La Transformación de las Metodologías de Desarrollo y la Tendencia Ágil

Cristian D. Castaño Román, Juan C. Cadavid Ruiz, Estudiantes del programa Ingeniería de Sistemas. Fundación Universitaria Católica Lumen Gentium

Abstract—Este documento explica las metodologías de desarrollo de software y evidencia la tendencia que han adoptado recientemente algunas compañías hacia las metodologías de desarrollo ágil. Al inicio del documento se detalla el origen de la ingeniería de software y como se adoptaron técnicas de otras disciplinas para minimizar los riesgos en la construcción de este. Además, se establece un paralelo explicando en detalle las principales características de la metodología de desarrollo tradicional y ágil. Finalmente, en la última parte del documento, se hace mención a la metodología de desarrollo ágil que es tendencia hoy en día y que es comúnmente usada no solo en proyectos de software, sino también que se ha llevado a casi todas las áreas de las organizaciones que la han implementado.

Index Terms— Desarrollo de software, ciclo de vida del software, ingeniería de software, metodologías ágiles, metodologías de desarrollo, metodologías tradicionales, Proyectos, Scrum, Software.

1 Introducción

La presente investigación hace referencia al uso de las metodologías de desarrollo usadas para la elaboración de software. Las metodologías se pueden definir como el procedimiento o los lineamientos a seguir para elaborar un producto de calidad. Aunque las metodologías descritas en el documento cuentan con características diferentes; es sumamente importante que antes de iniciar la construcción del software se elija una que se ajuste al tamaño del proyecto y el número de personas que conforman el equipo de trabajo, de esta forma se maximizará el potencial y aumentará la calidad del producto con los recursos, y el tiempo establecido.

Para comprender la importancia del uso de metodologías de desarrollo de software es necesario conocer las causas que llevaron a los primeros ingenieros a adoptar metodologías de otras disciplinas como la arquitectura o la ingeniería civil, una de estas fue la crisis del software. La crisis del software suscita una serie de acontecimientos que tuvieron repercusión negativa en la construcción del mismo en las décadas del 60 hasta el 80, llevando a los primeros ingenieros a detener la construcción artesanal de software para empezar a desarrollar basados en lineamientos y pasos para la elaboración del mismo.

- *C.D. Castaño Roman, Estudiante del programa Ingeniería de Sistemas. Fundación Universitaria Católica Lumen Gentium. Santiago de Cali, Colombia. E-mail: cricasro@gmail.com.*
- *J.C. Cadavid Ruiz, Estudiante del programa Ingeniería de Sistemas. Fundación Universitaria Católica Lumen Gentium. Santiago de Cali, Colombia. E-mail: jucacar28@hotmail.com.*

Lo descrito anteriormente incrementa el interés general de realizar este trabajo de investigación, dado que se desea conocer cuán importante resulta la implementación de metodologías de desarrollo de software, así mismo en el ámbito profesional, se presenta un gran interés por saber cuáles son las metodologías más usadas, cuales traen mayores beneficios, que metodología es la mejor para el desarrollo de software y sobre todo cuales son las metodologías que se están usando en los diferentes mercados en los que se incluye software.

Partiendo de investigaciones previas realizadas por otros estudiantes y basándonos en una serie de encuestas realizadas y charlas recibidas sobre las metodologías de desarrollo tradicional y metodologías de desarrollo ágil, se emprendió el trabajo de investigación para poder resolver las incógnitas que dejaban estas fuentes, pues aunque en la actualidad, en el ámbito profesional de la ingeniería de sistemas, ingeniería informática e ingeniería del software es común escuchar palabras como Ciclo de vida en cascada, Scrum, CMMI, Lean, Ágil, XP, etc. siempre se llegaba al mismo punto de preguntarse: ¿Cuál es la tendencia de las empresas que desarrollan software hoy en día? ¿Cómo lo hacen? ¿y cómo hacen para satisfacer las necesidades de un mercado tan cambiante como el actual?

Tratando de dar respuesta a las incógnitas planteadas en el anterior párrafo, el documento se divide en 5 partes. En la primera parte del documento se detallará el origen de la ingeniería del software a través de la implementación de la primera metodología de desarrollo, seguido de ello se detallarán cómo ha evolucionado esa metodología para dar nacimiento a una nueva, partiendo las metodologías en dos: tradicionales y ágiles. En la tercera parte se mencionarán los rasgos generales que tiene cada una de

estas metodologías y se detallará Scrum, la metodología de desarrollo de software más usada recientemente. En la cuarta parte del documento se presenta un caso de estudio en el cual se evidencia el uso de las metodologías agiles, y a continuación se realiza un comparativo de la aplicación de las metodologías ágiles versus las metodologías tradicionales, mostrando el impacto que estas tienen en cuanto a tiempo y dinero.

2 Origen De las Metodologías de Desarrollo

En los años de 1960 hasta 1980 existió algo llamado la crisis del software en donde se identificaron muchos problemas en el desarrollo del mismo, todos los proyectos que incluían software sobrepasaron el presupuesto y los tiempos estimados, algunos otros causaron daños a la propiedad e incluso algunos causaron pérdidas de vidas [1]. Esto se debía a la forma artesanal de cómo se elaboraba, no había ningún método o norma que permitiera establecer pautas para esta labor; esto era diferente a lo que hacían otras ingenierías como la mecánica o la civil, por esto se decidió tomar algunas ideas de otras disciplinas y aplicarlas al software.

Así es como nace la ingeniería de software y con ella aparecieron las primeras metodologías para establecer pautas sobre cómo desarrollar software. Los primeros ciclos de vida fueron tomados de otras ingenierías estableciendo etapas como: el levantamiento de requerimientos, diseño, construcción, pruebas, instalación y finalmente una fase de mantenimiento; una vez finalizada una etapa se generaba documentación para entregar a la siguiente y se creía que una vez superada la etapa no era necesario volver a ella.

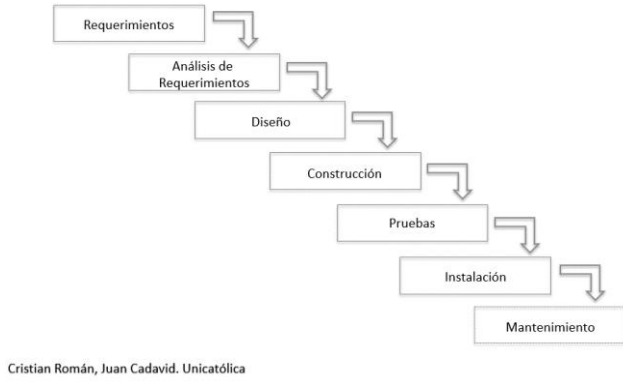


Fig. 1. Ciclo de vida en cascada.

En otro tipo de ingenierías o profesiones en las que se desarrollan proyectos que traen consigo como resultado objetos ya sea puentes, edificios o casas, es probable que sea una de las únicas formas que haya de hacerlo, dado que se basan en física, en matemáticas y requerimientos cerrados; esto indica que no pueden haber errores entre una etapa y la otra, por ejemplo: Si en el proyecto de construir un edificio se hizo mal el diseño y éste error se llevó a la siguiente etapa, puede suceder una catástrofe o que simplemente se disparen los costos porque el cambiar un elemento físico como una columna, escalera etc. es altamente costoso.

De forma similar se trató de implementar esta metodología en la ingeniería del software, incluso aún se usan estas metodologías hoy en día, y aunque han dado resultado para algunos proyectos, con el tiempo y también con las exigencias del mercado y los cambios del entorno en que se desenvuelve el proyecto (que son cada vez más frecuentes) trajo consigo inconvenientes para otros proyectos que involucraban software y que no se adaptaban a dicha metodología. Experiencias como la mencionada por Alejandra Moreno, Agile Coach de **Cencosud** [9] se habían vuelto el diario vivir de muchos ingenieros de sistemas: "Trabajé varios años con metodologías tradicionales y la experiencia no fue buena, todos los proyectos se retrasaban, teníamos muchos problemas de comunicación, los clientes estaban molestos e insatisfechos, hacíamos cosas que nadie usaba, trabajábamos desmotivados y bajo una altísima presión." Esto último se debe a que como se detallará más adelante, a las metodologías de desarrollo tradicional les cuesta bastante realizar cambios en cualquier fase del proyecto. Desde el punto de vista de la gestión de proyectos, este escenario aleja completamente al proyecto de conseguir objetivos de calidad, dado que modifica por completo todas las variables del triángulo de hierro (ver imagen Fig. 2).

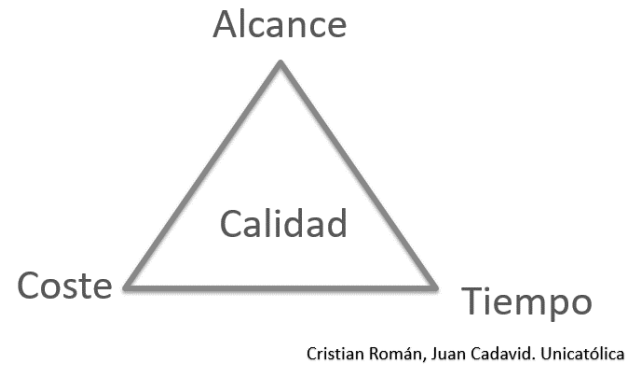


Fig. 2. Variables del triángulo de hierro.

Un proyecto exitoso no es solo el que cumple con las variables mencionadas anteriormente, dado que "el proyecto debe traer consigo un valor significativo para el usuario" [2], que le represente una mejora, una ayuda o

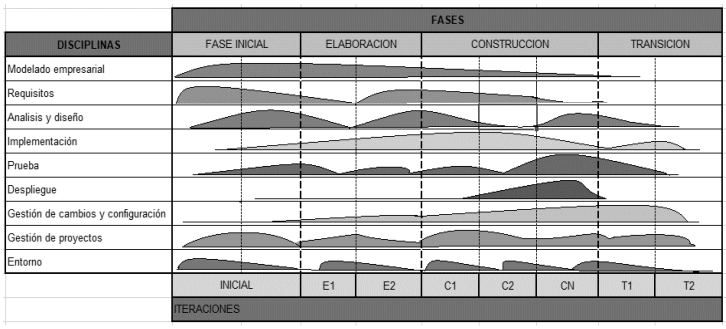
una alternativa a un proceso que demandaba mayor esfuerzo; de no ser así el software desarrollado será inoficioso para el cliente y será un proyecto que aunque se haya cumplido en el tiempo, el alcance y los costos establecidos de parte y parte, nunca se utilizará por parte del usuario final.

Para esta serie de situaciones que se presentaron surgió dentro de la ingeniería de sistemas un concepto llamado "agilidad" concebido como una reacción a la manera tradicional de construcción de software, teniendo como concepción realizar software maleable, modificable y fácilmente escalable, diferenciándose principalmente de lo tradicional en que en este se realizaran entregas tempranas de lo que el cliente requiere y no una entrega final, al término de un lapso de tiempo, como lo es la metodología tradicional.

3 METODOLOGÍAS TRADICIONALES

Las metodologías de desarrollo tradicional hacen mayor énfasis en cuanto a la estricta planificación y control del proyecto, adicional a una especificación exacta de lo que el cliente requiere, de esta forma se impone una disciplina bastante estricta en cuanto a los procesos de desarrollo de software. Una vez se encuentra detallado el requerimiento del cliente, se procede a analizar y diseñar rigurosamente una serie de roles, actividades, artefactos (diagramas UML), y documentación detallada, para entonces comenzar con el desarrollo del producto especificado inicialmente.

Todo este esfuerzo se realiza con el fin de conseguir un software más eficiente y de calidad, focalizándose en documentación, planificación y procesos. A continuación, se detalla el flujo de trabajo en una metodología tradicional como RUP (Proceso Unificado Relacional Fig. 3):



Cristian Roman, Juan Cadavid. Unicatolica

Los flujos de trabajo descritos en la imagen se dividen en dos grandes grupos: flujos de trabajo de proceso y flujos de trabajo de soporte. En los primeros está centralizado gran parte del esfuerzo del equipo de trabajo en donde se dedica mucho más tiempo al levantamiento de requerimientos, análisis y diseño e implementación que a las pruebas, estas últimas solo toman fuerza en la etapa final de la construcción. En el segundo grupo, los flujos de trabajo son mucho más bajos a excepción de la disciplina relacionada con la gestión del cambio y configuraciones en las etapas de construcción y transición, esto representa el alto impacto qué genera un cambio de cualquier índole en una de las fases ya mencionadas.

Algo que se debe tener en cuenta en este tipo de metodologías es que el cliente debe ser capaz de describir con gran detalle las funcionalidades de lo que desea para poder acordar un alcance del proyecto con él, además de que el entorno en el que se desenvuelva el proyecto influye significativamente y debe ser tenido muy en cuenta puesto que un cambio en el entorno, puede generar un cambio en el proyecto. Esto puede llegar a concluir que todo el esfuerzo en la planificación del proyecto se ha perdido y es necesario entonces modificar el alcance, el tiempo y seguramente aumentar los costos tanto para el cliente como para el equipo de trabajo.

4 METODOLOGÍAS AGILES

Las metodologías de desarrollo ágil han surgido como respuesta a los problemas detallados anteriormente y surgió en gran medida para cubrir necesidades de proyectos de desarrollo de software en la que aplicar las metodologías tradicionales se equipararía con “matar moscas a cañonazos” [3].

En febrero del 2001, 17 desarrolladores de software se reunieron en el Snowbird Ski & Summer Resort en Utah, Estados Unidos en donde crearon el Manifiesto Ágil y establecieron: "Estamos descubriendo formas mejores de desarrollar software tanto por nuestra propia experiencia como ayudando a terceros. A través de este trabajo hemos aprendido a valorar:

Individuos e interacciones, sobre procesos y herramientas.

Software funcionando, sobre documentación extensiva.

Colaboración con el cliente, sobre negociación contractual.

Respuesta ante el cambio, sobre seguir un plan.

Esto es, aunque valoramos los elementos de la derecha, valoramos más los de la izquierda." [4].

Para que esta metodología funcione se debe establecer primero un cambio de pensamiento organizacional en donde se afiance la cultura, estilos de liderazgo, creencias etc. este cambio de pensamiento permitirá al equipo el cumplimiento de los valores establecidos en el manifiesto, guiados por los siguientes principios:

1. Satisfacer al cliente: Se proyecta realizar entregas tempranas y continúas del software que le den valor al producto desde etapas tempranas.

2. Aceptación de cambio en los requerimientos: Aun en etapas tardías de desarrollo se aceptan cambios en los requerimientos pues esto le da al cliente ventajas competitivas en el mercado.

3. Entregas funcionales frecuentes: Se establece entregas tempranas entre dos semanas y dos meses siempre apuntando al periodo de tiempo más corto establecido.

4. Ritmo sostenible: Los responsables del negocio y los desarrolladores trabajan juntos de forma cotidiana durante todo el proyecto, no hay picos de esfuerzos en algunas fases y en otras no.

5. Equipos motivados y auto organizados: Se le da el apoyo al equipo de trabajo y se le da confianza en la ejecución del trabajo.

6. Comunicación directa: Se establecen reuniones presenciales para comunicar información al equipo de desarrollo.

7. Software funcionando: No se hacen entregas inoficiosas del software, el cliente en cada entrega podrá manipular correctamente software para la mejora del mismo en la siguiente iteración.

8. Desarrollo sostenible: Software y equipo de trabajo altamente adaptado a los cambios en cualquier etapa del desarrollo.

9. Excelencia técnica: Se cuenta con un equipo con conocimientos técnicos bastante fuertes que permiten la elaboración de software de calidad.

10. Simplicidad: Aprovechamiento del tiempo y los recursos a la hora de realizar una labor, relacionada con el proceso de desarrollo.

11. Equipos auto-organizados: Generan las mejores arquitecturas, diseños y requisitos.

12. Reflexión y adaptación en equipo: Posibilidad de revisar las falencias de las individualidades para convertirlas en oportunidades de mejora como equipo.

Cuando se ha logrado el cambio de pensamiento organizacional en la compañía o cuando se cuenta con un equipo comprometido bajo estos lineamientos de agilidad es posible manifestar esta metodología a través de prácticas como lo son SCRUM, XP, LEAN, CRYSTAL, KANBAN... entre otras. Se hace tanto énfasis en el cambio de pensamiento dado que es lo más importante pues "ágil" es una forma de pensar y una forma diferente de trabajar para aportar valor al negocio como lo conocíamos [2]. Sin importar cuál sea la práctica que se elija para implementar metodologías de desarrollo ágil cada una de ellas está orientada a la satisfacción del cliente y la adaptabilidad del software en cualquiera de las fases de su ciclo de vida. La planificación adaptativa le permite al equipo y al cliente estar preparado al cambio, por lo que en cuanto se haga una entrega parcial del software el cliente tendrá resultados anticipados (time to y un retorno de la inversión (ROI) desde muy temprano en el proyecto.

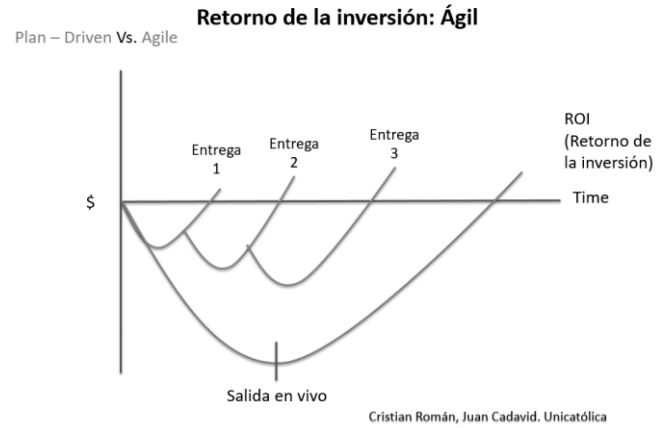


Fig. 3. Tradicional vs Ágil.

Como se observa en la imagen, en cada una de las entregas que se realiza del proyecto el cliente recupera parte de la inversión efectuada dado que cada entrega es completamente funcional, minimizando el riesgo desde el inicio, hasta la entrega final del proyecto.

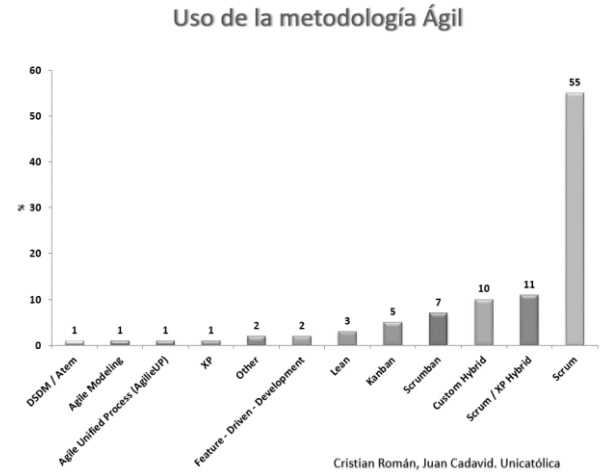


Fig. 4. Uso de la metodología Ágil.

De entre todas las diferentes metodologías y marcos de referencia para aplicar metodologías ágiles, Scrum y las que aplican alguna variación de Scrum junto con otras (xp, kanban), conocidas popularmente como metodologías hibridas de desarrollo ágil, son las más populares y en la

actualidad se usan no solo en empresas como casas de software, sino que algunas otras compañías prestadoras de diferentes servicios han adoptado prácticas de estas metodologías aplicándolas a diversas áreas de la que no necesariamente tienen que ver con sistemas.

A continuación, detallaremos Scrum, la metodología de desarrollo ágil más popular usada recientemente.

5 SCRUM

Es un proceso en el cual se aplican una serie de buenas prácticas para trabajar colaborativamente, en equipo y obtener el mejor resultado posible de un proyecto. De acuerdo a su definición, Scrum es más un framework que una metodología como tal, es por esto que es posible tomar ciertas prácticas para aplicarlas a diversos ámbitos de una compañía.

En Scrum se realizan entregas parciales y regulares, pero sobre todo funcionales de lo que es el producto final, priorizadas primero por el beneficio que le aportan al o los usuarios finales y todas las partes interesadas del proyecto; maximizando la utilidad de lo que se construye y el retorno de la inversión, así, cuando se realizan estas entregas, se mitiga altamente uno de los riesgos principales en un proyecto: Entregar al cliente lo que no necesita. Por otra parte, cuando se debe trabajar bajo entornos en los que los requisitos son altamente variables o que simplemente no se encuentran bien definidos, en donde el mercado exige innovación, competitividad, flexibilidad y sobre todo productividad, se puede adaptar en tiempo real el producto que se está construyendo a las necesidades del cliente. Sin embargo es importante establecer una diferencia entre lo que es la metodología de desarrollo ágil o framework Scrum para el desarrollo y la gestión de proyectos bajo Scrum, pues en la primera el equipo se focaliza en una única cosa: construir software de calidad, mientras que, la gestión de un proyecto Scrum se focaliza en definir cuáles son las características que debe tener el producto a construir (qué construir, qué no y en qué orden) y en remover cualquier obstáculo que pudiera entorpecer la tarea del equipo de desarrollo [5].

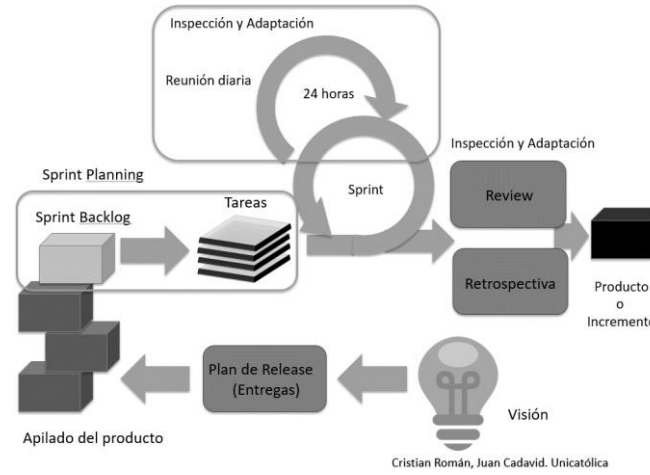


Fig. 5. Esquema de trabajo SCRUM.

Bajo un esquema de trabajo Scrum es importante que estén bien definidos los roles y el objetivo principal en esta definición es mantener comprometidos a los miembros del equipo, al Scrum Master (persona conocedora del proceso Scrum que se encarga de orientar al equipo y al propietario del producto) y al dueño del producto, además de involucrar a usuarios, al negocio y a todas las partes interesadas del proyecto.

A continuación, describiremos el proceso y los beneficios que trae usar la metodología Scrum.

5.1. El Proceso

Fig. 6. Proceso de la metodología Scrum [11].

Cuando se inicia un proyecto y se opta por usar la metodología Scrum primero se debe crear una lista de requerimientos la cual debe ser organizada en orden de prioridad, dándole más importancia a los requerimientos que sean más urgentes y que tengan un mayor impacto para el usuario final.

Cuando ya se tiene construida la lista de requerimientos se procede con las siguientes tres actividades:

1. **Planificación y Organización**

La planificación de la Iteración (Sprint) se debe crear en presencia de todo el equipo de trabajo con el fin de que todos estén enterados de los lineamientos del proyecto y los más importante en esta fase es que se deben responder dos preguntas que son fundamentales, la primera es ¿Que se va a entregar? y la segunda es ¿Cómo se va a ejecutar?

2. **Seguimiento y Dirección**

En esta fase se da seguimiento diario de las diferentes actividades realizadas por cada integrante del equipo de trabajo el cual expone los avances que ha tenido y las dificultas que se le han presentado para que en cabeza del Scrum master se plantee una solución la cual involucre a todos los miembros del equipo de trabajo garantizando siempre la productividad, estas actividades son plasmadas en un tablero físico (Tablero scrum), a partir de estas reuniones diarias cada miembro debe tener claridad cuáles van a ser las actividades que va a realizar y que actividades deben ser modificadas.

3. **Revisión**

En la fase de revisión se debe garantizar de que las actividades desarrollas no tengan ninguna modificación y se debe revisar tanto por el equipo de trabajo como por el cliente.

Con base en estos resultados se toman acciones correctivas las cuales se priorizan y se incluyen nuevamente en la lista de trabajo.

5.2. Beneficios

De acuerdo a la investigación realizada [10], se hace una selección de los principales beneficios que se tienen al usar la metodología Scrum, los cuales dan un valor agregado ya que permiten optimizar procesos de una forma ágil y segura, a continuación, detallaremos cada beneficio:

A. **Productividad:** Cabe destacar que la productividad de los integrantes del equipo es altamente efectiva debido al seguimiento y actualización que se realiza diariamente de cada una de las actividades permitiendo resolver los problemas que se presenten en cada una de estas de forma efectiva cumpliendo con los objetivos propuestos en el sprint.
B. **Flexibilidad:** Este punto es importante destacarlo ya que el cliente puede participar en la iteración de la actividad realizada, evaluándola y exponiendo sus argumentos para que el desarrollo cumpla con el objetivo planteado, estas actividades pueden ser reorganizadas para cumplir con los lineamientos del cliente.
C. **Cumplimiento:** Este beneficio se da ya que el cliente establece los objetivos que requiere con el desarrollo y mediante entregas cortas que realiza el equipo de desarrollo el cliente puede obtener resultados sin tener que esperar a que la solución esté terminada en su totalidad.
D. **Reducción de riesgos:** En cada una de las reuniones (Sprint) cada miembro socializa las actividades realizadas y de esta forma todo el equipo está alineado y entre todos se pueden detectar problemas con mayor facilidad y de esta forma se pueden resolver de manera efectiva lo que implica que a futuro se reduzcan riesgos en el desarrollo planteado, esto hace que se genere un valor agregado hacia el cliente.
E. **Comunicación Efectiva:** Uno de los puntos destacados en la metodología Scrum es la comunicación efectiva entre cada uno de los miembros del equipo de desarrollo y cliente. Esto se evidencia en cada una de los Sprint diarios realizados por el equipo y en cada una de las entregas cortas que se realizan al cliente, donde todos los miembros implicados están al tanto de cada uno de los movimientos que se deben realizar para llevar a cabo de un desarrollo del proyecto de forma efectiva.
F. **Motivación:** Este punto es importante ya que las personas implicadas en el proyecto pueden mostrar su creatividad resolviendo cada una de las actividades planteadas lo que hace que estos tengan mayor libertad para expresar sus ideas y puedan administrar el tiempo que van a invertir en el desarrollo de estas de forma más efectiva, respetando siempre los tiempos de entrega.

5.3 Caso de éxito

De acuerdo a los beneficios que trae usar la metodología Scrum, varias empresas de diversos sectores económicos ya están implementando la metodología en diferentes procesos, no solamente en procesos relacionados con el desarrollo de software, lo que hace que esta metodología tenga una gran adaptabilidad a diferentes entornos y condiciones de las empresas. A continuación, se describirá un caso de éxito presentado en la compañía Spotify [14] donde se podrá ver con más detalle cómo funciona la metodología Scrum y que beneficios ha traído para esta compañía.

En la actualidad Spotify es la empresa número uno en el negocio de música por Streaming, y cuenta con más de 10 millones de canciones con las cuales el usuario final puede interactuar ya sea a manera de compra o simplemente escuchándolas como una lista de reproducción. Esta plataforma está integrada con redes sociales lo que permite

la interacción con amigos, además de que cuenta con diferentes planes que tienen costos de acuerdo a las preferencias de los usuarios en cada uno de los más de 17 países donde tiene presencia.

Ahora veamos cual es la relación que tiene Spotify con la metodología Scrum, el negocio está dividido en pequeños grupos que funcionan como pequeñas empresas independientes, donde cada una tiene sus propias funciones específicas y trabajan desarrollando "productos mínimos viables" [15]. Cada grupo realiza entregas frecuentes y cada grupo cuenta con un líder el cual se encarga de la interacción con los otros grupos de trabajo, todo funciona como un engranaje. Ahora bien, la metodología Scrum aporta a cada grupo autonomía absoluta (lo que no quiere decir que los grupos trabajen de manera independiente ya que se establecen objetivos en común) pero debe adecuarse para conseguir los fines propuestos.

Para que la aplicación de la metodología Scrum funcione de manera correcta, Spotify aporta tres componentes claves a sus equipos de trabajo que son la confianza, el compromiso y la colaboración.

La confianza es un punto clave para que la metodología se adapte en la organización ya que esta aporta autonomía a los grupos de trabajo lo que hace que las personas que integran el proyecto se encuentren motivadas, centrándose en cada una de las tareas propuestas y de esta forma se le da un reconocimiento a su labor.

En cuanto al compromiso este se da debido a la responsabilidad que tiene cada equipo de trabajo para cumplir con los objetivos propuestos de forma eficiente, respetando los tiempos establecidos.

Por otro lado, está el factor de la colaboración que se da mediante la interacción entre cada grupo de trabajo con el firme objetivo de sacar el proyecto adelante.

Con base al anterior ejemplo se puede observar como la metodología Scrum es aplicada y modelada de forma efectiva en Spotify, y que a pesar del gran crecimiento de esta compañía el uso de la metodología ha impactado cada una de sus áreas; lo que ha desencadenado el cumplimiento de la gran demanda que tienen de manera efectiva siempre dándole valor a sus clientes. El uso de esta metodología no solo ha impactado su agilidad para el desarrollo de nuevos requerimientos si no que ha impactado en toda la operación de la compañía, que hoy en día requiere de mucha agilidad para cumplir con las exigencias de los usuarios y, esto al final se ve reflejado en sus estados financieros donde las ganancias año tras año se incrementan.

Adicional al ejemplo anterior, se detallarán algunas compañías que están usando la metodología Scrum.

TABLA1 [12].
EMPRESAS QUE USAN LA METODOLOGIA SCRUM

Sector Económico	Nombre de Empresa
Media y Telecom	BBC, BellSouth, British Telecom, DoubleYou, Motorola, Nokia, Palm, Qualcomm, Schibsted, Sony/Ericsson, Telefonica I+D, TeleAtlas, Verizon
Software, Hardware	Adobe, Austenite, Biko2, Central Desktop, Citrix, Gailén, IBM, Intel, Microfocus, Microsoft, Novell, OpenView Labs, Plain Concepts, Primavera, Proyectalis, Softhouse, Valtech, VersionOne.
Internet	Amazon, Google, MySpace, Yahoo!
ERP	SAP
Banca e Inversión	Bank of America, Barclays Global Investors, Key Bank, Merrill Lynch
Sanidad y Salud	Patientkeeper, Philips Medical
Defensa y Aeroespacial	Boeing, General Dynamics, Lockheed Martin
Juegos	Blizzard, High Moon Studios, Crytek, Ubisoft, Electronic Arts
Otros	3M, Bose, GE, UOC, Ferrari

6 Tradicional vs Ágil

Cuando se realiza una comparación entre las metodologías Tradicionales y las metodologías ágiles encontramos varias diferencias debido al enfoque de cada una, a continuación, se detallarán las diferencias más significativas entre las dos metodologías:

TABLA2.
EDIFERENCIAS EN EL CICLO DE DESARROLLO DE SOFTWARE

Ciclo/ Desarrollo de Software	Metodologías Tradicionales	Metodologías Agiles
Análisis de requerimientos	Se realiza el levantamiento de información mediante un documento estandarizado en el cual se relacionan los requerimientos.	Se crea una lista de requerimientos donde se categorizan por orden de prioridad, esta lista es modificable tanto por el equipo como por el cliente.

Diseño	El diseño es flexible pero extenso, la documentación se presenta de manera rigorosa y exhaustiva.	Diseño simple y fácil de modelar, la documentación es mínima, se le da prioridad a la comunicación.
Desarrollo	El desarrollo se maneja de forma individual, a cada miembro del equipo de trabajo se le entrega una tarea específica. Hasta que el desarrollo no esté completo no se entrega.	El desarrollo se maneja de forma colectiva, donde se presenta una gran transferencia de conocimiento para todos los miembros del equipo. Se realizan entregas periódicas.
pruebas	Las pruebas se realizan por desarrollos individuales, donde cada miembro se responsabiliza por su parte.	Se involucra a todo el equipo de trabajo en conjunto con el cliente y se realizan pruebas cortas pero de gran calidad.
Mantenimiento	El mantenimiento es exhaustivo ya que al no involucrar al cliente en los procesos de desarrollo y a hacer entregas tan largas pueden surgir errores que pueden cambiar el sistema.	El mantenimiento es mínimo ya que se mitigaron todos los errores posibles a lo largo del ciclo de vida del proceso, gracias a la comunicación efectiva y a las entregas frecuentes.

La principal diferencia entre ambas metodologías, es que en las metodologías agiles la iteración es un proceso fundamental y está inmerso en su ciclo de vida, esta característica es muy importante puesto que acorta el tiempo de ejecución de un proyecto estableciendo una comunicación efectiva entre sus partes impactando directamente en la curva de aprendizaje haciendo que el proyecto sea más efectivo y cumpla con los requerimientos del cliente, asegurando el proceso y generando un menor mantenimiento del producto terminado a futuro.

Por otro lado, encontramos que los procesos de mantenimiento y de enseñanza están estructurados de una manera más completa en las metodologías tradicionales, si bien el proceso tradicional es más lento tiene sus ventajas ya que se constituye desde su inicio de una manera más robusta y esto se ve reflejado en su documentación minuciosa y en sus procesos de enseñanza.

Fig. 7. Metodologia Ágil versus Metodologia tradicional

7 CONCLUSIONES

Con base en lo planteado en este documento sobre las metodologías tradicionales y las metodologías ágiles, teniendo en cuenta el caso de estudio de Spotify, y la evolución constante de nuestro entorno que ha desencadenado que las empresas desarrolladoras de software deban satisfacer la demanda de requerimientos que ejercen los usuarios finales de manera ágil y efectiva, acortando los tiempos para la consecución de objetivos en proyectos relacionados con el desarrollo de software y cumpliendo con los mismos estándares de calidad, se concluye que la mejor forma de soportar los nuevos lineamientos del entorno es el uso de metodologías ágiles sobre el uso de las metodologías tradicionales, puesto que este tipo de metodologías aparte de que otorgan una comunicación efectiva entre sus partes y dan flexibilidad a los desarrollos para que estos sean modificables y fácilmente escalables, también garantizan en el proyecto calidad, estabilidad en costos y cumplimiento de tiempos que necesitan los usuarios finales.

REFERENCIAS

[1] therac-25, disponible en: http://wikipedia.orgwikiTherac-25

[2] Miguel Viejo Maestre, Singergía Soluciones Software, Guayaquil, Ecuador. Noviembre 2017.

[3] Javier Moya, disponible en: https://pmi-mad.org/index.php/socios/articulos-direccion-proyectos/1288-metodologia-agil-vs-metodologia-tradicional

[4] Manifestó Ágil, disponible en: http://agilemanifesto.org/iso/es/manifesto.html

[5] ¿Proyectos complejos? ¿Necesitas resultados? Conoce Scrum. Disponible en: https://proyectosagiles.org

[6] Javier Garzás, CEO Kybele Consulting. Velneo, Life is soft, enero 2013, disponible en: https://www.youtube.com/watch?v=L9qlrgKWqfI

[7] Roberth G. Figueroa, Camilo J. Soils, Armando A. Cabrera, METODOLOGÍAS TRADICIONALES VS. METODOLOGÍAS ÁGILES, disponible en: https://engenhariasoftwareisutic.files.wordpress.com/2016/05/metodologias_tradicionales_vs-_metodologias_agiles.pdf

[8] Beneficios de la metodología, disponible en: https://proyectosagiles.org/beneficios-de-scrum/

[9] Historia Cencosud disponible en: https:// tiendasjumbo.co/cencosud

[10] Introducción a scrum y por qué debes usarlo, disponible en: https://highscalability.wordpress.com/2010/01/24/introduccion-a-scrum-y-por-que-debes-usarlo/

[11] ¿Para qué sirve el scrum en la metodología Ágil?, disponible en:

[12] Empresas que usan la metodologia Scrum, disponible en: http://www.i2btech.com/blog-i2b/tech-deployment/para-que-sirve-el-scrum-en-la-metogologia-agil/

[13] Historia de la metodología Scrum, disponible en: https://proyectosagiles.org/historia-de-scrum/

[14] Acerca de Spotify, disponible en: https://www.spotify.com/co/about-us/contact/

[15] Como Spotify organiza, de manera ágil, su departamento de desarrollo, disponible en: http://www.javiergarzas.com/2012/11/spotify-agil-desarrollo.htm

Cristian David Castaño Román, Estudiante de ingeniería de sistemas de último semestre de la Fundación Universitaria Católica Lumen Gentium, apasionado por el autoaprendizaje, Certificado en redes por D- link Latinoamérica y en gerencia de proyectos por la institución educativa politécnico de Colombia, quien actualmente labora para la compañía Plataforma Tecnológica S.A.S bajo el cargo de ingeniero de implementación y soporte desde hace 1 año y 4 meses.

Juan Carlos Cadavid Ruiz, Estudiante de ingeniería de sistemas de la Fundación Universitaria Católica Lumen Gentium, se ha desempeñado como en los cargos de Jefe de Sistemas en la empresa Almacenes La 13 S.A desde el año 2010 al 2014, trabajo en el centro de servicios informáticos de la Pontificia Universidad Javeriana Cali desempeñando el cargo de supervisor de primera línea dese el año 2014 al 2016, posterior a eso trabajo para Carvajal Servicios desempeñando el cargo de Ingeniero de implementación desde año 2015 al 2017 donde creo el software contable para que integra el sistema de nómina con la contabilidad de las empresas externas al grupo Carvajal, actualmente se desempeña como Specialist Payment para la empresa Price Smart.