ponerlo en práctica puede llegar a ser muy difícil y costoso, ya que el cambio en la forma de hacer las cosas puede llegar a ser un obstáculo al adoptar nuevas técnicas.
- La entrega continua ayuda a mejorar la calidad general del software, ayudando a entregar valor al cliente de una manera más rápida, porque es una técnica que permite pasar versiones de código con características nuevas de una manera más rápida y habilitarlas en el momento que se desea.
- El involucrar al cliente en todo el proceso de entrega continua permite que sus expectativas sean cubiertas casi en su totalidad, porque recibe valor continuamente.
- Para lograr la adopción de nuevas técnicas como entrega continua, es necesario que las personas estén abiertas al cambio y motivadas para lograr mejores resultados.

8. REFERENCIAS BIBLIOGRÁFICAS

- [1] "IEEE Standard Glossary of Software Engineering Terminology," *IEEE Std 610.12-1990*, pp. 1-84, 1990, doi: 10.1109/IEEESTD.1990.101064.
- [2] G. Lomphey and S. Hernandez, "La importancia de la calidad en el desarrollo de productos de software," *Technical Report COMP-018-2008, Mexico*, 2008.
- [3] C. Tarazona T., "Amenazas informáticas y seguridad de la información " *Derecho Penal y Criminología* vol. 28, no. 84, 2007.
- [4] J. R. L. Fuentes, *Desarrollo de Software ÁGIL: Extreme Programming y Scrum*. IT Campus Academy, 2015.
- [5] K. Beck *et al.*, "Manifiesto for agile software development," 2001.
- [6] J. P. Womack and D. T. Jones, "Lean thinking," *Banish waste and create wealth in your corporation*, ProdPress. com, Wroclaw, 2008.
- [7] M. Poppendieck and T. Poppendieck, *Lean Software Development: An Agile Toolkit: An Agile Toolkit.* Addison-Wesley, 2003.
- [8] "Continuous Delivery Pipeline. Scaled Agile Framework." <https://www.scaledagileframework.com/continuous-delivery-pipeline/>
- [9] "Continuous Exploration. Scaled Agile Framework." <https://www.scaledagileframework.com/continuous-exploration/>
- [10] Atlassian. "Business Value of Continuous Delivery | Atlassian." <https://www.atlassian.com/continuous-delivery/principles/business-value>
- [11] P. C. Martínez Carazo, "El método de estudio de caso: estrategia metodológica de la investigación científica," (in Español), *Pensamiento & Gestión*, no. 20, 2006.
- [12] R. K. Yin, *Case Study Research: Design and Methods*. SAGE Publications, 2003.
- [13] O. E. Dictionary. Oxford University Press (in English).
- [14] E. Ormala, "Impact Assessment: European Experience of Qualitative Methods and Practices," *Evaluation Review*, vol. 18, no. 1, pp. 41-51, 1994, doi: 10.1177/0193841x9401800105.
- [15] "5 Stages in the Design Thinking Process." <https://www.interaction-design.org/literature/article/5-stages-in-the-design-thinking-process>
- [16] "Continuous Integration. Scaled Agile Framework." <https://www.scaledagileframework.com/continuous-integration/>
- [17] "Continuous Deployment. Scaled Agile Framework." <https://www.scaledagileframework.com/continuous-deployment/>
- [18] P. Hodgson. "Feature Toggles (aka Feature Flags)." <https://martinfowler.com/articles/feature-toggles.html>
- [19] N. Vafaie, M. Arvidsson, informationsteknik, and Engineering, "Metrics to measure the impact of continuous integration- An empirical study," (in eng), Student essay 2015-05-05 2015, doi: <http://hdl.handle.net/2077/38845>.
- [20] R. Rodríguez Cardona and M. Cobas Aranda, "METODOLOGIA DE EVALUACION DE IMPACTOS DE PROYECTOS DE INVESTIGACIÓN," 2009, doi: https://inis.iaea.org/collection/NCLCollectionStore/_Public/45/078/45078486.pdf.
- [21] C. Mann and F. Maurer, "A case study on the impact of scrum on overtime and customer satisfaction," *Agile Development Conference (ADC'05)*, pp. 70-79, 2005.
- [22] L. Mendoza Morales, M. Perez, and A. Grimán, *Análisis del Impacto del Proceso de Desarrollo en las Características de Calidad del Software*, IDEAS. 2018.

- [23] T. Lehtonen, S. Suonsyrjä, T. Kilamo, and T. Mikkonen, "Defining metrics for continuous delivery and deployment pipeline," in *SPLST*, 2015.
- [24] L. Chen, "Continuous Delivery: Huge Benefits, but Challenges Too," *IEEE Software*, vol. 32, no. 2, pp. 50-54, 2015, doi: 10.1109/MS.2015.27.
- [25] L. Chen, "Continuous Delivery: Overcoming adoption challenges," *Journal of Systems and Software*, vol. 128, pp. 72-86, 2017/06/01/ 2017, doi: <https://doi.org/10.1016/j.jss.2017.02.013>.
- [26] E. Laukkanen and C. Lassenius, *How to Adopt Continuous Delivery? A Research Proposal*. 2015, p. 356.
- [27] E. Laukkanen, J. Itkonen, and C. Lassenius, "Problems, causes and solutions when adopting continuous delivery—A systematic literature review," *Information and Software Technology*, vol. 82, pp. 55-79, 2017/02/01/ 2017, doi: <https://doi.org/10.1016/j.infsof.2016.10.001>.
- [28] G. G. Claps, R. Berntsson Svensson, and A. Aurum, "On the journey to continuous deployment: Technical and social challenges along the way," *Information and Software Technology*, vol. 57, pp. 21-31, 2015/01/01/ 2015, doi: <https://doi.org/10.1016/j.infsof.2014.07.009>.
- [29] E. Yacuzzi, "El estudio de caso como metodología de investigación: teoría, mecanismos causales, validación," *Serie Documentos de Trabajo*, 2005.
- [30] S. Chetty, "The case study method for research in small-and medium-sized firms," *International small business journal*, vol. 15, no. 1, pp. 73-85, 1996.
- [31] M. A. Hernández, S. Cantin Garcia, N. Lopez Abejon, and M. Rodriguez Zazo, "Estudio de encuestas," *Estudio de Encuestas*, vol. 100, 2010.
- [32] M. F. Sánchez, *Calidad total: modelo EFQM de excelencia*. FC Editorial, 2005.