

TABLA 1. CARACTERIZACIÓN DE PRINCIPIOS

Id	Principio	Referencia	Definición
P1	Coordinación entre equipos	(Dingsøyr y Moe, 2014),	Para coordinar equipos en el desarrollo ágil a gran escala, se deben establecer normas y valores comunes y enfocarse en las redes de conocimiento eficaces, ya que estas son esenciales en el desarrollo a gran escala debido a la naturaleza intensiva en conocimiento del desarrollo de software.
P2	Gestión de portafolio	(Dingsøyr y Moe, 2014),	La gestión del portafolio consiste en manejar y ejecutar varios proyectos ágiles en la misma cartera bajo un entorno cambiante. Debe existir una realimentación continua del nivel del proyecto al nivel del portafolio que permita cambiar el portafolio en pro a optimizar el valor que actualmente se está generando. De la misma manera, debe existir una retroalimentación continua del portafolio hacia a los niveles de proyecto; con el fin de que los equipos tomen decisiones las cuales sean consistentes con los objetivos de la cartera.
P3	Prácticas de escalamiento	(Dingsøyr y Moe, 2014),	Comprender el contexto es esencial al elegir el enfoque correcto para mejorar la agilidad a gran escala de los equipos y la organización. Para el desarrollo de sistemas embebidos a gran escala, la agilidad debería escalar con respecto al número de equipos involucrados y las actividades de ingeniería de sistemas en cada iteración debido a la codependencia del desarrollo de software y hardware.
P4	Arquitectura de software	(Dingsøyr y Moe, 2014),	Se debe prestar atención al papel que representa la arquitectura y como esta coordina el trabajo del desarrollo a gran escala. Además, la organización del trabajo arquitectónico debe ser influenciado por el nivel de incertidumbre y el nivel de cambio.
P5	El contenido es la clave	(Laanti, 2014)	Use la retroalimentación de los usuarios y el conocimiento esencial basado en la experticia y experiencia para crear lo mejor que pueda soñar.
P6	Co creación	(Laanti, 2014)	Los grupos son más rápidos para resolver problemas que los individuos. Deje que el software evolucione juntos, ya que la suma del todo es más que sus partes. El desarrollo de software es un juego cooperativo.
P7	La retroalimentación es el combustible para aprender	(Laanti, 2014)	Utilice comentarios rápidos y concretos sobre todo el trabajo realizado. Estudie lo que crea el éxito y haga más de eso.
P8	Agilidad empresarial	(Laanti, 2014)	Los lanzamientos generan ingresos. El modelo de negocio debe dictar la tasa de lanzamiento y el interés del usuario define el modelo de negocio. Libere con menos frecuencia cuando el costo de la transacción sea alto.
P9	Uso de la automatización como apalancamiento	(Laanti, 2014)	Use la automatización para aprovechar el esfuerzo manual necesario. Desarrolle el sistema, de modo que brinde un mejor apalancamiento para la unidad de trabajo realizada.

P10	Escala usando fractales	(Laanti, 2014)	Los fractales son la forma natural de escalar y estructuras bastante permanentes. Utilice niveles de abstracción más altos y sistemas anidados, como los bucles de control anidados.
P11	Evitar explosiones combinadas	(Laanti, 2014)	Agregar más personas a un proyecto ralentiza el progreso, ya que la necesidad de comunicación crece casi exponencialmente cuando aumenta el número de interfaces. La complejidad se controla mejor dividiéndola en pedazos más pequeños.
P12	Secuencia para el máximo rendimiento	(Laanti, 2014)	La arquitectura modular aumenta la velocidad. Encuentre el rendimiento máximo para su cartera equilibrando lo que se puede hacer en paralelo y lo que se debe hacer en secuencia.
P13	Apreciar el conocimiento profundo	(Laanti, 2014)	Solo más de cinco años de experiencia crean un conocimiento profundo. Use los mejores expertos para abordar los problemas más importantes y complejos.
P14	Nivelación de trabajo	(Laanti, 2014)	Distribución uniforme del trabajo y eliminación del trabajo innecesario y el tiempo de espera en función del rendimiento medido. La priorización del trabajo y Kanban son las herramientas aquí.
P15	Simplicidad	(Laanti, 2014)	Buscar simplicidad en las soluciones. Simplicidad es un principio ágil.
P16	Situacionalidad	(Laanti, 2014)	Use el principio de Pareto para evitar que los procesos sean demasiado complejos. No todos los casos deben ser tratados por igual.
P17	Controlar procesos, no ítems	(Laanti, 2014)	Cree reglas simples para la toma de decisiones, en lugar de controlar cada decisión individualmente. Haz reglas claras del juego.
P18	Mentalidad de crecimiento	(Laanti, 2014)	Haz más de lo que creó el éxito. Los mejores líderes no rechazan los intentos defectuosos, sino que les dan un giro para crear más éxito. Los fracasos son la fuente secreta del éxito.
P19	Escuche a los empleados, ellos conocen todos los problemas.	(Laanti, 2014)	El valor se crea en primera línea. La velocidad a la que puede eliminar los impedimentos de progreso o servicio se correlaciona con la mejora realizada para los negocios. Comprende el problema que estás resolviendo junto con la ayuda de los empleados.
P20	Detectar y usar patrones	(Laanti, 2014)	Usa y aplica patrones. Sus problemas ya han sido resueltos por alguien y en alguna parte.
P21	El costo de la innovación	(Laanti, 2014)	Alivie la carga del usuario con una solución que cueste menos. Brinde un mejor servicio o llene los vacíos entre los flujos de valor. No ate el capital, permita flexibilidad en las inversiones y opciones de pensamiento a nivel de cartera. Optimizar el costo de la cartera.
P22	Utilizar el conocimiento tácito	(Laanti, 2014)	Utilice el conocimiento tácito de las personas para saber si se dirige en la dirección correcta o no. Cuando la gente se siente orgullosa del resultado, se dirige en la dirección correcta.

P23	El aprendizaje ocurre entre equipos	(Laanti, 2014)	La integración grupal permite una mejor empatía entre los miembros de los equipos, dando como resultado una retroalimentación más rápida y la creación de conocimiento colectivo bajo una misma visión y ambición.
P24	Rápido es mejor que la perfección.	(Laanti, 2014)	Maximiza el trabajo sin hacer. Si no está roto, no lo arregles. Tolerar pequeñas imperfecciones. Rápido es mejor que la perfección. Lo mejor es enemigo de lo bueno.
P25	Prevenir el problema cuando sea pequeño	(Laanti, 2014)	El éxito esconde pequeños problemas. Para mantenerse exitoso, no ignore los pequeños problemas.

TABLA 2. CARACTERIZACIÓN Y CLASIFICACIÓN DE PRÁCTICAS

Id	Prácticas	Referencia	Definición	Clasificación	
				Qué	Cómo
PR1	SoS semanales	(Kalenda, 2017)	Reunión de sincronización entre equipos. Informar sobre el estado actual del trabajo, impedimentos y gestión de dependencias.	X	
PR2	Enviar representantes del equipo a la reunión de SoS	(Kalenda, 2017)	Tal como sugiere LeSS o SAFe con la variación de usar un sistema rotativo de miembros representantes del equipo en cada reunión.		X
PR3	Llevar a cabo un Scaled Sprint Demo separadas por equipos individuales	(Kalenda, 2017)	Cada equipo tendrá su reunión de demostración por separado, sólo con propietarios de producto	X	
PR4	Modelo de feria de ciencias	(Kalenda, 2017)	Cada equipo configura una pantalla, y los empleados visitan las pantallas que les parecen interesantes. Cada 'stand' ofrece una presentación de 15 minutos sobre los esfuerzos de trabajo recientes		X
PR5	Llevar a cabo un reunión de retrospectiva a escala	(Kalenda, 2017)	La reunión de retrospectiva consiste en reflexionar sobre la ejecución de la iteración anterior por encima del nivel de un solo equipo. Esta reunión forma parte del taller de inspección y adaptación en SAFe, y LeSS lo llama retrospectiva general.	X	
PR6	Llevar a cabo scaled sprint planning	(Kalenda, 2017)	Scaled sprint planning es una reunión donde se discute el futuro de un producto. En SAFe se conoce como PI planning y LeSS; sprint planning one.	X	

PR7	Utilizar dos nuevas capas para gestionar los requisitos	(Kalenda, 2017)	En una primera capa, llamada equipos de productos se estructuran los responsables del trabajo: propietarios del producto, especialistas, expertos y miembros del equipo de desarrollo. La segunda capa, denominada equipos de productos de área, la cual se usa cuando se desarrolla un producto más grande y con más de siete equipos de desarrollo, se delegan responsabilidades de toma de decisiones sobre el presupuesto y decisiones operativas que afectan al producto		X
PR8	Usar el rol Product owner team	(Kalenda, 2017)	Incluir a todos los propietarios de productos de equipos individuales, gerente de cartera, gerente de pruebas y líder de UX. Con el fin de aclarar las responsabilidades y hacer posible el desarrollo de requisitos funcionales claros para los equipos basados en una comprensión profunda de los requisitos del negocio desde el punto de vista del cliente.		X
PR9	Crear equipos de componentes cruzados	(Kalenda, 2017)	Crear equipos interfuncionales especializados en flujos comerciales específicos, que les permita centrarse en un área de productos más estrecha		X
PR10	Establecer un programa de intercambio sistemático entre equipos	(Kalenda, 2017)	Capacitación en el aula, visitas de intercambio y sesiones de videoconferencia, así como visitas a largo plazo con objetivos claramente especificados.		X
PR11	Todos los equipos deben compartir el mismo modelo mental	(Bjørnson, Wijnmaalen, Stettina, y Dingsøyr, 2018)	Para trabajar juntos de manera efectiva, los equipos deben tener una comprensión clara del proceso de trabajo, las tareas y las capacidades de otros equipos. Es importante que todos los equipos compartan el mismo modelo mental para que puedan interpretar las señales contextuales de manera similar y tomar decisiones compatibles con respecto a su objetivo común.	X	
PR12	La comunicación en circuito cerrado	(Bjørnson et al., 2018)	Es más que simplemente desarrollar y enviar mensajes; también tiene que ver con la creación de un significado compartido. La comunicación es el simple intercambio de información, mientras que la comunicación de circuito cerrado agrega un ciclo de retroalimentación: ¿Se recibió e interpretó correctamente la información? Este ciclo de retroalimentación adicional es fundamental para una comunicación exitosa entre múltiples equipos.	X	

PR13	Confianza entre equipos	(Björnson et al., 2018)	Creencia compartida de que los equipos desempeñarán sus funciones y protegerán los intereses de sus compañeros de trabajo. La confianza mutua es la confianza del equipo en el carácter, la integridad, la fuerza y las habilidades de otro equipo o grupo. La confianza modera la relación entre el rendimiento del equipo.	X	
PR14	Asegurar el apoyo financiero	(Brown, 2011)	El progreso requiere un fuerte apoyo de la gerencia. Dadas las presiones financieras y políticas que existen, los proyectos no pueden luchar sin visibilidad y el apoyo del CIO y su equipo.	X	
PR15	Entrenamiento intensivo y acompañamiento durante los primeros 2-6 meses	(Brown, 2011)	Se debe capacitar a las personas en ágil, así mismo un entrenador debe estar disponible para resolver las dudas de los equipos.	X	
PR16	Tome una visión amplia de la habilitación y el conocimiento	(Brown, 2011)	La capacitación debe ser especializada para los equipos de desarrollo, las partes interesadas del negocio y los ejecutivos de gestión. Específicamente, debe incluirse la capacitación de sensibilización para la gestión de nivel superior.	X	
PR17	Tener total compromiso con la adopción y el despliegue	(Brown, 2011)	El compromiso con la adopción y el despliegue es crítico. Habrá desafíos que deben ser direcciones que requieren un nivel de fe y determinación para tener éxito	X	
PR18	Considere el ciclo de vida completo del sistema	(Ambler, 2007)	Hay varias razones por las que es importante adoptar el ciclo de vida de: comienzo, elaboración y construcción, transición y producción. Primero, muchos equipos ágiles se centran en los aspectos de construcción del SDLC (Ciclo de vida del desarrollo) sin tener en cuenta las complejidades de iniciar un proyecto. Implementarlo en producción o incluso ejecutar el sistema una vez que está en producción, los riesgos abordados por estas fases son críticos independientemente de la escala, pero aumentan la importancia en proporción al aumento de la complejidad resultante de los factores de escala mencionados anteriormente. En segundo lugar, el ciclo de vida incluye explícitamente técnicas de escalado importantes, como los requisitos iniciales y la visión de la arquitectura, así como pruebas independientes continuas.		X

PR19	Continuous Independent Testing	(Ambler, 2007)	TDD es un enfoque para las pruebas confirmatorias en el que valida el sistema al nivel de su comprensión de los requisitos. Este es el equivalente de "prueba de humo" o prueba contra la especificación, aunque es importante, no es toda la imagen de validación. El desafío fundamental con las pruebas confirmatorias, y por lo tanto TDD, es que asume que las partes interesadas entienden y pueden describir sus requisitos. Aunque los enfoques iterativos aumentan la posibilidad de esto, no hay garantías. Una segunda suposición de TDD es que los desarrolladores tienen las habilidades para escribir y ejecutar las pruebas, habilidades que se pueden obtener con el tiempo pero que pueden no tener hoy en día.	X	
PR20	Desarrollo impulsado por el riesgo y el valor	(Ambler, 2007)	Las fases explícitas del Proceso Unificado (UP) y sus hitos son estrategias importantes para escalar el desarrollo de software ágil para satisfacer las necesidades del mundo real de las organizaciones modernas. El ciclo de vida de UP está basado en el riesgo y el valor. Lo que esto significa es que los equipos del proyecto UP se esfuerzan activamente por reducir tanto el riesgo comercial como el técnico al inicio del ciclo de vida, al tiempo que brindan comentarios concretos durante todo el ciclo de vida en forma de software de trabajo. Cuando los procesos ágiles como XP y Scrum están claramente basados en valores, pueden mejorarse para abordar el riesgo de manera más efectiva. Esto es particularmente importante a escala debido al mayor riesgo asociado con la mayor complejidad de tales proyectos.	X	
PR21	Técnicas ágiles de bases de datos	(Ambler, 2007)	Los datos son un aspecto importante de cualquier aplicación comercial y en mayor medida de los activos de la organización en su conjunto. Así como la lógica de su aplicación puede desarrollarse de manera ágil, también lo pueden hacer sus activos orientados a datos. Para escalar ágilmente, todos los miembros del equipo deben trabajar de manera ágil, incluidos los profesionales de datos.	X	
PR22	Gobernanza de desarrollo Lean	(Ambler, 2007)	La gobernanza es fundamental para el éxito de cualquier departamento de TI, y es particularmente importante a escala. La gobernanza efectiva no se trata de comando y control, sino que se centra en habilitar las conductas y prácticas correctas a través de técnicas colaborativas y de apoyo. Es mucho más efectivo motivar a las personas a hacer lo correcto que tratar de obligarlas a hacerlo.	X	

PR23	Hablar su idioma	(Nyfjord, Bathallath, y Kjellin, 2014)	Un obstáculo importante al tratar con la colaboración en contextos complejos es el contenido emocional de varios tipos de comunicación. Por lo tanto, la forma más efectiva de asegurar que la comunicación funcione bien cuando varios equipos están coordinados es usar su lenguaje y también dar retroalimentación específica a los equipos sobre cómo describen su trabajo, sus necesidades y su progreso. En una etapa posterior, se pueden desarrollar estándares de comunicación sobre cómo los equipos deben comunicarse entre sí, pero si se introducen prematuramente, crean más confusión de la que resuelven.	X	
PR24	Crear una cultura de Benchmarking público	(Nyfjord et al., 2014)	Uno de los métodos más eficientes para compartir conocimiento es comparar el desempeño entre equipos. Sin embargo, si los gerentes no brindan retroalimentación, crítica y apreciación de manera similar a varios equipos, es difícil crear una cultura de comunicación justa y abierta. Para que los equipos ajusten las descripciones de su trabajo de acuerdo a cómo otros equipos describen su trabajo, es necesario desarrollar un diálogo continuo sobre las comparaciones entre los equipos. Se ha probado con éxito una versión específica de este llamado "Sitios de pares", donde un sitio de desarrollo de software da una respuesta personalizada a otro sitio de desarrollo.	X	
PR25	Motivar desde la experiencia personal	(Nyfjord et al., 2014)	Los gerentes que no pueden defender su motivación al proporcionar referencias a la experiencia práctica pueden arriesgarse a la trampa de la comunicación de tener subordinados que no los entiendan. Esto es especialmente importante cuando hay pocas rutinas fijas, lo que ejerce una gran presión sobre la capacidad de las personas para aceptar ideas abstractas. Proporcionar argumentos racionales y consistentes también puede ser difícil cuando se discuten varias teorías o perspectivas. En tales casos, el uso de ejemplos aclara las ideas. Por lo tanto, el gerente que motiva sus argumentos combinados con ejemplos reales y concretos tiene más probabilidades de ser entendido que el que usa teorías abstractas		X

PR26	Incluir subordinados en reuniones	(Nyfjord et al., 2014)	Los representantes de varios niveles jerárquicos deben incluirse en las reuniones. Las reuniones entre la gerencia y los líderes del equipo siempre deben incluir al menos un miembro del equipo y preferiblemente debe ser un miembro diferente en cada reunión. Esto permite a cada miembro del equipo comunicar su visión del proyecto y motiva a todos a ser parte del proceso de decisión. Al tratar siempre de incluir tres niveles jerárquicos en las reuniones, se asegura que los aspectos relevantes se mueven entre todos los niveles de la organización. En el modelo de sistema propuesto de "Planificación interactiva".		X
PR27	Combinar varios métodos de modelado	(Nyfjord et al., 2014)	Tener una actitud flexible hacia los modelos y usarlos cuando sea necesario, y en cualquier combinación de modelos, y usar los modelos como herramientas de apoyo en lugar de sistemas de gobierno. Los gerentes deben ser hábiles para dibujar "Imágenes enriquecidas" que proporcionen modelos del problema que puedan explicarse fácilmente a los legos y hacerlos conscientes de las diversas perspectivas entre las partes interesadas en un proyecto complejo. Un gerente puede aplicar varias perspectivas diferentes al mismo tiempo y para cada perspectiva hacer modelos conceptuales. Cuando se han presentado varios modelos conceptuales, es posible comparar todos los modelos con la situación de desarrollo de software y sacar conclusiones basadas en más de una perspectiva.		X
PR28	Usar definiciones globales	(Nyfjord et al., 2014)	El uso de diferentes definiciones en varios niveles organizacionales causará confusión. Por lo tanto, el uso de descripciones recursivas respalda la comunicación global en proyectos grandes porque se utilizan protocolos de comunicación similares en todos los niveles del proyecto.		X
PR29	Aceptar el caos como motor de desarrollo	(Nyfjord et al., 2014)	Aceptar cierta cantidad de caos sin infringir los objetivos básicos del proyecto. Para ello, el gerente debe lidiar con las relaciones, el dinamismo y la imprevisibilidad. No controlar sino liderar.		X
PR30	Centralizar reglas críticas	(Nyfjord et al., 2014)	El uso de un enfoque estricto de "gestión por objetivo" en un contexto complejo probablemente limitará la productividad de los equipos. Por lo tanto, la formalización de objetivos y limitaciones como reglas o principios genéricos que se pueden implementar de cualquier manera que los equipos consideren apropiados proporcionará una dirección más suave que permitirá la creatividad y productividad del equipo.		X

PR31	Conocimiento compartido	(Berntzen, Moe, y Stray, 2019)	Informar a los participantes de cómo sus propias tareas y las de los demás contribuyen al proceso de trabajo general. Una comprensión compartida del proceso de trabajo y las áreas de experiencia de otros facilitan la coordinación del conocimiento	X	
PR32	Objetivos compartidos	(Berntzen et al., 2019)	Dirigir la atención y el esfuerzo de individuos y grupos. Trascender los objetivos funcionales de las diferentes unidades de trabajo y permitir un esfuerzo unificado hacia un resultado colectivo.	X	
PR33	Respeto mutuo	(Berntzen et al., 2019)	Valorar las contribuciones de los demás y considerar el impacto de sus propias acciones en el trabajo de los demás.	X	
PR34	Comunicación de alta calidad	(Berntzen et al., 2019)	Comunicación frecuente, precisa, oportuna y de resolución de problemas.	X	

TABLA 3. CARACTERIZACIÓN Y CLASIFICACIÓN DE DESAFÍOS Y FACTORES DE ÉXITO

Id	Factor	Referencia	Definición	Clasificación	
				Qué	Cómo
FE1	Unificación de vistas y valores	(Kalenda, 2017)	La empresa de estudio de caso definió valores compartidos: se logró mediante la introducción de un "Manifiesto ágil". El Manifiesto contenía doce declaraciones sobre los valores y principios básicos que la compañía quería seguir. Además, contenía varias reglas que guiaban la forma de trabajar en la empresa. Estas reglas definieron las prioridades en la organización, por ejemplo, i) cooperación y coherencia entre equipos antes de la autonomía y ii) especialización con sustituibilidad antes de la distribución de responsabilidades del equipo.	X	
FE2	Apoyo de la gerencia	(Kalenda, 2017; Dikert, Paasivaara, y Lassenius, 2016; Ebert y Paasivaara, 2017)	Obtener el apoyo y el compromiso de los altos mandos y administrativos de la organización para implementar un enfoque	X	

FE3	Cultura empresarial y previa experiencia en ágil	(Kalenda, 2017; Barroca, Dingsøyr, Falessi, y Power, 2019)	Valores, actitudes, patrones y hábitos dentro y entre los grupos existentes en la organización, que junto con la experiencia previa de algunos integrantes de los equipos facilita la difusión del conocimiento en desarrollo ágil de software.	X	
FE4	Preparar el primer evento de planificación del incremento de programa	(Ebert y Paasivaara, 2017)	Con la ayuda de un entrenador externo, crear agendas e instrucciones para los participantes del evento de planificación del incremento de programa		X
FE5	Tener un ingeniero de tren de lanzamiento (RTE) de tiempo completo	(Ebert y Paasivaara, 2017)	Tener un ingeniero de tren de lanzamiento de tiempo completo que gestione la coordinación organizando y liderando las reuniones Scrum-of-Scrums (SoS) y cuidando los elementos de mejora continua.		X
FE6	Prestar especial atención a los elementos de mejora	(Ebert y Paasivaara, 2017)	Asigne responsables a cada elemento de mejora, y monitoree su implementación.		X
FE7	Informar y tener compromiso con el cambio	(Ebert y Paasivaara, 2017; Dikert et al., 2016)	Asegurar desde el comienzo que todos entiendan las razones del cambio y por qué es importante.	X	
FE8	Capacitar y entrenar al personal	(Ebert y Paasivaara, 2017; Dikert et al., 2016; Barroca et al., 2019)	Capacitar tanto a los gerentes como a los miembros del equipo sobre la adopción del nuevo enfoque.	X	
FE9	Involucrar a los líderes del cambio	(Ebert y Paasivaara, 2017; Dikert et al., 2016; Barroca et al., 2019)	Factor de éxito el cual resalta que la coordinación se ve favorecida gracias a los portavoces del cambio, líderes que contribuyen en la adopción de la nueva forma de trabajo.		X
FE10	Elegir y personalizar un enfoque ágil	(Dikert et al., 2016)	Los equipos deben innovar y encontrar las prácticas que mejor se adapten a sus procesos. Esto indica personalizar el enfoque ágil de una manera pragmática y no siguiendo una interpretación estricta de los libros de texto, no obstante, siempre teniendo en cuenta y especial cuidado de no contradecir los principios ágiles al momento de personalizar dicho enfoque. Asimismo, se debe ver la 'transformación' como un proceso ágil constante	X	

			incentivando su refinamiento a las necesidades actuales de la organización.		
FE11	Empezar con una adopción piloto	(Dikert et al., 2016)	Implementar un enfoque ágil en un proyecto piloto. Esto ayuda a crear confianza en que la forma ágil de trabajar sería adecuada para la organización y aumenta la aceptación tanto en el desarrollo como en la gestión ágil.		X
FE12	Contar con un entrenador externo	(Dikert et al., 2016)	Un entrenador puede marcar la diferencia entre el éxito y el fracaso en la transformación, ya que puede observar y corregir los problemas cuando surgen, ayudando a desviar la atención del enfoque en las herramientas para comprender los principios ágiles. Se recomienda que el entrenador sea externo, ya que este tendrá una visión objetiva sobre la organización.	X	
FE13	Comunicación y transparencia	(Dikert et al., 2016)	Tratar de comunicar el cambio a la mayoría de personas de la organización, buscando que todos conozcan (o refuerzan sus conocimientos) en torno a la transformación, la nueva forma de trabajo y los objetivos que se desean alcanzar con ella. Adicional a ello, se debe hacer el cambio transparente y facilitar el intercambio de información entre los implicados con el fin de capturar y compartir recomendaciones, comentarios, experiencias, entre otras.	X	
FE14	Mentalidad y alineación ágil	(Dikert et al., 2016)	Entender los valores ágiles para dar sentido a las razones que motivan el cambio o transformación de una organización.	X	
FE15	Autonomía del equipo	(Dikert et al., 2016)	Permitir la autoorganización en equipos crea un compromiso y motivación con el cambio, que los equipos se apropien del proceso de desarrollo, lo mejoren voluntariamente, aumenta la productividad y la moral.		X
FE16	Gestión de requerimientos	(Dikert et al., 2016)	El compromiso y el buen desempeño del rol de propietario	X	

			del producto, influyó en el correcto desempeño de los equipos.		
Desafíos reescritos como factores de éxito					
FD1	<i>Incentivar el compromiso y el trabajo en equipo</i>	(Kalenda, 2017)	El trabajo en equipo puede verse afectado cuando los especialistas del equipo no se sienten cómodos compartiendo sus conocimientos especializados con los demás integrantes del equipo ya que creen que puede perder su calidad de especialista si alguien más aprende el cómo hacer su trabajo. También se dio el caso en que los desarrolladores al no entender el trabajo realizado por sus otros compañeros no confiaban en dicho trabajo. En algunas ocasiones la definición de 'hecho' no era grupal sino interpretada individualmente por los miembros de los equipos.		X
FD2	Medir el progreso	(Kalenda, 2017)	Medir el progreso consiste en monitorear el avance y medir la tendencia del cambio. Se torna un desafío, en la medida en que para algunas organizaciones no fue fácil encontrar qué métricas medir para obtener resultados significativos.	X	
FD3	<i>Manejar la presión y la sobrecarga de trabajo a los equipos</i>	(Kalenda, 2017)	La nueva forma de trabajo junto con la presión constante del mercado pueden ser detonantes de la tensión en los equipos.	X	
FD4	<i>Persuadir a los equipos para que transformen su forma, procesos y herramientas de trabajo</i>	(Ebert y Paasivaara, 2017; Dikert et al., 2016; Kalenda, 2017; Barroca et al., 2019; Conboy y Carroll, 2019)	La resistencia al cambio se da cuando en una organización que se va a transformar, hay personas que pueden estar familiarizadas con el desarrollo ágil, pero que no gustan de esa forma de trabajo, ya que muchas veces no quieren cambiar los procesos y herramientas que ya han seleccionado.	X	

FD5	<i>Contar con un enfoque estructurado de requisitos</i>	(Dikert et al., 2016; Kalenda, 2017; Barroca et al., 2019)	Algunos frameworks no tienen un enfoque estructurado de requisitos y a menudo los grandes proyectos de desarrollo exigen una gestión de requisitos de alto nivel. Al estar distribuidos, los requisitos son creados por diferentes partes interesadas y los equipos no pueden estar al tanto de todos ellos. Se complica aún más cuando los propietarios del producto tienen dificultades para desglosarlos a un tamaño que sea posible estimar el esfuerzo.	X	
FD6	Aseguramiento de la calidad	(Dikert et al., 2016; Kalenda, 2017; Barroca et al., 2019)	Actividades de prueba no se pueden realizar dentro de los límites de las historias de usuario y requieren más recursos. Cuando las tareas de prueba se superponen a los límites del equipo, es sensato tener equipos separados para estas tareas, pero se debe definir la coordinación entre los equipos especializados de control de calidad y los equipos de desarrollo.		X
FD7	<i>Apoyar la integración de ágil con no ágil</i>	(Dikert et al., 2016; Kalenda, 2017; Barroca et al., 2019)	Los beneficios completos de la transformación ágil no podrán lograrse a menos que toda la organización esté configurada para trabajar siguiendo el mismo paradigma. La integración de ágil con no ágil puede resolverse mediante la inclusión de partes de la organización en cascada en el proceso de planificación e involucrando a los equipos no ágiles al principio del proceso de planificación.	X	
FD8	Coordinación entre equipos en ambientes distribuidos	(Dikert et al., 2016; Kalenda, 2017; Barroca et al., 2019; Dingsøyr et al., 2019; Figalst et al., 2019)	En un ambiente de trabajo distribuido, los esfuerzos deben centrarse en la administración de dependencias, el equilibrio con los objetivos de la organización y el mantenimiento de la consistencia técnica.	X	

FD9	<i>Asegurar el apoyo financiero durante la adopción</i>	(Ebert y Paasivaara, 2017; Dikert et al., 2016; Kalenda, 2017; Barroca et al., 2019)	Es necesario invertir en entrenamiento y coaching en pro de mejorar los resultados de la transformación ágil (factores motivación y aplicación de métodos ágiles); además, es necesario permitir y/o crear los espacios físicos de trabajo necesarios para el desarrollo ágil, es decir, espacios compartidos para los desarrolladores en vez de oficinas separadas, espacios para los meetings como puede ser una sala de conferencias, entre otras.	X	
FD10	<i>Tener una idea compartida de implementación ágil</i>	(Dikert et al., 2016)	La organización no entiende los conceptos y el propósito del desarrollo ágil de software realizando una implementación no exitosa.	X	
FD11	<i>Contar con una guía de implementación y garantizar el entrenamiento y acompañamiento de expertos</i>	(Dikert et al., 2016; Barroca et al., 2019)	Debido a factores tales como la falta de comprensión de los conceptos de desarrollo ágil de software, o la falta de orientación tanto de la literatura como de un entrenador, entre otros, hacen que implementar métodos ágiles se convierta realmente difícil.	X	
FD12	Gestión jerárquica y balance estructura organizacional	(Dikert et al., 2016; Barroca et al., 2019; Conboy y Carroll, 2019; Figalíst, Elsner, Bosch, y Olsson, 2019)	Se plantean 3 escenarios en la implementación ágil a escala: (i) confusión y desorganización al enfrentarse con roles adicionales propuestos por la nueva forma de trabajar, (ii) el nuevo papel de los gerentes intermedios en ágiles no está claro. Hubo casos en los que la gestión continuó trabajando de acuerdo con el antiguo modelo de cascada. También hubo problemas para duplicar la burocracia cuando había dos formas diferentes de trabajar, (iii) en algunos casos, la organización inicial tenía límites internos, o conocimientos especializados en silos, causando problemas en la implementación ágil.	X	
FD13	Comparar y contrastar frameworks	(Conboy y Carroll, 2019)	Elegir entre los marcos de desarrollo ágil de software existentes no es una tarea fácil, se debe comparar frameworks		X

			para determinar cuál se ajusta mejor a las necesidades de la organización y justificar la inversión.		
FD14	<i>Elegir y apoyar una forma de implementación</i>	(Conboy y Carroll, 2019)	Se evidencia como un desafío, debido a que la implementación de un framework ágil (según el autor) fue un reto tanto en una implementación top-down como en una implementación bottom-up. Para la primera, debido a que no quedaba claro cuándo era necesario el apoyo y participación de la alta dirección y a que los equipos terminaron escalando prácticas distintas por las diversas interpretaciones; Para la segunda, al ser de arriba hacia abajo, algunos los miembros del equipo sintieron que les imponían una nueva forma de trabajo y que ésta no proporcionaba un camino claro de implementación.		X
FD15	<i>Buscar diferentes fuentes de conocimiento sobre la adopción ágil</i>	(Conboy y Carroll, 2019)	Falta de estudios empíricos que apliquen algún marco de desarrollo ágil de software y permitan a futuras organizaciones orientarse sobre las formas de hacer ágil. Si bien existen documentos básicos para SAFe, Scrum, LeSS y Scrum at Scale, faltan estudios de casos empíricos que posteriormente apliquen esos marcos "en la naturaleza". Allí hubo muchos ejemplos de "muros de ladrillo" en todas las organizaciones: casos en los que el personal afirmó que había un problema particularmente difícil o un problema contextual para el cual los documentos marco originales no tenían orientación		X
FD16	<i>Medir el progreso de la transformación ágil mediante el valor proporcionado</i>	(Conboy y Carroll, 2019)	Se tiende a medir la transformación ágil por el porcentaje de adopción de un marco, en vez de medirlo por ejemplo por el valor que proporciona. Los gerentes tienen poca orientación sobre cómo encontrar el grado óptimo de transformación.		X

FD17	Mantener la autonomía del desarrollador	(Conboy y Carroll, 2019)	Autonomía y flexibilidad es de lo que se trata ágil, pero cuando se trata de a gran escala y se es permisivo con los desarrolladores entonces el enfoque de adopción se fragmenta hasta que no queda nada de él	X	
FD18	Definir conceptos y términos comunes	(Conboy y Carroll, 2019; Dingsøy, Falessi, y Power, 2019)	Las diferentes interpretaciones del nuevo enfoque dan lugar a la inconsistencia en la adopción y el manejo de conceptos claves del framework. Los hallazgos indican que algunos marcos usan una "terminología abstracta" y falta explicación exhaustiva.		X
FD19	Asegurar la colaboración del cliente cuando existen numerosos stakeholders	(Dingsøy, Falessi, y Power, 2019)	En el desarrollo ágil a gran escala, los product owners soportan múltiples equipos que trabajan sobre un mismo producto, por lo que adquieren nuevas responsabilidades y ya no es un rol que cumple una persona sino un equipo de product owners.		X
FD20	Eficiencia en la toma de decisiones en grandes esfuerzos de desarrollo de productos o proyectos	(Dingsøy, Falessi, y Power, 2019)	Buscar un modo de toma de decisiones que preserve la autonomía del equipo en una organización distribuida globalmente trabajando en un producto.		X
FD21	<i>Mantener una comunicación adecuada y evitar la pérdida y alteración de información</i>	(Figalist, Elsner, Bosch, y Olsson, 2019)	Se debe priorizar la comunicación adecuada, ya que en entornos a gran escala, la información puede sufrir múltiples alteraciones cuando se difunde a través largas rutas.	X	

REFERENCIAS

- Ambler, S. W. (2007). Agile software development at scale. En Ifip central and east european conference on software engineering techniques (pp. 1–12).
- Barroca, L., Dingsøy, T., y Mikalsen, M. (2019). Agile transformation: A summary and research agenda from the first international workshop. En International conference on agile software development (pp. 3–9).
- Berntzen, M., Moe, N. B., y Stray, V. (2019). The product owner in large-scale agile: An empirical study through the lens of relational coordination theory. En International conference on agile software development (pp. 121–136).

- Bjørnson, F. O., Wijnmaalen, J., Stettina, C. J., y Dingsøyr, T. (2018). Inter-team coordination in large-scale agile development: A case study of three enabling mechanisms. En International conference on agile software development (pp. 216–231).
- Brown, A. W. (2011). A case study in agile-at-scale delivery. En International conference on agile software development (pp. 266–281).
- Conboy, K., y Carroll, N. (2019). Implementing large-scale agile frameworks: Challenges and recommendations. *IEEE Software*, 36(2), 44–50.
- Dikert, K., Paasivaara, M., y Lassenius, C. (2016). Challenges and success factors for large-scale agile transformations: A systematic literature review. *Journal of Systems and Software*, 119, 87–108.
- Dingsøyr, T., Falessi, D., y Power, K. (2019). Agile development at scale: The next frontier. *IEEE Software*, 36(2), 30–38.
- Dingsøyr, T., y Moe, N. B. (2014). Towards principles of large-scale agile development. En International conference on agile software development (pp. 1-8). Cham, Alemania.
- Ebert, C., y Paasivaara, M. (2017). Scaling agile. *IEEE Software*, 34(6), 98–103.
- Figalist, I., Elsnér, C., Bosch, J., y Olsson, H. H. (2019). Scaling agile beyond organizational boundaries: Coordination challenges in software ecosystems. En International conference on agile software development (pp. 189–206).
- Kalenda, M. (2017). Scaling agile software development in large organizations (Tesis de maestría). Masaryk University, Brno, República Checa.
- Laanti, M. (2014). Characteristics and principles of scaled agile. En International conference on agile software development (pp. 9-20). Cham.
- Nyffjord, J., Bathallath, S., y Kjellin, H. (2014). Conventions for coordinating large agile projects. En International conference on agile software development (pp. 58–72).