



Universidad Nacional Autónoma de México
Programa de Posgrado en Ciencias de la Administración

**“DEVOPS COMO MARCO DE TRABAJO PARA LA
MEJORA CONTINUA DE SOFTWARE PARA BANCA
MÓVIL”**

TESIS

Que para optar por el grado de:

Maestra en Informática Administrativa
**Campo de conocimiento: Gestión de los Servicios de Tecnologías
de la Información**

Presenta:
Marisol Valdovinos Sánchez

Tutora: Mtra. Angélica María Ramírez Bedolla
Facultad de Contaduría y Administración

Ciudad de México, noviembre de 2020.



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

AGRADECIMIENTOS

A la Mtra. Angélica María Ramírez Bedolla por su dedicación y paciencia y sobre todo por ser una guía en mi camino.

A todos mis maestros y compañeros del posgrado por sus enseñanzas y experiencias.

Al Mtro. Alfredo Corona Carrera por su calidad humana con los alumnos.

A la UNAM por todas las oportunidades que me ha brindado durante mi trayectoria profesional.

Al CONACYT por sus programas de apoyo para los alumnos de posgrado.

DEDICATORIA

A mi familia:

Gracias Ricardo por tu valiosa ayuda y apoyo durante el proceso.

Richie y Fer por ser unos hijos maravillosos y ser parte de mi vida.

Perita por tus enseñanzas y valores para superarme cada día y por tu incondicional apoyo en mi trayectoria académica.

I. Introducción	6
1.1 Antecedentes	6
1.2 Planteamiento del problema	9
1.3 Objetivos	12
1.4 Diseño de la investigación	13
1.5 Metodología	16
1.5.1 Descripción de la investigación	16
1.5.2 Recolección e integración de datos	17
1.5.3 Análisis de la información	28
II. Marco teórico de la investigación	¡Error! Marcador no definido.
2.1 Introducción	36
2.1.1 Sistema Financiero Mexicano	38
2.1.2 Estructura del sistema financiero	41
2.1.3 Banca Móvil	48
2.1.4 Usuarios de Banca Móvil	56
2.2 Metodologías tradicionales de desarrollo de software	58
2.2.1 Antecedentes	58
2.2.2 Metodologías tradicionales de desarrollo de software	60
2.2.3 Metodología en Cascada	61
2.2.4 Metodología basada en prototipos	64
2.2.5 Metodología Iterativa e Incremental	67
2.2.6 Modelo en espiral	70
2.3 Marco de desarrollo ágil de soluciones para tecnología móvil	72
2.3.1 Introducción	72
2.3.2 Scrum	74

2.3.3 Kanban	77
2.3.4 Extreme Programming (XP)	79
2.3.5 DevOps	81
2.3.5.1 Origen	81
2.3.5.2 Concepto	84
2.3.5.3 Ventajas del enfoque DevOps	87
2.3.5.4 Principios y prácticas DevOps	90
2.3.5.5 DASA/Modelo de competencias DevOps	95
2.3.5.6 Herramientas y Plataformas de automatización	98
2.3.5.7 Experiencias de adopción del enfoque	103
III. Resultados	106
3.1 Situación actual de Bancatec	106
3.1.1 Misión y visión	106
3.1.2 Estrategia de transformación digital.	107
3.1.3 Matriz de factores que limitan la entrega ágil de aplicaciones móviles.	108
3.2 Marco de trabajo “Framework DevOps for Mobile Banking”	110
3.2.1 Definición, objetivos, principios y componentes	110
3.2.2 Equipos de Entrega Ágil (E2A)	117
3.2.3 Proceso unificado y de mejora continua de desarrollo y despliegue de banca móvil.	123
3.2.4 Tubería Automatizada de Entrega Continua (TAG)	133
3.2.5 Indicadores clave de rendimiento	138
3.2.6 Gestión del Cambio	143
IV. Conclusiones	149
Apéndices	153
Apéndice A: Cuestionario Diagnóstico sobre despliegue y entrega continua de aplicaciones de Banca Móvil	153

Apéndice B: Entrevista Diagnóstica sobre despliegue y entrega continua de aplicaciones de Banca Móvil	156
Fuentes consultadas	157

I. Introducción

1.1 Antecedentes

En el mundo actual estamos experimentando una revolución relacionada con los servicios financieros, ya que cada vez es más imperante tener acceso rápido a nuestros recursos, facilidades para realizar transacciones con otras empresas y/o personas, mantener la seguridad en los más altos niveles y hacerlo desde cualquier punto donde nos encontremos las 24 horas del día y los 7 días de la semana. Esto implica múltiples retos tecnológicos para las empresas que brindan este tipo de servicios. ¿Cómo mantener la evolución de los servicios haciéndolos cada vez más ágiles, seguros e innovadores y que a su vez esta evolución sea rentable? Es una de las preguntas que ronda repetidamente en los comités estratégicos de las instituciones financieras.

El constante cambio en las necesidades de los clientes, aunado a la sofisticación de dispositivos móviles y a las amenazas de seguridad en términos de información han impulsado la demanda de nuevos servicios o productos para satisfacer las necesidades del mercado y del entorno financiero (Gasca, Camargo y Medina, 2014). Parte de esta evolución tecnológica ha motivado a las organizaciones financieras a cambiar su tradicional enfoque de servicio basado en sucursales y buscar modelos de negocio orientados al autoservicio a través de dispositivos móviles que les permitan movilidad e interacción en línea con sus clientes. Esto puede representar nuevos retos para el sector financiero tanto en el ámbito tecnológico como en el de servicio al cliente.

Por su parte, Deloitte (2012) refiere que la banca móvil puede aportar beneficios para las instituciones financieras como ampliar el área de oportunidad para ofertar productos y servicios financieros, la reducción de costos en la entrega de productos y/o servicios a través de canales móviles y la mejora continua en el nivel de servicio del cliente. Además, los usuarios pueden obtener ventajas con el uso de canales móviles como la reducción en el costo de las comisiones bancarias, movilidad e interacción en línea así como simplicidad en trámites y servicios bancarios.

Dicha corporación señala que la banca móvil será un componente integral de la estrategia comercial de los bancos, ya que se convertirá en una necesidad más que en una elección por lo que las organizaciones deberán contemplar su inclusión en las plataformas de operación financiera con una visión de mejora continua que les permita impulsar oportunidades de crecimiento.

En este contexto, el estudio Global Mobile Banking Benchmark (2017), en el que se evaluaron 53 bancos de 18 diferentes países, señaló que el corporativo bancario mejor evaluado a nivel mundial, BBVA Bancomer, dispone de una banca móvil enfocada a proporcionar constante funcionalidad al cliente a través de procesos de mejora continua.

Subsecuentemente, los premios World Finance Digital Banking Awards 2018 que resaltan a las instituciones con sólidas estrategias digitales, por segundo año consecutivo destacaron a BBVA Bancomer, México, como mejor banco digital por su estrategia de innovación tecnológica que le permitió ampliar en un 35% su cartera de clientes digitales y posicionar el 70% de los servicios a través de canales digitales. En este sentido, también se le otorgó el distintivo a la organización bancaria por la mejor aplicación móvil, centrada en productos,

funcionalidades y servicios digitales basados en acciones de innovación tecnológica (World Finance, 2018).

Sin embargo, en la búsqueda de innovación y mejora continua de la banca móvil, algunas instituciones financieras presentan ciertos inconvenientes en el ciclo de fabricación de aplicaciones móviles en ambientes de desarrollo tradicionales, es decir, en donde las áreas de desarrollo y operaciones funcionan de manera aislada.

Tal situación deriva de la necesidad de implementar constantemente nuevos paquetes de software móvil debido a cambios de funcionalidad, mejoras de seguridad y/evolución de las necesidades en el mercado, lo que hace más complejo y lento el proceso de integración, pruebas, despliegue y mejora de aplicaciones, provocando a su vez el aumento del consumo de recursos tecnológicos y humanos.

Los marcos ágiles de desarrollo representan una oportunidad para optimizar la gestión del ciclo de desarrollo de software en la producción y entrega de productos funcionales al cliente en tiempos más cortos (Molina, 2012). No obstante, la falta de alineación, coordinación y colaboración entre las áreas de TI que intervienen en dicho proceso representa grandes retos tanto en la mejora continua de software, como en la entrega del valor al cliente en los tiempos estimados, lo que puede representar obstáculos para la rentabilidad y competitividad en el modelo de negocio.

Derivado de lo anterior, es recomendable que las áreas de TI involucradas en el proceso de fabricación, pruebas y despliegue de banca móvil cuenten con prácticas y herramientas que les permitan reaccionar de manera ágil y coordinada en contextos de constante cambio. También es importante que dichas áreas cuenten con las capacidades necesarias para

mantener ambientes estables y confiables que impulsen la mejora continua de software y aporten valor al modelo de negocio.

1.2 Planteamiento del problema

Las plataformas móviles como tendencia digital son un factor clave que las empresas deben considerar en su estrategia tecnológica para el posicionamiento de productos y servicios a través de soluciones digitales competitivas e innovadoras además de incrementar su abanico de canales de comunicación con el cliente final.

No obstante, las aplicaciones de banca móvil requieren de una permanente revisión y actualización de sus componentes por cambios en las necesidades o por la aparición de vulnerabilidades que ponen en riesgo la información de los clientes, lo que implica constante interacción entre diversas áreas de negocio, desarrollo y operativas consumiendo grandes cantidades de tiempo y recursos tecnológicos de la empresa. Dicha dinámica, además puede mermar esfuerzos encaminados a la mejora continua e innovación tecnológica, parte de la estrategia digital.

La creciente demanda de nuevas funciones y/o correcciones en las aplicaciones móviles de la banca, también ha provocado que las áreas involucradas en la liberación a producción de tecnología móvil se vean desbordadas en sus capacidades de atención, lo cual impacta en tiempos y costos de entrega al negocio. En otros casos, ciertas actividades manuales pueden ocasionar errores en el despliegue o lanzamiento de las aplicaciones generando retrasos significativos en la entrega de valor al cliente.

Aunado a ello, las áreas de desarrollo concentran sus esfuerzos en fabricar soluciones tecnológicas móviles con entregas frecuentes en tiempo y forma, sin embargo, al momento

de ejecutar el proceso de despliegue se pueden detener por una serie de procedimientos repetitivos y manuales que en ocasiones son más complejos que el propio desarrollo de la solución.

Asimismo, los departamentos de desarrollo, pruebas, seguridad e infraestructura se encargan de instrumentar, configurar y optimizar recursos para garantizar ambientes de desarrollo y producción confiables, operables y escalables, no obstante, esta tarea se vuelve cada vez más compleja por el incremento de plataformas, aplicaciones y las interacciones entre ellas, lo que representa un esfuerzo de gestión cada vez mayor para mantener el ecosistema tecnológico.

Finalmente, el tiempo de salida al mercado, también conocido como time-to-market¹ de los productos financieros móviles tiende a ser más corto debido a la competencia en el mercado, por lo tanto los mecanismos tradicionales de desarrollo de software pueden presentar limitaciones para responder de manera ágil al continuo flujo de actualizaciones.

En este sentido, Gasca, Camargo y Medina (2014) denotan que la implementación de aplicaciones móviles requiere de nuevas tendencias de desarrollo que se enfoquen en la solución de problemas específicos para la implementación de servicios con altos niveles de calidad en los tiempos esperados.

Bajo una nueva visión de colaboración y entendimiento organizacional, esta investigación busca identificar cuáles son los principales factores que impiden la entrega ágil y continua

¹ En marketing es el tiempo que pasa desde que se lanza un producto hasta que una actualización llega para sustituirlo. En un contexto tecnológico, la empresa deberá buscar el equilibrio para que esos tiempos se ajusten a las necesidades del consumidor, alineado con la estrategia de ventas. Recuperado de <https://www.reasonwhy.es/>

de aplicaciones de banca móvil y plantear un marco de trabajo basado en DevOps² que permita desarrollar capacidades de respuesta ágil para integrar, desplegar, entregar y mejorar continuamente productos y servicios de banca móvil para “Bancatec”³ con la calidad esperada.

El resultado principal de esta investigación es contar con un conjunto de principios, mejores prácticas y procesos para optimizar el ciclo de desarrollo de software de banca móvil a través de ambientes de trabajo flexibles y colaborativos que permitan dar continuidad y agilidad al modelo de negocio.

² Conjunto de prácticas de ingeniería de software que integran las áreas de desarrollo y operaciones para lograr ciclos de desarrollo más cortos, implementaciones más frecuentes, productos de mayor calidad y clientes más satisfechos. Recuperado de <https://devops.com/definition-devops-masses/>

³ Organización financiera que por razones de confidencialidad se denomina “Bancatec”.

1.3 Objetivos

Objetivo general

Identificar los principales factores que impiden la entrega ágil y continua de software de banca móvil entre las áreas que participan en el desarrollo y despliegue de productos financieros en Bancotec para desarrollar un entorno de trabajo “*Framework DevOps for Mobile Banking*” enfocado a la alineación, transformación y optimización de equipos de trabajo, procesos, tecnología y planes de seguimiento que permitan agilizar la entrega y mejora continua de aplicaciones móviles financieras.

Objetivos específicos

- Integrar una matriz de factores para identificar los principales aspectos que impiden la entrega ágil y continua de aplicaciones de banca móvil en Bancatec.
- Definición, principios y objetivos del “*Framework DevOps for Mobile Banking*” bajo directrices DevOps para alinear, transformar y optimizar los ambientes de trabajo que impulsen la entrega ágil y mejora continua de aplicaciones móviles.
- Diseño de los componentes del “*Framework DevOps for Mobile Banking*” centrados en equipos de trabajo, procesos de entrega continua, tecnología e indicadores clave que permitan mejorar el rendimiento de los equipos, desde la planificación hasta la liberación, eficientar la entrega de valor y dar seguimiento puntual a los resultados obtenidos.

1.4 Diseño de la investigación

Pregunta de investigación

¿Cuáles son los principales factores que impiden la entrega ágil y continua de aplicaciones de banca móvil en Bancatec?

La hipótesis

Los factores que impiden la entrega ágil y continua de aplicaciones de banca móvil en Bancatec pueden estar relacionados con la falta de alineación entre los equipos y la ausencia de automatización en los flujos de trabajo operativos y repetitivos.

La variable de estudio de la presente investigación se describe en la Tabla 1.

Tabla 1. **Descripción de la variable de estudio.**

Variable	Factores que impiden la entrega ágil y continua de aplicaciones de banca móvil en Bancatec.
Definición conceptual	Entorno a los elementos o aspectos que impiden el desarrollo y despliegue ágil de tecnología móvil.
Definición operacional	Se esperan obtener de los valores que arrojen los instrumentos de medición Cuestionario Diagnóstico y Entrevista Diagnóstica, ambos sobre despliegue y entrega continua de aplicaciones de Banca Móvil

La investigación se llevará a cabo a través de un modelo no experimental, transversal descriptivo⁴, con alcance exploratorio⁵ (Sampieri, Collado & Lucio, 2010) conformado por cinco fases de trabajo señaladas en la Tabla 2.

Tabla 2. Fases de la investigación.

1	Planteamiento del problema
2	Revisión de la literatura
3	Recolección de datos.
4	Análisis de los datos.
5	Reporte de resultados.

Con base en la investigación documental y el reporte de resultados producto del presente protocolo se propondrá un marco de trabajo basado en DevOps para mejorar el rendimiento de los ambientes de trabajo y agilizar la entrega y mejora continua de aplicaciones de banca móvil en el departamento de Servicios de Información Tecnológica adscrita a la Dirección General de Desarrollo y Pruebas. En dicha área se lleva a cabo parte de la planeación de la estrategia digital móvil, la administración del Data Warehouse⁶ y la gestión de aplicaciones relacionadas con Business Intelligence⁷.

⁴ No implica manipulación deliberada de variables, solo se observan los fenómenos en su ambiente natural. La recopilación de datos se hace en un único momento. Indaga la incidencia de las modalidades, categorías o niveles de una o más variables en una población. (Sampieri, Collado & Lucio, 2010).

⁵ Estudio centrado en analizar temas o problemas de investigación poco estudiados del cual se tienen dudas.

⁶ Base de datos corporativa alimentada por diversas fuentes de información y analizada desde diferentes perspectivas para facilitar la toma de decisiones en determinado ámbito, empresa u organización (Sinnexus).

⁷ Capacidad para transformar los datos en información y ésta en conocimiento con el fin de optimizar el proceso de toma de decisiones en los negocios (Sinnexus).

Los participantes serán integrantes de las áreas de sistemas, pruebas, infraestructura aplicativa de TI, seguridad y canales de negocio que intervienen en el proceso de integración, despliegue y aprobación de aplicaciones de banca móvil.

Los instrumentos de medición del presente protocolo de investigación estarán basados en:

- Cuestionario Diagnóstico de Despliegue de Aplicaciones de Banca Móvil que estará integrado por preguntas cerradas de opción múltiple, basado en la Escala de Likert⁸ (Sampieri, Collado & Lucio, 2010) con el fin de identificar la problemática que afecta el proceso de integración, entrega y despliegue ágil de aplicaciones móviles y la mejora continua así como la interacción entre los equipo de trabajo que participan. Se planea aplicarlo al personal de las áreas de desarrollo de sistemas, pruebas, seguridad de la información e infraestructura aplicativa con el fin de identificar los aspectos en los ambientes de trabajo que afectan la entrega ágil y mejora continua de aplicaciones móviles.
- Entrevista Diagnóstica sobre despliegue y entrega continua de aplicaciones de Banca Móvil integrado por una serie de preguntas abiertas dirigidas a los responsables de las áreas de desarrollo, pruebas, seguridad de la información e infraestructura. Con su aplicación se planea indagar sobre el flujo actual de actividades y procesos que implican la integración, despliegue y entrega continua de aplicaciones móviles y la interacción entre los equipos de trabajo.

⁸ Conjunto de ítems que se presentan en forma de afirmaciones para medir la reacción del sujeto en tres, cinco o siete categorías (Sampieri, Collado & Lucio, 2010).

La validez de los instrumentos de medición se llevará a cabo a través de un grupo de tres expertos en materia de tecnología ágil. Los instrumentos se ajustarán en caso de observaciones y se elaborará la versión final.

La aplicación de los instrumentos se llevará a cabo vía web a través de la plataforma Google Forms y por ese mismo medio se obtendrá la estadística de las respuestas para el análisis de los resultados.

1.5 Metodología

1.5.1 Descripción de la investigación

El presente protocolo se basa en un modelo no experimental transversal descriptivo, como se describió en el primer capítulo, y para la recolección de datos se diseñaron dos procedimientos a fin de recabar información vinculada con el proceso de integración, despliegue y entrega continua de aplicaciones de banca móvil en las áreas de Desarrollo de Sistemas, Pruebas, Seguridad e Infraestructura aplicativa de TI en la organización Bancatec. Los procedimientos se citan a continuación.

- a) Aplicación del instrumento de medición Cuestionario Diagnóstico sobre despliegue y entrega continua de aplicaciones de Banca Móvil, basado en la Escala de Likert⁹ (Sampieri, Collado & Lucio, 2010), integrado por preguntas cerradas de opción múltiple enfocado al personal de las áreas de Desarrollo de Sistemas, Pruebas,

⁹ Conjunto de ítems que se presentan en forma de afirmaciones para medir la reacción del sujeto en tres, cinco o siete categorías (Sampieri, Collado & Lucio, 2010).

Seguridad de la Información e Infraestructura Aplicativa del Departamento de Servicios de Información Tecnológica (DSIT) adscrita a la Dirección General de Desarrollo y Pruebas con el fin de identificar los principales factores que afectan el proceso de integración, despliegue y entrega ágil de aplicaciones móviles limitando la mejora continua así como la interacción entre los equipo de trabajo que intervienen. Ver Apéndice A.

- b) Aplicación del instrumento de medición Entrevista Diagnóstica sobre despliegue y entrega continua de aplicaciones de Banca Móvil al personal responsable de las áreas de Desarrollo de Sistemas, Pruebas, Seguridad de la Información e Infraestructura Aplicativa del DSIT con el fin de identificar a detalle el flujo de actividades y procesos que implican la integración, despliegue y entrega continua de aplicaciones móviles y la interacción entre los equipos de trabajo. Ver Apéndice B.

1.5.2 Recolección e integración de datos

El instrumento de medición Cuestionario Diagnóstico sobre despliegue y entrega continua de aplicaciones de Banca Móvil se aplicó al personal que participa en los departamentos de Desarrollo de Sistemas, Pruebas, Seguridad de la Información e Infraestructura Aplicativa del DSIT, de acuerdo con el orden de la Tabla 3.

Tabla 3. **Participantes del protocolo de estudio.**

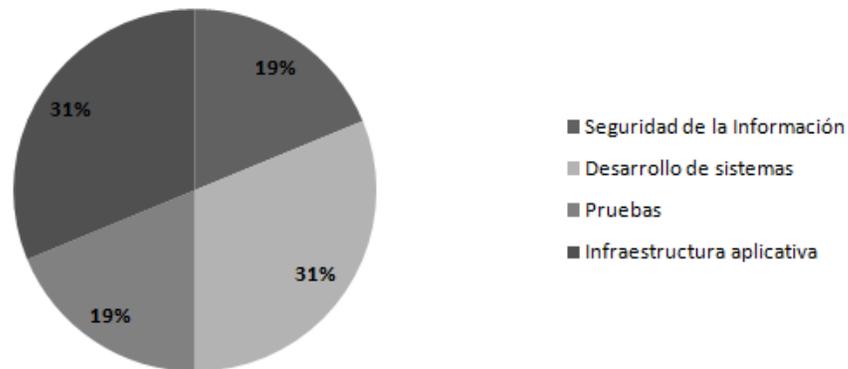
Equipo de trabajo	Perfil	No. de participantes	Coordinadores por área
-------------------	--------	----------------------	------------------------

● Desarrollo de sistemas	de Ing./Lic. Sistemas	5	1
● Pruebas	Ing. Pruebas	3	1
● Seguridad de la información	Experto Seguridad/Ing. SW	3	1
● Infraestructura	Ing. Redes/Sistemas	5	1
		Total	20

El cuestionario se aplicó a través de la plataforma Google Forms y, de la misma forma, por medio de esa interfaz se obtuvieron los resultados de la estadística con los siguientes hallazgos.

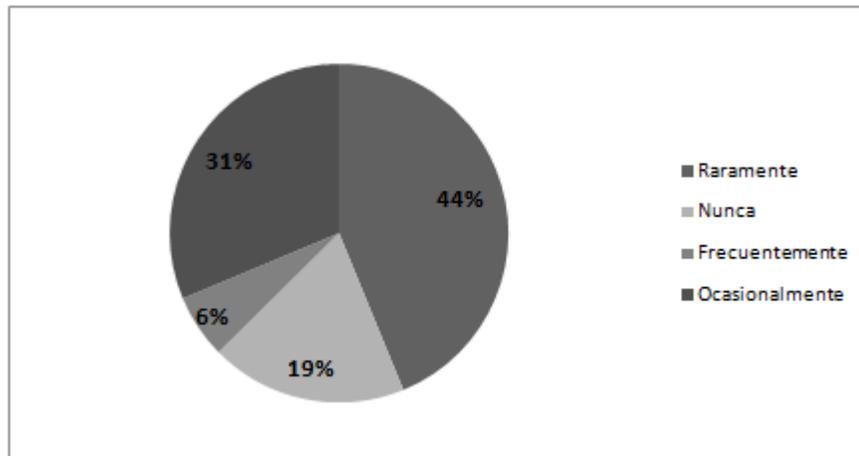
La distribución de participantes por área refleja que el 31% pertenece al área de desarrollo, el 31% a infraestructura, el 19% a pruebas y, de igual forma, el 19% a seguridad. Ver figura 1.

Figura 1. **Distribución de participantes por área.**



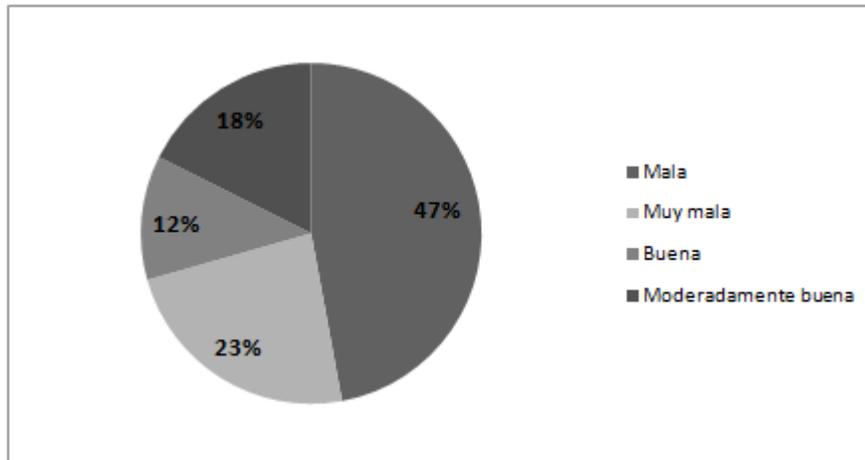
Se identificó que el 44% de los participantes consideran que los objetivos y principios de TI en su área están poco alineados con los equipos que intervienen en el proceso de integración, despliegue y entrega de software móvil. El 31% opinan que ocasionalmente los objetivos se alinean con todos los equipos de trabajo del DSIT. Ver figura 2.

Figura 2. **Percepción de alineación de objetivos de TI.**



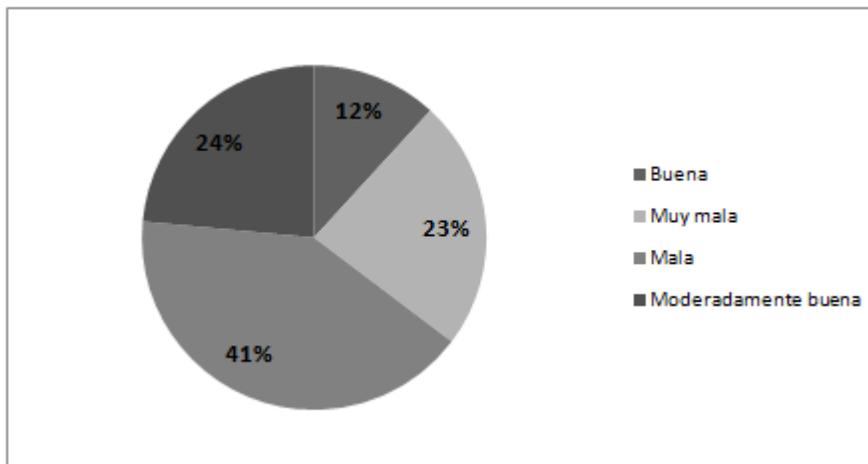
En cuanto a la gestión de promoción de ambiente para nuevas versiones de software móvil, el 47% de los integrantes de las áreas opinan que la comunicación y/o retroalimentación entre los equipos del DSIT es mala y poco clara. Subsecuentemente, el 23 % perciben que es muy deficiente. Ver figura 3.

Figura 3. **Percepción de la calidad de comunicación y retroalimentación entre equipos.**



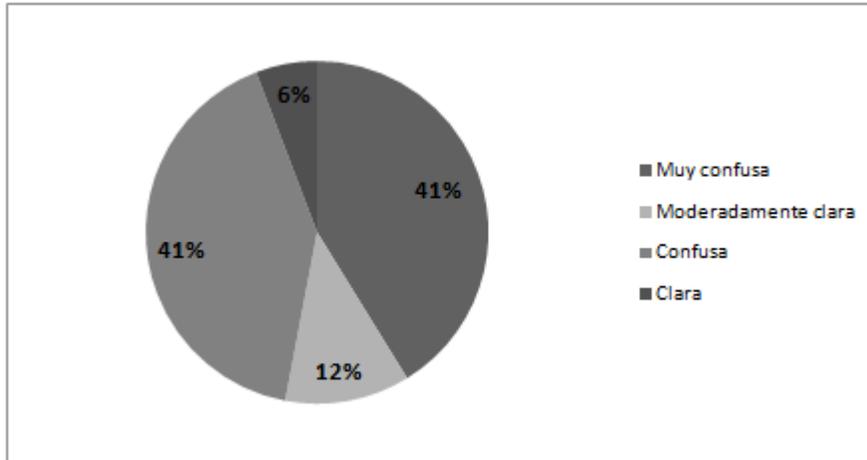
Con respecto al nivel de colaboración entre los diferentes equipos de trabajo, el 41% opinan que es deficiente y el 23% consideran que prácticamente no existe esa forma de trabajo entre ellos. Ver figura 4.

Figura 4. Percepción del nivel de colaboración entre los equipos de trabajo.



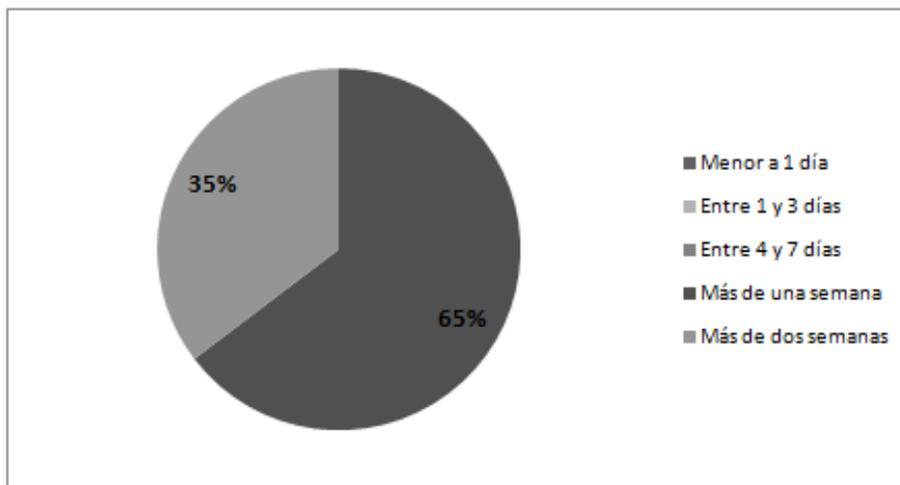
Los participantes perciben que falta claridad y definición puntual en los roles y las responsabilidades entre los integrantes de los equipos, el 41 % estiman que la distribución es confusa y el 41% valoran que es muy confusa. Ver figura 5.

Figura 5. Percepción en definición de roles y responsabilidades entre los equipos de trabajo.



Los equipos de trabajo estiman que el tiempo de respuesta por parte de infraestructura y seguridad para la promoción entre ambientes de software móvil es muy lento. El 65% consideran que pueden tardar más de una semana para responder a los requerimientos, por su parte, el 35% señalaron que pueden transcurrir hasta dos semanas para su liberación. Ver figura 6.

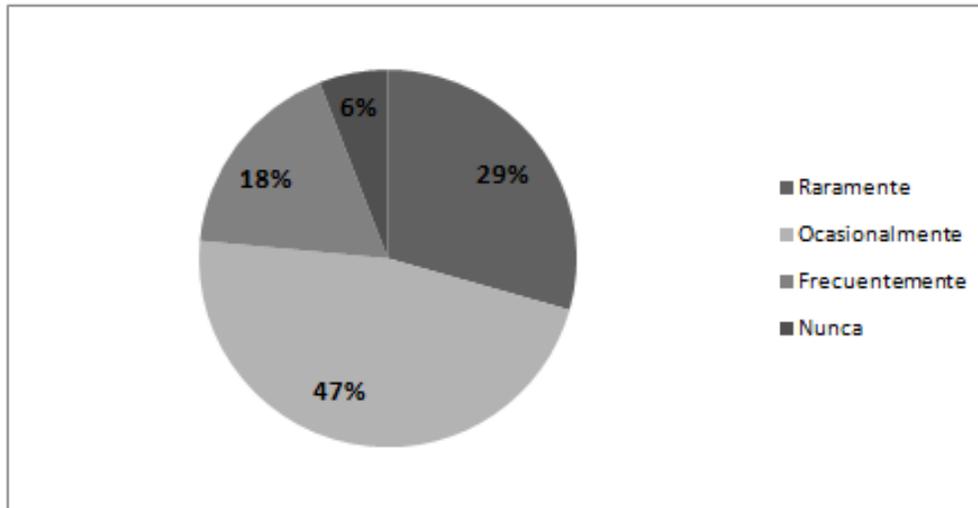
Figura 6. **Tiempo de respuesta de los equipos de infraestructura y seguridad para la promoción ente ambientes.**



El 47% del equipo de desarrollo consideran que ocasionalmente se ha logrado acelerar el desarrollo de nuevas funcionalidades de software móvil pero la puesta en producción no ha

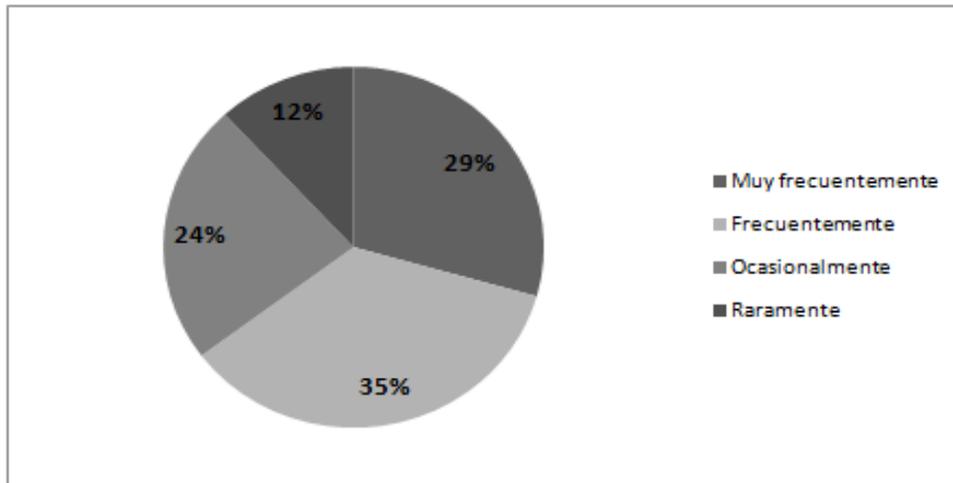
podido seguir el mismo ritmo. El 29% señalaron que esto ha sido menos frecuente. Ver figura 7.

Figura 7. Se logra acelerar el desarrollo de versiones de software móvil pero la puesta en producción no sigue el mismo ritmo.



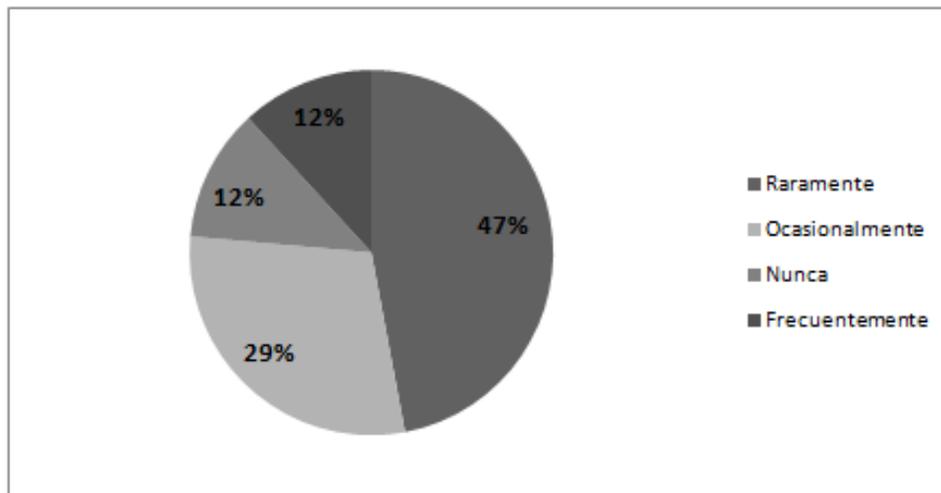
El equipo de infraestructura considera que frecuentemente (35%) se presentan incidencias vinculadas con la calidad del software de banca móvil que se despliega, generando trabajo adicional a los equipos de infraestructura e incrementando el trabajo en proceso. Otros señalaron que son ocasionales (23%) este tipo de incidencias. Ver figura 8.

Figura 8. Identificación, por parte de infraestructura, de incidencias vinculadas con la calidad del software de banca móvil que se despliega, generando trabajo adicional.



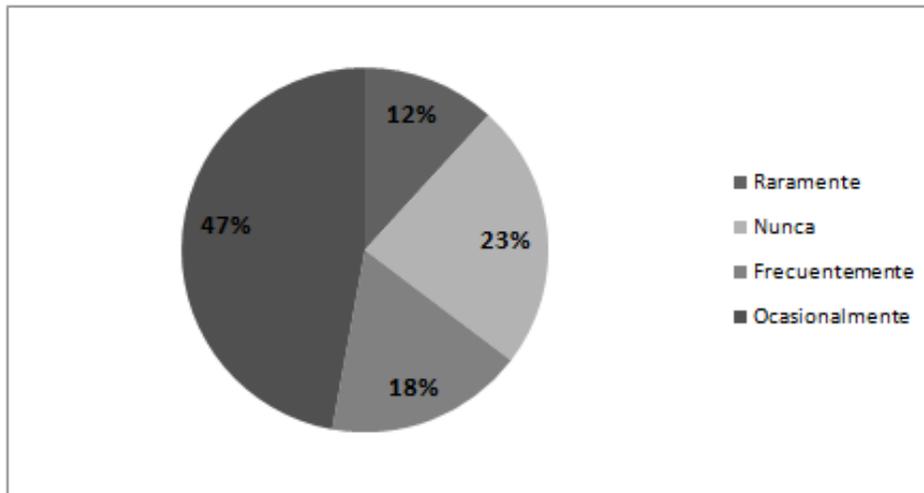
El 47% estiman que el área de seguridad no interactúa oportunamente a lo largo del proceso de integración, despliegue y entrega de paquetes de software móvil. El 29% consideran que ocasionalmente interactúa en etapas tempranas. Ver figura 9.

Figura 9. Participación del equipo de Seguridad a lo largo del proceso de integración, despliegue y entrega de software móvil.



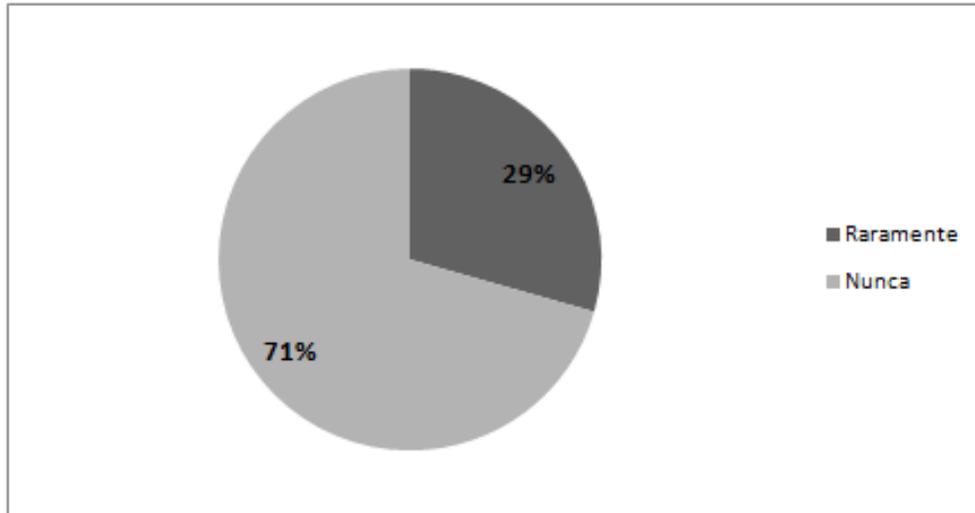
Los equipos de trabajo opinan, 47%, que ocasionalmente se detectan errores/incidencias de software móvil en etapas tempranas (desarrollo, integración y pruebas) y dan atención rápida. Por su parte, el 18% indican que la detección y atención es frecuente. Ver figura 10.

Figura 10. **Detección de errores/incidencias de software móvil en etapas tempranas (desarrollo, integración y pruebas) y atención rápida.**



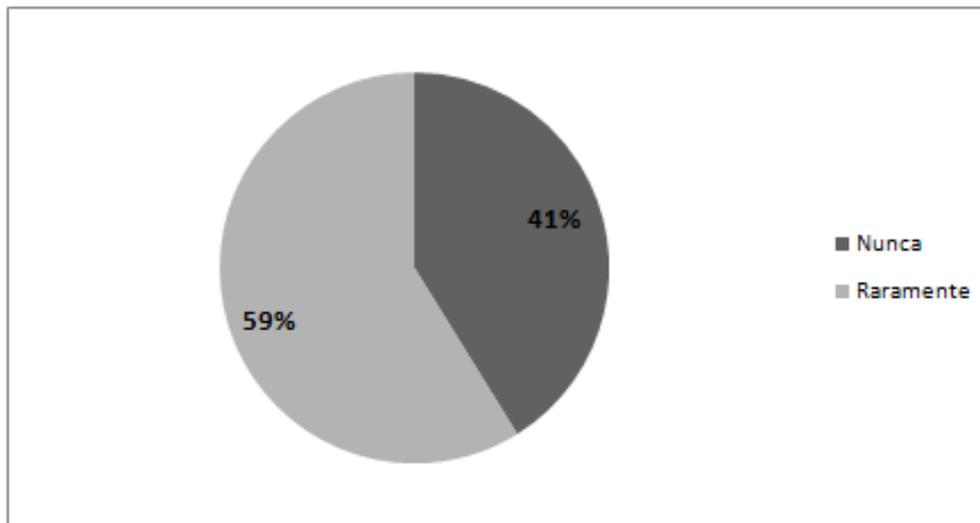
En cuanto al empleo de herramientas para automatizar la promoción entre ambientes (versionamiento/copiado de código, compilación y deploy) para nuevas versiones de software móvil, el 70% aseguraron que no tiene acceso a ellas y el 29% señalaron que esporádicamente las emplean para ese fin. Ver figura 11.

Figura 11. **Herramientas que automatizan la promoción entre ambientes (versionamiento/copiado de código, compilación y deploy) para nuevas versiones de software móvil.**



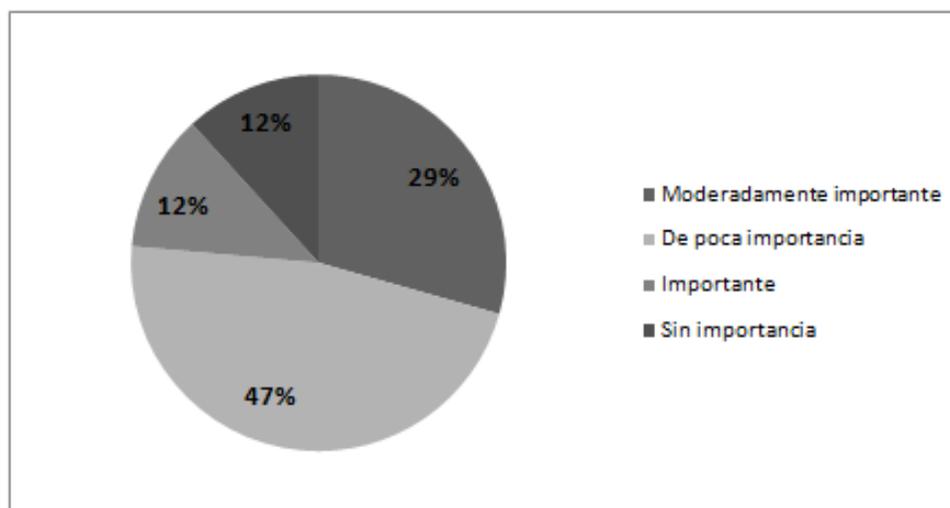
Con respecto al uso de herramientas para la automatización de pruebas en nuevas versiones de software móvil, en los ambientes de desarrollo y operaciones, el 59% consideran que esporádicamente las emplea y el 41% señalaron que no tienen acceso a ellas. Ver figura 12.

Figura 12. **Herramientas que automatizan las pruebas de nuevas versiones de software por desplegar en los ambientes de desarrollo u operaciones.**



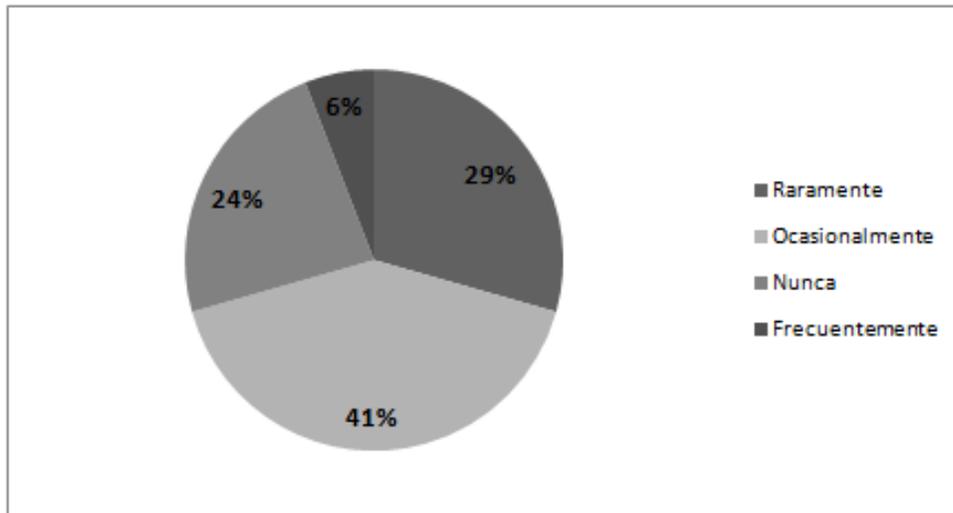
El 47% de los integrantes de los equipos de trabajo estiman que su participación en el proceso de integración, despliegue y entrega de software móvil para impulsar la mejora continua es de poca importancia. El 29% consideran que es moderadamente importante. Ver figura 13.

Figura 13. **La relevancia de las actividades de los integrantes de los equipos de trabajo en el proceso de integración, despliegue y entrega de software móvil para impulsar la mejora continua.**



El 41% de los integrantes de los equipos de trabajo señalaron que se sienten ocasionalmente motivados al realizar las actividades vinculadas con su rol. El 29% estiman que sienten poco motivados. Ver figura 14.

Figura 14. **Nivel de motivación de los integrantes de los equipos de trabajo al realizar las actividades vinculadas con su rol.**



El instrumento de medición Entrevista Diagnóstica sobre despliegue y entrega continua de soluciones de Banca Móvil se aplicó al personal responsable de las áreas del DSIT, de acuerdo con el orden de la Tabla 4.

Tabla 4. **Participantes del protocolo de estudio.**

Equipo de trabajo	Perfil	Coordinadores por área
● Desarrollo de sistemas	Ing, Sistemas	1
● Pruebas	Ing. Pruebas	1
● Seguridad de la información	Ing. Redes/SW	1
● Infraestructura	Ing. Sistemas/Redes	1
Total		4

La entrevista se aplicó *in situ* en las instalaciones de TI de la organización financiera y se identificaron puntualmente las siguientes problemáticas o áreas de oportunidad.

- Las áreas de desarrollo de sistemas e infraestructura aplicativa trabajan como unidades independientes, en donde la primera busca la constante promoción de nuevas funcionalidades y la segunda trata de disminuir la ejecución de dichos cambios/incidencias para mantener la estabilidad y confiabilidad de las plataformas.
- A pesar de que se llevan a cabo pruebas en etapas tempranas durante el ciclo de desarrollo, diversos incidentes son detectados hasta el entorno de producción, generando cargas excesivas de trabajo para infraestructura.
- El proceso de promoción de nuevas funcionalidades de aplicaciones de banca móvil se enfrenta a retrasos significativos para el despliegue y la entrega. Por un lado, implica actividades manuales y repetitivas y, por el otro, el área de infraestructura tiene mucho trabajo acumulado por atender.
- Aunque existe comunicación y retroalimentación básica entre los coordinadores de las áreas, los equipos de trabajo no se visualizan como parte de un gran equipo que apunta hacia el mismo objetivo.

Así mismo, los coordinadores describieron los componentes de arquitectura que intervienen en el proceso de despliegue y entrega continua de software móvil, la metodología que emplean durante el ciclo de desarrollo de aplicaciones de Banca Móvil y las principales problemáticas a las que se enfrentan para el despliegue ágil e impulsar la mejora continua. Dicha información se esquematiza en el siguiente apartado.

1.5.3 Análisis de la información

Con base en la aplicación de los instrumentos de medición descritos anteriormente se consolidaron los datos y se llegaron a los siguientes resultados.

Diversos son los componentes de arquitectura que se requieren para garantizar los servicios de la banca móvil (consulta de saldo, transferencias interbancarias, estados de cuenta, etc.) y son administrados por las áreas de operaciones e infraestructura de TI, en coordinación con los departamentos de sistemas, pruebas y seguridad en Bancatec.

En el diagrama de la figura 34 se ilustran los principales componentes de arquitectura y el flujo de información que se requieren para desplegar dichos servicios. Para realizar una operación desde el aplicativo móvil se requiere una conexión a Internet para ejecutar llamadas a los servicios de banca móvil las cuales son atendidas por un conjunto de adaptadores montados en Mobile First Platform¹⁰, dichos adaptadores contienen la lógica transaccional de cada operación e interactúan con un modelo de datos optimizado y diseñado con los mínimos datos del cliente necesarios que permiten realizar la transacción. Para explotar los datos generados por la aplicación se integra un proceso ETL¹¹ que permite almacenar la historia de manera organizada en el Data Warehouse¹², con lo cual se puede combinar la información transaccional del cliente con el resto de sus datos procedentes de los sistemas Core.

Al realizar una operación desde el aplicativo móvil los adaptadores interactúan con los servicios de los sistemas Core, por ejemplo, al realizar una transferencia interbancaria los

¹⁰Es una plataforma IBM abierta y completa para teléfonos inteligentes y tabletas que permite desarrollar, probar, conectar, ejecutar y gestionar de manera eficiente aplicaciones móviles. en organizaciones de todo tipo. Recuperado de <https://www.ibm.com/support/knowledgecenter/es/SSNJPX/welcome.html>

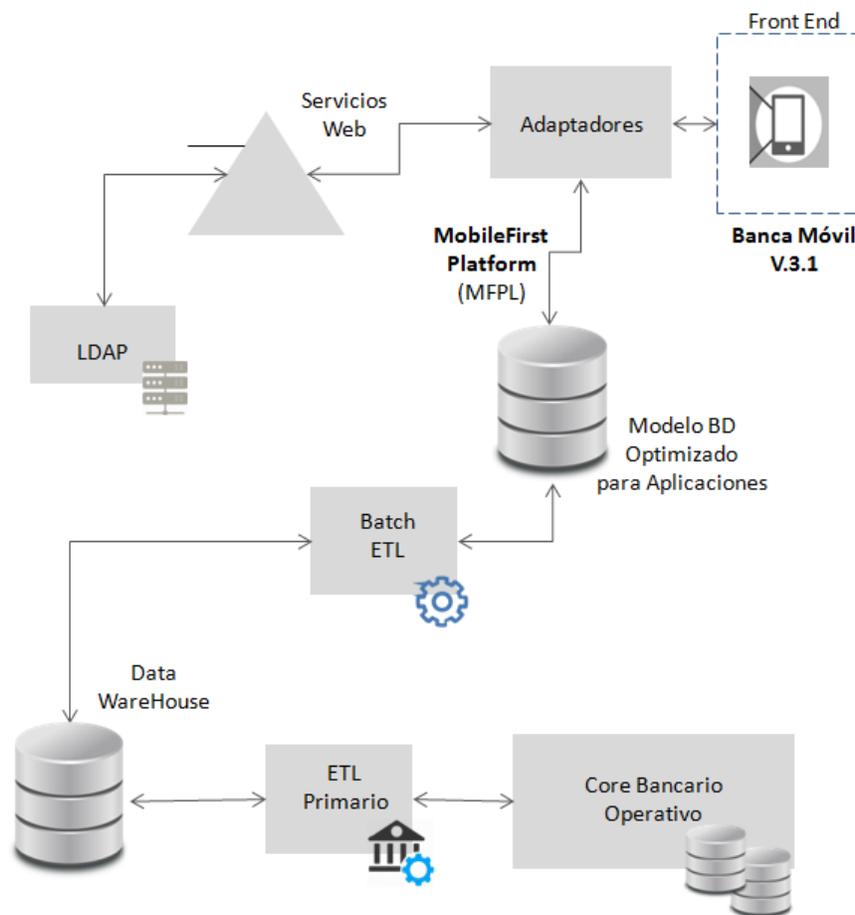
¹¹ Proceso de extracción, transformación y carga, frecuentemente abreviado ETL, busca integrar datos para combinarlos desde múltiples fuentes y construir un almacén de datos. Recuperado de https://www.sas.com/en_us/insights/data-management/what-is-etl.html

¹² Es un repositorio o área de almacenamiento donde todos los datos de la empresa se guardan en un solo lugar. Puede incluir datos de diferentes fuentes, así como datos actuales e históricos de plataformas heredadas.

servicios deben orquestar las transacciones necesarias para realizar el cargo en la cuenta del cliente y la transferencia al destinatario.

El coordinador del área de infraestructura aplicativa indicó que cuando se requiere desplegar nuevas versiones de banca móvil el proceso de integración se vuelve complejo generando interdependencia entre las áreas y cuellos de botella, como se detalla más adelante.

Figura 15. Diagrama de arquitectura de componentes de aplicaciones móviles en Bancatec.



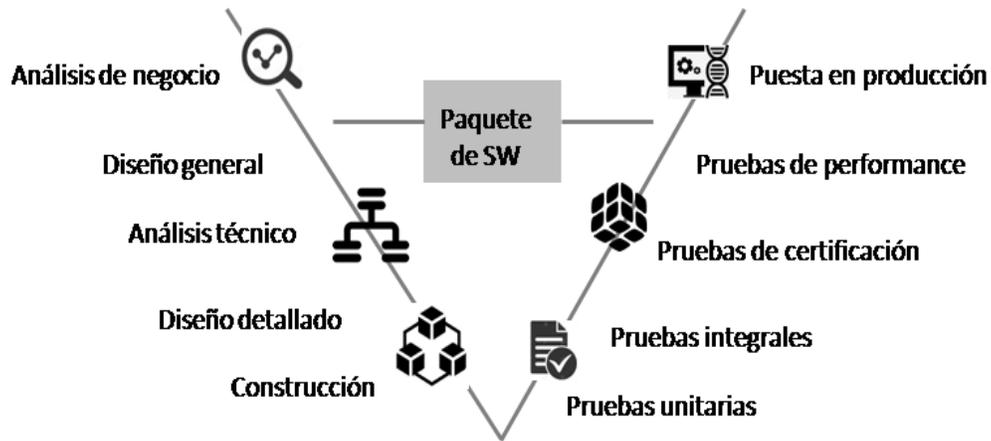
Por su parte, el coordinador del área de sistemas refirió que el personal involucrado en el proceso de desarrollo de software de banca móvil en Bancatec emplean el método V¹³, el cual es considerado una abstracción simplificada de las actividades del ciclo más que un modelo completo y preciso de desarrollo.

En el primer vector se presentan a modo de síntesis las actividades de análisis que descomponen las necesidades de negocio en partes pequeñas y manejables. En el vector opuesto se despliegan las actividades que agregan y prueban los paquetes de software en construcción.

Buscan hacer énfasis en la verificación y validación de los requerimientos con actividades de prueba tempranas para facilitar la detección de errores de manera anticipada. Es decir, una vez que cuentan con el diseño general, conforme se construye el paquete de software, el departamento de sistemas aplica las pruebas unitarias y, subsecuentemente, las áreas de operación y calidad realizan las pruebas integrales, de certificación y performance para concluir con el despliegue del paquete en producción. Ver diagrama de la figura 16.

Figura 16. **Modelo V para el ciclo de desarrollo de software móvil en Bancatec.**

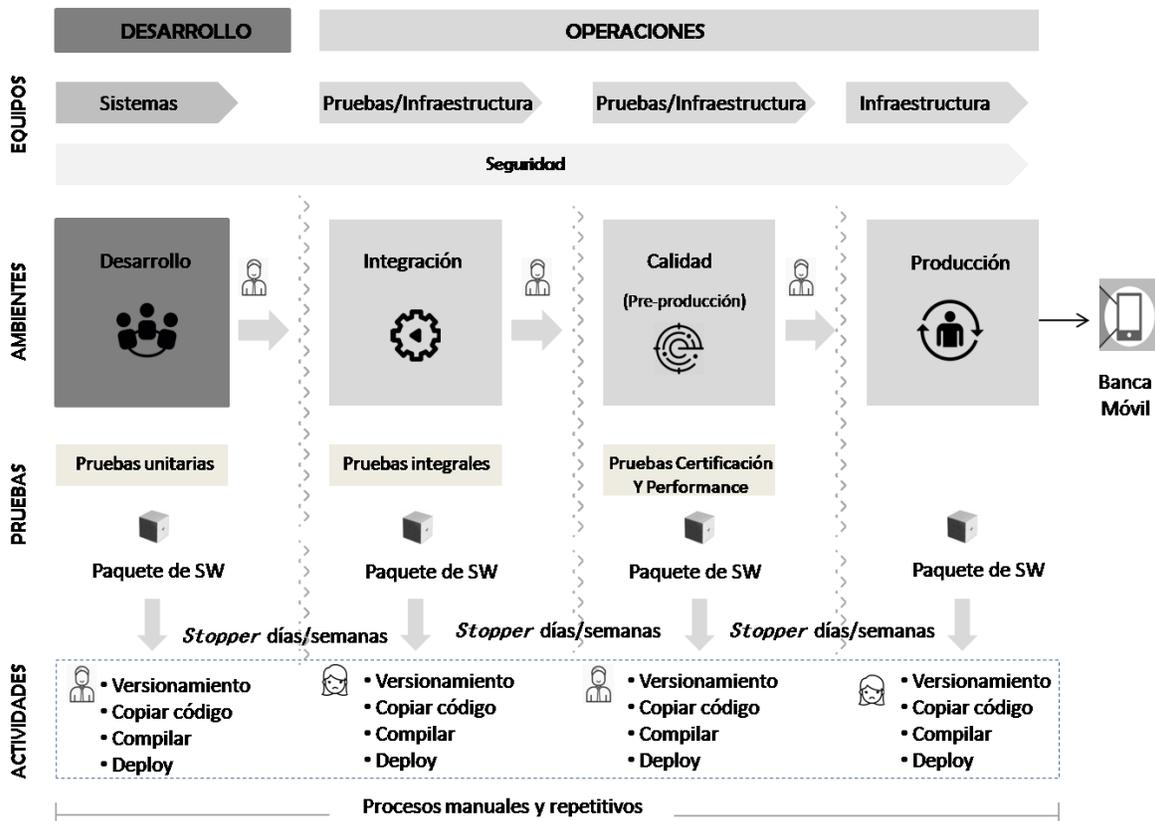
¹³ Donald Firesmith (2013). Recuperado de https://insights.sei.cmu.edu/sei_blog/2013/11/using-v-models-for-testing.html



Sin embargo, cuando se trata de incrementos o actualizaciones en aplicaciones móviles, por cuestiones de funcionalidad, seguridad o cambio de necesidades en el mercado y considerando la complejidad de la arquitectura de los componentes, el número de promociones entre entornos se dispara, haciendo lento y complejo el proceso de integración, entrega y despliegue de aplicaciones móviles.

Según los datos recolectados, la promoción entre ambientes se hace de forma manual y requiere un elevado número de pasos. El flujo de trabajo entre sistemas, pruebas, seguridad de la información e infraestructura aplicativa se convierte en una serie de actividades manuales y repetitivas que obstaculizan el despliegue continuo de paquetes de software. El principal problema es que los tiempos del proceso de ejecución de los pasos de ambiente tardan días e incluso semanas en lugar de horas y/o minutos, como se observa en la figura 17.

Figura 17. Ciclo de despliegue de software móvil en Bancatec.



El área de sistemas se enfoca en construir piezas de software con base en los requerimientos de negocio y efectuar las pruebas unitarias correspondientes en el ambiente de desarrollo. Una vez concluidas de manera exitosa ejecuta la copia del código, la compilación y el deploy del paquete. Después, solicita la promoción en el entorno de integración al área de seguridad e infraestructura. Sin embargo, la autorización puede tardar días o semanas debido a la carga de trabajo que tiene el área de operaciones. Una vez que el equipo de sistemas obtiene los permisos, deposita el paquete en el ambiente de integración.

El equipo de infraestructura toma el paquete del código y lo implementa en el ambiente de integración con los componentes de arquitectura necesarios que aseguren su correcta funcionalidad. Posteriormente, el personal de **pruebas** integrales realiza los test programados

y una vez concluidos, ejecuta la copia del código, la compilación y el deploy del paquete. Subsecuentemente, solicita permiso al área de seguridad e infraestructura para la promoción en el ambiente de calidad, lo cual también puede tardar días o semanas. Una vez que lo aprueban, el equipo de pruebas deposita el paquete en el ambiente de calidad.

El equipo de infraestructura toma la pieza de código y la implementa en el ambiente de calidad con los componentes de arquitectura necesarios que aseguren su correcta operación. El equipo de **calidad** realiza las pruebas de certificación y performance programadas. Cuando son concluidas de manera exitosa, dicha área realiza la copia del código, la compilación y el deploy del paquete. Solicita la autorización de seguridad e infraestructura para la promoción en el ambiente de producción. Finalmente, después de días o semanas el equipo de infraestructura, con el visto bueno del área de seguridad, despliega el paquete de software de banca móvil en ambiente de producción.

En síntesis y con base en la información analizada, se identificó que las actividades rutinarias y manuales durante el proceso de despliegue y entrega de software móvil restan agilidad a la entrega continua y merman la calidad del software final.

En cuanto a la forma de organización entre las áreas de trabajo, se detectaron diversas áreas de oportunidad para lograr entornos laborales colaborativos y más eficientes.

- **Silos entre desarrollo, infraestructura y seguridad.** Los equipos de trabajo operan de manera aislada y de forma independiente sin una estrategia en común generando procesos burocráticos para cada promoción. Se consideran equipos opuestos.
- **Falta de automatización de actividades manuales y repetitivas.** A lo largo del proceso de integración, despliegue y entrega continua de software móvil se realizan

diversas actividades manuales reiterativamente generando saturación para ciertas áreas y tiempos ociosos para otras. El trabajo está propenso a errores manuales restando agilidad al negocio.

- **Generación de desperdicios y demoras en despliegue de paquetes.** Las incidencias son detectadas hasta el entorno de producción encolando el trabajo que estaba planificado, ocasionando desperdicio de recursos y disminuyendo esfuerzos para mejora continua. Hay sobrecargas de trabajo para operaciones producto de errores manuales.
- **Ausencia de responsabilidades punto a punto.** Es complicado identificar si los problemas provienen de sistemas, seguridad, pruebas o infraestructura, ya que la mayoría surgen en la integración y promoción de entorno. Sistemas considera que su trabajo y responsabilidad ha terminado cuando se promociona a producción.
- **Ausencia de cultura integradora.** El pensamiento de los equipos de trabajo es limitando, se enfocan en ejecutar procesos aislados y no consideran la construcción completa de un producto o servicio. Falta de colaboración y objetivos en común. Se enfocan en buscar culpables más que en identificar soluciones.
- **Falta de motivación.** Los equipos trabajan con falta de estímulo para realizar el rol de sus actividades y no tienen clara la importancia de su participación en el proceso de integración, despliegue y entrega de software móvil para impulsar la calidad y mejora continua.

II. Marco teórico de la investigación

2.1 Introducción

Con la consolidación del Banco de México en la década de los años veinte (Banxico, 2018) como organismo central regulador y gestor financiero del gobierno federal inicia la constitución formal del sistema financiero mexicano lo que conlleva la delineación de directrices a seguir con el fin de reactivar el crédito a nivel nacional e instrumentar la monetización en el país.

Con el tiempo, la transición de la moneda a la emisión de billetes, la inestabilidad internacional durante la Segunda Guerra Mundial, la creación de la banca múltiple, los procesos de privatización, la devaluación del peso mexicano, la apertura a la banca extranjera, la llegada de nuevas tecnologías, así como el fortalecimiento del marco jurídico regulatorio mexicano, entre otros factores, reconfiguraron el sistema financiero mexicano (Clavellina, 2013).

La llegada de nuevas tecnologías como Internet y posteriormente los teléfonos inteligentes con software móvil, fungieron como propulsores del cambio en las instituciones bancarias, para revolucionar el enfoque de servicio basado en sucursales o centros de atención a plataformas móviles.

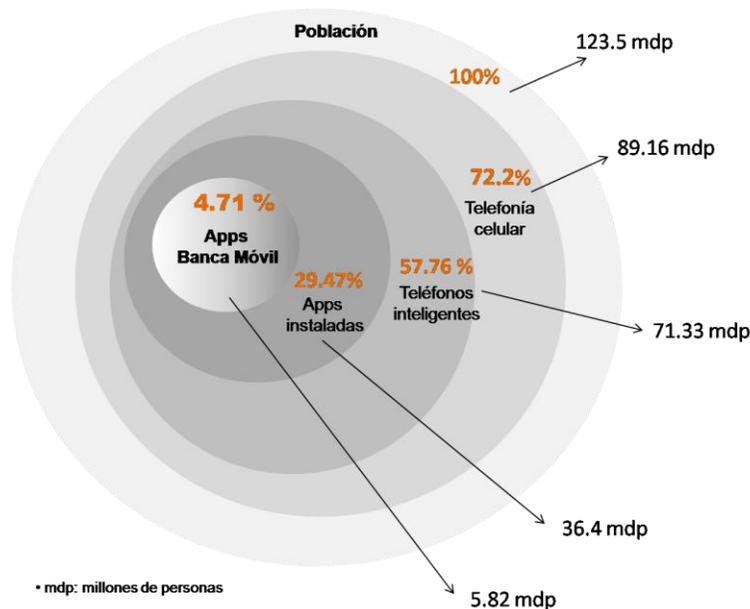
Según Forbes México¹⁴, en 2006 se lleva a cabo la implementación de los primeros canales de acceso a servicios y productos bancarios a través de aplicaciones móviles. En 2013, se marca un hito en la banca mexicana con la consolidación de la banca móvil nacional para

¹⁴ Forbes México. Recuperado de <https://www.forbes.com.mx/brand-voice/historia-banca-perfeccionamiento-las-sucursales-desde-siglo-xix-al-xxi/>

facilitar y optimizar la gestión de recursos financieros. Ejemplo de ello es la institución BBVA Bancomer que logra posicionarse en ese año como líder en servicios digitales móviles.

Con respecto a la distribución de TI entre la población en 2017, el Instituto Nacional de Estadística y Geografía indicó en la Encuesta Nacional sobre Disponibilidad y Uso de Tecnologías de la Información en Hogares (ENDUTIH, 2017), considerando una población de 123.5 millones de personas (mdp) en ese año, que el 57.73% (71.3 mdp) cuenta con internet, el 72.2 (89.16 mdp) tiene un teléfono celular y el 57.76% (71.33 mdp) dispone de un celular inteligente. Asimismo, el 29.47% (36.4 mdp) instaló aplicaciones en sus teléfonos y el 4.71 0% (5.82 mdp) instaló alguna aplicación de acceso a la banca móvil. Ver figura 18.

Figura 18. Gráfica de distribución de telefonía celular inteligente y uso de la banca móvil según datos ENDUTIH (2017).



Con base en la estadística se estima que existen 71.3 millones de teléfonos inteligentes en uso al 2017, uno de cada 2 mexicanos (57.76%) tiene acceso a este tipo de dispositivos, aunado al 57.73% de los habitantes que cuentan con servicio de datos y/o Internet y solo el 4.7% de la población total cuenta con servicios bancarios móviles, por lo tanto las aplicaciones móviles representan un canal potencial para penetrar la población mexicana con la banca digital, partiendo de una brecha inicial u oportunidad de mercado aproximadamente del 53%.

Las soluciones móviles han reinventado la manera en que el cliente puede interactuar con la banca, convirtiéndose en un canal dinámico de acceso al sistema financiero que prácticamente puede llevarse en el bolsillo. No obstante, para las instituciones que soportan la infraestructura de estas aplicaciones se ha convertido en un reto tanto tecnológico como de seguridad, ya que la arquitectura que las soporta es cada vez más compleja, los lenguajes de programación y los dispositivos móviles evolucionan a velocidades vertiginosas y la competencia entre instituciones tiende a ser más cerrada. Por lo tanto, es imprescindible contar con mecanismos ágiles, precisos y confiables para el desarrollo y liberación de estas aplicaciones.

2.1.1 Sistema Financiero Mexicano

El sistema financiero mexicano es un conjunto de organismos y mecanismos que permiten la gestión de la inversión del ahorro y está integrado por diversos sectores y entidades, tanto públicas como privadas. El objeto de dicho sistema es buscar la asignación eficiente de recursos entre ahorradores y demandantes de crédito. De acuerdo con la Comisión Nacional

Bancaria y de Valores (CNBV)¹⁵, órgano supervisor y desconcentrado de la Secretaría de Hacienda y Crédito Público con autonomía y facultades ejecutivas, los integrantes del sistema se enlistan en la Tabla 5.

Tabla 5. Sectores y entidades del sistema financiero mexicano (Septiembre 2019).

Sector	Entidades en operación
Entidades supervisadas de manera integral	2,327
Sociedades controladoras de Grupos Financieros	23
Instituciones de Banca Múltiple	51
Banca de Desarrollo y Entidades de Fomento	16
Sociedades Financieras de Objeto Múltiple (Sofom ER)	47
Organizaciones y Actividades auxiliares de Crédito	25
Uniones de Crédito	85
Sociedades Cooperativas de Ahorro y Préstamo (Socaps)	159
Sociedades Financieras Populares (Sofipos)	44
Otras entidades del Sector de Ahorro y Crédito Popular	7
Casas de Bolsa	36
Fondos de Inversión	763

¹⁵ Órgano desconcentrado de la SHCP, con facultades en materia de autorización, regulación, supervisión y sanción sobre los sectores que integran el sistema financiero mexicano <https://www.gob.mx/cnbv>

Instituciones Calificadoras de Valores	7
Emisoras	648
Bolsas de Valores y de Contratos de Derivados	2
Contrapartes centrales	2
Instituciones para el Depósito de Valores	1
Otros participantes del mercado de Valores y de Derivados	65
Participantes en redes de medios de disposición relevantes	126
Sociedades de Información Crediticia	3
Asesores en Inversión	109
Otras entidades supervisadas	108
Entidades sujetas únicamente a supervisión en materiales de PLD/FT	2,628
Sociedades Financieras de Objeto Múltiple, Entidades no Reguladas (Sofom ENR)	1,628
Centros Cambiarios	976
Transmisores de dinero	44
TOTAL	4,955

Nota: Fuente <https://www.gob.mx/cnbv/acciones-y-programas/sectores-supervisados?idiom=es>

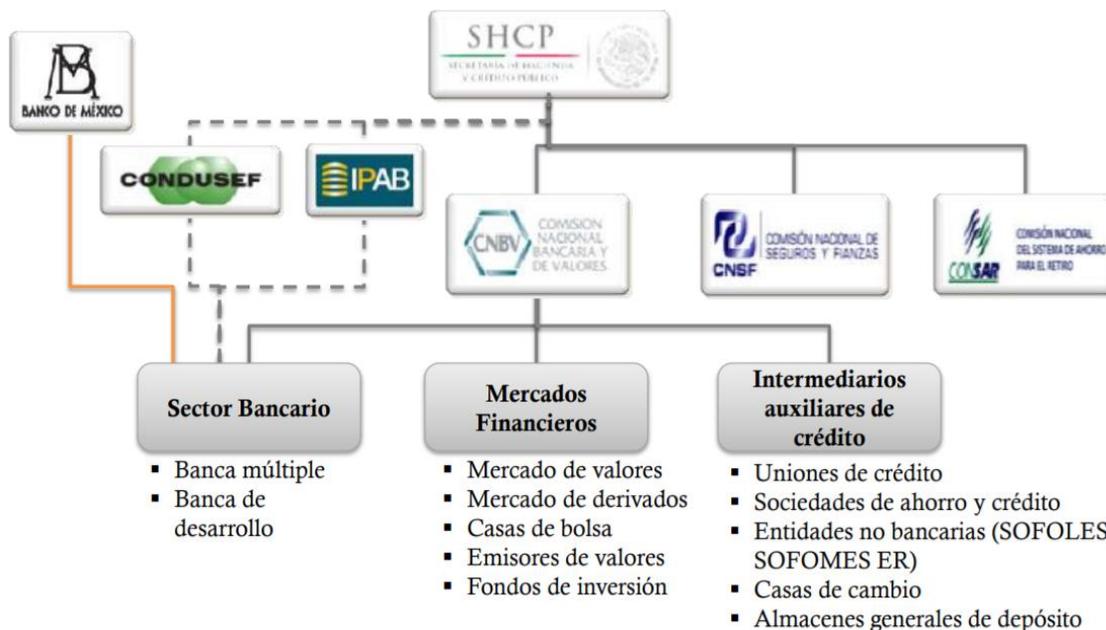
Asimismo, De Alba (2005) refiere que el sistema financiero mexicano se integra por la Secretaría de Hacienda y Crédito Público (SHCP), el Banco de México, las instituciones de la banca múltiple y las instancias de la banca de desarrollo, las sociedades financieras autorizadas para operar como entidades de ahorro y crédito popular, las sociedades cooperativas de ahorro y préstamo, los fideicomisos públicos del gobierno federal y los organismos o las instancias autorizadas para la supervisión la operación financiera.

2.1.2 Estructura del sistema financiero

La organización del sistema financiero mexicano tiene su base en la SHCP de la cual se desprenden la CNBV, la Comisión Nacional de Seguros y Finanzas (CNSF) y la Comisión Nacional del Sistema de Ahorro para el Retiro (CONSAR). Dicha estructura también cuenta con instancias auxiliares de supervisión y vigilancia como son la Comisión Nacional para la Protección y Defensa de los Usuarios de Servicios Financieros (CONDUSEF) y el Instituto para la Protección al Ahorro Bancario (IPAB).

Asimismo, el Banco de México y la CNBV son las instancias reguladoras del sector bancario y éste a su vez se integra por la banca múltiple y la banca de desarrollo, como se muestra en la figura 19.

Figura 19. **Estructura del sistema financiero mexicano.**



Nota. Fuente:

cnbv.gob.mx/PRENSA/Presentaciones%20y%20Discursos/20130826%20ITAM%20Sistema%20financiero%20mexicano.pdf

En el presente protocolo de investigación, nos enfocaremos en la banca múltiple, la cual consiste en el conjunto de bancos con un giro comercial que ofrecen productos y servicios para cubrir las necesidades financieras de la población. Actualmente¹⁶ se conforma por 51 instituciones bancarias autorizadas y en operación, que cumplen con estándares internacionales en materia de regulación y son supervisados por la CNBV. Las instituciones bancarias que conforman dicha banca se despliegan en la tabla 6.

La banca múltiple se agrupa en dos grandes rubros, ahorro y crédito, que gestiona según los requerimientos de la población desde distintos niveles de atención. Además, ejerce una importante tarea dentro del sistema financiero porque permite contactar a proveedores y

¹⁶ <https://www.gob.mx/cnbv/acciones-y-programas/banca-multiple>

consumidores de recursos financieros, con apego al sistema nacional de producción y consumo de bienes y servicios.

Tabla 6. Instituciones de la banca múltiple (septiembre 2019).

BANCOS EN OPERACIÓN			
1	ABC Capital	26	Credit Suisse
2	Accendo Banco	27	Deutsche Bank
3	Actinver	27	Famsa
4	Afirme	29	Finterra
5	American Express	30	Forjadores
6	Autofin	31	Fundación Dondé
7	Azteca	32	HSBC
8	Bajío	33	ICBC
9	Banco S3	34	Inbursa
10	Bancoppel	35	Inmobiliario Mexicano
11	Bancrea	36	Intercam
12	Bank of America	37	Invex
13	Bank of China	38	JP Morgan
14	Bankaool	39	Mifel

15	Banorte	40	Mizuho Bank
16	Banregio	41	Monex
17	Bansi	42	MUFG
18	Bardays	43	Multiva
19	Base	44	Pagatodo
20	BBVA Bancomer	45	Sabadell
21	BIA Afirme	46	Santander
22	CI Banco	47	Scotiabank
23	Citibanamex	48	Shinhan
24	Compartamos	49	Ve por Más
25	Consubanco	50	VW Bank
		51	KEB Hana México

Nota: Fuente <https://www.gob.mx/cnbv/acciones-y-programas/banca-multiple>

Por su parte, la banca de desarrollo está compuesta por instituciones gubernamentales que apoyan a sectores de forma particular y su misión es generar las condiciones para lograr un México más productivo. Ver Tabla 7.

Tabla 7. **Instituciones de la banca de desarrollo (septiembre 2019).**

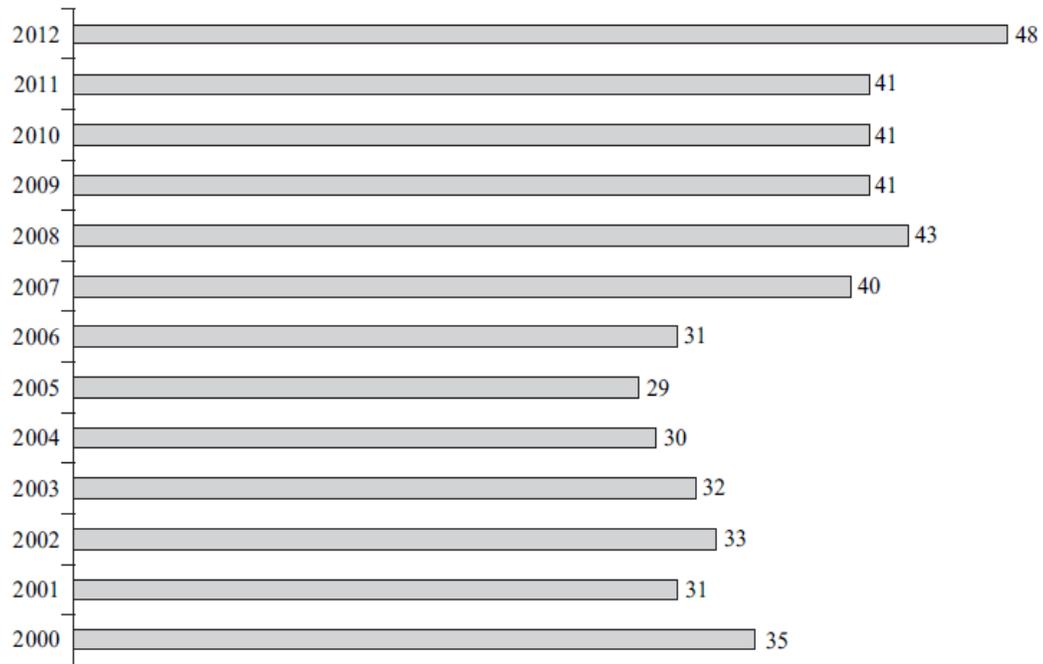
BANCA DE DESARROLLO

1	Financiera Nacional de Desarrollo Agropecuario, Rural, Forestal y Pesquero (FND).
2	Banco del Ahorro Nacional y Servicios Financieros, S.N.C. (BANSEFI)
3	Banco Nacional de Comercio Exterior, S.N.C. (BANCOMEXT)
4	Banco Nacional de Obras y Servicios Públicos, S.N.C (BANOBRAS)
5	Banco Nacional del Ejército, Fuerza Aérea y Armada, S.N.C. (BANJERCITO)
6	Nacional Financiera, S.N.C (NAFIN)
7	Sociedad Hipotecaria Federal, S.N.C. (SHF)
8	Fondo de Capitalización e Inversión del Sector Rural (FOCIR)
9	Fideicomisos Instituidos en Relación con la Agricultura (FIRA)

Nota: Fuente <https://www.gob.mx/shcp/articulos/estas-son-las-9-instituciones-que-integran-la-banca-de-desarrollo?idiom=es>

Bajo un enfoque de competitividad, el número de entidades que integran el sistema financiero mexicano ha tenido un relevante incremento en las últimas décadas (Morales, García & Ángeles, 2013). De acuerdo con la CNVB en el año 2000 existían 35 bancos autorizados incrementando a 48 instancias para el año 2012, presentando especial crecimiento durante el gobierno de Felipe Calderón Hinojosa. Ver figura 20.

Figura 20. **Sistema Financiero Mexicano: 2000 – 2012**



Nota. Fuente: Morales, García & Ángeles, 2013.

El fortalecimiento de la banca mexicana se puede adjudicar a las estrategias de diversificación de productos, servicios y canales de comunicación, así como a los métodos de penetración de servicios financieros en diferentes segmentos de la población. Además, a la competitividad de las tasas de interés en los rubros de crédito y de inversión o ahorro. La especialización de los bancos en nichos de mercado para la atención de sectores socioeconómicos específicos también ha influido positivamente en dicho crecimiento.

En el informe de 2017¹⁷ del Banco de México en torno a la estabilidad del sistema financiero se señaló que pese a factores financieros adversos, como el proceso de liberación de las gasolinas, la depreciación cambiaria, el incremento de la inflación y el entorno internacional

¹⁷ Banco de México. Reporte sobre el sistema financiero mexicano. Octubre de 2017. Fuente: <http://www.banxico.org.mx/publicaciones-y-prensa/reportes-sobre-el-sistema-financiero/%7B4EC1681B-CA21-165F-5AFF-3585D846F541%7D.pdf> (02/09/2018)

incierto, entre otros, las instituciones financieras mexicanas presentan solidez, es decir, una capitalización elevada y niveles adecuados de liquidez, derivado de constantes mejoras en regulación y supervisión por parte de la autoridades financieras.

Sin embargo, Avendaño (2018) refiere que de las 50 instituciones financieras que operan en México, sólo 8 entidades, en su mayoría extranjeras, concentran el 80% de la cartera total de clientes, como se puede observar en la Tabla 8.

Tabla 8. **Concentración del mercado bancario nacional.**

Banco	% de activo total	Capital
BBVA Bancomer	22.0	Extranjero
Santander	15.3	Extranjero
Banamex	14.5	Extranjero
Banorte	11.2	Nacional
HSBC	7.7	Extranjero
Scotiabank	4.6	Extranjero
Inbursa	3.8	Nacional
Total	79.4 (64.2% extranjero vs. 15.2% nacional)	

Nota. Fuente: Avendaño Carbellido, Los retos de la banca digital en México, 2018.

2.1.3 Banca Móvil

La Banca Móvil (CNBV, 2017) es una plataforma digital que permite ofertar servicios o productos financieros a través de dispositivos móviles asociados a un número de teléfono móvil y a una cuenta de usuario. Es decir, es un esquema donde los clientes tienen acceso a sus recursos y funge como un canal no tradicional que ha ido evolucionando de acuerdo a las facilidades que prestan las nuevas tecnologías. Para algunos países desarrollados (Kopicki & Miller, 2008) la banca móvil es un medio de bajo costo para la prestación de servicios financieros y un esquema económico para la transferencia de remesas.

Algunos ejemplos de productos y servicios financieros se muestran en la tabla 9.

Tabla 9. Ejemplos de productos y servicios financieros.

Servicios de Banca Móvil	Productos de Banca Móvil
Transferencias interbancarias	Tarjeta de crédito
Emisión de estados de cuenta	Cuenta de cheques
Consulta de saldos	Créditos hipotecarios
Pago de tarjetas o servicios	Inversiones
Disposición de efectivo	Créditos personales

En relación con los productos y servicios financieros enlistados, la banca móvil tiene una función esencial como facilitadora de servicios bancarios a través de un canal tecnológico

transnacional entre las entidades financieras y los usuarios finales, mismo que se describirán en el siguiente apartado.

Con respecto a los componentes tecnológicos y el flujo de información que se puede generar en un esquema digital financiero, en la figura 21 se despliega un Diagrama de Banca Móvil con el fin de describir cómo las transacciones financieras en un ecosistema bancario se llevan a cabo en sistemas conocidos como transaccionales *core*¹⁸ que llevan a cabo los movimientos entre las cuentas bancarias. Los sistemas *core* interactúan con los canales de acceso¹⁹ bancario (portal web, sucursales, ATM²⁰, Call Center²¹, SMS²², etc.) a través de una capa de integración que pone a disposición los principales servicios para permitir acceder al sistema bancario desde cualquier tecnología, por ejemplo, vía web.

En el caso de la banca móvil, una plataforma de servicios móviles distribuye la aplicación para los principales sistemas operativos (IOS, Android, Windows, etc.) de los dispositivos móviles para acceder a servicios y productos financieros como transferencias bancarias, consulta de saldos, tarjetas de crédito, pago de servicios, créditos hipotecarios, etc.

Figura 21. **Diagrama de Banca Móvil**

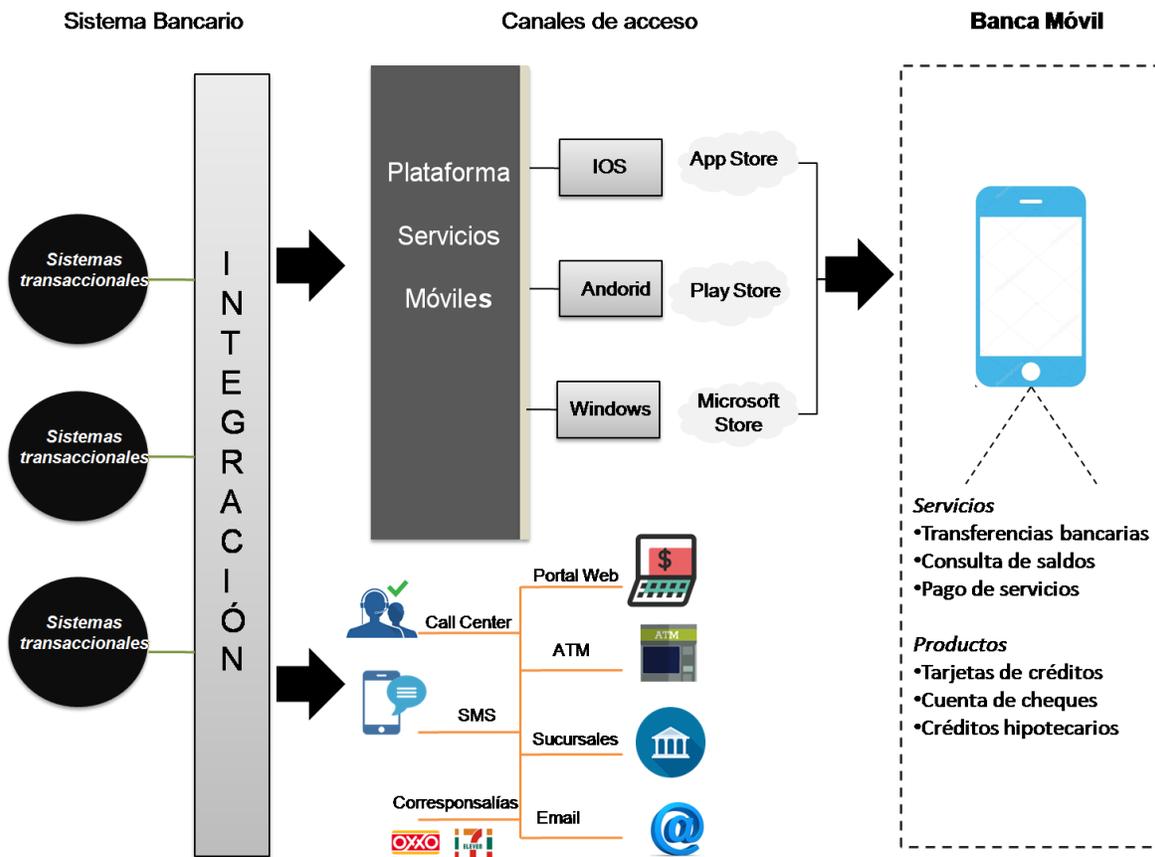
¹⁸ Sistemas centrales de la banca en donde se llevan a cabo las transacciones financieras.

¹⁹ Puntos de acceso al sistema financiero.

²⁰ El modo de transferencia asíncrona (Asynchronous Transfer Mode, ATM) o dispositivo electrónico que permite llevar a cabo transacciones financieras básicas, como retiro de efectivo, consulta de saldo, compra de tiempo aire y pago de servicios básicos.

²¹ Centro de llamadas en donde ejecutivos financieros atienden a los clientes con requerimientos financieros.

²² Acrónimo del inglés Short Message Service, “Servicio de Mensajes Cortos” es un servicio disponible en los teléfonos móviles que permite el envío de mensajes cortos, conocidos como mensajes de texto.



Las regulaciones financieras en el 2012 (Alonso et al., 2013) en materia de simplificación de apertura de cuentas, así como las actualizaciones de la SHCP en torno a esquemas de crédito con nuevas reglas de identificación de clientes y conformación de expedientes para la apertura de cuentas ligadas a la banca móvil motivaron nuevos modelos de negocio bancario. Diversas instituciones bancarias decidieron abrir al mercado aplicaciones para pagos móviles a través de plataformas de teléfonos inteligentes observando una favorable aceptación por parte de la población e identificando un interés en el consumo de servicios financieros a través del celular en ciertos sectores de la población.

Dichos ajustes legales generaron un ecosistema favorable para el desarrollo de la banca móvil ya que simplificaron los requisitos para la gestión de nuevas cuentas con requerimientos mínimos de documentación, vinculadas a un número de teléfono y con montos de transferencia restringidos por aspectos de seguridad.

En general, el progreso de la banca móvil puede acelerar el crecimiento empresarial aprovechando la tendencia generalizada del uso de telefonía móvil y la implantación de modelos de negocio orientados al autoservicio.

Sin embargo, el gobierno sigue representando un rol clave como facilitador de un contexto normativo favorable que fomente la competencia y el crecimiento en torno a plataformas financieras digitales y el fortalecimiento de instituciones supervisoras sólidas e independientes.

Asimismo, hay que considerar mantener un equilibrio en la profundización de servicios digitales financieros con salvaguardias para la estabilidad financiera y la protección de los consumidores. Un avance excesivamente rápido en el proceso de ampliación de los servicios digitales, sin una supervisión adecuada, conlleva riesgos de inestabilidad financiera.

Con respecto a los beneficios del consumo de servicios de banca móvil, Deloitte²³ refiere que dicho esquema puede representar varias oportunidades para las instituciones financieras como ampliar el área de oportunidad o de mercado para ofertar productos y servicios financieros, la reducción de costos en la entrega de productos a través de canales móviles y la mejora continua en el nivel de servicio del cliente.

²³ Deloitte. “El futuro de la banca móvil en América Latina. Perspectivas desde Argentina, Brasil y México”. 2012.

A su vez, los clientes o usuarios finales consumidores de productos móviles pueden obtener ventajas como la reducción en el costo de las comisiones bancarias, movilidad e interacción en línea las 24 horas, comodidad y simplicidad en trámites y servicios bancarios.

De acuerdo con el análisis del Centro de Servicios Financieros de Deloitte dicha plataforma será un componente integral de la estrategia comercial de los bancos, ya que se convertirá en una necesidad más que en una elección en el sistema económico cuya tendencia es la digitalización de gran parte de productos y servicios financieros.

En un contexto regional, es importante resaltar que la banca móvil puede impulsar la inclusión financiera de una nación o región y como consecuencia promover el desarrollo y el empoderamiento de los ciudadanos (Findex, 2017). Algunos estudios hechos en ciertos países subdesarrollados señalan que el acceso a ciertos servicios financieros digitales vía teléfonos celulares puede contribuir a mejorar el potencial de generación de ingresos de las personas y por tanto reducir la pobreza.

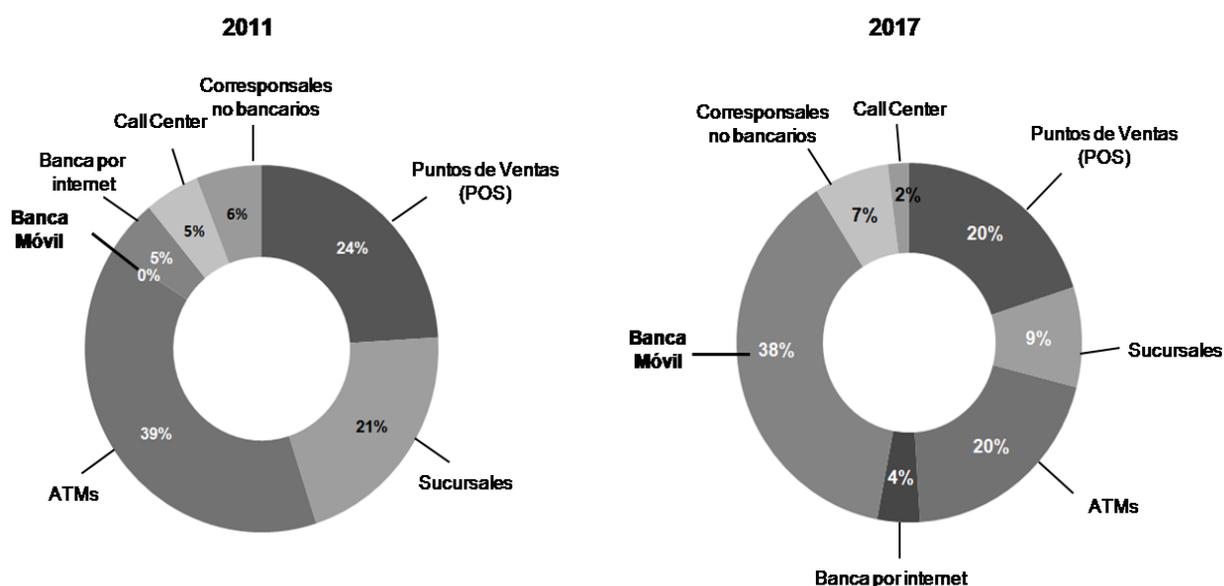
Bajo esta visión, se precisa que si se tiene especial enfoque en las necesidades locales tanto de las empresas como de las personas para la implementación ágil de esquemas de banca móvil se pueden potenciar los resultados bidireccionalmente, tanto para los sistemas bancarios como para la sociedad en general. Los servicios financieros digitales bajo una óptica de crecimiento inclusivo pueden ayudar a que las personas:

- Aumenten los ahorros per cápita
- Impulsen negocios o actividades minoristas en zonas marginadas en lugar de tareas menos remuneradas.
- Mejorar la calidad de vida

- Faciliten el entendimiento y manejo de riesgos financieros
- Disminuir las tasas de pobreza

En este sentido, la Federación Latinoamericana de Bancos (FELABAN, 2018) refiere en su IV Informe de Inclusión Financiera que en América Latina, como se puede observar en la figura 22, la banca móvil tuvo un crecimiento promedio del 38% en el número de operaciones en 6 años, promoviendo la inclusión financiera²⁴ de la región e impactando positivamente en diferentes indicadores económicos y sociales, como el permanente crecimiento económico, la disminución de los índices de pobreza y el engrosamiento de la recaudación fiscal. Parte de la estrategia tecnológica, impulsada por los sectores público y privado, que ha permitido dicho avance son la penetración de la telefonía móvil y el acceso a internet en regiones donde no contaban con la infraestructura necesaria.

Figura 22. Crecimiento de la Banca Móvil.



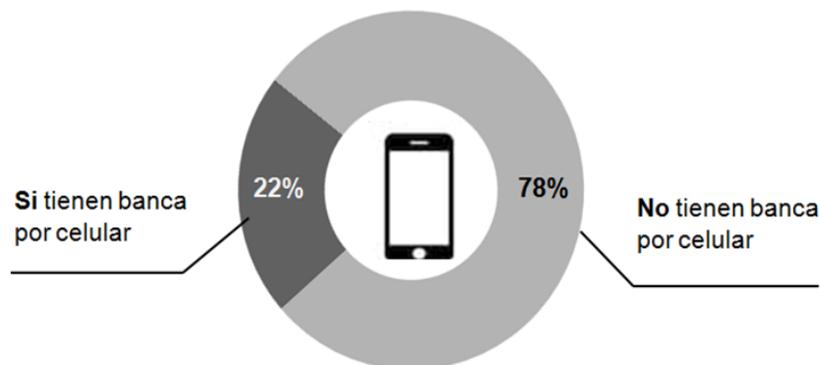
²⁴ Acceso que tienen los agentes económicos (personas, familias, empresas) a una gama de productos y servicios financieros útiles y asequibles que satisfacen sus necesidades, como transferencias, pagos, crédito, ahorro, seguros, etc. (FELABAN, 2018).

Nota. Fuente: <https://www.felaban.net/inclusion/>

En el caso de México, la Tercera Encuesta de Inclusión Financiera (ENIF, 2018) del INEGI indica que el 22% de los clientes entre 18 y 70 años de edad que cuentan con un contrato en una organización bancaria tienen contratado el servicio de banca móvil.

El 78% de los clientes que aún no cuentan con la plataforma móvil, en muchos casos, es porque prefieren el uso de canales tradicionales bancarios, como las sucursales, los cajeros automáticos o la banca por internet, ya sea por falta de conocimientos y habilidades para el manejo de tecnología móvil o por ausencia de posibilidades para acceder a recursos tecnológicos como teléfonos inteligentes. Ver figura 23.

Figura 23. Distribución porcentual de la población con cuenta formal según contrato de servicio de banca móvil.



Nota. Fuente: <https://www.inegi.org.mx/>

Bajo este panorama, el INEGI reitera a las organizaciones financieras la relevancia de diseñar y ejecutar estrategias digitales inclusivas que impulsen el uso de productos y servicios

financieros móviles a través de herramientas ágiles y fáciles de manejar a fin de fomentar el desarrollo y crecimiento nacional.

FELABAN, por su parte, subraya la importancia de la expansión de la banca móvil como un instrumento estratégico que puede generar múltiples beneficios para todos los actores parte del modelo digital (FELABAN, 2018). Por un lado, los clientes pueden tener mayor certeza, transparencia y agilidad al llevar a cabo transacciones financieras desde cualquier punto de la región así como optimización de costos y tiempos en los servicios financieros.

Para las instituciones bancarias es posible incrementar los datos sobre el conocimiento del cliente para aportar valor al negocio ofertando una gama de productos y servicios financieros más competitivos, con una mayor penetración del mercado y reducción de los costos de operación. A su vez, el sector gubernamental puede fomentar programas de educación financiera entre las poblaciones menos favorecidas para su empoderamiento y conocimiento económico que impulse el consumo de productos y servicios financieros en el marco de la economía formal.

Cabe resaltar que la representante del Fondo Monetario Internacional (FMI, 2019), Christine Lagarde, subrayó los avances del gobierno mexicano en materia de inclusión financiera:

- Iniciativa digital CoDi²⁵ que busca mejorar la banca móvil impulsando los pagos electrónicos en tiempo real desde teléfonos móviles.

²⁵CoDi® es una plataforma desarrollada por Banco de México para facilitar las transacciones de pago y cobro a través de transferencias electrónicas, de forma rápida, segura y eficiente, a través de teléfonos móviles. Lo anterior en un esquema 24x7. Usa la tecnología de los códigos QR y NFC, para facilitar que tanto comercios como usuarios, puedan realizar transacciones sin dinero en efectivo y sin costo para los usuarios o clientes finales.

- Esfuerzos por ampliar la red móvil que puedan acercar más servicios financieros a las zonas más remotas.
- La decisión de otorgar a los fondos de pensiones mayor flexibilidad en sus inversiones para contribuir a canalizar ahorros hacia usos más productivos.
- Promover la competencia en el sector bancario y racionalizar los procesos de quiebra a fin de fomentar para los bancos más incentivos para otorgar créditos.
- Aprobación de la Ley para Regular las Instituciones de Tecnología Financiera (2018) para establecer un marco integral centrado en la innovación protegiendo a la vez la estabilidad financiera.
- Iniciativa para promover el uso de tecnologías innovadoras en la prestación de servicios financieros a través de modelos de negocio emergentes (menú de servicios como pagos y remesas, microfinanzas y microfinanciamiento colectivo).

Finalmente, es importante visualizar a la banca móvil como parte de una estrategia integral en el largo plazo para promover la inclusión financiera a gran escala, disminuir la brecha tecnológica y acelerar el proceso de innovación y transformación hacia una economía digital a fin de promover el desarrollo económico del país.

2.1.4 Usuarios de Banca Móvil

En el modelo de banca móvil (Deloitte, 2012), los usuarios pueden agruparse en dos segmentos básicamente. Los clientes como parte formal del sistema bancarizado y los

usuarios que eventualmente realizan transacciones a través de plataformas de mensajes de texto como SMS²⁶, servicios USSD²⁷ o de sistemas móviles.

En el primer grupo, las instituciones financieras enfocan su estrategia de entrega de valor a al cliente²⁸ captando sus necesidades y concentrando esfuerzos para el desarrollo y mejora continua de aplicaciones móviles, ya que los usuarios acceden a la banca móvil a través de teléfonos móviles.

En el segundo segmento, los usuarios no son parte formal del sistema financiero y suelen vivir en regiones con infraestructura bancaria limitada (sucursales, ATMs, centros de atención, etc.), sin embargo, pueden demandar servicios o productos bancarios, ejemplo de ello es el envío de recursos a través de transferencias o depósitos entre cuentas bancarias tradicionales.

Las entidades financieras estiman que la banca móvil puede ser una alternativa fácil y de bajo costo para penetrar con servicios financieros al segundo grupo de usuarios, ya que estos sólo requieren contar con un teléfono móvil con internet como vía de acceso al sistema financiero. Ejemplo de ello es el envío de remesas a través de la banca móvil a un costo menor para los usuarios.

²⁶ Acrónimo del inglés Short Message Service, "Servicio de Mensajes Cortos" es un servicio disponible en los teléfonos móviles que permite el envío de mensajes cortos, conocidos como mensajes de texto.

²⁷ Acrónimo del inglés Unstructured Supplementary Service Data, "Servicio Suplementario de Datos no Estructurados" es un servicio para el envío de datos a través de móviles GSM.

²⁸ Valor al cliente incluye: rapidez de entrega, riqueza de funcionalidad y calidad global del producto (Pressman).

2.2 Metodologías tradicionales de desarrollo de software

2.2.1 Antecedentes

Las metodologías de desarrollo de software se pueden entender como un conjunto de pautas, reglas y recomendaciones (Pantaleo & Riaudo, 2016) que permiten gestionar, planificar y controlar el proceso de fabricación de software a través del uso de herramientas, métodos y modelos de desarrollo. La evolución tecnológica ha permitido la consolidación de diversos mecanismos de desarrollo, cuyo uso y aplicación puede depender de una gama de factores según el contexto y requerimiento de cada empresa u organización. Parte de dicha evolución se explica a continuación.

A mediados del siglo pasado, la incipiente capacidad tecnológica para el desarrollo de programas y la falta de lineamientos o modelos a seguir, impedía a los programadores efectuar un previo análisis de requerimientos y elaboración de un plan de trabajo, por lo que se limitaban a implementar código pequeño y eficiente con base en los recursos disponibles (Gacitúa, 2003).

En la década de los sesentas, surgen los primeros procedimientos para desarrollar aplicaciones centrándose en las técnicas de programación, más que en el control y gestión de las actividades involucrada en el desarrollo. Los programadores invertían grandes cantidades de tiempo corrigiendo y perfeccionando el software que generaban lo que sobrepasaba el presupuesto estimado (Aveson & Fitzgerald, 2006). Posteriormente se desarrollaron aplicaciones más funcionales y complejas atendiendo a ciertos requerimientos y considerando estimaciones más acertadas sobre el costo y el mantenimiento del software.

Aparecen lenguajes de programación de alto nivel como BASIC, PL/I y SNOBOL (Pantaleo & Riaudo, 2016).

Una década más tarde, en los setentas, los modelos de desarrollo de software tuvieron mayor énfasis en estructurar las actividades de desarrollo para lograr productos únicos para cada cliente y con esto se consolidó el desarrollo de software estructurado o en cascada basado en técnicas Top-Down²⁹ y estructuras formales de programación (secuencia, iteración, etc.) a través de lenguajes como Pascal, C, Forth (Pantaleo & Riaudo, 2016).

En los ochenta y con la aparición de los sistemas distribuidos³⁰, los mecanismos de desarrollo se perfeccionaron y se consolidaron algunas metodologías como Structured Analysis and Design Technique (SADT)³¹ e Ingeniería de la información (IE) o Metodología de Ingeniería de la información (IEM)³² los cuales consideraban diagramas más detallados, estándares y modelos de madurez (Pantaleo & Riaudo, 2016). Además, se desarrollan interfaces para la producción y evaluación del software.

En los noventas, se buscaba garantizar el éxito de las aplicaciones enfocándose en la identificación adecuada de requerimiento y elaboración acertada de diseños o prototipos con el fin de llegar a software con la funcionalidad requerida (Arias, 2006). Aparecen nuevos métodos de desarrollo Programación Orientadas a Objetos, Método de Desarrollo de Sistemas Dinámicos y técnicas de desarrollo rápido de software. Se emplea al uso de patrones

²⁹ Técnica que plantea que las dependencias entre los módulos, que conforman un programa, se verifiquen de niveles superiores hacia niveles inferiores (Pauta & Moscoso, 2017).

³⁰ Conjunto de recursos de hardware y software conectados a través de una red para ser compartidos en actividades de desarrollo de aplicaciones.

³¹ Herramienta de análisis funcional y estructurado que permite identificar las necesidades de los usuarios a través de diagramas con niveles sucesivos de detalle para solucionar un problema en particular de una organización. (Ben, 2015).

³² Enfoque de desarrollo de software que considera procesos de planificación, análisis, diseño y aplicaciones de ejecución.

de diseño y del Unified Modeling Language³³ (UML) para modelado de software (Papajorgji & Pardalos, 2014).

En el marco de un contexto global marcado por constante sofisticación tecnológica, a finales de los noventa, se observa un cambio de enfoque con el surgimiento de un grupo de mecanismos de desarrollo ágil como una alternativa al desarrollo lineal y rígido para la fabricación de software (Cervantes, 2016).

Los mecanismos ágiles, bajo un nuevo enfoque de integración y colaboración, concentran esfuerzos para construir equipos de trabajo más integrales, innovadores y eficientes para producción de paquetes de software en tiempos más cortos con entrega de valor al cliente (Lwakatare, Kuvaja, & Oivo, 2015).

2.2.2 Metodologías tradicionales de desarrollo de software

El conjunto de etapas ordenadas, estructuradas y alineadas a los requerimientos de un proyecto, el cual se ejecuta a través de planes de trabajo secuenciales en ambientes controlados, se conocen como metodologías tradicionales de desarrollo de software. Están enfocados a proyectos de duración amplia, orientados a procesos, mantienen poca comunicación con el cliente, tienen baja aceptación a cambios de requerimientos y la entrega de paquetes de software se lleva a cabo hasta la conclusión del proyecto (Navarro, Fernández & Morales, 2013).

Es búsqueda de garantizar el producto final se lleva a cabo una rigurosa planeación en el proceso de desarrollo de software por lo que el grado de control y la dependencia entre las

³³ Lenguaje Unificado de Modelado que funge como un estándar para visualizar, especificar, construir y documentar el ciclo de desarrollo de software (Papajorgji & Pardalos, 2014).

fases no permite que esta metodología se adapte fácilmente a cambios en el alcance, por lo que muchas organizaciones buscan enfoques más flexibles de desarrollo ante escenarios en constante cambio (Rodríguez & Dorado, 2015).

2.2.3 Metodología en Cascada

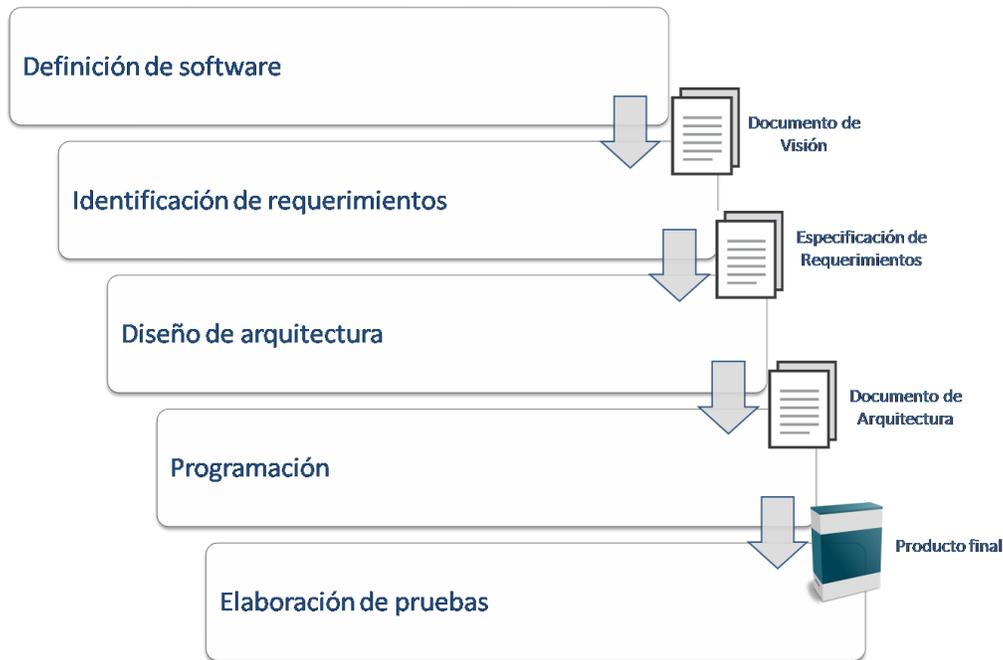
Se trata de un conjunto **de** fases, con objetivos y actividades definidas, que permiten el desarrollo de aplicaciones de manera secuencial o lineal. Las etapas o fases son dependientes entre sí, es decir, cada una depende de la ejecución de su predecesora y así sucesivamente hasta contar con el producto final.

Es una metodología con un marco documental claro y preciso, en donde cada ciclo de actividades genera un documento como entregable que le sirve de entrada al siguiente ciclo.

Las pautas que propone este modelo abarcan definición de software, análisis de requerimientos, diseño de la arquitectura, codificación y pruebas (Pantaleo & Riaudo, 2016).

En el diagrama de la figura 24 se observa la lógica y la secuencia que sigue este modelo y el documento entregable que se genera en cada fase.

Figura 24. Metodología en Cascada por Pantaleo & Riaudo (2016).



En la definición de software se identifican los principales requerimientos de negocio con base en las necesidades comerciales o los requerimientos del usuario (Documento de Visión). En la siguiente etapa se lleva a cabo el análisis de los requerimientos funcionales, técnicos y el grado de factibilidad del producto que se busca desarrollar (Especificación de Requerimientos). Posteriormente se elabora la estructura de los componentes de la solución y su relación para integrar el diseño de arquitectura que servirá de guía para la siguiente etapa (Documento de Arquitectura). La implementación del software se lleva a cabo a través de actividades de codificación y verificación en un ambiente cerrado y controlado (Producto final). Finalmente, al software se le aplica una serie de pruebas de funcionalidad, rendimiento y calidad. Posteriormente se llevan a cabo actividades de mantenimiento del software.

Structured System Analysis and Design Methodology (SSADM)³⁴ (Pantaleo & Riaudo, 2016) y Yourdon System Method³⁵ (Molina, 2012) son ejemplos perfeccionados de modelos en cascada.

En la Tabla 10 se aprecian fortalezas y limitaciones la metodología en cuestión.

Tabla 10. Fortalezas y limitaciones de la Metodología en Cascada.

Fortalezas	Limitaciones
<input checked="" type="checkbox"/> División clara de fases y actividades durante el desarrollo.	<input checked="" type="checkbox"/> Inflexibilidad para aceptar cambios.
<input checked="" type="checkbox"/> Gestión eficiente de riesgos a través de un control riguroso (Pantaleo & Riaudo, 2016).	<input checked="" type="checkbox"/> Exceso de documentación. <input checked="" type="checkbox"/> Retrasos en la entrega del producto final (Pressman, 2010).
<input checked="" type="checkbox"/> Proceso documental sólido y continuo que facilita el mantenimiento y la gestión.	

Asimismo, entre mayor sea la complejidad del proyecto que emplea este enfoque más difícil puede ser su aplicación ya que un cambio de requerimientos detectado en una fase avanzada

³⁴ Método de Análisis y Diseño Estructurado de Sistemas, riguroso modelado que divide el ciclo de desarrollo en ocho etapas, comenzando con la planificación estratégica y los estudios de factibilidad y finaliza con la producción, el mantenimiento y la revisión del producto final. Adecuada para implementar proyectos complejos en entidades gubernamentales (Isaias & Issa, 2015).

³⁵ Metodología estructurada, diseñada por E. Yourdon en 1989, incluye una serie de pasos para el desarrollo de software tales como análisis, diseño, codificación, pruebas y retroalimentación con base en un diagrama jerárquico que incluye el diseño lógico y físico de la solución (Mishra, kumar & Garg, 2013).

puede tener impactos negativos durante el ciclo de desarrollo, repercutiendo en el incremento de costos de producción, falta de garantías en la funcionalidad del producto final, etc.

Finalmente, se recomienda para sistemas en donde se cuenta con la totalidad de requerimientos en una etapa inicial y las actividades de desarrollo se pueden ejecutar de una manera lineal.

2.2.4 Metodología basada en prototipos

Está enfocado a la construcción de prototipos o maquetas con los requerimientos a alto nivel del cliente con el fin de tener un acercamiento rápido al diseño del producto final, pero sin considerar la arquitectura integral (Rodríguez & Dorado, 2015).

Esta metodología trata de ir de la mano con el cliente en el proceso de especificación de los requisitos a través de modelados cuando no se tiene claridad en el detalle de la solución que necesitan. Durante el ciclo de desarrollo, considera varias etapas, representadas en la figura 25.

Figura 25. Metodología basada en prototipos por Pressman (2010).



Con la participación de todos los interesados del proyecto, los objetivos generales y requerimientos iniciales de la aplicación se plasman en un plan rápido, información que servirá para efectuar un diseño ligero o modelado rápido con los requerimientos a alto nivel, enfocándose en el aspecto visual. Posteriormente se construye un prototipo que puede incluir componentes de prueba como pantallas de captura, formularios y reportes gráficos. El prototipo se presenta a los interesados para la correspondiente retroalimentación o visto bueno (Pressman, 2010). Con base en dichas observaciones el prototipo se va perfeccionado hasta que los participantes lo aprueben y entonces se inicia la codificación e implementación.

Rapid Application Development (RAD)³⁶ es considerado un modelado por prototipos ya que se basa en la construcción de componentes o módulos de software, con grupos de trabajo secuenciales y en paralelo, para luego reutilizarlos en sistemas (Molina, 2012).

³⁶ Desarrollo Rápido de Aplicaciones diseñado por James Martin en 1981.

Las fortalezas y limitaciones de esta metodología se señalan en la Tabla 11.

Tabla 11. Fortalezas y limitaciones de la Metodología basado en prototipos.

Fortalezas	Limitaciones
<input checked="" type="checkbox"/> Comprensión rápida de requerimientos al alto nivel a través de prototipos.	<input checked="" type="checkbox"/> Falta de cumplimiento de las expectativas del cliente por falta
<input checked="" type="checkbox"/> Reducción de costos de desarrollo derivado de la reutilización de componentes.	de inclusión de la arquitectura integral. <input checked="" type="checkbox"/> Falta de visión en temas de
<input checked="" type="checkbox"/> Disminución de riesgos al detectar oportunamente los requerimientos iniciales.	funcionalidad, calidad y mantenimiento del software.
<input checked="" type="checkbox"/> Modelado visual de software en etapas tempranas.	

La aplicación de este enfoque se recomienda cuando se tiene un requerimiento concreto, pero no se cuenta con detalles de funcionalidad, lógica y arquitectura del software a desarrollar (Pressman, 2010). A través de prototipos perfeccionados se ofrece una idea al cliente de lo que se puede fabricar y una vez que se aprueba el prototipo final se enfocan esfuerzos en la construcción e implementación.

2.2.5 Metodología Iterativa e Incremental

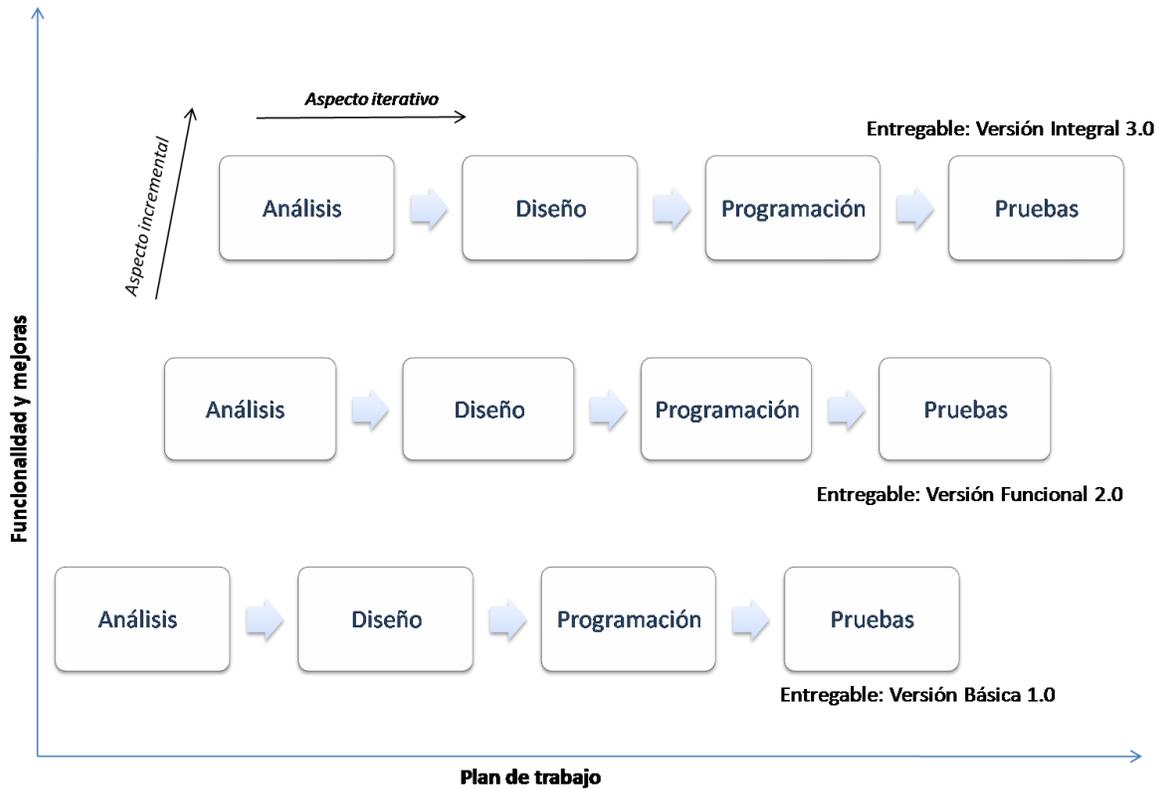
Orientado en proporcionar paquetes o módulos de software con cierta funcionalidad al cliente o usuario y posteriormente en perfeccionarlos a través de fases incrementales e interactivas basadas en procesos secuenciales y en paralelo (Pressman, 2010).

Este enfoque emplea el modelo en cascada para el ciclo de desarrollo (Molina, 2012), iniciando por la identificación de requerimientos funcionales, técnicos y lógicos del proyecto y después la construcción del versionamiento de componentes mejorados, con el fin de reducir tiempos de desarrollo y agilizar la entrega del producto final

La factorización de las actividades del plan de trabajo, con un alcance bien delimitado, emplea secuencias lineales en forma escalonada y cada una genera versiones de software mejorado o adicionado, el cual será evaluado por los interesados en el proyecto hasta alcanzar el producto final.

En la figura 26 se presenta el diagrama de la Metodología Iterativa e Incremental con una serie de secuencias lineales cuya meta es la entrega de componentes tecnológicos funcionales.

Figura 26. Diagrama del Metodología Iterativa e Incremental (Pantaleo & Riaudo, 2016).



El aspecto incremental garantiza un avance en los aspectos funcionales en búsqueda de la reducción del tiempo de desarrollo por lo que permitirá fabricar varias versiones corregidas del software a un menor tiempo. El iterativo apoya la identificación y gestión de requisitos, así como la administración de riesgos ya que está en constante contacto con los requerimientos de los interesados en el proyecto.

El Unified Process (UP) o Rational Unified Process (RUP)³⁷ emplea casos de uso³⁸ para la identificación de requerimientos del usuario y está centrado en la arquitectura (diseño, despliegue e implementación) de los componentes de la solución a implementar. Se considera

³⁷ Proceso Unificado o Proceso unificado de Racional.

³⁸ Describe las acciones o iteraciones que un usuario requiere del sistema que se desarrollará a través de la elaboración de diagramas (Pérez, 2011).

una metodología iterativa e incremental (Molina, 2012) porque permite la liberación de versiones o paquetes funcionales de software.

En la Tabla 11 se aprecian las fortalezas y limitaciones de la Metodología Iterativa e Incremental.

Tabla 11. Fortalezas y limitaciones de la Metodología Iterativa e Incremental.

Fortalezas	Limitaciones
<input checked="" type="checkbox"/> Adaptable a cambios constantes.	
<input checked="" type="checkbox"/> Arquitectura basada en componentes.	<input checked="" type="checkbox"/> Estructura del sistema tiende a degradarse tras una nueva versión.
<input checked="" type="checkbox"/> Modelado visual de software.	<input checked="" type="checkbox"/> Desfases en la documentación
<input checked="" type="checkbox"/> Reducción de costos por reutilización de componentes.	<input checked="" type="checkbox"/> ante mejoras incrementales o versionamientos.
<input checked="" type="checkbox"/> Control de riesgos desde etapas tempranas.	

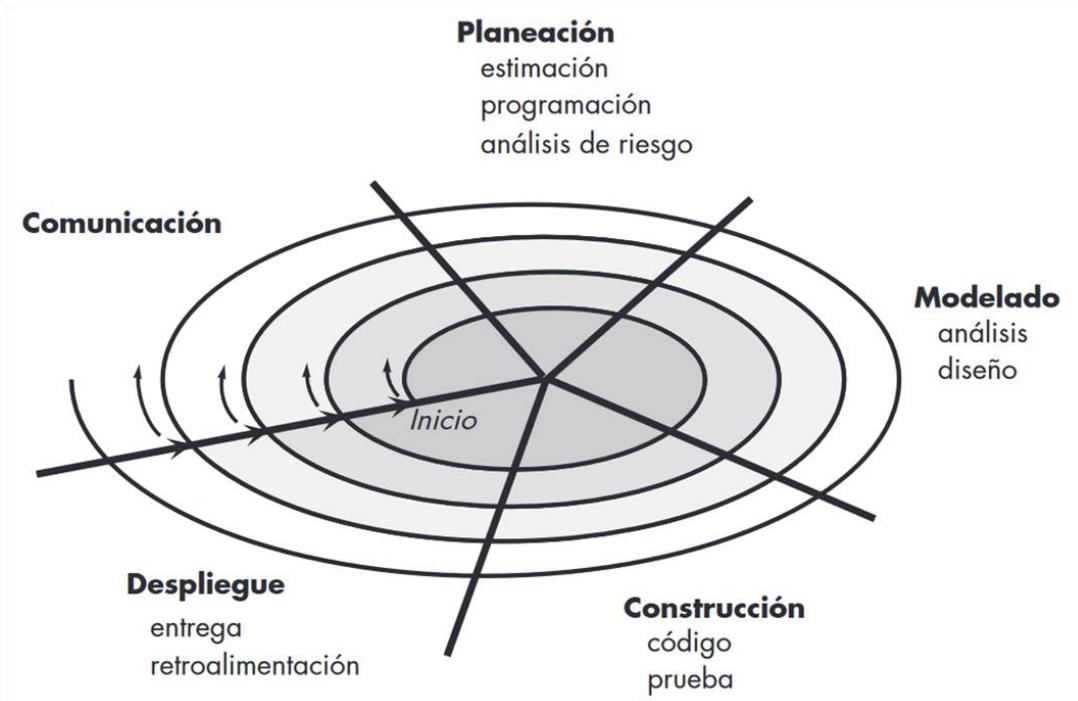
El Modelo Iterativo e Incremental es recomendable en contextos donde los requerimientos permitan operar con un número mínimo de componentes o módulos funcionales y posteriormente se puedan liberar el resto de paquetes de software. También se aconseja en situaciones en donde no se tiene claridad en los objetivos de la solución final, y se puede ir construyendo gradualmente e iterativamente.

2.2.6 Modelo en espiral

Basado en los modelos descritos anteriormente, cascada, iterativo e incremental y por prototipos, se enfoca en la integración de prototipos aprobados y posteriormente en la producción de componentes o módulos de software mejorados a través de procesos iterativos e incrementales para garantizar un proceso de madurez en la construcción de la solución (Pressman, 2010).

Este modelo emplea un plan de trabajo, dividido en “mini-proyectos”, centrado en la generación de entregas evolutivas de software buscando la reducción de riesgos en torno a los factores costo, alcance y tiempo (Pantaleo & Riaudo, 2016).

Figura 27. Diagrama del Modelo en Espiral por Pressman, 2010.



En primera instancia, la fase de la comunicación permite identificar los requerimientos iniciales, funcionales o básicos de la solución. En la planeación se estiman las actividades y se realiza una evaluación de riesgos. Entonces se inicia la construcción de prototipos o modelados, que serán aprobados por los interesados en el proyecto. Posteriormente, se ejecuta la construcción o codificación de componentes cada vez más robustos, conforme a los circuitos de la espiral, y luego se preparan para el despliegue, validación y retroalimentación (Ver diagrama de la figura 27).

Este modelo permite que las primeras iteraciones de la espiral impliquen menor costo y tiempo y con base en el contexto, los interesados y la dirección del proyecto se ejecutarán tantos circuitos en espiral hasta contar con la solución que cumpla con los requerimientos.

Microsoft Solutions Framework (MSF) es un marco de trabajo basado en el modelo espiral y cascada para el desarrollo de software y se centra justamente en la optimización de los factores costos, calidad y riesgos. Asimismo, se alinea a ocho directrices para la entrega exitosa de los componentes tecnológicos. Entre las más relevantes están mantener una comunicación abierta en los equipos de trabajo, impulsar la difusión de metas y objetivos claros del proyecto, mantener un enfoque en el valor agregado al modelo de negocio e invertir en estándares de calidad (Pérez, 2011).

En la Tabla 12 se exponen algunas fortalezas y limitaciones del Modelo en Espiral.

Tabla 12. Fortalezas y limitaciones del Modelo Iterativo e Incremental.

Fortalezas	Limitaciones
------------	--------------

- | | |
|--|---|
| <input checked="" type="checkbox"/> Adaptación rápida a procesos de sofisticación constante de los componentes del software. | <input checked="" type="checkbox"/> Complejidad en su implementación. |
| <input checked="" type="checkbox"/> Modelado visual de software en etapas tempranas. | <input checked="" type="checkbox"/> Dificultad para identificar riesgos potenciales en etapas tempranas |
| <input checked="" type="checkbox"/> Reducción de costos derivado de la reutilización de componentes. | |

Se recomienda aplicar el Modelo en Espiral en problemas que impliquen una solución a gran escala, con altos niveles de complejidad y que impliquen un gran esfuerzo para ejecutar el plan de trabajo.

2.3 Marco de desarrollo ágil de soluciones para tecnología móvil

2.3.1 Introducción

Ante un contexto de variabilidad en los mercados, surgimiento de modelos de negocio emergentes y un permanente proceso de sofisticación tecnológico nace el desarrollo ágil de aplicaciones que busca satisfacer en tiempos más cortos las expectativas del cliente. En este sentido, el aumento de la productividad de aplicaciones y el cumplimiento de los objetivos del cliente en el menor tiempo posible aporta mayor valor al negocio lo que representa una ventaja competitiva para las organizaciones (Molina, 2012).

Bajo este escenario, las metodologías tradicionales de software presentan algunas dificultades en la producción de software ya que contemplan largos plazos para las entregas, generación excesiva de documentación y ambientes rígidos de desarrollo (Pantaleo & Riaudo, 2016).

El enfoque ágil se enmarca en el Manifiesto Ágil (2001) que suscribe las bases para el desarrollo rápido de software a través de una serie de lineamientos priorizando los requerimientos del cliente. Dichas directrices resaltan la importancia del equipo de trabajo sobre los procesos y las herramientas, la elaboración de la documentación mínima posible, la constante comunicación entre todos los interesados en el proyecto y la generación de ambientes rápidos y flexibles de trabajo. Asimismo, estiman que los procesos de desarrollo ágil deben contemplar la constante entrega de paquetes de software al cliente para validar los requerimientos en etapas tempranas y si es necesario gestionar los cambios necesarios mediante estructuras de desarrollo flexible (Letelier & Penadés, 2012).

El marco ágil busca integrar equipos de trabajo centrados y sincronizados en los mismos objetivos para desarrollar y desplegar, en periodos de producción más cortos, productos de alta calidad al cliente alineando el desarrollo de software con el modelo de negocio (Molina, 2012). Se basa en una planeación sencilla para efectuar entregas pequeñas, iterativas e incrementales bajo ambientes rápidos y flexibles de trabajo. Su orientación es hacia las personas o clientes y un poco menos hacia los procesos, además de centrarse en la producción de software, dejando en segundo término la documentación (Navarro, Fernández & Morales, 2013). Este enfoque fomenta una comunicación fluida con el cliente y una constante retroalimentación entre los interesados del proyecto para optimizar el rendimiento del equipo.

En la actualidad, el desarrollo de aplicaciones móviles está en gran medida sujeto a las exigencias del mercado, al constante cambio en los requerimientos del cliente y a la variabilidad de las plataformas tecnológicas. Esto ha provocado la reducción de la vida útil de la tecnología móvil por lo que las organizaciones requieren modificar o adoptar nuevos enfoques de desarrollo que permitan la pronta entrega de valor al cliente (Molina, 2012). Contar con mayor número de productos móviles con la funcionalidad y calidad esperada en el mercado con un despliegue ágil y continuo, representa una ventaja competitiva para las empresas a través de la reducción del time-to-market y mejores márgenes en el retorno de inversión

2.3.2 Scrum

Considerado como un marco de trabajo, Scrum permite desarrollar software ágilmente a través de equipos de trabajo pequeños, interdisciplinarios y auto-organizados implementando una serie de lineamientos y roles con plazos fijos de entrega (Dimes, 2015).

Las actividades del proyecto son divididas y priorizadas en un listado de entregables pequeños y concretos. Se desarrollan en ciclos iterativos conocidos como sprints con duración fija de 1 a 4 semanas para hacer entregas de paquetes de software o incrementos ejecutables (Canós, Letelier & Penadés, 2012).

Se recomienda efectuar reuniones diarias con las partes interesadas para dar seguimiento y mantener comunicación constante (Becerra & Sanjuan, 2014). A continuación se describen roles, activos y flujo de trabajo que contempla Scrum (Pantaleo & Riaudo, 2016).

Roles

- Product Owner: conoce el negocio y define el producto. Responsable de priorizar y definir alcance y requerimientos.
- Scrum Master: coordinador y facilitador del grupo de trabajo. Se encarga de negociar con el Product Owner la operatividad del desarrollo.
- Team: grupo de desarrolladores de 3 a 7 personas que ejecutan el Sprint Planning. A medida que ejecuta iteraciones aprende y calcula estimaciones más certeras.
- Project Manager: se encarga de integrar el proyecto al flujo de trabajo de la organización.

Activos

- Product Backlog: listado priorizado de características y funcionalidades del producto cuyo alcance puede cambiar a lo largo de la ejecución.
- Sprint Backlog: subconjunto de requerimientos del Product Backlog a ejecutar en un sprint sin posibles cambios.

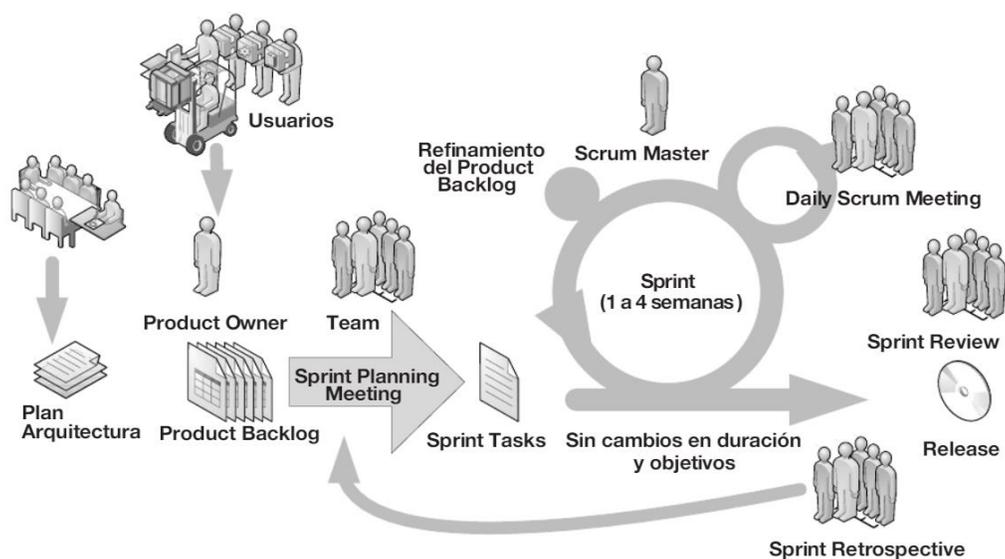
Flujo de trabajo

- Sprint: iteración de duración fija, de 1 a 4 semanas.
- Project Planning: actividades de planificación del proyecto.
- Sprint Planning: actividades de planificación de un Sprint.
- Release: código funcionando entregado al cliente.
- Sprint Retrospective: reunión para retroalimentación del Sprint anterior e identificación de oportunidades de mejora.

En la figura 28 se describe la dinámica de los componentes de Scrum en donde a partir de un listado priorizado del producto, acordado por las partes interesadas del proyecto, se integra

el Sprint Backlog, inamovible en alcance y duración, que se ejecutará en un Sprint. Una vez desarrollado, probado y entregado cada paquete de software con valor al cliente (Release), el equipo se reunirá y revisará en retrospectiva los puntos a mejorar para ajustar el siguiente Sprint Backlog a ejecutar y así lograr optimizar el rendimiento de los recursos.

Figura 28. Flujo de componentes de Scrum (Pantaleo & Riaudo, 2016).



En la Tabla 13 se exponen algunas ventajas y limitaciones de Scrum.

Tabla 13. Ventajas y limitaciones de Scrum.

Ventajas	Limitaciones
<input checked="" type="checkbox"/> Optimización y seguimiento puntual de las expectativas del cliente (Kniberg & Skarin, 2010).	<input checked="" type="checkbox"/> Difícil implementación en proyectos grandes y complejos.

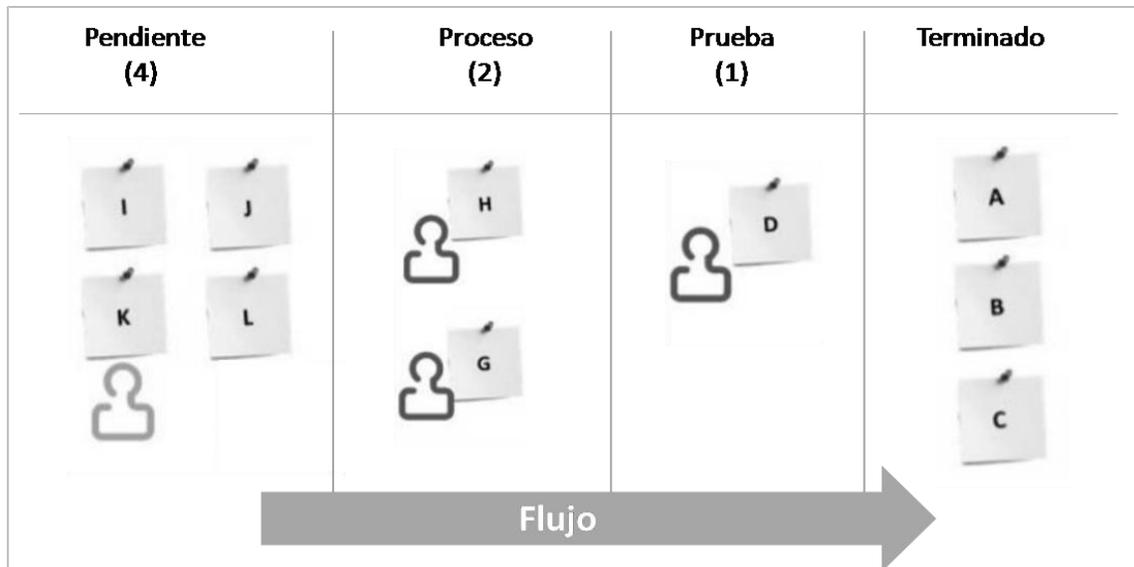
<input checked="" type="checkbox"/> Flexibilización y adaptación a negocios emergentes.	<input checked="" type="checkbox"/> Estimación correcta de plazos fijos para asegurar el éxito del Release.
<input checked="" type="checkbox"/> Constante entrega de valor al cliente, construcción rápida de componentes.	<input checked="" type="checkbox"/> Contar con equipos altamente calificados, integrados y motivados.
<input checked="" type="checkbox"/> A mayor versionamiento mayor estabilidad en el software.	

2.3.3 Kanban

Método de origen japonés para la producción de autos en serie que fue adoptado en 2004 por la empresa Microsoft (Guzmán, Islas, Corona, & Méndez, 2014) para resolver proyectos con incidencias en los plazos de entrega y errores en la calidad del producto.

Kanban busca fomentar el trabajo en equipo con un flujo permanente de tareas, las cuales son divididas y priorizadas en paquetes de trabajo, limitando su número en cada estado del flujo de actividades según la capacidad de producción del equipo (Kniberg & Skarin, 2010). Los paquetes priorizados pueden organizarse en cuatro apartados: Pendiente, Proceso, Prueba y Terminado, según las necesidades del proyecto, para conformar el tablero Kanban, el cual es considerado una herramienta de visualización y comunicación para mostrar el flujo del proyecto.

Figura 29. Tablero clásico Kanban (Guzmán, Islas, Corona & Méndez, 2014).



El tablero de la figura 29 muestra una vista clara y sencilla del flujo de las actividades del proyecto, las prioridades, los plazos de entrega, los responsables y las posibles incidencias (Ponce, Moreno, Yáñez & Soto, 2014). En este caso, la restricción limita tener únicamente 4 paquetes en Pendiente, 2 en Proceso y 1 en Prueba para garantizar las entregas funcionales y en tiempo con la optimización del rendimiento del equipo.

Esta herramienta es de gran utilidad para hacer siempre visible el trabajo y restringir el volumen de paquetes en operación que permita generar un flujo estable, reducir el tiempo de desarrollo y agilizar la entrega de paquetes con valor al cliente. En la Tabla 14 se exponen algunas fortalezas y limitaciones de la metodología Kanban.

Tabla 14. Ventajas y limitaciones de Kanban

Fortalezas	Limitaciones
<input checked="" type="checkbox"/> Optimiza el flujo de trabajo del proyecto.	<input checked="" type="checkbox"/> Dificultad para balancear recursos sin tiempos ociosos.

<input checked="" type="checkbox"/> Entrega de paquetes funcionales a un ritmo constante.	<input checked="" type="checkbox"/> Dificultad para gestionar paquetes de trabajo complejos con estimaciones largas.
<input checked="" type="checkbox"/> Flexibilidad en el desarrollo, pocos lineamientos a seguir.	

2.3.4 Extreme Programming (XP)

XP o Programación Extrema busca integrar equipos de desarrollo sincronizados y colaborativos en ambientes de trabajo adaptables e integrados que permitan la satisfacción del cliente y potenciar al máximo el rendimiento del equipo. Es un enfoque iterativo e incremental donde el cliente tiene un rol muy importante al definir el valor de negocio que se requiere a través de historias de usuario que son analizadas por el equipo para estimar el esfuerzo y los recursos necesarios. Posteriormente el cliente define el orden de prioridades a entregar y los desarrolladores integran dichos requerimientos (Letelier & Penadés, 2012).

El equipo del proyecto, organizado en parejas de desarrolladores, trabaja intensamente en un diseño simple para implementar entregas pequeñas e incrementales en plazos cortos de tiempo, con la restricción de emplear máximo 40 horas de trabajo a la semana para mantener un ritmo y una calidad constante de producción. Se lleva a cabo una integración continua de paquetes de software en donde todos los integrantes del equipo conocen y pueden modificar la composición de los mismos, conocido como propiedad colectiva (Gómez, Duarte y Güevara, 2014).

Figura 30. Esquema conceptual de XP (Pantaleo & Riaudo, 2016).



En el esquema de la figura 30 se puede observar que XP propone un desarrollo en pares con el fin de aumentar la calidad de los paquetes de software, efectuar entregas incrementales y disminuir los tiempos de desarrollo, sin embargo, es fundamental la participación activa del cliente durante todo el proceso de desarrollo e implementación. Además, este método ágil a través de la refactorización³⁹ en los componentes de los paquetes desarrollados permite obtener un diseño sencillo del producto y efectuar pruebas unitarias y de usuario de manera simple para garantizar la satisfacción del cliente.

Las ventajas y limitaciones de la metodología XP se muestran en la Tabla 15.

Tabla 15. **Ventajas y limitaciones de XP.**

Ventajas	Limitaciones
----------	--------------

³⁹ Es el proceso en el cual se aplican cambios en un sistema de software para mejorar su estructura interna, pero sin alterar el comportamiento externo del código.

- | | |
|---|--|
| <input checked="" type="checkbox"/> Capacidad de respuesta ante constantes cambios. | <input checked="" type="checkbox"/> Complejidad para estimar el costo y el tiempo de desarrollo. |
| <input checked="" type="checkbox"/> Optimizar el rendimiento del equipo. | <input checked="" type="checkbox"/> Disponibilidad del cliente. |
| <input checked="" type="checkbox"/> Gestión de la complejidad del software. | |

2.3.5 DevOps

2.3.5.1 Origen

En un contexto en donde la mejora continua⁴⁰ y la velocidad de entrega del producto son cruciales para contar con ventajas competitivas en empresas como Google y Amazon, el éxito del modelo de negocio puede depender en gran medida de la capacidad para implementar software nuevo varias veces al día, o incluso cada hora según las necesidades del cliente (Thoms, 2018).

Como se describió en el capítulo anterior, el desarrollo ágil busca optimizar la producción y entrega continua de aplicaciones dividiéndola en paquetes de software más pequeños y periodos de tiempo cortos a través de equipos de trabajo sincronizados y enfocados a la entrega de valor al cliente.

⁴⁰ Enfoque para la mejora de procesos que busca minimizar el desperdicio, optimizar la velocidad, los costos y la facilidad de entrega de productos y/o servicios ofrecidos. <https://www.devopsagileskills.org/dasa-devops-principles/>

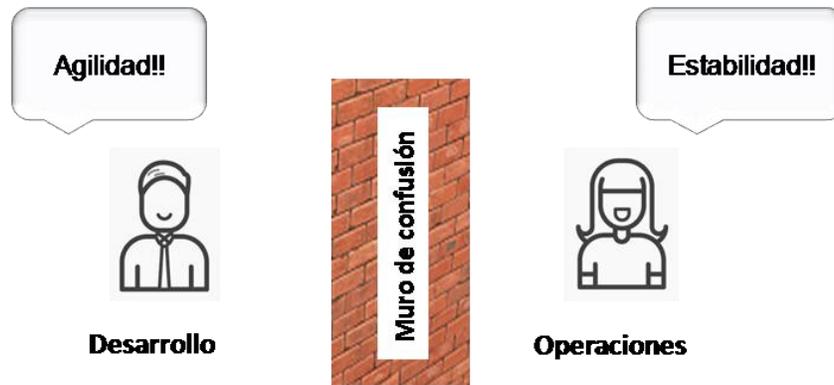
Sin embargo, históricamente el desarrollo, la integración y las operaciones se han hecho de manera aislada y con distintos objetivos a perseguir (IMB, 2019). En muchos casos, dicho proceso se lleva a cabo de forma manual entre diferentes ambientes de trabajo generando una serie de errores o cuellos de botella en la liberación de software funcional y restando agilidad al negocio.

Asimismo, cuando la integración de los componentes del software a liberar depende de diversas plataformas y/o tecnologías (BD, WS, ETL, Web, etc.) operadas por diferentes áreas, resulta compleja la completa y correcta ejecución de la implantación de manera manual, ya que el proceso se vuelve propenso a errores humanos originando conflictos e ineficiencia.

Lee Thompson⁴¹ lo describe como una especie de “Muro de confusión” entre desarrollo y operaciones que impide que las interacciones se ejecuten en el marco de una metodología ágil. El área de desarrollo se enfoca más a la creación y producción de cambios en el software, sin conocer por completo el flujo necesario del despliegue. En cambio, el departamento de operaciones piensa en términos de estabilidad y confiabilidad de las plataformas, tratando de disminuir la ejecución de cambios. Dicha dinámica se representa en la figura 31.

Figura 31. **Representación del “Muro de Confusión”.**

⁴¹ Ex tecnólogo jefe de E*TRADE Financia, <http://dev2ops.org>.



Nota. Fuente: <http://dev2ops.org/>

Es decir, en ambientes de desarrollo ágil los desarrolladores se enfocan en la producción constante y en incremento de paquetes, sin embargo, los operadores en muchos casos no tienen la capacidad de integrar y desplegar los paquetes de manera rápida generando un desfase en la entrega de la funcionalidad al cliente.

Además, la falta de alineación entre las áreas de TI en el ciclo de desarrollo, inclusive con las unidades de negocio y aun utilizando enfoques ágiles, puede demorar el proceso de despliegue de aplicaciones y la mejora continua de software, generando insatisfacción del cliente y restando rentabilidad al negocio.

En 2009⁴², el belga Patrick Debois logró discernir este problema y propuso las bases de una nueva forma de colaboración y cooperación entre desarrollo y operaciones con el término DevOps, del inglés "Development & Operations", el cual plantea un conjunto de directrices y mejores prácticas que permitan la adopción de herramientas para fomentar la automatización de procesos y optimización del flujo de trabajo del equipo que interviene en el ciclo de desarrollo e implantación de software.

⁴² <https://devops.com/the-origins-of-devops-whats-in-a-name/>

Posteriormente en 2013 se publicó el libro "The Phoenix Project" que expone una historia en torno a la búsqueda de nuevas formas desarrollo basadas en DevOps para proyectos de TI con retrasos en las entregas. En 2010 en EE.UU. por primera vez se celebra la conferencia "DevopsDays"⁴³ sobre temas de desarrollo de software, operaciones de infraestructura de TI y mejora continua. Posteriormente se convierte en un intercambio global de mejores prácticas en ingeniería de software y para el 2019 suman 34 eventos programados en diferentes ciudades del mundo.

En un reciente estudio conocido como "*Informe sobre el Estado de la Nube*" (RightScale's, 2016) se pone de manifiesto la importancia que ha adoptado esta nueva corriente de organización para mejorar el rendimiento de los equipos de trabajo ya que indica que el 70% de las Pymes en EE.UU. está adaptando prácticas basadas en DevOps para mejorar el desarrollo de aplicaciones.

2.3.5.2 Concepto

DevOps es un conjunto de principios, prácticas y herramientas orientadas a alinear las áreas de desarrollo, pruebas y operaciones de TI para lograr ciclos de desarrollo más cortos, entregas más frecuentes, productos de mayor calidad y clientes más satisfechos (Belalcázar, Díaz & Rodríguez, 2016).

⁴³ www.devopsdays.org

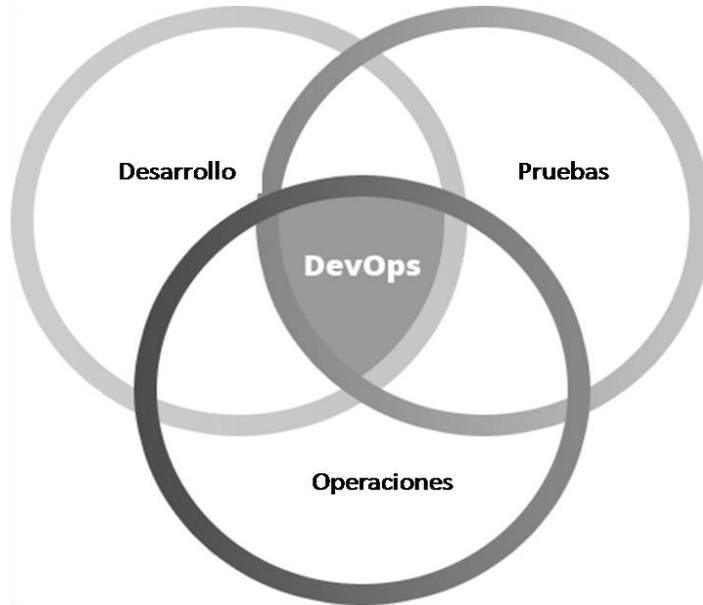
Este enfoque además implica un cambio en la cultura organizacional para lograr una interacción ágil y colaborativa de las personas que permita optimizar la entrega de servicios con valor en el marco de las metodologías ágiles de desarrollo de software (Gartner, 2018).

Por su parte, IBM (2017) conceptualiza a DevOps como una capacidad empresarial para el desarrollo constante de innovación que permita satisfacer en el menor tiempo posible las necesidades del cliente a través de entornos de trabajo enfocados en la mejora continua y la agilidad en la entrega de tecnología.

De acuerdo con Machiraju y Gaurav (2018), las prácticas DevOps buscan integrar la funcionalidad de los equipos de desarrollo y operaciones para disminuir los silos entre ambos durante el proceso de producción de software ya que, en las estructuras tradicionales, dichas áreas tienden a visualizar metas y funciones separadas. Asimismo, Thoms (2018) asevera que su alcance no solo es lograr equipos de trabajo más colaborativos, sino crear equipos de trabajo unificados que interactúen en cada etapa del ciclo de vida de un producto.

Estas prácticas facilitan que las áreas claves del ciclo de fabricación de software, desarrollo, pruebas y operaciones converjan en ambientes flexibles, ágiles y estables de trabajo orientados al valor del negocio como se puede observar en el diagrama de la figura 32. Se busca compartir y comunicar objetivos y métricas entre las áreas clave con una nueva base cultural y de colaboración para todos los interesados en el proyecto que permita contar con capacidades competitivas para aprovechar las oportunidades del mercado.

Figura 32. **Intersección de áreas clave en DevOps”.**

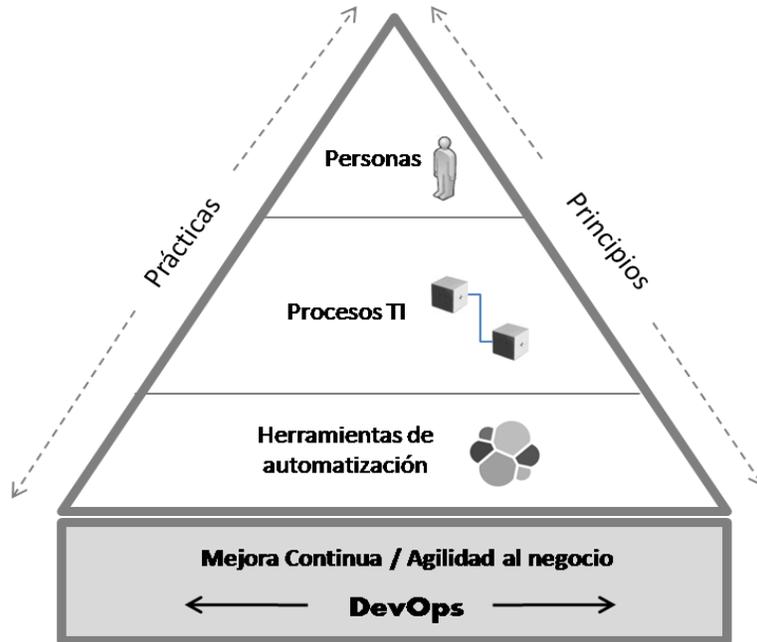


Nota. Fuente: <http://dev2ops.org/>

En un entorno DevOps, los desarrolladores trabajan más de cerca y en colaboración con las operaciones de TI para agilizar la codificación, las pruebas y el versionamiento de software, sin descuidar la confiabilidad del producto.

DevOps se puede contextualizar como una forma de trabajo que busca integrar y unificar tres componentes básicos: los *equipos de trabajo* (desarrollo, pruebas y operaciones) que se encargan de ejecutar los *procesos* necesarios para la operación de TI a través de una gama de *herramientas de automatización* a fin de impulsar la mejora continua de software y darle agilidad al negocio. Ver diagrama de la figura 33.

Figura 33. **Diagrama DevOps.**



Cuando la estructura de TI aplica prácticas basadas en DevOps su rendimiento se vuelve más eficiente y confiable (Machiraju y Gaurav, 2018), particularmente por el uso de herramientas de automatización que permiten incrementar la velocidad de ejecución del ciclo de desarrollo entre distintos ambientes, unificar el proceso de despliegue de la aplicación, disminuir errores en la implantación y garantizar la estabilidad de los sistemas de producción. Es decir, en un contexto DevOps se puede contar con un equipo sincronizado y una estructura automatizada que permita implementar ciclos ágiles de despliegue con mayor frecuencia y eficiencia garantizando la continuidad del negocio.

2.3.5.3 Ventajas del enfoque DevOps

De acuerdo con Boulton (2018), una de las mayores fortalezas de esta filosofía es la automatización de la construcción de software con un impacto positivo en el rendimiento de TI ya que facilita el despliegue continuo, mejora la experiencia del cliente y fomenta un

equilibrio entre las necesidades de los diferentes ambientes de desarrollo, pruebas y operaciones.

DevOps puede mejorar la experiencia del cliente desarrollando equipos con capacidad de respuesta ágil para adaptarse a las necesidades reales, lo que permite captar de forma continua los comentarios de los usuarios y fomentar la lealtad de los clientes.

Si se logra automatizar la integración, la entrega y el despliegue de aplicaciones, aunado con la práctica de una permanente retroalimentación con el cliente, se puede reducir el time-to-market y mejorar la calidad de los productos y servicios ofertados. Por lo tanto, este enfoque permite que los equipos de trabajo sean más eficientes centrándose en actividades entorno a la innovación y mejora continua, en lugar de ejecutar el ciclo de integración y despliegue de manera manual.

Desde una perspectiva de inteligencia de negocio, el pensamiento DevOps puede representar dos grandes ventajas, la alineación estratégica de TI e impulsar la agilidad del negocio (Edwards, 2010). Al fomentar objetivos de negocio compartidos, esta forma de trabajo permite alinear las áreas de TI desde un entendimiento organizacional en donde se subraya la importancia de cada área al formar parte de un proceso empresarial unificado. Por otro lado, a través del proceso de automatización se busca la entrega ágil de productos y servicios al cliente con las funcionalidades esperadas.

También es considerado un enfoque de trabajo adaptable y flexible para cualquier tipo de negocio según lo requiera la operación, ya sea en entornos que interactúen con metodologías tradicionales, ágiles o híbridas que busquen la mejora continua y la optimización de recursos.

En ambientes DevOps es posible disminuir el silo entre desarrollo, pruebas y operaciones para agilizar el ciclo de desarrollo de software y al mismo tiempo mantener estabilidad y confiabilidad en las plataformas. En la Tabla 16 se puede apreciar un listado con las ventajas de la adopción de DevOps considerando también algunas limitaciones para su implantación.

Tabla 16. Ventajas y limitaciones del enfoque DevOps.

DEVOPS	
Ventajas	Limitaciones
<ul style="list-style-type: none"> <input checked="" type="checkbox"/> Mejorar la experiencia del cliente. <input checked="" type="checkbox"/> Disminuir time-to-market del producto. <input checked="" type="checkbox"/> Impulsar mejora continua. <input checked="" type="checkbox"/> Reducir errores, sobrecarga y repetición de tareas. <input checked="" type="checkbox"/> Agilidad al negocio, implementaciones con mayor frecuencia y eficiencia. <input checked="" type="checkbox"/> Alineación estratégica de TI con el modelo de negocio. <input checked="" type="checkbox"/> Ambientes de trabajo colaborativos, ágiles y flexibles. <input checked="" type="checkbox"/> Compatible con ambientes de trabajo tradicionales o ágiles. 	<ul style="list-style-type: none"> <input checked="" type="checkbox"/> Resistencia al cambio por parte de todos los interesados. <input checked="" type="checkbox"/> Eliminar herramientas y sistemas que impidan la unificación del flujo de trabajo. <input checked="" type="checkbox"/> Presupuesto para migración a nuevas herramientas o plataformas de automatización. <input checked="" type="checkbox"/> Tiempo para la transición y adopción del nuevo enfoque.

2.3.5.4 Principios y prácticas DevOps

El enfoque DevOps propone una serie de principios para optimizar el ciclo de desarrollo de aplicaciones, impulsar la mejora continua y fomentar ambientes de trabajo ágiles, colaborativos y alineados en torno al negocio.

Planificación centrada en el negocio. Busca identificar un conjunto mínimo de funcionalidades que permitan aportar valor y organiza el trabajo en iteraciones cortas enfocadas a la entrega continua. El pensamiento *lea* (Womack & Jones, 2008) es una excelente alternativa para especificar valor de manera ágil con la cantidad mínima de recursos. La planificación en cuestión incluye la gestión de entrega de versiones con las funcionalidades y los plazos estimados.

Integración Continua. Representa un cambio de paradigma en el desarrollo de software porque es una práctica que permite que los desarrolladores integren el código con alta frecuencia y validen su funcionalidad constantemente, sin esperar a concluir todos los componentes y detectar errores en fases avanzadas (Fowler, 2014). Los desarrolladores crean el código o componentes del producto con cierta estructura para integrarlo de forma continua y automatizada. Posteriormente a través de mecanismos de control de versiones, con otros elementos del equipo ejecutan pruebas integrales de la aplicación.

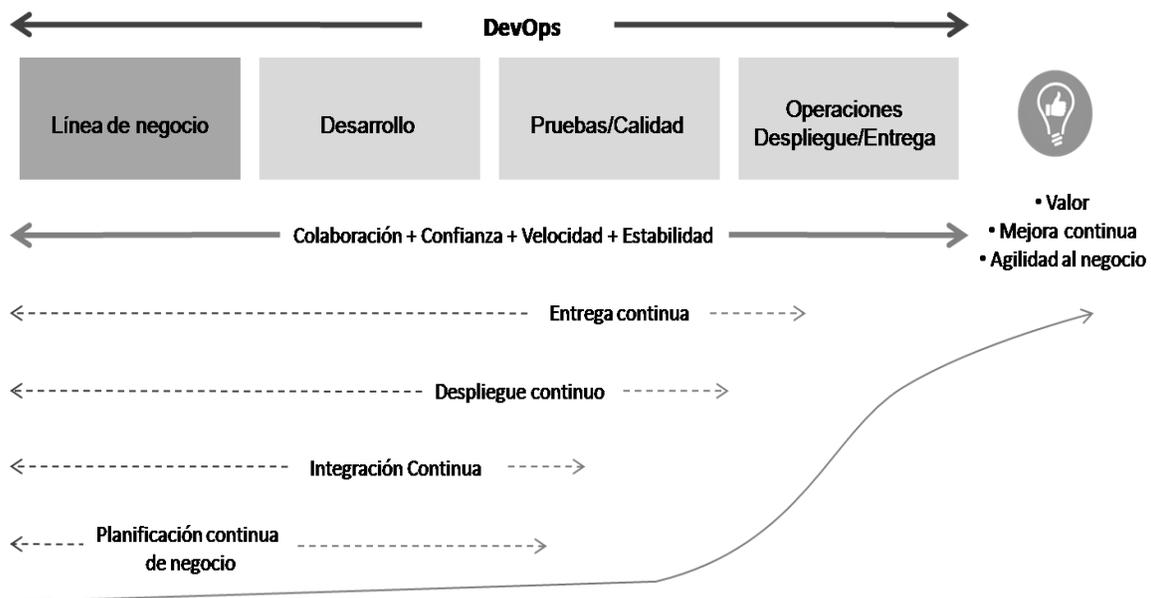
Despliegue Continuo. Consiste en preparar los paquetes de software integrados a través de ambientes automatizados (“*pipeline*”) en un entorno de operaciones garantizando el funcionamiento efectivo de las versiones estables para ser desplegadas. Su objetivo es aumentar el número de despliegues frecuentes conforme la operación lo requiera y mantener un enfoque de "falla rápida" que facilite la identificación de errores o incidentes en etapas

tempranas de desarrollo para darles atención. Evita la instalación de versiones fallidas en ambientes de producción.

Entrega Continua. Permite probar la aplicación en cualquier momento durante el ciclo de desarrollo y que esté lista para pasar a producción, aunque no siempre se requiera liberar en ambiente de operaciones. A diferencia del despliegue continuo que siempre es automatizado, en la entrega hay un componente humano, el responsable del negocio, por ejemplo, que autoriza que la aplicación sea puesta en producción. Se busca brindar un mayor rendimiento entre los ambientes y que los equipos se mantengan en constante retroalimentación sobre su aplicación. Las aplicaciones deben estar siempre en un estado disponible para la entrega.

En el diagrama de la figura 34 se muestra la interacción de estos principios con las fases de desarrollo y producción de aplicaciones en un entorno DevOps.

Figura 34. Principios DevOps.



En este contexto, se busca optimizar el trabajo colaborativo y la mejora continua en función de:

- Disminuir silos entre desarrolladores, pruebas y operaciones.
- Contar con paquetes de software incrementales listos para desplegar en cualquier ambiente, disminuyendo los ciclos de retroalimentación con el cliente.
- Ejecutar pruebas integrales a través de herramientas de automatización que aseguren requerimientos y funcionalidad.
- Desarrollar aplicaciones en menor tiempo y con menor número de fallas o incidencias.
- Identificar en etapas tempranas errores y tener respuesta rápida de atención.
- Minimizar incidencias por instalación de versiones obsoletas ó erróneas en ambientes de producción.
- Mayor comunicación y retroalimentación en el ciclo de desarrollo de aplicaciones.

Por su parte, la Asociación de Aptitudes Ágiles y DevOps (DASA por sus siglas en inglés) propone una serie de prácticas a seguir para la implantación exitosa de ambientes DevOps que incluyen diferentes perspectivas (DASA, 2017).

1. **Acción centrada en el cliente.** La estrategia de desarrollo se enfoca en actividades orientadas a garantizar la entrega constante y ágil de aplicaciones de calidad y valor, buscando experiencias más personalizadas con el cliente. Toda la actividad de creación de productos se centra en el cliente con ciclos cortos de retroalimentación.
2. **Automatización.** Busca automatizar la integración, despliegue y entrega continua de aplicaciones, así como la infraestructura necesaria para su operación. Permite la

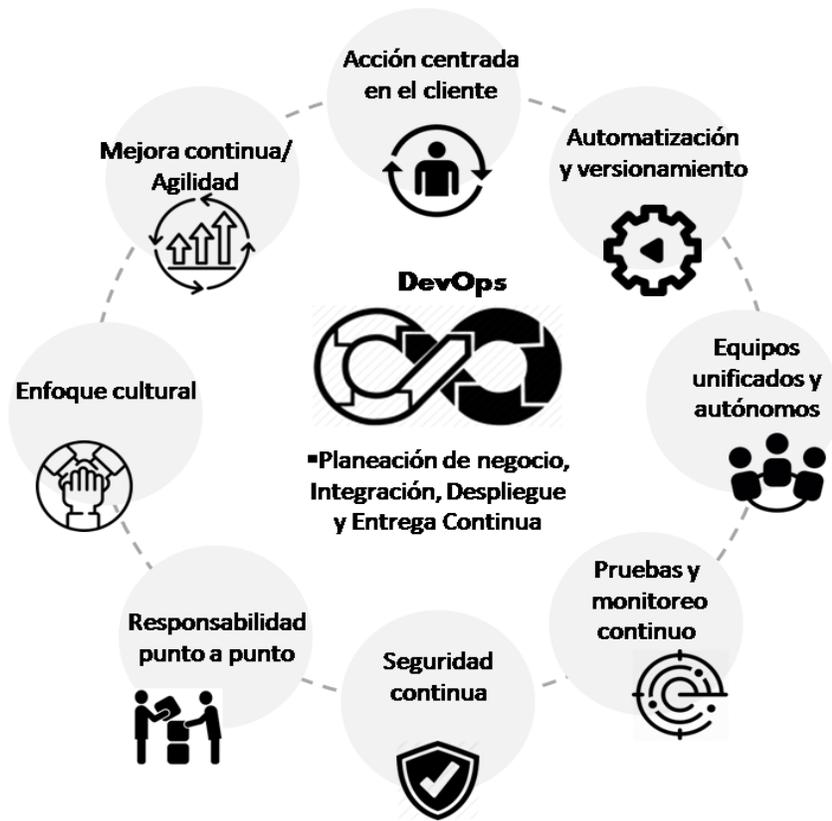
configuración de plataformas o herramientas para el versionamiento y reutilización de código a fin de optimizar la infraestructura de TI. Impulsa la mejora continua al disminuir silos entre las áreas clave, reducir tiempos y costos de entrega durante el proceso y minimizar el número de errores.

3. **Mejora continua/ Agilidad.** DevOps hace hincapié en la mejora continua de los procesos de desarrollo de software para minimizar el desperdicio, optimizar la velocidad, los costos y la facilidad de entrega. Se mantiene una constante retroalimentación con el cliente y se acelera la optimización de las aplicaciones dando agilidad al negocio.
4. **Enfoque cultural.** Se requiere una evolución hacia una cultura de colaboración, transparencia y comunicación entre las áreas clave. Para lograr dicha transformación y aportar valor busca la integración y alineación de desarrollo, pruebas y operaciones, así como del responsable del negocio y el cliente. Plantea una visión más amplia del producto a desarrollar adoptando nuevos métodos y herramientas, en lugar de colaborar como unidades de trabajo independientes.
5. **Equipos de trabajo unificados y autónomos.** Integrar equipos multifuncionales con perfiles genéricos que contemplen una gama equilibrada de competencias, en lugar de especialistas de TI con perfiles específicos (programador, analista, operación). Cada elemento del equipo es autónomo en la cobertura de su servicio y a la vez interactúa colaborando y comunicándose permanentemente.
6. **Responsabilidad de extremo a extremo.** El equipo DevOps debe asumir el compromiso de proporcionar atención y soporte desde la planificación hasta la operación del producto para garantizar el flujo completo del servicio y de los procesos que aseguren su entrega.

7. **Pruebas continuas.** Se requiere validar continuamente la configuración de los ambientes de desarrollo, pruebas y operaciones, los scripts de despliegue y las aplicaciones a fin de garantizar la calidad del producto final (IBM, 2017).
8. **Monitoreo continuo.** Orientado a la evaluación de los procesos de desarrollo a fin de verificar el comportamiento y rendimiento de las aplicaciones para detectar oportunamente incidencias en entornos de prueba y producción. También busca monitorear al personal en términos de desarrollo de competencias (conocimiento, habilidades y actitud).
9. **Seguridad continua.** En la transición de enfoque a DevOps, la seguridad asume un papel muy relevante ya que busca estar presente en toda la cadena de desarrollo del producto a medida que es construido, integrado, desplegado y entregado, sin que eso signifique retrasar al equipo de desarrollo o restar desempeño a la infraestructura de TI. Bajo un nuevo paradigma, trata de aplicar una óptica unificada para asegurar y operar las aplicaciones o productos desarrollados.

En la figura 35 se plantea la interacción entre los principios y las prácticas del enfoque DevOps.

Figura 35. **Prácticas del enfoque DevOps.**



2.3.5.5 DASA/Modelo de competencias DevOps

DASA estima que el perfil del ingeniero de TI está evolucionando hacia una orientación cada vez más genérica a medida que se consolidan los equipos DevOps en las organizaciones. La Asociación señala la importancia de conocer la combinación adecuada de habilidades y conocimientos necesarios para colaborar en ambientes DevOps y propone un Modelo de Competencias integrado por ciertas habilidades y áreas de conocimientos, mismas que están orientadas en cuatro direcciones.

- Motivación. Capacidad de adaptación y asimilación de valores y principios de la cultura organizacional alineada con los objetivos de TI y negocio. Facultad de recibir capacitación para desarrollar confianza y proactividad en la ejecución de funciones. Mejorar capacidad de análisis, tolerancia al fracaso y aceptación al cambio.

- ❑ Colaboración en equipo. Logro de metas comunes a través de la cooperación, la responsabilidad mutua, el propósito común y la capacidad de brindar soporte integral a lo largo del ciclo de desarrollo del producto.
- ❑ Liderazgo. Guiar al equipo de trabajo para lograr desempeño de alto rendimiento con valores y transparencia. Mantener una visión integral del ciclo de vida del producto, así como la importancia del rol de todos los interesados.
- ❑ Mejora continua. Mantener un enfoque de trabajo abierto, dinámico y flexible que permita la ejecución del proceso de mejora continua para desarrollar productos de calidad que garantice la satisfacción del cliente.

Las áreas de conocimiento para potenciar las capacidades de los equipos de trabajo se enlistan a continuación.

- Optimización del valor de negocio. Enfocado a eficientar el uso de la infraestructura de TI, incluyendo el ciclo de retroalimentación del cliente, la gestión del nivel de servicio, el monitoreo del rendimiento y la gestión de casos de negocio.
- Análisis de negocios. Consiste en la correcta identificación de requisitos de negocio, funcionales, no funcionales y de desarrollo a largo plazo de los procesos de negocio, análisis de datos y refinamiento de especificaciones.
- Arquitectura y diseño. Es la definición de la estructura de los componentes de la aplicación y las plataformas y/o ambientes necesarios para la operación de producto final.
- Programación. Conocimientos en codificación, compilación e integración de paquetes de software. Gestión de datos para la construcción de los componentes de la aplicación bajo los lineamientos vigentes de gobierno de datos.

- Entrega continua. Experiencia en pruebas automatizadas, administración de despliegue y lanzamiento, gestión de configuración, control de versiones y soluciones en la nube.
- Especificación de pruebas. Capacidad para elaborar casos de prueba con base en los requerimientos funcionales, de calidad y estructurales de la aplicación.
- Ingeniería de infraestructura. Experiencia en supervisión técnica, gestión del rendimiento TI, soporte de aplicaciones local y en la nube. Gestión de capacidad y disponibilidad.
- Seguridad, Riesgo y Cumplimiento. Capacidad para fusionar los componentes de la aplicación con los estándares y modelos de seguridad de TI conforme evoluciona el ciclo de desarrollo para garantizar la entrega y disponibilidad continua.

En el diagrama de la figura 36 se ejemplifica el Modelo de Competencias DASA entorno a las habilidades y capacidades requeridas.

Figura 36. **Modelo de Competencias DASA.**



2.3.5.6 Herramientas y Plataformas de automatización

Como parte de una estrategia eficiente de gestión de recursos, DevOps contempla entre sus máximas disminuir los silos entre desarrollo, pruebas, seguridad y operaciones a través de la implantación de herramientas y/o plataformas automatizadas para unificar el proceso de desarrollo y operaciones. Al mismo tiempo, la unificación garantiza el uso exclusivo de las herramientas durante las actividades de desarrollo y posteriormente en el proceso de pruebas, seguridad y despliegue, agilizando la cadena de suministro de software, disminuyendo errores manuales durante la ejecución de ambientes y optimizando la infraestructura de TI (Belalcázar, Díaz, Molinari & Rodríguez, 2016).

Entre los beneficios que DevOps proporciona al automatizar el ciclo de desarrollo y la configuración de la infraestructura están la integración continua, el control de versiones de

productos, las pruebas automáticas y el despliegue automático y continuo, ejecución automática de scripts y, al mismo tiempo, garantiza la estabilidad y seguridad de aplicaciones e infraestructura de TI para dar agilidad y confiabilidad al negocio.

Algunos ejemplos de las herramientas y/o plataformas de automatización para operar ambientes DevOps se describen en la Tabla 17.

Tabla 17. **Ejemplo de herramientas de automatización en entornos DevOps.**

Herramienta o Plataforma	Descripción
Puppet ⁴⁴	Puppet Enterprise gestiona la infraestructura como código, proporcionando la base para las prácticas DevOps como el control de versiones, las pruebas automatizadas, la integración continua y la entrega continua. Implementa los cambios en ambientes confiables y se recupera más rápidamente de las fallas, liberando al equipo DevOps para que sea más ágil y responda a las necesidades comerciales. Puppet puede ayudar a automatizar todo el ciclo de vida del desarrollo de la aplicación, desde la infraestructura central hasta el mantenimiento de la aplicación.
Chef ⁴⁵	En ambientes DevOps, la herramienta Chef configura a la infraestructura como un código, lo que garantiza que la política de configuración sea flexible, versionar el software, probar y asegurar

⁴⁴ <https://puppet.com/solutions/devops>

⁴⁵ <https://www.chef.io/chef/>

	<p>constantemente. Los servidores administrados por esta aplicación se evalúan continuamente en función del estado deseado, lo que garantiza que la modificación de la configuración se corrija automáticamente y que los cambios de configuración se apliquen universalmente. Evita el aumento constante del costo de la infraestructura de TI.</p>
<p>Accenture DevOps Platform (ADOP)⁴⁶</p>	<p>Es una plataforma diseñada para ayudar a los clientes a aumentar su agilidad para afrontar mejor las interrupciones. A través de su práctica de arquitectura nativa de cloud, ayuda a crear nuevas aplicaciones, aprovechando las arquitecturas basadas en contenedores y microservicios, con una implementación automatizada en la plataforma OpenShift Container de Red Hat. Accenture tiene como objetivo permitir a los clientes pasar a un modelo de TI de varias velocidades que les permita innovar y desarrollar nuevos servicios más rápidamente.</p>
<p>IBM Bluemix Continuous Delivery⁴⁷</p>	<p>Es una plataforma desarrollada por IBM en donde los equipos de entrega de software pueden crear aplicaciones nativas en la nube y aprovechar las API para conectarse a sistemas empresariales de registro, sistemas móviles e IoT. IBM Bluemix Continuous Delivery</p>

⁴⁶<https://www.redhat.com/es/about/press-releases/los-principales-proveedores-recurren-la-plataforma-de-contenedores-openshift-de-red-hat-para-impulsar-la-innovaci%C3%B3n>

⁴⁷ <https://www.ibm.com/blogs/bluemix/2016/11/simplify-devops-transformation-bluemix-continuous-delivery/>

	<p>es un nuevo servicio para permitir la entrega continua de aplicaciones en la nube brindando a los equipos de desarrollo una forma inteligente para la empresa de practicar DevOps. Contempla una gama de componentes para crear, implementar y administrar sus aplicaciones que permite que el desarrollo y las operaciones sean repetibles y más fáciles de gestionar.</p>
<p>Ansible⁴⁸</p>	<p>Herramienta de “Gestión de la Configuración”, enfocada a configurar la infraestructura para automatizar el despliegue de la misma. Tiene interfaces sencillas y de fácil uso con funcionalidades de aprovisionamiento, despliegue de aplicaciones (Ansible Tower) y orquestación a través de los despliegues de OpenStack. Impulsada principalmente por Red Hat, la herramienta tiene soporte para la mayoría de las distribuciones de Linux. Utilizada por compañías como Rackspace, CSC, HP, Cisco o IBM.</p>
<p>Docker⁴⁹</p>	<p>Es una referencia del segmento de los contenedores en la nube derivado de la necesidad por agilizar, simplificar y asegurar la distribución de aplicaciones en entornos escalables. Docker busca automatizar la instalación de aplicaciones dentro de contenedores de software, proporcionando una capa de virtualización que permite trabajar con esas aplicaciones en múltiples sistemas operativos. Es capaz de empaquetar cualquier aplicación y sus dependencias en un contenedor virtual que se puede posteriormente, ejecutar en cualquier servidor. Proporciona flexibilidad y portabilidad en las</p>

⁴⁸ <https://www.ansible.com/integrations/devops-tools>

⁴⁹ <https://www.ansible.com/integrations/devops-tools>

	aplicaciones en instalaciones físicas, cloud pública, cloud privada, etc.
Gradle ⁵⁰	Herramienta de desarrollo y automatización basada en código abierto enfocada a la construcción, automatización y entrega ágil de software de cualquier índole (aplicaciones móviles, sitios web, sistemas transaccionales, microservicios, etc.). Dirigido tanto a pequeñas start ups como a grandes empresas con diferentes necesidades, aplicaciones móviles, sitios web, sistemas transaccionales, microservicios, etc. Sobresale por su versatilidad para soportar diferentes tipos de código Java, C++, Python y otros lenguajes de programación, además de su soporte para IDEs como Netbeans, Eclipse, o IntelliJ IDEA. Otro diferenciador importante es su capacidad para crear paquetes incrementales, lo que ahorra a los desarrolladores tiempo en el proceso de compilación.
Kubernetes ⁵¹	Kubernetes Engine es un entorno de producción administrado que permite implementar aplicaciones en contenedores. Ofrece innovaciones recientes en productividad para desarrolladores, eficiencia de recursos, automatización de operaciones y flexibilidad de código abierto para acelerar el tiempo de salida al mercado. Lanzado por Google en 2015 tomando la experiencia de ejecución de Gmail y YouTube en contenedores. Entre su principal

⁵⁰ <https://gradle.org/>

⁵¹ https://cloud.google.com/kubernetes-engine/?utm_source=google&utm_medium=cpc&utm_campaign=latam-MX-all-es-dr-skws-all-all-trial-e-dr-1007178-LUAC0009353&utm_content=text-ad-none-none-DEV_c-CRE_346533890099-ADGP_SKWS+%7C+Multi+~+Compute+%7C+Kubernetes-KWID_43700043800334816-kwd-299675360776-userloc_1010042&utm_term=KW_kubernetes-ST_Kubernetes&gclid=CjwKCAjw7anqBRALeIwAgvGgmy8MUwzIttacQtw4VQ1eHr5K2ahqaya4nm8uLG07SxE5WI7XJLxR0xoCQ6AQAvD_BwE&gclsrc=aw.ds

	diferenciador destaca su soporte para diferentes ambientes de ejecución de contenedores, como puede ser la propia Docker.
--	---

Sin embargo, el equipo DevOps podrá analizar la gama de herramientas de automatización existentes y valorar cuál es la más adecuada en alineación con los objetivos estratégicos de TI y de negocio. Es decir, no existe una recomendación genérica de interfaces a implantar bajo el enfoque DevOps, depende de diversos factores como tipo de organización, objetivos de negocio, sistemas de gestión propietarios, heredados o en la nube, modelo de operación e infraestructura de TI, tipo de productos y servicios desplegados, entre otros.

2.3.5.7 Experiencias de adopción del enfoque

DevOps ha ido tomando relevancia en el mercado a nivel global mostrando impactos positivos en las estructuras de TI y dando agilidad al negocio. En la Tabla 18 se describen cuatro experiencias de su adopción en organizaciones con diferentes contextos empresariales (Boulton, 2018).

Tabla 18. Experiencias de adopción del enfoque DevOps.

Empresa	Experiencia
 <p>Es un corporativo dedicado al desarrollo y comercialización de software empresarial y</p>	<ul style="list-style-type: none"> Contemplaron un área directiva denominada DevSecOps (Boulton, 2018). La adopción del enfoque fue práctica y exitosa.

<p>financiero⁵², con sede en Mountain View, California. Tiene más de 30 años en el mercado y cuenta con una plantilla de 4,500 desarrolladores.</p>	<ul style="list-style-type: none"> ● Implicó concientizar al personal de ser un factor clave en el modelo de negocio para crear valor al cliente. ● Lograron transitar del pensamiento individual al colaborativo. ● El ciclo de satisfacción no solo incluyó al cliente, sino también el personal que participó en la creación o innovación, con un impacto positivo en el desempeño.
Empresa	Experiencia
 <p>Aseguradora⁵³ con 150 años de servicio ofrece seguros de personas y servicios financieros. Tiene operaciones alrededor del mundo y cuenta con una cartera de 90 millones de clientes y 56 mil empleados.</p>	<ul style="list-style-type: none"> ● Primero migraron los sistemas tradicionales hacia plataformas en la nube (Microsoft e IBM) y contenedores ligeros y portables (Boulton, 2018). ● Después, integraron equipos interdisciplinarios para operar dichas plataformas en combinación con metodologías ágiles y prácticas DevOps. ● Enfocados al desarrollo rápido, detección pronta de errores y despliegue ágil.

⁵² <https://www.intuit.com/>

⁵³ <https://www.metlife.com.mx/acerca-de-metlife/conocenos/>

<p>En México se estiman cerca de 10 millones de clientes.</p>	
Empresa	Experiencia
 <p>Entre⁵⁴ los principales proveedores mundiales de soluciones de TI empresariales de código abierto, Red Hat lleva 26 años en el mercado. Ofrece a sus clientes modelos de suscripción predecibles y asequibles para adquirir tecnologías fiables y de alto rendimiento en la nube.</p>	<ul style="list-style-type: none"> • Adoptaron DevOps para modernizar la estructura de TI a través de la automatización integral digital (Boulton, 2018). • También para facilitar el proceso de manipulación de software empaquetado que suelen comprar pero que es difícil de adaptar a otros ambientes. • La adquisición de Red Hat por parte de IBM en 2018⁵⁵ busca que dicha marca sea referente global de proveedores en nubes híbridas y en servicios con entornos DevOps.
	<ul style="list-style-type: none"> • Su objetivo con DevOps residió en eliminar silos entre plataformas convencionales, heredadas e

⁵⁴ <https://www.redhat.com/es>

⁵⁵ <https://www.xataka.com/empresas-y-economia/ibm-compra-red-hat-34-000-millones-dolares-hito-historico-para-empresa-que-triunfo-linux-open-source>

<p>Fundada⁵⁶ en 2004, con sede en New York, contempla una cartera de redes de televisión, una compañía cinematográfica y actividades de gestión de parques temáticos. Integrada por 17 compañías individuales y unidades de negocios.</p>	<p>híbridas, públicas y privadas con las que operaban (IBM, 2018).</p> <ul style="list-style-type: none"> ● Además de incrementar su cartera de clientes a través de la mejora continua de productos y servicios tecnológicos, de agilizar la capacidad de respuesta y de lograr reducir el número de errores en producción. ● Lograron contar con una interfaz unificada para ver el flujo de trabajo completo de construcción e integración de paquetes de software, fechas de implantación, etc. ● También reducir el tiempo de comercialización de productos de TV de 20 a 5 semanas agilizando el modelo de negocio.
--	--

III. Resultados

3.1 Situación actual de Bancatec

3.1.1 Misión y visión

El grupo financiero al que refiere el presente protocolo está enlistado entre las principales organizaciones del sector en México y ofrece una amplia gama de productos y servicios a

⁵⁶ <http://www.nbcuniversal.com/who-we-are>

través de su banco, casa de bolsa, compañías de pensiones y seguros. Además dicha entidad está entre los proveedores más importantes en el ámbito crediticio gubernamental e hipotecario. Históricamente se ha posicionado como una de las instituciones bancarias más rentables en México, debido al fortalecimiento en su nivel de capitalización y la calidad de sus activos, entre otros factores.

La visión de Bancatec está orientada a convertirse en un gran aliado en el crecimiento del país y la misión se enfoca en generar confianza y fortaleza financiera para los clientes. Entre sus valores resaltan la integridad, confianza y lealtad, así como solidaridad, innovación, respeto y responsabilidad.

3.1.2 Estrategia de transformación digital.

Parte de la estrategia de transformación digital de Bancatec que busca posicionarse como uno de los bancos líderes de banca digital en el mercado, incluye la mejora de la funcionalidad en los aplicativos webs para implementar navegaciones tipo *header* que impulsen la usabilidad de la banca en línea y mejorar la satisfacción del cliente.

La organización también busca adoptar arquitecturas tecnológicas flexibles y multicanal que permitan generar experiencias únicas a fin de que los clientes nuevos adopten al 100% la plataforma de banca digital.

En la banca móvil son diversos los servicios, mejoras y nuevas funcionalidades por implementar en su totalidad para incrementar el compromiso del cliente y aumentar las transacciones financieras que le permita a la organización escalar posiciones en el ranking competitivo del sector financiero en México.

- Tarjeta digital con códigos dinámicos para agilizar las compras electrónicas con los datos mínimos del cliente y dar mayor certeza y seguridad en las transacciones.
- Autenticación biométrica para acceder a la banca móvil a través de huella digital o selfie.
- Seguimiento “*Mobile engagement*”⁵⁷ a través del teléfono móvil del cliente para atender incidencias o solicitudes de manera personalizada para impulsar el compromiso continuo.
- Retiro de efectivo sin tarjeta física a través de códigos seguros generados desde la aplicación móvil.

3.1.3 Matriz de factores que limitan la entrega ágil de aplicaciones móviles.

Bajo el esquema actual de operación de las áreas de desarrollo e infraestructura de TI de la organización se presentan diversas problemáticas y/o aspectos en el proceso de integración, despliegue y entrega de aplicaciones de banca móvil, descritas a detalle en el capítulo anterior, que limitan la implementación ágil de nuevas funcionalidades y mejoras obstaculizando la entrega de valor al cliente.

En la matriz de la Tabla 19 se presenta un análisis de los principales factores que limitan la entrega ágil y la mejora continua de productos de banca móvil en Bancatec con el nivel de incidencia o presencia y las áreas con mayor afectación.

⁵⁷ El compromiso móvil es el acto de involucrar a un usuario a través de los canales de mensajería disponibles dentro y fuera de una aplicación. Refiere la capacidad de ofrecer experiencias de marca positivas, respaldar los objetivos comerciales y construir relaciones valiosas con los clientes a largo plazo.
Recuperado de https://www.arubanetworks.com/assets/so/SO_MobileEngagement.pdf

Tabla 19. **Matriz de factores que limitan la entrega ágil y la mejora continua de productos de banca móvil en Bancatec.**

Factor	Áreas afectadas				Nivel de incidencia
	Desarrollo	Calidad	Operaciones	Seguridad	
Procesos manuales y repetitivos que restan agilidad al negocio	X	X	X	X	Alto
Falta de claridad en objetivos comunes.	X	X	X		Alto
Deficiente comunicación entre equipos de trabajo.	X	X	X	X	Alto
Silos entre desarrollo, infraestructura y seguridad.	X		X	X	Alto
Sobrecarga de actividades en áreas operativas.			X		Medio
Ausencia de una cultura organizacional integradora y colaborativa.	X	X	X	X	Alto
Claridad en los tramos de responsabilidad	X	X	X		Medio
Generación de desperdicios y demoras en despliegue de paquetes	X	X	X		Medio

Como se puede observar en la matriz son cinco los factores con mayor incidencia que impactan y restan agilidad al proceso de desarrollo y despliegue de aplicación móviles el cual involucra a las áreas de desarrollo, calidad, seguridad y operaciones de la organización. Entre los más sobresalientes están las actividades manuales y repetitivas que restan agilidad al negocio y calidad al producto final. Además de propiciar cuellos de botella en el área de operaciones o infraestructura generando desperdicios y demoras en el despliegue de aplicaciones móviles.

También, resalta la falta de enfoque común entre las diferentes áreas involucradas en la construcción y despliegue del producto final, generando silos entre los equipos. Otro factor que impacta negativamente es la ausencia de una cultura organizacional integradora en donde los ambientes de trabajo operan de manera aislada y sin una visión integral de colaboración.

Se recomienda implementar en Bancatec un marco de trabajo basado en la filosofía DevOps que agilice e impulse el proceso de construcción y entrega de productos de banca móvil bajo un enfoque de calidad y mejora continua y al mismo tiempo alinee a las áreas compartiendo objetivos comunes con esquemas de trabajo unificado, mismo que se describe en el siguiente apartado.

3.2 Marco de trabajo “*Framework DevOps for Mobile Banking*”

3.2.1 Definición, objetivos, principios y componentes

“*Framework DevOps for Mobile Banking*” es un marco de referencia enfocado a la alineación, transformación y optimización de equipos de trabajo, procesos de desarrollo y entrega de productos móviles financieros, tecnología y planes de seguimiento, bajo

directrices DevOps, para agilizar la entrega continua, impulsar la calidad y la mejora de aplicaciones de banca móvil.

Dicho marco busca evolucionar el entorno de los equipos de trabajo, desde la planificación hasta la entrega, hacia ambientes estrechamente colaborativos, automatizados y eficientes vinculando oportunamente a las áreas de negocio en todo el flujo que implica la entrega continua de aplicaciones móviles para mejorar las capacidades de respuesta de despliegue ágil y de mejora continua.

“*Framework DevOps for Mobile Banking*” contempla cinco objetivos a seguir por las organizaciones para asegurar la entrega ágil y mejora continua de aplicaciones móviles con valor al cliente. Consultar figura 37.

1. **Acción centrada en el negocio.** Alinear el esfuerzo de los equipos de trabajo bajo una misma visión de objetivos y métricas orientados por el **valor** del negocio.
2. **Simplificar y agilizar** el proceso de integración, despliegue y **entrega continua** de productos de banca móvil.
3. **Reducir el tiempo de comercialización** del producto final de banca móvil desde la planificación hasta la entrega final.
4. Contar con ambientes automatizados que permitan **implementaciones con mayor frecuencia y eficiencia** en entornos de trabajo estables y ágiles, disminuyendo el número de defectos o errores manuales e impulsando la calidad del producto final.
5. Impulsar la **mejora continua** y la evolución de procesos DevOps a través del seguimiento de indicadores clave que permitan medir y monitorear eficazmente.

Figura 37. Objetivos “*Framework DevOps for Mobile Banking*”



Como marco de referencia “*Framework DevOps for Mobile Banking*” plantea una serie de principios que sirven de guía a la organización para aprovechar todos los beneficios de los entornos de trabajo basados en DevOps. Ver diagrama de la figura 38.

- **Practicar una cultura flexible y colaborativa con responsabilidad compartida.**

Evolución hacia una cultura de colaboración entre las áreas clave integrando y alineando a los equipos de trabajo bajo una visión más amplia del producto y el valor que cada uno aporta.

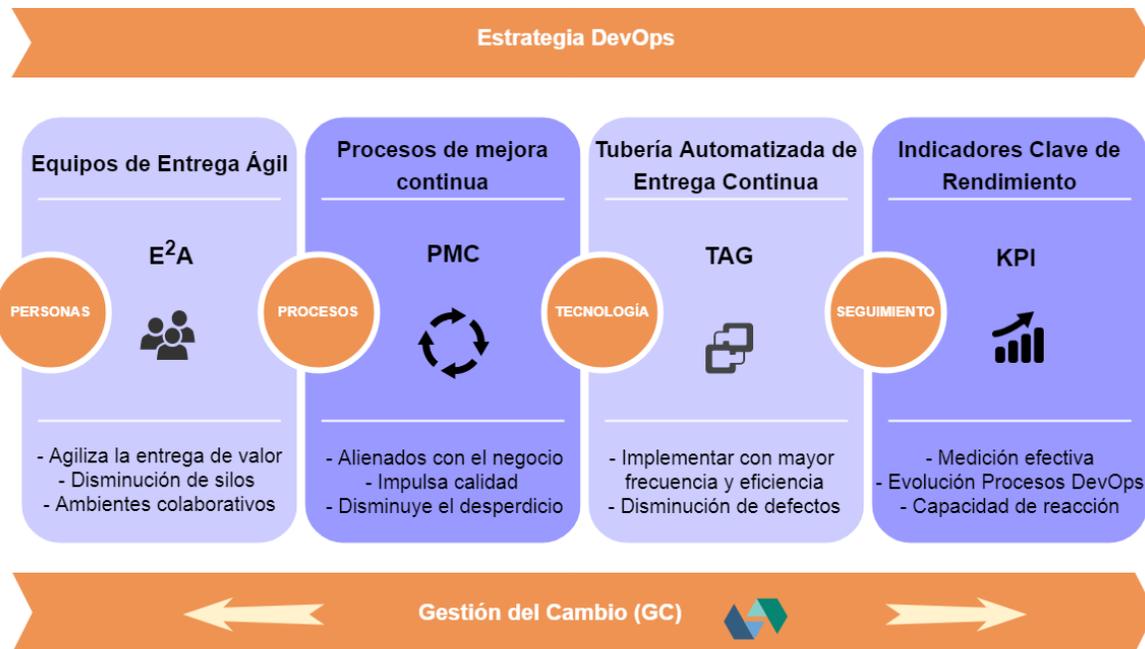
- **Construir Paquetes Mínimos Viables (PMV).** Fabricación ágil de paquetes de software incrementales listos para desplegar en cualquier ambiente, disminuyendo los ciclos de retroalimentación con el cliente.
- **Equipos de Entrega Continua.** Integrar equipos multifuncionales con perfiles genéricos que contemplen una gama equilibrada de competencias,
- **Integración y Prueba Continua.** Integración de código identificando dependencias, análisis de código estático, pruebas de seguridad y detección temprana de defectos manteniendo un enfoque de "falla rápida".
- **Despliegue Continuo a través de tuberías (“*pipeline*”) automatizadas.** Preparar los paquetes de software integrados a través de ambientes automatizados para asegurar la disponibilidad de las aplicaciones y aumentar el número de despliegues frecuentes conforme la operación lo requiera disminuyendo el tiempo de comercialización e impulsando la calidad final del producto.
- **Entrega Continua.** Estandarizar proceso de aprobación y entrega final de PMV asegurando la calidad con periodos cortos de retrospectiva con el cliente. Hay un componente humano o responsable de negocio que decide la puesta en producción.
- **Uso continuo de métricas.** Medir la evolución de la automatización de los procesos es clave para corregir oportunamente la dirección de los planes de transformación de los ambientes de trabajo alineados con las metas técnicas y de negocio.

Figura 38. Principios “*Framework DevOps for Mobile Banking*”.



“*Framework DevOps for Mobile Banking*” considera cinco componentes para su implementación en la organización basados en la estrategia DevOps: Procesos enfocados a la Mejora Continua (PMC), Equipos de Entrega Ágil (E²A), Tubería de Automatización de Entrega Continua (TAG), Indicadores clave de rendimiento (KPI) y Gestión del cambio (GC). Ver mapa conceptual de la figura 39.

Figura 39. Componentes de “*Framework DevOps for Mobile Banking*”.



Una de las fortalezas del marco de referencia es su capacidad de adaptabilidad en la fase de definición y desarrollo del producto a las metodologías de desarrollo ágil como Lean, SCRUM, XP y Kanba, entre otras, que permiten generar valor de manera ágil e impulsar la entrega continua. Asimismo, en caso de que el contexto tecnológico y de negocio sugiera una implementación del marco de trabajo en fases o por componentes es posible aplicarlo en entornos tradicionales centrándose únicamente en la automatización y mejora de procesos vinculados con la integración, despliegue y entrega continua de aplicaciones.

Para tales efectos, la implementación de todos los componentes del marco no es obligatoria ni secuencial. Por ejemplo, se puede operar en una primera etapa con una tubería automatizada de entrega continua acompañada de una serie básica de KPI. En una segunda fase, se puede implementar el resto de los componentes, pero siempre alineados con los objetivos de TI y de negocio y en su conjunto bajo los principios y las prácticas del marco de trabajo en cuestión.

La implementación del “*Framework DevOps for Mobile Banking*” en la organización puede representar diversos beneficios y ventajas en los ámbitos técnico y de negocio. Entre ellos se encuentran la entrega ágil y continua de valor, pensamiento centrado en agregar valor y mejora continua al producto en lugar de mantenerlo estable, disminución del time-to-market, ambientes de desarrollo y operaciones veloces y estables, disminución de defectos, capacidades de respuesta bajo el enfoque “falla rápida”, estandarización de procesos de integración, despliegue y entrega continua e implementaciones frecuentes con periodos cortos de retrospectiva. Ver diagrama de la figura 40.

Figura 40. **Beneficios para la organización “*Framework DevOps for Mobile Banking*”.**



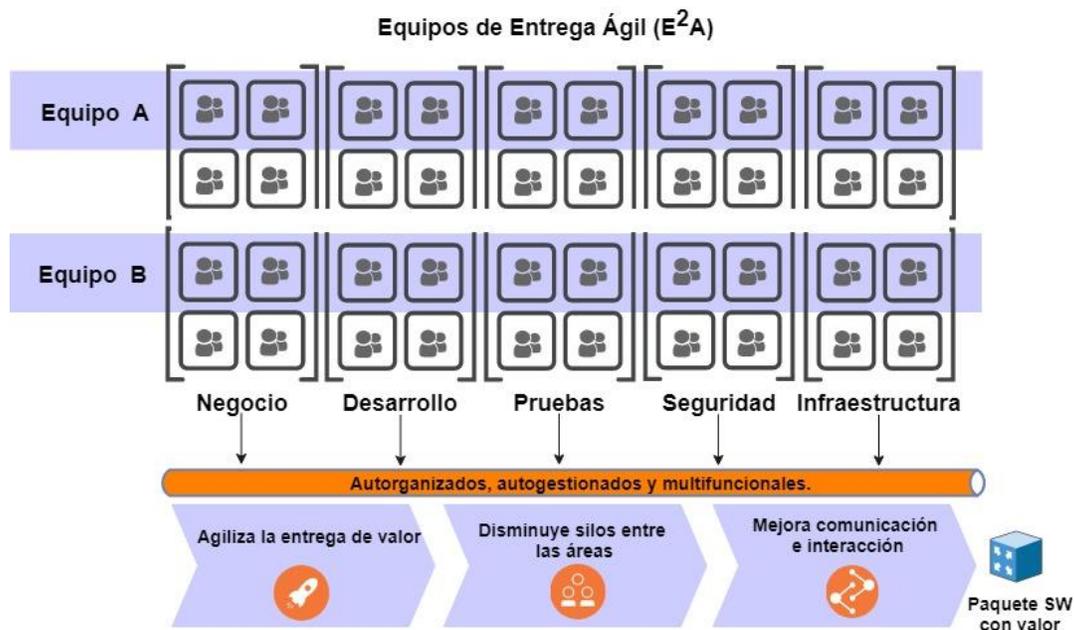
En el caso de la organización de estudio y debido al estado actual en que operan los ambientes de trabajo se sugiere instrumentar los componentes del marco de trabajo en su totalidad para

lograr una mayor cohesión entre todos los elementos y así obtener un mayor número de beneficios para la empresa.

3.2.2 Equipos de Entrega Ágil (E²A)

Bajo un nuevo pensamiento organizacional, se plantea formar equipos multifuncionales, autónomos y alineados con los objetivos de negocio para optimizar y agilizar la entrega de valor al cliente en cortos intervalos de tiempo, conocidos como iteraciones. Se busca reducir los silos entre los equipos, mejorar la colaboración y la comunicación y sobre todo, agilizar el despliegue y la entrega del producto al cliente en entornos DevOps.

Figura 41. Integración de Equipos de Entrega Ágil (E²A).



En la figura 41 se esquematiza la manera en que los grupos E²A se organizan, considerando a los integrantes de las áreas clave que participan en la definición, la construcción, las

pruebas, la implementación y la operación del producto final visualizando como meta final la entrega de Paquetes Mínimos Viables con valor al cliente.

Se asigna por lo menos a un elemento de cada departamento para la conformación de los equipos ágiles y flexibles según su perfil, y en caso de ser necesario se contratan perfiles específicos con habilidades basadas en DevOps. Se trata de que los grupos E²A colaboren de manera muy cercana planeando, integrando, implementando y desplegando los paquetes funcionales y, a su vez, obteniendo una retroalimentación rápida y efectiva.

Los roles basados en DevOps que se plantean para la implementación de los grupos E²A, se describen a continuación. Ver figura 42.

- **Líder de Negocio.** Conoce el negocio y define el producto. Responsable de definir el alcance y requerimientos, así como de la priorización de los productos de banca móvil. En el presente marco de trabajo es importante contextualizar al líder de negocio con las prácticas DevOps con el objeto de conocer los beneficios y aportaciones de “*Framework DevOps for Mobile Banking*” y sea parte del proceso ágil en la definición y priorización de tareas.
- **Líder de Proyecto DevOps.** Coordinador y facilitador del grupo de trabajo que implementa y organiza DevOps en toda la organización. Este rol funge como evangelizador de los beneficios de “*Framework DevOps for Mobile Banking*” entre todos los integrantes del equipo, identificando y cuantificando las ventajas acumuladas de dichas prácticas. Se busca que este rol tenga la aceptación del representante de negocio así como de los equipos de desarrollo y operaciones para que la transición hacia los principios DevOps sea posible. También funge como líder responsable de

la correcta articulación entre las personas, los procesos y las herramientas de automatización que permitan la formalización del uso del ambiente de trabajo.

- **Gerente de Lanzamiento.** Es el gerente de proyecto responsable de la administración y coordinación del proyecto desde la definición, el desarrollo hasta la producción del producto final de banca móvil. Se enfoca en impulsar la entrega continua del producto a través de la supervisión de la integración y la coordinación del desarrollo, las pruebas y la implementación. A diferencia de los gerentes de proyecto tradicionales, este rol requiere de conocimientos técnicos y habilidades especializadas para ejecutar y mantener una cadena de herramientas de entrega de aplicaciones de extremo a extremo. Proporcionan visibilidad en todo el proceso DevOps midiendo e interpretando métricas en todas las tareas.
- **Experiencia de Garantía.** El control de calidad en los productos de banca móvil desempeña un papel integral en la entrega exitosa del producto final en el desarrollo de software. Este rol es clave para asegurar una óptima experiencia del usuario sobre el producto terminado. Bajo la perspectiva del “*Framework DevOps for Mobile Banking*”, surge la necesidad de un nuevo tipo de control que no solo pruebe la funcionalidad del producto, sino que también incluya pruebas de experiencia del usuario. Además de asegurar que el producto final tenga todas las características definidas en la especificación inicial, también incluye que la construcción del producto proporcione una buena experiencia de usuario.
- **Arquitecto de Automatización.** Es un papel fundamental que tiene como objetivo identificar las herramientas y los procesos correctos para construir un entorno

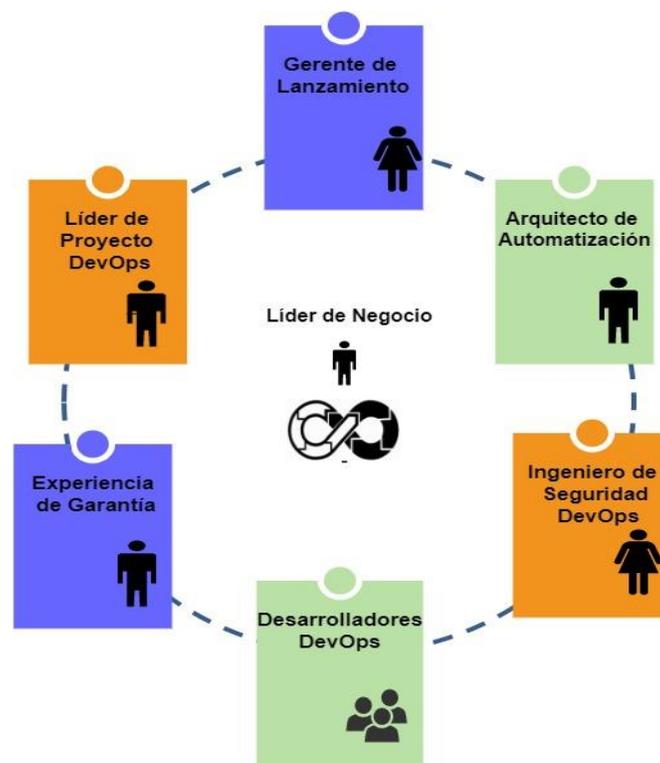
DevOps automatizado, eficiente y eficaz en la organización. Diseña, analiza e instaure estrategias para la implementación y entrega continua, además de garantizar una alta disponibilidad en los sistemas de desarrollo, pruebas y producción. Conforme se implementa una mayor automatización, este rol impulsa la entrega de código con mayor calidad y mayor frecuencia.

- **Ingenieros de Seguridad DevOps.** Este rol es responsable de impulsar la seguridad de las aplicaciones y de la infraestructura desde la definición inicial del producto de banco móvil hasta la entrega final. El Ingeniero de Seguridad DevOps requiere seleccionar las herramientas adecuadas para integrar la seguridad de manera permanente. Para tales efectos, este rol puede adoptar entornos de desarrollo integrados que permitan ejecutar permanentemente acciones de verificación a lo largo del proceso de desarrollo, integración y entrega. Esta figura trabaja en conjunto con los desarrolladores con el objetivo de ofrecer un producto que sea resistente no solo a los ataques técnicos, sino también uno que ofrezca cierta protección al usuario contra los ataques de ingeniería social⁵⁸.
- **Desarrollador de software.** Requiere de una visión integral de los entornos técnicos para construir productos de banca móvil, capacidades para utilizar variedad de tecnologías, habilidades para codificar y realizar scripts, y conocimiento de herramientas de automatización en entornos DevOps. Este rol además requiere de capacidad de gestión de datos y enfoque en los resultados del negocio. En

⁵⁸ Conjunto de técnicas que permiten manipular o influenciar las acciones de un individuo o grupo de personas con el fin de obtener información confidencial para su explotación con fines ilícitos (De Salvador, 2011).

“*Framework DevOps for Mobile Banking*” las responsabilidades del desarrollador de software aumentan en alcance ya que no solo se espera que se dedique a escribir código, sino que además realice pruebas unitarias, implementación y monitoreo continuo a través de plataformas automatizadas de desarrollo y despliegue con el fin de mejorar la calidad y la eficiencia.

Figura 42. Roles en un contexto “*Framework DevOps for Mobile Banking*”.



La estructura de un equipo DevOps depende del contexto del proyecto y del tipo de organización. Bajo la perspectiva de “*Framework DevOps for Mobile Banking*” para crear y preservar equipos funcionales y eficaces que aporten resultados ágiles con la calidad y la funcionalidad esperada, se busca que evolucionen, como parte del paradigma organizacional, de lo tradicional a lo ágil para instaurar mejores ambientes de trabajo en beneficio de todos

los integrantes y de la organización. En este sentido, la gestión del cambio se detalla en un apartado posterior del presente capítulo.

“*Framework DevOps for Mobile Banking*” contempla una serie de habilidades y áreas de conocimiento, basadas en el Modelo de Competencias DASA, para potenciar y fomentar las capacidades de los integrantes en los E²A. Ver esquema de la figura 43.

Figura 43. **Modelo de Competencias de los E²A.**

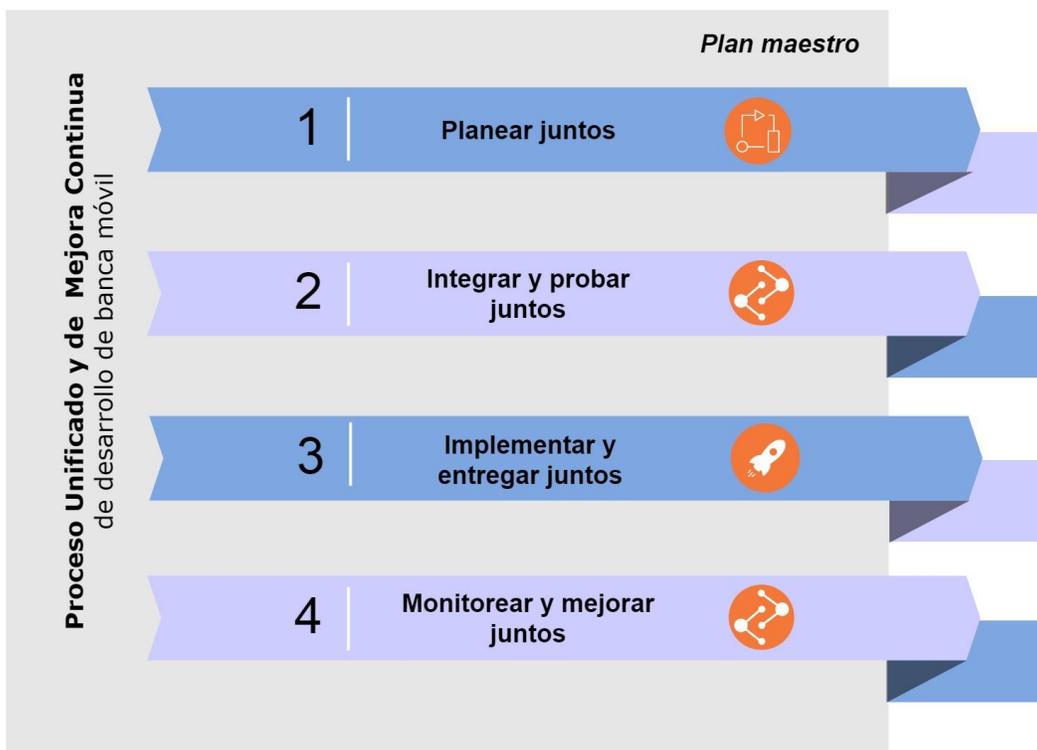


Nota. Fuente: Modelo de Competencias DASA. <https://www.devopsagileskills.org/dasa-competence-model/>

3.2.3 Proceso unificado y de mejora continua de desarrollo y despliegue de banca móvil.

Para el ciclo de desarrollo de soluciones móviles se plantea un proceso unificado y de mejora continua que tiene como eje un plan maestro, contextualizado con el negocio y la tecnología, en donde los diferentes ambientes y equipos interactúan de una manera colaborativa y alineada bajo objetivos comunes. Se trata de cuatro fases simplificadas y centradas en la entrega y mejora continua del producto en donde el enfoque, la colaboración y la retroalimentación son una constante. El proceso unificado se despliega en el diagrama de la figura 44.

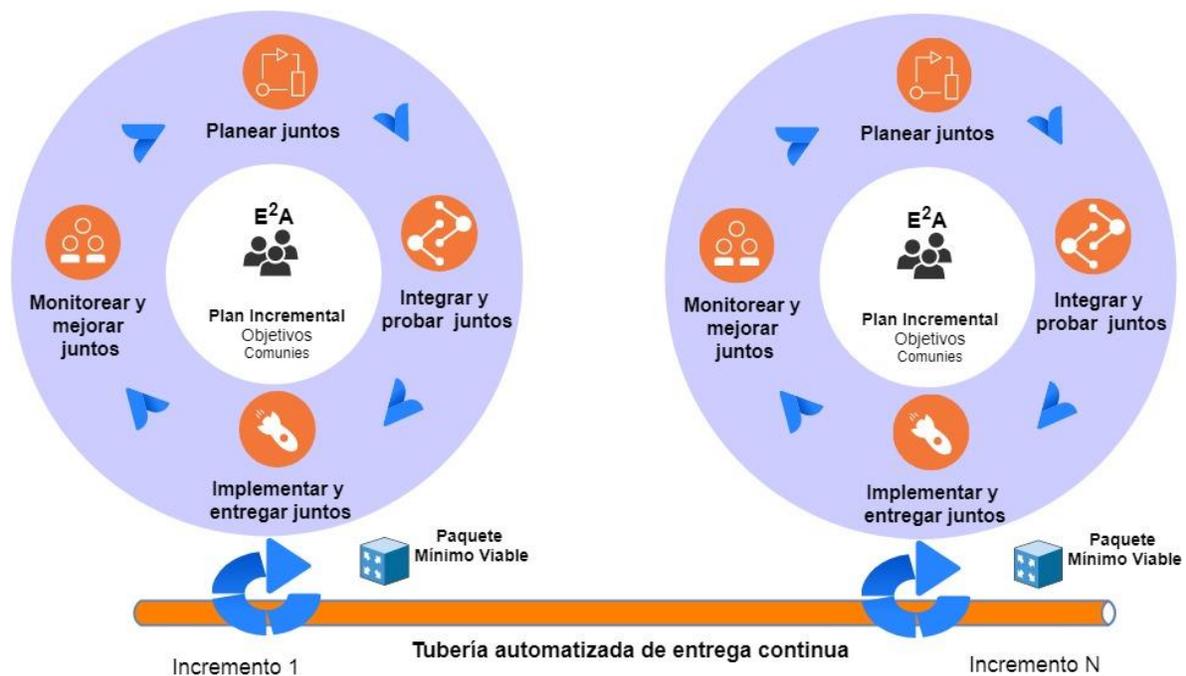
Figura 44. Proceso unificado y de mejora continua de desarrollo de banca móvil.



El proceso en cuestión plantea un esquema de trabajo unificado y ágil que se ejecuta en periodos de tiempo cortos iterativamente según el contexto lo sugiera. Su base es el plan

maestro incremental que permite articular y coordinar al E²A para que los elementos del equipo participen de manera coordinada en la definición de las actividades en cada iteración asegurando la calidad y el valor incremental en la entrega continua. Ver esquema de la figura 45.

Figura 45. Esquema Incremental del Proceso Unificado de Entrega y Mejora Continua.



Parte del proceso se ejecuta a través de la Tubería automatizada de entrega continua, pieza fundamental del pensamiento DevOps y descrita en el siguiente apartado, la cual permite unificar y darle agilidad a la integración, despliegue y entrega continua de los paquetes mínimos viables.

1. PLANIFICAR JUNTOS. Con la participación de todos los integrantes del E²A se busca tener claridad en los requerimientos de negocio y el contexto tecnológico para definir un plan

maestro incremental. El líder o representante de negocio prioriza la entrega del producto en **Paquetes Mínimos Viables (PMV)** con valor incremental. Tomando como base los planes de desarrollo ágil en ambientes SCRUM, se plantea un programa de trabajo con entregas en cortos periodos de tiempo (p. ej. 2 o 4 semanas) y con sesiones periódicas entre los integrantes de los equipos para dar seguimiento al plan maestro y asegurar la entrega de los PMV en tiempo y forma. La periodicidad de las entregas y del seguimiento dependerá de la naturaleza de cada proyecto o de las necesidades del cliente.

La participación temprana y permanente del área de negocio en la definición inicial facilita la dirección del proyecto y la rápida toma de decisiones en función del alcance, la lista priorizada de objetivos y la identificación oportuna de riesgos.

Se trata de una planificación estratégica que evoluciona a lo largo de toda la vida del producto debido a cambios de necesidades del cliente, del mercado, aparición de nuevas ideas, competitividad tecnológica, entre otros aspectos. La información mínima del plan maestro para cada iteración puede considerar una serie de artefactos, también basados en SCRUM, mismo que se detallan a continuación:

- A. **Project Planning.** Actividades de planificación del proyecto con estimación del esfuerzo y tiempos de entrega.
- B. **Mapa Funcional/ Diagrama de Vistas.** Representación gráfica del análisis funcional que permite delimitar y desagregar sucesivamente las funcionalidades del producto de una manera ágil y visual. El diagrama consiste en el diseño Front - End de la aplicación que permita la navegación rápida a las funcionalidades mínimas viables.

- C. **Product Backlog.** Listado priorizado de características y funcionalidades de la aplicación móvil que representa la visión y expectativas del cliente respecto a la entrega del producto. Su alcance puede cambiar a lo largo de la ejecución.
- D. **Sprint Backlog.** Subconjunto de requerimientos del Product Backlog a ejecutar en un sprint sin posibles cambios para asegurar la entrega en el tiempo estimado.
- E. **Modelo de Uso / Modelo y Notación de Procesos de Negocio.** Según el contexto del proyecto se emplearán casos de uso o diagramas de procesos de negocio para estructurar los componentes del PMV de una manera ágil y de fácil entendimiento.
- F. **Historias de usuarios.** Representación de requisitos escritos de las necesidades o funcionalidades que solicita el cliente utilizando el lenguaje común del usuario. Permite administrar los requerimientos para el desarrollo y entrega de aplicaciones sin excesivos formatos.

2. INTEGRAR Y PROBAR JUNTOS. En contexto al plan incremental los equipos ágiles cooperan y colaboran en la construcción, la integración y las pruebas considerando tanto las metas funcionales como la calidad del paquete incremental por desplegar. Es decir, se busca lograr la meta pero con incrementos medibles basados en la calidad que eviten el desperdicio de recursos. La iteración consiste en integrar y probar continuamente con la ayuda de herramientas automatizadas para asegurar el éxito del despliegue del paquete en entornos de calidad que a su vez permita fortalecer las capacidades de respuesta rápida ante posibles fallos. En esta etapa es crucial la automatización de la infraestructura, los despliegues y gestión de la configuración.

Durante esta fase que abarca el desarrollo, integración, pruebas y despliegue continuo se puede considerar una gama de acciones en su mayoría ejecutadas a través de la Tubería Automatizada de Entrega Continua (TAG), las cuales se enlistan a continuación:

- Desarrollo y/o actualización del código de los componentes del PMV.
- Compilación e integración del código en el repositorio base e identificación de dependencias - TAG.
- Análisis de código estático y optimización del control de versiones - TAG.
- Implementación del código en los ambientes de pruebas y calidad - TAG.
- Automatización de pruebas de funcionalidad y seguridad con secuencias de comandos de ejecución sin intervención manual - TAG.
- Detección temprana de defectos manteniendo un enfoque de "falla rápida".
- Despliegue Continuo a través de TAG para asegurar la disponibilidad de las aplicaciones y aumentar el número de despliegues frecuentes.

Se plantea un conjunto de artefactos para ir desarrollando durante esta fase como parte del seguimiento al plan maestro incremental, que puede ser parte del paquete de entregables finales, los cuales se detallan a continuación:

- A. Código fuente del PMV. Paquete de código versionado, compilado e integrado con una estructura lógica (código fuente en sí, archivos de recursos y vistas, librerías de código, etc.).
- B. Arquitectura con los componentes de la solución. Diagrama de los componentes del aplicativo móvil por capas (Front-End/Back-End, Esquemas de Escalabilidad/Modularidad).

- C. Matriz de casos de prueba de Aceptación. Permite validar la correcta función del PMV en forma unitaria, integral, de certificación y performance. Se prueba la interacción con otros elementos, dispositivos o escenarios necesarios para su funcionamiento. Se ejecutan, se observan los resultados obtenidos y se registran en un Reporte de Aceptación de Pruebas asegurando la funcionalidad y calidad esperada del producto.
- D. Documentación del código. A medida que se codifica y se prueba cada elemento, se redacta la documentación sobre lo desarrollado.

3. IMPLEMENTAR Y ENTREGAR JUNTOS. Los equipos ágiles se concentran en las actividades diarias de desarrollo con la seguridad de que pueden mandar a producción un lanzamiento de calidad, en cualquier momento que deseen, sin una complicada implementación y sin tener que seguir pasos de manuales ejecutados por un especialista.

Esta fase es para implementar continuamente los paquetes incrementales y asegurar la entrega al cliente final generando valor a través de todo el proceso de entrega continua. Con base en el plan incremental y una comunicación constante y efectiva, los equipos se sincronizan buscando aumentar el número de entregas frecuentes y exitosas conforme la operación y el contexto de proyecto lo requiera. Con la implementación temprana y frecuente en producción se estima reducir tiempos muertos y desperdicio de recursos.

Los equipos continúan con las tareas diarias de desarrollo con la confianza de que pueden mandar a producción un lanzamiento de calidad, en cualquier momento que deseen, sin una complicada implementación y sin tener que seguir pasos de manuales ejecutados por un especialista.

Entre las acciones a considerar durante esta etapa están:

- Entrega o Liberación a producción de los componentes técnicos del PMV para disposición y uso de los usuarios o clientes finales.
- Asegurar la aprobación y entrega final del PMV con la calidad esperada.
- Primera fase para atención de incidentes y *feedback* del cliente.
- Estabilizar y operar el ambiente de producción. Verificar que cubre con los requerimientos necesarios para garantizar que la solución funcione bien desde una perspectiva funcional y no funcional.
- Capacidad para corregir un lanzamiento implementado o solucionar un problema de seguridad en cualquier momento.

Entre los artefactos que se pueden considerar durante esta fase están:

- A. El aplicativo o PMV en operación. Paquete de software implementado en ambiente de producción aprobado por el Product Owner.
- B. Diagramas de Estados/Actividad. Descripción de cómo el PMV implementa su funcionalidad y facilita el entendimiento de las transiciones entre diversos objetos para desplegar fácilmente los flujos clave de negocio.
- C. Manual de Usuario. Documento de comunicación técnica que explica la funcionalidad, interacción y operación de la aplicación móvil final.
- D. Memoria técnica de implementación. Describe el proceso de instrumentación y las especificaciones técnicas mínimas de hardware y software que se requiere para el funcionamiento adecuado del aplicativo desarrollado.

4. MONITOREAR Y MEJORAR JUNTOS. Posterior a la entrega del paquete incremental probado, el equipo ágil se preparará para recibir los comentarios por parte del cliente o dueño del producto. Se enlistan las observaciones o ajustes al producto y se revisan nuevos requerimientos del cliente y se integran a un plan de mejora.

También se analiza la retrospectiva del equipo del ciclo incremental ejecutado. Todos los miembros del E²A revisan sus prácticas, identifican áreas de oportunidad e indagan nuevas formas para mejorar. Se busca que el equipo esté constantemente mejorando la ejecución del siguiente plan incremental.

Con todos los puntos analizados se concentra una lista priorizada para el subsecuente plan incremental considerando la lista con peticiones clave y de mejora del cliente, así como las acciones correctivas y/o mejoras a las prácticas que el equipo acordó realizar. El plan con las mejoras se ejecuta en la siguiente iteración desde una perspectiva general, tanto en los equipo ágiles como en el producto final.

El diseño de un plan de mejora continua (Ver figura 46) enfocado al producto final e impulsado por la alta dirección de la organización permitirá implementar constantemente mejoras continuas de funcionalidades, performance y/o experiencia del cliente a fin de desarrollar productos más competitivos y que respondan mejor a las exigencias del cliente. Los elementos del plan pueden basarse en los elementos descritos a continuación:

Figura 46. **Plan de Mejora Continua.**



- 1) Indicadores clave que permitan analizar y medir la funcionalidad, uso y performance de la aplicación para enfocar nuevas versiones y funcionalidades que generen mayor valor al cliente. Algunos indicadores clave se enlistan a continuación
- Usuarios activos (usuarios activos mensuales y diarios para su segmentación).
 - Nuevos usuarios (por campaña o promoción).
 - Uso de la aplicación (flujo de la app).
 - Duración de la sesión (período de tiempo que transcurre desde que el usuario abre la sesión hasta que la cierra).
 - Retención (el porcentaje de usuarios que vuelven a la app luego del primer uso).
 - Registro de eventos (acciones que el usuario realiza desde la app).
 - Tasa de abandono (porcentaje de usuarios que dejan de utilizar la app dentro de un periodo determinado respecto al anterior)

- h) Descargas de la app (conocer la fuente de tráfico que desencadena la instalación para direccionar nuevas campañas de marketing).
 - i) Tiempos de respuesta de peticiones (tiempo que un usuario espera al utilizar la app).
 - j) Tasa de desinstalación (indica cuántos usuarios interrumpen el uso de la aplicación por posibles defectos, problemas o desinterés en la app).
- 2) *Feedback* continuo del cliente. A través de encuestas móviles, campañas por email, mensajes SMS u otras herramientas se puede monitorear la experiencia del cliente con la aplicación móvil.
 - 3) Roles y responsabilidades claras y precisas. Los equipos ágiles deben asumir el compromiso de proporcionar atención y soporte desde la definición, la planificación hasta la operación del producto para garantizar el flujo del servicio y de los procesos que aseguren su entrega.
 - 4) Herramientas que permitan la visibilidad y trazabilidad del desarrollo de las aplicaciones móviles. Contar con aplicativos que permitan estandarizar y automatizar el flujo del ciclo de desarrollo de aplicaciones que impulsen la entrega continua.
 - 5) Base de conocimiento de mejora continua. Integrar, analizar y explotar la información con el *feedback* del cliente inmediatamente después de los lanzamientos o liberaciones a producción y compartir el conocimiento con los integrantes del equipo, especialmente con los roles de desarrollo para la implementación ágil de mejoras o innovaciones al producto.

Para lograr la mejora continua, los indicadores clave fungen un papel vital durante esta fase, ya que lo que no se puede medir no se puede mejorar⁵⁹. En el apartado de Indicadores Clave de Rendimiento se describe a detalle una propuesta para implementarlos de manera exitosa.

Las actividades que involucran el monitoreo y mejora continua se describen a continuación:

- Asegurar la disponibilidad de las aplicaciones a través de ambientes estables y de rápida respuesta.
- Optimizar el comportamiento de las aplicaciones poniendo énfasis en los resultados de las pruebas de performance y certificación.
- Elaborar plan de mejora continua y ejecutarlo.
- Generar bitácoras con las acciones permitidas por usuarios o grupos de usuarios en caso de incidentes entre pase de ambientes o detección de defectos.
- Mantener un enfoque transversal de monitoreo para compartir nuevos conocimientos con los integrantes de los equipos ágiles e identificar áreas de oportunidad y probar nuevas formas para mejorar.

3.2.4 Tubería Automatizada de Entrega Continua (TAG)

La tubería automatizada de entrega continua, pieza clave del “*Framework DevOps for Mobile Banking*” y de la estrategia DevOps, representa los flujos de trabajo, las actividades y la automatización necesarios para guiar los paquetes de software de banca móvil desde la planeación hasta la liberación de valor para el usuario o cliente final.

⁵⁹ William Thomson Kelvin , físico y matemático británico (1824 – 1907).

TAG abarca la automatización de la integración, despliegue y entrega continua de los PMV de banca móvil según lo requiera el negocio. Permite desarrollar la capacidad de ofrecer nuevas funciones al cliente con mucha más frecuencia que con los procesos actuales.

La entrega continua puede tener distintas frecuencias para los lanzamientos. Según lo requiera el entorno de negocio y mercado pueden ser liberaciones diarias o múltiples lanzamientos por día, semanales o mensuales, entre otras periodicidades. En suma, TAG permite efectuar entregas del aplicativo, en su totalidad o en partes, de manera rápida, automatizada, eficiente y continua.

Según el contexto tecnológico de la organización en cuanto a escalabilidad, adaptabilidad y disponibilidad de los servicios de TI, se plantea el uso de la herramienta **CA Harvest Software Change Manager** (CA Harvest SCM)⁶⁰ para la instrumentación de la tubería automatizada, la cual contempla capacidades orientadas por procesos para la administración de los equipos de desarrollo en toda la organización, abarcando múltiples plataformas y herramientas de administración de versiones. También está enfocada en simplificar el proceso de administración de la versión de software, así como el desarrollo y mantenimiento de las aplicaciones de negocio con flujos de trabajo optimizados y control automatizado de versiones de código fuente.

Adicionalmente incluye el módulo **CA Continuous Delivery Director** que permite facilitar la planificación, organización y optimización de la entrega continua para todos los equipos, específicamente en los siguientes rubros:

⁶⁰ <https://www.ca.com/ar/products/ca-harvest-software-change-manager.html>

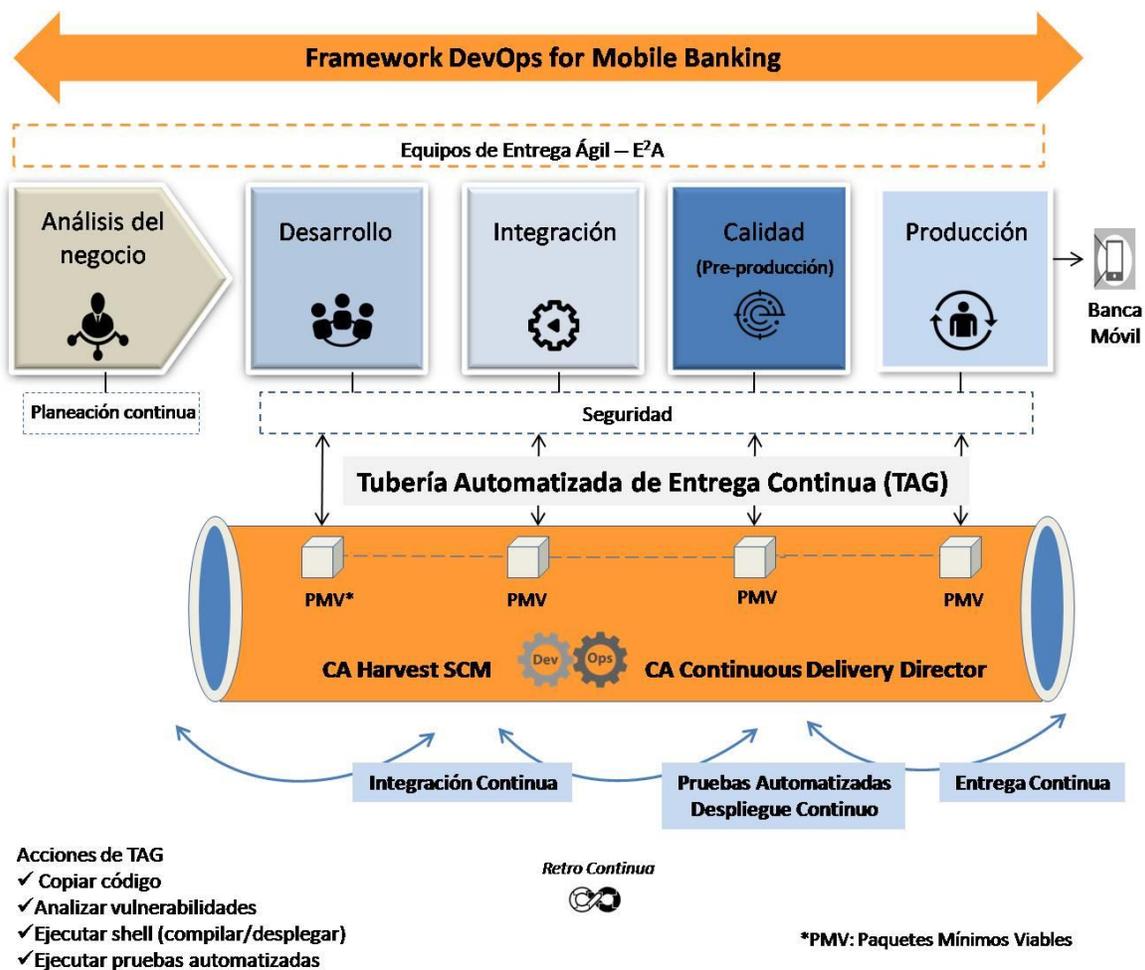
- 1) Implementaciones continuas confiables. Permite organizar y administrar el proceso para generar un círculo virtuoso de mejoras.
- 2) Coordinación avanzada de las pruebas. Facilita la integración de la calidad en las aplicaciones desde el principio y ofrece la mejor experiencia al cliente de manera continua.
- 3) Mejora continua. Aplica análisis para mejorar los procesos, el tiempo de resolución, la productividad y la calidad.

Para la instrumentación de TAG con la herramienta Harvest se pueden considerar las siguientes actividades:

- Instalar en cada ambiente de trabajo (desarrollo integración, calidad y producción), un agente Harvest y definir las rutas de trabajo para la construcción de la tubería.
- Configurar la herramienta para ejecutar una secuencia de comandos en los servidores de los ambientes de trabajo para llevar a cabo las acciones de automatización TAG:
 - Copiado de código.
 - Análisis de Vulnerabilidades.
 - Ejecución de shell (compilación y despliegue)
 - Ejecución de pruebas automatizadas (unitarias, integrales, de performance y certificación)
- Habilitación de esquemas de *rollback* de versiones estables a través de la ejecución automatizada de shells en caso de versiones con errores o defectos.
- Habilitar la interfaz gráfica de Harvest para que los integrantes de los E²A ejecuten de manera ágil y automatizada en cada ambiente de trabajo las acciones de la tubería para los PMV, incluyendo el versionamiento automatizado por la herramienta.

- Configuración de bitácoras y log de los resultados de las acciones ejecutadas en los entornos de trabajo.
- Ejecución de pruebas de transporte y entrega continua de paquetes de software móvil sin funcionalidad a través de la tubería para garantizar su correcto funcionamiento.
- Definir usuarios o grupos de usuarios para facilitar las aprobaciones para los pases entre ambiente que agilice la gestión de despliegues y lanzamientos de los paquetes de software.

Figura 47. Tubería Automatizada de Entrega Continua (TAG) con CA Harvest SCM.



De acuerdo al diagrama de la figura 47, una vez que se logra la definición del producto de banca móvil con la participación activa del área de negocio, se priorizan las necesidades o funcionalidades y se elabora el plan de trabajo dividido en iteraciones para la entrega incremental del valor al cliente, los E²A inician el desarrollo de los PMV.

Cuando los equipos cuentan con paquetes de software listos para ser integrados y desplegados entre ambientes se conectan a la interfaz gráfica de Harvest, identifican el ambiente y el paquete de software a desplegar y dan la instrucción para ejecutar acciones de automatización TAG minimizando en gran medida actividades manuales y rutinarias propensas a constantes errores humanos.

Harvest enruta los PMV al servidor del ambiente correspondiente ejecutando las acciones automatizadas de la tubería en segundos o minutos (copiar código, descomprimir, compilar, desplegar) además de ejecutar las pruebas automatizadas reduciendo potencialmente la pérdida de tiempo resultante de la ejecución de pruebas manuales.

La herramienta envía las notificaciones a los usuarios correspondientes para la autorización de los despliegues o liberaciones, y de esta manera agiliza el proceso de integración, prueba, despliegue y entrega ágil al cliente impulsando la mejora constante de funcionalidades, performance y experiencia del cliente.

En caso de detección de defectos o errores en los PM, Harvest mantiene estables y sincronizados los ambientes de trabajo para la operación a través de sus capacidades de versionamiento automatizado y regresión de versiones previas. La herramienta evita lanzamientos de PMV con defectos garantizando la *calidad* de los productos.

Conforme la organización se aproxima a la implementación de la tubería de automatización, los equipos ágiles tendrán una lista sólida de oportunidades de mejora:

- Entregar o liberar código de forma inmediata, sin esperar a acumular muchas funcionalidades, incidencias o desarrollos independientes.
- Mayor estabilidad en los entornos de integración, calidad y producción.
- Incorporación ágil de nuevas funciones o necesidades del cliente.
- Mayor visibilidad y trazabilidad del código que se está desarrollando.
- Identificación de errores en etapas tempranas y atención rápida.
- El tiempo de Integración, Despliegue y Entrega Continua se reduce minutos o segundos acortando el tiempo de comercialización.

3.2.5 Indicadores clave de rendimiento

“*Framework DevOps for Mobile Banking*” implica cambios sustantivos en la organización y forma de operación de los equipos de trabajo, así como la implementación de herramientas para la automatización de procesos vinculados con el desarrollo de aplicaciones móviles. Contar con indicadores que permitan dar seguimiento puntual al resultado de la transformación es un componente clave para evaluar el desempeño de los equipos, el rendimiento de la tubería automatizada de entrega continua y los resultados de la estandarización de procesos que impactan el desarrollo de aplicaciones de banca móvil.

Un plan de indicadores clave permitirá medir y monitorear detalladamente la evolución de la automatización del ciclo de integración, despliegue y entrega continua de aplicaciones en

entornos DevOps, la satisfacción o experiencia del cliente, así como el rendimiento de las aplicaciones a fin de validar la efectividad del marco de trabajo en cuestión.

Considerando como base la estrategia DevOps, se propone el siguiente plan indicadores de rendimiento y desempeño (Pfeiffer, 2017) para efectos del “*Framework DevOps for Mobile Banking*” cuya instrumentación dependerá de las características y entorno del negocio.

Indicadores clave de rendimiento:

- A. **Frecuencia de implementación.** Permite estimar la frecuencia de lanzamientos o liberaciones con nuevas funcionalidades o mejoras del producto. Cuando se observa una tendencia estable o ascendente en el tiempo apunta a un proceso ágil de entrega y mejora continua. En cambio, una disminución podría significar cuellos de botella dentro del flujo de trabajo existente.
- B. **Tiempo de entrega.** Estima el tiempo que implica resolver incidentes, integrar nuevas funcionalidades o características, desde el inicio hasta la entrega del producto para su uso final. El monitoreo se enfoca en observar los plazos de entrega, si son largos pueden sugerir retrasos en ciertos flujos del ciclo de desarrollo e implementación.
- C. **Tasa de error.** Este indicador refiere no solo problemas de calidad, sino también incidentes relacionados con el rendimiento continuo y el tiempo de disponibilidad del producto final. Cuando la tendencia del indicador es ascendente sugiere revisar la estrategia de pruebas y de puesta en operación para asegurar la calidad del código y sus dependencias antes del lanzamiento.

- D. **Tasa de implementación fallida.** Este indicador determina con qué frecuencia las implementaciones provocan interrupciones o incidentes que afectan la entrega continua del producto. Se busca que el indicador sea lo más bajo posible. Los fallos por implementaciones a veces están vinculados con la frecuencia de despliegue por lo que se sugiere revisar los flujos vinculados con dicho proceso.
- E. **Tiempo de detección.** Es esencial detectar los fallos rápidamente si se producen y buscar minimizarlos o erradicarlos. Este indicador permite monitorear el tiempo para detectarlos, a mayor rango de detección, los esfuerzos de respuesta pueden no ser oportunos e interrumpir el flujo de la operación.
- F. **Tiempo medio de recuperación.** Después de la detección de fallas o defectos en los despliegues, este indicador permite monitorear el tiempo de atención y solución después de la detección de fallas en los despliegues. Indica la capacidad de responder adecuadamente a los incidentes identificados por lo que la detección rápida debe ir acompañada de un tiempo de recuperación igualmente ágil.
- G. **Porcentaje de éxito en tests automáticos.** La automatización de los casos de prueba optimiza y reduce los tiempos de entrega, no obstante, si la tendencia de este indicador es a la baja, es importante la revisión de la construcción de pruebas y la implementación de la automatización para que no afecte los despliegues. Ver diagrama de la figura 48.

Indicadores clave de desempeño:

- A. **Priorización de funciones.** Es un indicador enfocado a monitorear el uso de funcionalidades o características por parte del usuario. Si el indicador es bajo o está

disminuyendo es recomendable evaluar el ciclo de retroalimentación de los usuarios para la repriorización de las funciones o identificación de mejoras con mayor valor.

- B. **Disponibilidad del servicio.** Permite medir el tiempo que las aplicaciones están disponibles y funcionan correctamente. Se busca que el indicador mantenga tiempos de disponibilidad de casi el 100%. En caso contrario, es necesario revisar los planes de infraestructura, soporte y servicios para asegurar el servicio.
- C. **Rendimiento de la aplicación.** Este indicador permite dar seguimiento a los incidentes vinculados con el rendimiento de las aplicaciones y mejora continua. Mantenerlo al mínimo permitirá una alta disponibilidad y una buena experiencia para el usuario final. En caso de identificar incrementos replantear la estrategia de optimización de rendimiento de la aplicación, así como de los servidores o contenedores y validar que las herramientas de monitoreo son las adecuadas.
- D. **Tickets de atención.** Los tickets de soporte al cliente y los comentarios son indicadores que sugieren que la calidad de la aplicación y/o la satisfacción del cliente no se logró como se planeaba. Cuando el indicador va en incremento es importante revisar el ciclo completo de desarrollo de la aplicación, en cuanto a funcionalidades, test, despliegue y liberación para identificar defectos o inconsistencias y asegurar la entrega continua con la calidad esperada.
- E. **Tiempo medio de atención de tickets.** Es un indicador vinculado con el tiempo de respuesta de los equipos para resolver incidentes reportados por los usuarios sin afectar el flujo de la operación. Cuando el indicador se dispara se pueden probar los flujos de trabajo de los equipos, así como buscar mejoras en los procesos que afecten el ciclo de desarrollo y lanzamiento de aplicaciones. Ver diagrama de la figura 48.

Figura 48. Plan de indicadores clave de desempeño y de rendimiento en entornos de trabajo

DevOps.



El seguimiento cuidadoso de los indicadores planteados puede garantizar mayor eficiencia en el ciclo de desarrollo, integración y entrega continua de los productos financieros móviles e impulsar una mejor experiencia del usuario final. Además de servir como guía o brújula para hacer los ajustes necesarios en la implementación del ambiente de trabajo del presente protocolo en caso de identificar complicaciones o fallas en la evolución hacia entornos de trabajo más colaborativos, veloces y eficientes en función de la entrega y mejora continua de valor al cliente.

3.2.6 Gestión del Cambio

Al principio puede resultar desalentador promover que los equipos de trabajo de TI en una organización transiten hacia nuevas formas de interacción, métodos y/o herramientas de trabajo, influenciados por prácticas y principios DevOps, ya que el cambio implica salir de la zona de confort, aprender nuevas formas de organización, desapegarse de viejas rutinas y adoptar otras formas de pensamiento. Sin embargo, una estrategia clara y enfocada en los objetivos de TI y de negocio puede coadyuvar al cambio del paradigma organizacional de manera continua y eficiente.

Cuando se inicia un cambio, los equipos de trabajo pueden pasar por una fase de resistencia y caos que les hace pensar que, en la situación inicial, estaban mejor. Es importante que el equipo líder del cambio esté consciente de ello y tenga la convicción que la evolución hacia contextos diferentes le permitirá instaurar mejores ambientes de trabajo en beneficio de todos los integrantes y de la organización.

“Framework DevOps for Mobile Banking” busca impulsar un cambio estratégico en los ambientes de trabajo encaminado hacia estructuras de trabajo colaborativas, flexibles y ágiles

para que los equipos desarrollen capacidades más competitivas, centradas en el cliente, la agilidad, la calidad y la mejora continua de los productos financieros móviles.

En el marco de trabajo se plantea que la Gestión del Cambio (GC) puede incidir la manera en que se comunican e interactúan los equipos de trabajo de TI haciendo énfasis en lograr estructuras organizacionales más transparentes, comunicadas y eficientes. La GC puede direccionar a los equipos clave que impacta el desarrollo y entrega de aplicaciones móviles para que asuman un rol más consciente de las dependencias y necesidades de los otros.

Sin perder de vista la posibilidad de flexibilizar y adaptar cualquier nueva forma de trabajo según lo requiera el contexto, la GC enfatiza que para lograr un verdadero cambio cultural en torno a las directrices DevOps es importante considerar tanto la perspectiva del grupo que lidere la transformación como las demás áreas que participen en el cambio. Se busca con oportunidad darle la relevancia a cada uno de los equipos implicados y enfatizar la apertura y transparencia entre ellos.

Una vez iniciada la transición, se propone la adopción del Modelo Kotter (Duque, 2014) para gestionar y asegurar que la transición impulse los objetivos que se persiguen a través de “*Framework DevOps for Mobile Banking*” en la organización. En la figura 49 se esquematizan los elementos a considerar en dicho modelo, mismos que se pueden adoptar según lo requiera el contexto tecnológico y de negocio.

Figura 49. Gestión de Cambio basado en Modelo Kotter (Duque, 2014).



En el punto número 1 que refiere “Establecer sentido de la urgencia” es importante plantear una justificación real y precisar para llevar a cabo un cambio en el entorno de la organización. En este sentido, la **Matriz de factores que limitan la entrega ágil de aplicaciones móviles** desarrollada en el cuarto capítulo del presente trabajo de investigación permite identificar claramente cuáles son los aspectos que afectan el proceso de integración, despliegue y entrega ágil de aplicaciones móviles restando competitividad y valor a las áreas de TI.

El análisis de la matriz por área de trabajo permite identificar cuáles son los equipos con mayor impacto en el proceso así como los más afectados con sobrecargas de trabajo, entre otros aspectos. Esto permitirá difundir un “sentido de urgencia” explicando las ventajas organizacionales de instaurar ambientes de trabajo más equilibrados y alineados.

En el punto 2 se menciona la necesidad de crear una coalición con los equipos para liderar el cambio. Existen diferentes estrategias para introducir la transformación en una organización, puede iniciar desde la alta dirección, con los equipos actuales o creando nuevos grupos de trabajo. Para efectos del “*Framework DevOps for Mobile Banking*” se plantea crear un **grupo semillero en DevOps** que fomente el cambio de mejores principios y prácticas con perfiles multifuncionales para crear y entregar valor de manera automática e independiente. Se considera contar con DevOps Master que acompañe al equipo durante la transición y asegure que se cumplan los principios y prácticas, actuando como un facilitador y solucionador de problemas. Posteriormente replicar las prácticas en nuevos ambientes de trabajo con los equipos restantes.

En el punto 3 se plantea la importancia de crear una nueva visión alineada con los nuevos objetivos establecidos haciendo partícipe a los equipos de trabajo y resaltando la importancia de fomentar el desempeño de alto rendimiento a través de:

- Una **visión y objetivos compartidos** conocidos, interiorizados y aceptados por todos los integrantes del equipo a fin de lograr una visión verdaderamente compartida.
- Un liderazgo claro, reconocido y asumido por los integrantes de las áreas de trabajo, en donde las buenas prácticas y los principios son constantemente replicados por el líder.
- Retrospectivas continuas que permitan identificar a través de la experiencia de todos los integrantes del equipo y los indicadores clave las áreas de oportunidad o mejora para incentivar el rendimiento de los equipos.

- Cultura de Transparencia para fomentar espacios donde el conocimiento y las ideas se comparten abiertamente y en todas direcciones que motiven la proactividad y la mejora.
- Indagar en las diferencias fomentando ambientes tolerantes y flexibles para recibir sugerencias y opiniones de manera respetuosa y abierta.
- Mejora de la comunicación identificando brechas de conocimiento entre los integrantes del equipo, impedimentos para realizar las actividades, necesidades específicas, etc. Se busca mantener un diálogo fluido y efectivo basado en la colaboración, acompañamiento y apoyo profesional.

Una vez que se tiene la definición de dichos elementos y su difusión oficial adicionalmente se plantea su divulgación a través de Lienzos Canvas o Tableros Kanban con el objetivo de maximizar la propagación y entendimiento en la organización.

En el siguiente punto “Asegurar los resultados a corto plazo” se trata de crear metas a corto plazo y no sólo un objetivo a largo plazo. Se busca generar valor en el mínimo tiempo posible, es decir, garantizar que los equipos de trabajo colaboren alineados y enfocados en la construcción de productos móviles de manera ágil y continua minimizando el ciclo de desarrollo a través de una estrategia de entrega definiendo e implementando pequeñas o mínimos incrementos de funcionalidad que aseguren la entrega continua de valor al cliente. También se hace hincapié dentro del equipo que entre más grandes sean los incrementos a desarrollar, la complejidad crece e impacta en el ciclo de entrega por lo que el tamaño de las liberaciones condiciona la capacidad de entrega continua.

Durante esta fase, los equipos pueden estar receptivos al continuo cambio y tener presente que la planeación a corto plazo puede tener ajustes, por lo que deberán responder lo más rápido posible pero controlando los cambios según lo requiera el contexto.

En el punto 5 referente a la “Consolidación de las mejoras” implica un paso muy importante en la gestión del cambio en una organización. Frecuentemente se tiene la falsa percepción, en un inicio, que al alcanzar los resultados esperados el cambio se ha logrado lo que puede generar que el equipo rápidamente regrese a sus antiguas prácticas o métodos de trabajo.

Para lograr la consolidación de las mejoras en la organización es importante que no se disminuyan los esfuerzos del despliegue inicial, a medida que se alcanzan éxitos en el proceso de cambio se debe seguir impulsado a los equipos para que permanentemente se enfoquen en los objetivos y utilicen las prácticas de mejora para lograr los resultados en lo subsecuente, evitando la relajación y/o la desmotivación por el cambio entre los integrantes.

Fomentar un clima de crecimiento del cambio entre los equipos a través de un proceso de evolución continua permitirá generar valor efectivo al cliente asimilando y adoptando las nuevas prácticas de manera natural y constante.

En el punto 6 referente a “Institucionalizar los nuevos métodos” se busca institucionalizar y consolidar los logros obtenidos a fin de que la evolución forme parte de la cultura de la organización y garantice su permanencia en el tiempo. Se logra normalizar las mejoras en los equipos de trabajo sosteniendo las nuevas formas de comportamiento e integración de los miembros del equipo aplicando las nuevas prácticas. El objetivo de esta etapa es que día a día se mantenga el cambio de enfoque y de las prácticas hasta naturalizarlas a través de medios físicos o digitales en donde se institucionalicen y formalicen los nuevos valores,

objetivos y métodos de la organización y se acepten y ejecuten de manera natural por parte de todos los integrantes del equipo.

Parte fundamental de dicha normalización es que al crear nuevas células o equipos de trabajo se aplique un liderazgo que gestione el cambio basado en la nueva cultura organizacional para que se dé continuidad al proceso de transformación.

También se trata de divulgar los éxitos del equipo, fomentar retrospectivas positivas y reconocimientos directos entre los integrantes del equipo, así como resaltar la colaboración individual y mostrar reconocimiento público a las personas claves en el proceso de cambio. Se espera que el equipo al asimilar y aceptar la nueva dinámica irá mejorando y evolucionando constantemente.

IV. Conclusiones

Los enfoques tradicionales para el desarrollo y la entrega de software ya no son suficientes en contextos de cambio continuo. Las organizaciones buscan reducir cada vez más los procesos manuales ya que inducen a error, se cortan y crean desperdicios y demoras en la entrega de resultados. Se requieren nuevas prácticas que lleven a la entrega ágil de software de calidad y que satisfaga las necesidades del cliente de manera continua.

En este escenario, DevOps resulta un enfoque de pensamiento poderoso que permite automatizar los procesos relacionados con la integración, el despliegue y la entrega continua de software asegurando la transparencia y trazabilidad en todo momento y alineando a los equipos de trabajo e impulsando la mejora continua.

En el nuevo paradigma se pone de manifiesto la importancia de integrar y alinear a las personas, las herramientas y los procesos para crear ambientes de desarrollo ágiles y flexibles que permitan la generación continua de valor. En entornos DevOps las herramientas de automatización permiten implementar ciclos ágiles de despliegue con mayor frecuencia y eficiencia.

En entornos de banca móvil en donde la entrega ágil al cliente con la calidad, servicios y funcionalidades esperadas, las prácticas basadas en DevOps pueden ser clave para darle continuidad y agilidad al negocio.

En el caso de Bancatec, después de aplicar los instrumentos de medición y analizar la información correspondiente, se confirmó la hipótesis planteada en el presente protocolo “Los factores que impiden la entrega ágil y continua de aplicaciones de banca móvil en Bancatec pueden estar relacionados con la falta de alineación entre los equipos y la ausencia de automatización en los flujos de trabajo operativos y repetitivos” ya que se identificó que entre los factores con mayor impacto negativo en el proceso de desarrollo y despliegue de aplicación móviles están: los procesos manuales y repetitivos, la falta de claridad en objetivos comunes, la presencia de silos entre desarrollo, infraestructura y seguridad, la sobrecarga de actividades en áreas operativas y la ausencia de una cultura organizacional integradora y colaborativa.

Asimismo y de acuerdo con los objetivos planteados en la presente investigación, se logró integrar una matriz de factores que impiden la entrega ágil y continua de aplicaciones de banca móvil en Bancatec descrita en la Tabla 19 y en la cual se indican por orden de relevancia: Procesos manuales y repetitivos que restan agilidad al negocio, Falta de claridad

en objetivos comunes, Deficiente comunicación entre equipos de trabajo, Silos entre desarrollo, infraestructura y seguridad, Sobrecarga de actividades en áreas operativas, Ausencia de una cultura organizacional integradora y colaborativa, Claridad en los tramos de responsabilidad y Generación de desperdicios y demoras en despliegue de paquetes

Con los resultados obtenidos y como parte de los objetivos del presente protocolo de investigación se plantea un marco de trabajo denominado “*Framework DevOps for Mobile Banking*” con una serie de metas, principios y componentes enfocados a la alineación y transformación de equipos de trabajo, procesos de desarrollo y herramientas de automatización para agilizar la entrega, impulsar la calidad y la mejora continua de aplicaciones de banca móvil.

Resultado del análisis de las prácticas y principios ágiles en torno a DevOps, se propone que las bases de dicho marco de trabajo se enfoquen en la generación de valor al cliente, simplificar y agilizar el proceso de desarrollo y entrega continua del producto, la reducción del tiempo de comercialización, las implementaciones con mayor frecuencia y eficiencia a través de la automatización de ambientes e impulsar la mejora continua.

Los Equipos de Entrega Ágil (E²A) son uno de los componentes fundamentales del “*Framework DevOps for Mobile Banking*” para transitar hacia ambientes automatizados y eficientes. En las organizaciones, las personas son clave para lograr y sostener la evolución hacia lo ágil, por lo que es importante conocer las fases necesarias para lograr el cambio organizacional y asegurar la transformación esperada. La integración y adiestramiento de los E²A implican al menos cuatro fases, la formación o capacitación, el conflicto entre los integrantes debido al cambio, la normalización o aceptación de las nuevas prácticas ágiles y

como resultado final, la optimización del desempeño del equipo en ambientes más colaborativos y eficientes.

Una vez que se logran alinear y autogestionar a los (E²A) en entornos ágiles y colaborativos la implementación y mantenimiento de la Tubería Automatizada de Entrega Continua (TAG) viene por defecto, la aceptación y el uso por parte de los equipos de trabajo suele ser fácil y de manera continua.

Finalmente, “*Framework DevOps for Mobile Banking*” puede impulsar un cambio estratégico en los ambientes de trabajo encaminado hacia estructuras de trabajo colaborativas, flexibles y ágiles para que los equipos desarrollen capacidades más competitivas, centradas en el cliente, la agilidad, la calidad y la mejora continua de los productos financieros móviles.

Apéndices

Apéndice A: Cuestionario Diagnóstico sobre despliegue y entrega continua de aplicaciones de Banca Móvil

Cuestionario Diagnóstico sobre despliegue y entrega continua de aplicaciones de Banca Móvil

Objetivo del instrumento: identificar los principales factores que afectan el proceso de integración, despliegue y entrega ágil de aplicaciones móviles limitando la mejora continua así como la interacción entre los equipo de trabajo.
La importancia de tu participación es crucial para fomentar entornos de trabajo colaborativos y dinámicos que agilicen el proceso de despliegue y entrega continua de aplicaciones de banca móvil.
Tiempo estimado de respuesta: 3 minutos por ítem.

*** Required**

1. Marca el área a la que perteneces *

- Desarrollo de sistemas
- Integración
- Pruebas
- Seguridad de la información
- Infraestructura aplicativa

2. Los objetivos y principios de TI en tu área están alineados con los equipos que intervienen en el proceso de integración, despliegue y entrega de software móvil : *

- Muy frecuentemente
- Frecuentemente
- Ocasionalmente
- Raramente
- Nunca

3. ¿Cómo consideras la comunicación/retroalimentación entre las áreas de Desarrollo, Pruebas, Seguridad e Infraestructura cuando requieres de una promoción de ambiente de nuevas versiones de software móvil? *

- Excelente
- Buena
- Moderadamente buena
- Mala
- Muy mala

4. La colaboración entre los roles de Desarrollo, Pruebas, Seguridad e Infraestructura para la integración, despliegue y entrega de nuevos paquetes de software móvil es: *

- Excelente
- Buena
- Moderadamente buena
- Mala
- Muy mala

5. La definición de roles y responsabilidades entre Desarrollo, Pruebas, Seguridad e Infraestructura en el ciclo de integración, despliegue y entrega de software móvil es: *

- Muy clara
- Clara
- Moderadamente clara
- Confusa
- Muy confusa

6. El tiempo de respuesta por parte de los equipos de infraestructura y seguridad para la promoción entre ambientes de nuevas versiones de software móvil es: *

- Menor a 1 día
- Entre 1 y 3 días
- Entre 4 y 7 días
- Más de una semana
- Más de 2 semanas

7. Se logra acelerar el desarrollo de versiones de software móvil pero la puesta en producción no sigue el mismo ritmo (Solo para Desarrollo): *

- Muy frecuentemente
- Frecuentemente
- Ocasionalmente
- Raramente
- Nunca

8. Se presentan incidencias vinculadas con la calidad del software de banca móvil que se despliega, generando trabajo adicional a los equipos de infraestructura y cuellos de botella. (Solo para Infraestructura) *

- Muy frecuentemente
- Frecuentemente
- Ocasionalmente
- Raramente
- Nunca

9. El equipo de trabajo de Seguridad interactúa oportunamente a lo largo del proceso de integración, despliegue y entrega de paquetes de software móvil (Solo para Seguridad): *

- Muy frecuentemente
- Frecuentemente
- Ocasionalmente
- Raramente
- Nunca

10. En etapas tempranas (desarrollo, integración y pruebas) es posible detectar errores/incidencias del software móvil y dar atención rápida a ellos: *

- Muy frecuentemente
- Frecuentemente
- Ocasionalmente
- Raramente
- Nunca

11. ¿Cuentas con herramientas que automaticen la promoción entre ambientes (versionamiento/copiado de código, compilación y deploy) para nuevas versiones de software móvil? *

- Muy frecuentemente
- Frecuentemente
- Ocasionalmente
- Raramente
- Nunca

12. ¿En tu ambiente de desarrollo u operación, cuentas con herramientas que automaticen las pruebas de nuevas versiones de software por desplegar? *

- Muy frecuentemente
- Frecuentemente
- Ocasionalmente
- Raramente
- Nunca

13. La relevancia de tus actividades en el proceso de integración, despliegue y entrega de software móvil para impulsar la mejora continua es: *

- Muy importante
- Importante
- Moderadamente importante
- De poca importancia
- Sin importancia

14. ¿Te sientes motivada(o) al realizar las actividades vinculadas con tu rol? *

- Muy frecuentemente
- Frecuentemente
- Ocasionalmente
- Raramente
- Nunca

Muchas gracias por tu valiosa participación. Clausula de confidencialidad: los datos recabados son de carácter confidencial estrictamente para fines del protocolo de estudio "DEVOPS COMO MARCO DE TRABAJO PARA LA MEJORA CONTINUA DE SOFTWARE PARA BANCA MÓVIL".

SUBMIT

Apéndice B: Entrevista Diagnóstica sobre despliegue y entrega continua de aplicaciones de Banca Móvil

Entrevista Diagnóstica sobre despliegue y entrega continua de aplicaciones de Banca Móvil

Departamento de Servicios de Información Tecnológica adscrita a la Dirección General de Desarrollo y Pruebas.
Entrevista dirigida a los COORDINADORES de Desarrollo de Sistemas, Pruebas, Seguridad e Infraestructura.

Objetivo del instrumento: identificar el flujo de actividades y procesos que implican la integración, despliegue y entrega continua de aplicaciones móviles y la interacción entre los equipos de trabajo.

La importancia de tu participación es crucial para fomentar entornos de trabajo colaborativos y dinámicos que agilicen el proceso de despliegue y entrega continua de aplicaciones de banca móvil.

* Required

Indica tu área de gestión:

- Desarrollo de Sistemas
- Pruebas
- Seguridad de la Información
- Infraestructura aplicativa

Para Desarrollo de Sistemas:

¿Qué tipo de metodología emplean para el análisis y construcción de aplicaciones de banca móvil? ¿Han probado métodos ágiles? Describe. *

Your answer

Para Infraestructura:

¿Se busca disminuir la ejecución de cambios o atención de incidencias para mantener la estabilidad y confiabilidad de las plataformas? Si/No. Explica. *

Your answer

Para todas las áreas:

¿Cuál es el proceso a seguir para promocionar nuevas versiones/control de cambios de software de bsnavs móvil entre los ambientes? *

Your answer

¿Consideras que el proceso es ágil? Si/No. Explica. *

Your answer

¿Cuáles son los limitantes que enfrentan para hacer más ágil el proceso?

Your answer

¿Consideras que tu equipo de trabajo está integrado y colabora apuntando hacia los mismos objetivos? Si/No. Explica.

Your answer

¿Cómo es la comunicación/retroalimentación con las áreas que participan en el proceso de integración, despliegue y entrega de software de banca móvil? Describe.

Your answer

Muchas gracias por tu valiosa participación. Clausula de confidencialidad: los datos recabados son de carácter

Fuentes consultadas

- Alonso, J., de Lis, S. F., Hoyo, C., Lopez-Moctezuma, C., y Tuesta, D. (2013). *La banca móvil en México como mecanismo de inclusión financiera. Desarrollos recientes y aproximación al mercado potencial* (No. 1319). BVA Research.
- Arias Chaves, M. (2005). La ingeniería de requerimientos y su importancia en el desarrollo de proyectos de software. *InterSedes: Revista de las Sedes Regionales*, VI (10), 1-13.
- Avendaño, C. O. (2018). Los retos de la banca digital en México. *Revista IUS*, 12(41), 87-108. Recuperado en 10 de septiembre de 2018, de http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S1870-21472018000100087&lng=es&tlng=es.
- Aveson, D., y Fitzgerald, G. (2006). Methodologies for developing information systems: A historical perspective. In *The Past and Future of Information Systems: 1976–2006 and Beyond* (pp. 27-38). Springer, Boston, MA.
- BBVA. (2013). Análisis Económico. Dirección electrónica https://www.bbvaresearch.com/wp-content/uploads/migrados/WP_1319_Mexico_BancaMovil_tcm346-390713.pdf
- Becerra, P., y Sanjuan, M. (2017). Revisión de estado del arte del ciclo de vida de desarrollo de software seguro con la metodología SCRUM. *Revista Investigación y Desarrollo en TIC*, 5(2).
- Belalcázar, A., Díaz, F. J., Molinari, L., & Rodríguez, C. A. (2016). Principios, Roles y Métricas en alineamiento estratégico de nuevos requerimientos utilizando DevOps. In *XXII Congreso Argentino de Ciencias de la Computación (CACIC 2016)*.
- Ben, U. C. (2015). Design and Implementation of a Combinational Logic Design Tutor. *American-Eurasian Journal of Scientific Research*, 10(2), 73-87.
- Boulton (2018). DevOps: 12 consejos para impulsar la transformación digital. Recuperado de <http://devopscolombia.com/devops-12-consejos-impulsar-la-transformacion-digital/>
- Canós, J. H., y Letelier, M. C. P. (2012). Metodologías ágiles en el desarrollo de software. Universidad Politécnica de Valencia. Dirección electrónica <http://hdl.handle.net/123456789/476>

- Cervantes, M. H., Velasco-Helizondo, P. y Castro C.L. (2016). *Arquitectura de software. Conceptos y ciclo de desarrollo*. CDMX, México: Cengage Learning.
- Clavellina, M. J.L. (2013). Crédito bancario y crecimiento económico en México. *Economía Informa*. 378. (14-36).
- CNBV. (2017). Concepto de Banca Múltiple. Dirección electrónica <https://www.gob.mx/cnbv/acciones-y-programas/banca-multiple>
- DASA. (2017). Principios de Desarrollo de DASA. DevOps trata sobre experiencias, ideas y cultura para crear organizaciones de TI de alto rendimiento. Recuperado de <https://www.devopsagileskills.org/dasa-devops-principles/>
- De Alba, J. J. (2005). Marco Legal y Normativo del Sistema Financiero Mexicano. CDMX, México: Ediciones Ruz.
- De Salvador, L. (2011). Ingeniería Social y Operaciones Psicológicas en Internet. Instituto Español de Estudios Estratégicos. Recuperado de http://www.ieee.es/en/Galerias/fichero/docs_opinion/2011/DIEEEO74-2011.IngenieriaSocial_LuisdeSalvador.pdf
- Deloitte. (2012). “El futuro de la banca móvil en América Latina. Perspectivas desde Argentina, Brasil y México”. Dirección electrónica https://www2.deloitte.com/content/dam/Deloitte/py/Documents/about-deloitte/Futuro_banca_movil2012.pdf
- DevOps. (2018). Concepto del enfoque DevOps. Dirección electrónica <https://devops.com/>
- Dimes, T. (2015). *Conceptos Básicos de Scrum: Desarrollo de software Agile y manejo de proyectos Agile*. Babelcube Inc.
- Duque, J. (2014). Los procesos de cambio organizacional y la generación de valor. *Estudios Gerenciales*, 30(131), 162-171.
- Edwards, D. (2010). What is DevOps? Recuperado de <http://dev2ops.org/2010/02/what-is-devops/>
- FELABAN (2018). IV Informe de Inclusión Financiera. Recuperado de <http://felaban.s3-website-us-west-2.amazonaws.com/publicaciones/archivo20181211205300.pdf>

- Findex, G. (2017). Medición de la inclusión financiera y la revolución de la tecnología financiera. Recuperado de <https://openknowledge.worldbank.org/bitstream/handle/10986/29510/211259ovSP.pdf>
- FMI (2019). México: Una oportunidad para la inclusión financiera. Recuperado de <https://www.imf.org/es/News/Articles/2019/05/29/sp052919-mexico-an-opportunity-for-financial-inclusion>
- Fowler, M. (2014). Microservices a definition of this new architectural term. Recuperado de <https://martinfowler.com/articles/microservices.html>
- Gacitúa B. R. (2003). Métodos de desarrollo de software: El desafío pendiente de la estandarización. Software Development Methodologies: A Duel Pending for Standardization. *Theoria*, 12 (1), 23-42.
- Gartner (2018). Glosario IT de Gartner. Recuperado de <https://www.gartner.com/it-glossary/devops>
- Global Mobile Banking Benchmark (2017). Dirección electrónica <https://www.bbva.com/es/bbva-numero-uno-mundial-banca-movil/>
- Gómez, A. R., Duarte, A. Q., y Güevara, C. D. M. (2014). Desarrollo ágil de software aplicando programación extrema. *Revista Ingenio UFPSO*, 5(1), 24-29.
- Gutiérrez, M. F., Muñoz G. G., y Ángeles U. E. (2013). Sistema financiero y actividad económica en México: negocio y divergencia del sector bancario. *Análisis Económico*. 67 (XXVII), 171-198.
- Guzmán, D. I., Islas, U. C., Corona, C. P., y Méndez, B. E. P. (2014). Metodología ágil Scrumban en el proceso de desarrollo y mantenimiento de software de la norma MoProSoft. *Research in Computing Science*, 79, 97-107.
- Hernández, S. R., Fernández, C. C., y Baptista, L. P. (2010). *Metodología de la Investigación*. Quinta Edición. México: Mc. Graw -Hill/ Interamericana Editores S.A. de C.V.
- IBM (2017). DevOps for hybrid cloud: an IBM point of view. How DevOps for hybrid cloud can help organizations succeed with digital reinvention. Recuerado de <https://www-01.ibm.com/common/ssi/cgi-bin/ssialias?htmlfid=RAW14389USEN>
- IBM (2018). NBCUniversal scales DevOps across a large multi-speed IT Enterprise. Recuperado de <https://www.ibm.com/case-studies/nbcuniversal>

- IBM. (2019). ¿Qué es DevOps? Recuperado de <https://www.ibm.com/cloud/devops>
- Isaias P., y Issa T. (2015). Introduction to Information Systems Models and Methodologies. In: High Level Models and Methodologies for Information Systems. Springer, New York, NY.
- INEGI (2017). Encuesta Nacional sobre Disponibilidad y Uso de Tecnologías de la Información en Hogares. Recuperado de <http://www.beta.inegi.org.mx/proyectos/enchogares/regulares/dutih/2015/default.html>
- INEGI (2018). Tercera Encuesta Nacional de Inclusión Financiera. Recuperado de <https://www.inegi.org.mx/contenidos/saladeprensa/boletines/2018/OtrTemEcon/ENIF2018.pdf>
- Kniberg, H., y Skarin, M. (2010). *Kanban y Scrum – obteniendo lo mejor de ambos*. Washington, EE.UU.: C4Media
- Kopicki, Ron. y Miller, C. (2008) Banca móvil. Programa de aprendizaje sobre políticas de la FAO. Recuperado el 11 de septiembre de 2018, de http://www.fao.org/docs/up/easypol/686/3-4_mobilebanking_149ES.pdf
- Letelier, P., y Penadés, M. C. (2012). Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP). Laboratorio de Sistemas de Información. Departamento de Sistemas Informáticos y Computación. Facultad de Informática. Universidad Politécnica de Valencia.
- Lwakatare, L.E., Kuvaja, P. y Oivo, M. (2015). Dimensions of DevOps. *Agile Processes in Software Engineering and Extreme Programming*, (16), 212-217.
- Machiraju, S., & Gaurav, S. (2018). *DevOps for Azure Applications. Deploy Web Applications on Azure*. E.E. U.U., Washington: Apress.
- Mendoza, E.V. (2011). 10 razones para probar la banca móvil. Expansión. Dirección electrónica https://expansion.mx/mi-dinero/2011/07/27/10-razones-para-probar-la-banca-movil?internal_source=PLAYLIST
- Mishra, O., Kumar, V., y Garg, D. (2013). JIT supply chain; an investigation through general system theory. *Management Science Letters* , 3(3), 743-752.
- Molina, S.G.R. (2012). Metodologías ágiles enfocadas al modelado de requerimientos. *Informes Científicos-Técnicos UNPA*, 5(1), 1-29.

- Navarro C., A., y Fernández M., J., & Morales V., J. (2013). Revisión de metodologías ágiles para el desarrollo de software. *PROSPECTIVA*, 11 (2), 30-39.
- Pantaleo, G.L., y Rinaudo, L. (2016). *Ingeniería de Software*. Buenos Aires, Argentina: Alfaomega.
- Papajorgji, P. J., y Pardalos, P. M. (2014). *Software engineering techniques applied to agricultural systems*. USA,Florida: Springer.
- Pauta, A. L., y Moscoso, B. S. (2018). Verificación y validación de software. *Killkana Técnica*, 1(3), 25-32. https://doi.org/10.26871/killkana_tecnica.v1i3.112
- Pérez, A. O. A. (2011). Cuatro enfoques metodológicos para el desarrollo de Software RUP–MSF–XP–SCRUM. *INVENTUM*, 6(10), 64-78. Recuperado de <http://biblioteca.uniminuto.edu/ojs/index.php/Inventum/article/view/9>
- Pfeiffer, M (2017). Nueve métricas DevOps que debería usar para estimar las mejoras. SearchDataCenter en Español. Recuperado de <https://searchdatacenter.techtarget.com/es/consejo/Nueve-metricas-DevOps-que-deberia-usar-para-estimar-las-mejoras>
- Pressman, R.S. (2010). *Ingeniería del Software. Un enfoque práctico*. CDMX, México: McGraw-Hill.
- Ponce, M.U., Moreno, Yáñez, M. V., y Soto, B. R. (2014). Propuesta Metodológica para Desarrollo de Aplicaciones Móviles para Dispositivos Android. *Congreso Internacional de Investigación Academia Journals 2014*, 6,(3), 1429-1434.
- Quint (2016). Cinco preguntas clave sobre DevOps. Cerrando el círculo de la agilidad empresarial. Digital Biz. 19. Recuperado de <https://www.digitalbizmagazine.com/cinco-preguntas-clave-sobre-devops/>
- RightScale's, 2016. "State of the Cloud Report". Santa Bárbara, CA. Recuperado de <https://www.rightscale.com/press-releases/rightscale-2016-state-of-the-cloud-report>
- Rodríguez, C., y Dorado, R. (2015). ¿Por qué implementar Scrum?. *Revista Ontare*, 3(1), 125-144. <https://doi.org/10.21158/23823399.v3.n1.2015.1253>
- Sampieri, R. H., Collado, C. F., y Lucio, P. B.(2010). *Metodología de la investigación*. Quinta Edición. CDMX, México: McGraw-Hill.

- Sanchis, R., Poler, R. y Ortiz, A. (2009). Técnicas para el Modelado de Procesos de Negocio en Cadenas de Suministro. *Información tecnológica*, 20(2), 29-40. <https://dx.doi.org/10.4067/S0718-07642009000200005>
- Scaled Agile (2018). Scaled Agile Framework Enterprise (SAFe). Recuperado de <https://www.scaledagileframework.com/>
- Thoms, N. (2018). A Definition of DevOps for the Masses. Recuperado de <https://devops.com/definition-devops-masses/>
- Walls, M. (2013) Building a DevOps Culture, Sebastopol, California: O'Reilly.
- Womack, J. P., & Jones, D. T. (2008). Lean thinking. Banish waste and create wealth in your corporation, ProdPress. com, Wroclaw.
- World Finance (2018). World Finance Digital Banking Awards 2018. Recuperado de <https://www.worldfinance.com/awards/world-finance-digital-banking-awards-2018>
- Womack, J. P., & Jones, D. T. (2008). Lean thinking. *Banish waste and create wealth in your corporation*, ProdPress. com, Wroclaw.