



UNIVERSIDAD TÉCNICA DE BABAHOYO

FACULTAD DE ADMINISTRACIÓN, FINANZAS E INFORMÁTICA

PROCESO DE TITULACIÓN

MAYO – SEPTIEMBRE 2019

ESTUDIO DE CASO

PREVIA A LA OBTENCIÓN DEL TÍTULO DE

“INGENIERO EN SISTEMAS”

TEMA:

ANÁLISIS DE LAS METODOLOGÍAS ÁGILES DE DESARROLLO SOFTWARE

EGRESADO.

MORAYMA JENISER ORAMAS MARQUEZ

TUTOR:

ING ANA DEL ROCIO FERNANDEZ TORRES

BABAHOYO – LOS RÍOS

AÑO: 2019

INDICE GENERAL

INTRODUCCIÓN.....	3
DESARROLLO.....	5
CONCLUSIONES.....	26
RECOMENDACIONES	28
BIBLIOGRAFÍA	29
ANEXOS	30

INTRODUCCIÓN

En el presente caso de estudio es un análisis de las diferentes metodologías ágiles de software que permite comparar que metodología es más conveniente para el desarrollo de software en las empresas, por lo cual se ha basado la línea de investigación de la Universidad Técnica de Babahoyo que especifica el desarrollo de Sistemas de la información, comunicación y emprendimientos empresariales y tecnológicos, alineándose a esto es que se ha escogido el tema a tratar y su sublínea es el desarrollo de Sistemas Informáticos.

Se ha determinado para este caso que el método de investigación es deductivo porque va de lo general a lo particular, esto quiere decir que todos los datos secundarios que se van a recolectar son necesarios para conocer cuál es la mejor metodología ágil de desarrollo de software que existen por medio de un análisis que se va a realizar, determinando la mejor para que las empresas la puedan aplicar y ahorren tiempo en los procesos que realizan.

Las metodologías ágiles de desarrollo de software es una nueva tendencia que se ha implementado no hace mucho tiempo, debido a esto muchas empresas la ignoran porque se basan en las metodologías tradicionales y otras ya están utilizando estas metodologías, pero no tienen claro cuál de ellas es la mejor para su uso con el análisis que se realizará se quiere evidenciar cuál es la más factible. También estas Metodologías Ágiles de Desarrollo Softwares son una herramienta que se aplica para el análisis de trabajos de codificación de software, que permite la integración de equipos de trabajo, de esta forma se distribuyen tareas para la solución eficaz de los softwares a desarrollar, las necesidades evolucionan de acuerdo a las exigencias del cliente. Una característica principal de las metodologías ágiles es la

participación activa del cliente en el desarrollo del software. Las metodologías ágiles hacen hincapié en las comunicaciones cara a cara en vez de la documentación.

Lo que se quiere destacar es que las metodologías ágiles de desarrollo de software, son adecuadas de acuerdo a sus funcionalidades que permiten lograr una interacción ente la empresa, el desarrollador y el agente que solicita el servicio y/o producto. Esta metodología ágil de desarrollo de Software es un campo abierto que permite un desarrollo amplio de la estructura del software adaptando los requerimientos sugeridos por el cliente, además que se distribuye la codificación en grupos y el trabajo se agiliza, no tiene una estructura establecida como las metodologías tradicionales.

DESARROLLO

Es un estudio de las mejores herramientas aplicable debido a la complejidad del desarrollo de los softwares, los procesos tradicionales de las tecnologías de los softwares eran muy extensos y requerían de mucho tiempo para realizar las tareas asignadas en una empresa, generando un gran costo. Es importante concluir que estos métodos comunes han sido reemplazados debido a su funcionabilidad compleja.

En el mundo de la programación de lenguajes el desarrollo del software se requiere las aplicaciones de herramientas factibles y precisas para su ejecución para que su funcionalidad sea más accesible por el operador de sistemas, mediante la evaluación de los métodos analizados rigurosamente se determinará la mejor opción.

METODOLOGÍA

(Pantaleo & Rinaudo, 2016) Una metodología es un marco de trabajo que puede ser utilizado como guía de las actividades al llevar a cabo. Por lo tanto, una metodología de desarrollo de software no es más que una forma de trabajo para desarrollar software, donde se especifica las tareas a llevar a cabo, los artefactos a generar y las relaciones entre ambas.

Este concepto especifica las tareas a realizar de una manera detallada, los procesos a seguir, los requerimientos de la empresa la cual solicita soluciones informáticas, así se garantiza un resultado eficaz en menor tiempo y espacio posible.

DESARROLLO DE SOFTWARE

(Laínez Fuentes, 2015) “El desarrollo de software es una actividad diferente de toda la industria construida desde los tiempos de la revolución industrial.” (pág. 6) Según mi criterio porque rompe los paradigmas en la codificación de los softwares para darle simplicidad a los proyectos a realizar olvidándose de los procesos complejos que tardaban en ejecutarse.

DIFERENCIAS ENTRE METODOLOGÍAS CONDUCCIDAS POR LOS PLANES Y METODOLOGÍAS ÁGILES

(Pantaleo & Rinaudo, 2016) Las diferencias entre las Metodologías conducidas por los planes y las Metodologías Ágiles

En el presente cuadro se puede apreciar que la diferencia de estas dos metodologías muestra la opción del porque escoger las Metodologías Agiles, ya que son adaptables a las personas, a los cambios e interactúa el programador con los clientes, es una herramienta muy accesible sin determinaciones drásticas al momento de realizar un proyecto de codificación de software para una empresa.

PARA ENTENDER MEJOR LAS METODOLOGÍAS CONDUCCIDAS SE ANALIZARÁ LAS SIGUIENTES:

METODOLOGÍAS CONDUCCIDAS POR LOS PLANES

(Pantaleo & Rinaudo, 2016) Previamente mencionado un listado cronológico de las metodologías conducidas por los planes. A continuación, los modelos genéricos en los cuales estas se basan.

- Cascada: Es un modelo de desarrollo lineal secuencial. El proyecto de software es dividido en fases que deben procederse en forma secuencial. Estas fases han servido de base para la definición de otras metodologías
- Prototipado: Es un modelo de desarrollo iterativo en el cual se desarrolla una maqueta del producto. Esta maqueta o prototipo desarrollada por el equipo de proyecto y refinada junto al cliente permite idealmente especificar los requerimientos del producto. Inicialmente se comienza con la definición de alto nivel de los requisitos junto al cliente.
- DRA (Desarrollo Rápido de Aplicaciones): Este método, también conocido como RAD (en inglés Rapid Application Development), implica un proceso de desarrollo lineal secuencial llevado a cabo en paralelo por distintos equipos de trabajo. Cada uno de estos equipos de trabajo será responsable por una parte de las funcionalidades del proyecto. El objetivo es crear en un periodo corto de tiempo un producto final. Asimismo, el modelo se basa fuertemente en la reutilización de componentes y en caso de que esto no sea posible, en desarrollar componentes reutilizables.
- Incremental: Las metodologías mencionadas hasta aquí no consideran la evolución del producto debido al tiempo. En contraposición, el modelo incremental permite el aumento iterativo de la funcionalidad del producto. Será conveniente utilizar este modelo cuando el producto a construir no esté definido completamente, cuando haya un gran número de requerimientos.
- Espiral: Este modelo combina los modelos de cascada y la esencia iterativa del prototipado descriptos previamente. Por lo tanto, es una metodología iterativa incremental.

En las empresas cada vez que aplica un modelo este va a desempeñar y ejecutar un rol importante para realizar el proceso de la construcción de los productos:

Cascada: Es un modelo secuencial, el proyecto de software es dividido en fases para la facilidad de un trabajo más comprensible y fácil de desempeñar para determinar las obligaciones con mayor precisión.

Prototipado: Este modelo realiza una estructura o modelo del producto presentada por el grupo de trabajo detalla específicamente los requerimientos del producto, muestra las expectativas que tiene el cliente este es el objetivo principal del prototipado.

DRA (Desarrollo Rápido de Aplicaciones): Este modelo se establecen dos equipos de trabajo para realizarlo cada uno es responsable del producto terminado, es una herramienta que se utiliza para ahorrar tiempo este tipo de trabajo requiere que se utilicen productos reutilizables o crear, es un desarrollador rápido de aplicaciones como su nombre lo indica.

Incremental: Esta metodología estudia la evolución del producto en dirección al tiempo. En este tipo de metodología se aplica cuando no están definidos los requerimientos del proyecto porque ayuda a dar una orientación de lo que se requiere proyectar para la debida solución.

Espiral: En este método aplicamos una combinación para ejecutar la solución de nuestros proyectos, cada uno de los métodos tienen herramientas específicas que contribuyen con el desarrollo de los productos para la calidad de proceso final que se requiere obtener.

ANALISIS DE LAS METODOLOGÍAS ÁGILES

(Pantaleo & Rinaudo, 2016) El modelo es evolutivo, iterativo y trabaja con timeboxed. Cuando decimos que el modelo de desarrollo trabaja con timeboxed nos referimos a que las

iteraciones poseen tiempos fijos. La planificación es adaptativa y los entregables evolutivos. Esto permite brindar valor al negocio en forma rápida y adaptarse fácilmente a los cambios. Además, estas prácticas existen otras. Todas se basan en valores ágiles presentados previamente

La metodología ágil permite una planificación configurable a los sistemas y a su evolución, ayuda a realizar con agilidad los trabajos. Las Metodologías Ágiles es un modelo que ha generado un gran cambio en el desarrollo de los softwares, contribuyen fundamentalmente a los procesos simplificados.

CARACTERÍSTICAS DE LAS METODOLOGÍAS ÁGILES

(Laínez Fuentes, 2015) Las principales características de las metodologías ágiles son:

- Satisfacer al cliente entregando, rápidamente y a menudo, sistemas que tengan valor
- Proyectos iterativos e incrementales.
- Los requisitos no se especifican en detalles en el inicio del proyecto. La solución evolucionará a partir del feedback del cliente/ usuario.
- Reconocer si las estimaciones a largo plazo son graseras e imprecisas. Estas evolucionan a lo largo del proyecto.
- El esfuerzo de planificación se concentra en el futuro más próximo, es decir, en el corto plazo.
- El equipo de desarrollo se auto-organiza y participa en la planificación del proyecto.
- Poca formalidad, es decir, los roles en la jerarquía se simplifican.
- El cliente participa activamente del proceso.

- Trabaja con líderes en vez de con jefes.

Las características de las metodologías ágiles especifican brevemente y dan a entender al desarrollador los pasos que debe seguir para mejor aplicación y ejecución de los trabajos. Las metodologías Ágiles no requieren de una formalidad estricta en la jerarquía, los clientes son partes fundamentales del proceso porque el desarrollo del trabajo los realizan líderes.

Las metodologías más conocidas son:

Extreme Programming (XP):

(Laínez Fuentes, 2015) Concebida para proyecto de pequeño y medio tamaño, prevé participación activa y frecuente con el cliente. Describe detalladamente las prácticas de desarrollo que se van a utilizar como, por ejemplo: Lenguaje de programación, refactorización, pruebas unitarias, etc.

(Pantaleo & Rinaudo, 2016) Su nombre se promueve la disciplina extrema de cada una de las personas que componen el grupo de desarrollo en la realización de un conjunto de prácticas que son consideradas esenciales.

La metodología ágil Extreme Programming (XP) está designada a un campo más práctico y accesible, numérico para el manejo de softwares a desarrollar, el trabajo se facilita al desempeñar funciones que no requieren de componentes complejos. Por la determinación que nos brindan los autores exponentes.

Prácticas

(Pantaleo & Rinaudo, 2016) Las prácticas son las actividades llevadas adelante por los miembros del grupo del proyecto. Se dividen en dos categorías:

Prácticas Primarias: son aquellas aplicables a cualquier actividad del proyecto y las que se recomienda empezar.

Prácticas Corolarias: prácticas que se recomiendan adoptar una vez que se haya adquirido madurez en las Prácticas Primarias

Flujo de Trabajo

(Pantaleo & Rinaudo, 2016) En el aspecto estrictamente técnico se le suma a este proceso el diseño incremental y conducido por las pruebas [10],[11], los cuales hacen que los desarrolladores estén inmersos en un proceso de aprendizaje continuo del negocio y elaboren incrementalmente la solución. Esto consiste en la construcción de tests que plantean el modo de uso de los diferentes componentes previamente a su construcción. A partir del desarrollo de los tests, según este enfoque, se desarrolla el software de manera que las pruebas sean ejecutadas en forma exitosa.



Figura 1: Esquema conceptual que muestra las prácticas emblemáticas de XP

Fuentes: Pantaleo & Rinaudo (2016)

Cuando se habla de XP es un proceso sistemático de integración laboral continua donde se planifica el lugar de trabajo, espacio de trabajo para un mejor desarrollo y comodidad. En el sistema el proceso integral determina un constante aprendizaje, los tests forman un papel fundamental que da como resultado la eficiencia de la codificación del software.

Scrum:

(Laínez Fuentes, 2015) “Comparte muchas características con XP, como la participación activa del cliente, pero tiene su principal foco en las actividades de gestión del proyecto”.

(Pantaleo & Rinaudo, 2016) Esta metodología debe su nombre a la formación de rugby donde los jugadores miembros de un equipo se agrupan y suman sus fuerzas empujando al obstáculo que en ese caso es el equipo adversario. Esta forma de trabajo ataca los problemas asociados a la gestión de proyectos y como tal necesita de un complemento que se ocupe de los aspectos técnicos.

(Tardivo & Caymes-Scutari, 2016) Scrum es una metodología ágil que se basa “en la teoría de control de procesos empírica o empirismo. El empirismo asegura que el conocimiento procede de la experiencia y de la toma de decisiones basándose en lo que se conoce. Scrum emplea un enfoque iterativo e incremental para optimizar la predictibilidad y el control del riesgo” [10].

Los enunciados destacan la gestión de procesos de forma compartible, cada uno desempeña el respectivo rol. Cabe recalcar que unos de los conceptos destacan lo empírico que se basa en el conocimiento de las experiencias vividas dando un enfoque más preciso al

desarrollo de los softwares, evita mayores errores mediante la construcción del trabajo. Scrum es una metodología aplicada a la interacción que previene riesgos de gran magnitud.

CARACTERÍSTICAS DE SCRUM

(Salazar, Tovar, Linares, Lozano, & Valbuena, 2018) Scrum es una metodología para el desarrollo de software iterativa e incremental, debe su nombre a la jugada de rugby llamada de la misma manera, se dice que es iterativa ya que se ejecuta en bloques temporales cortos y fijos (de no menos de dos semanas) que reciben el nombre de sprints y es incremental en tanto se obtienen funcionalidades del producto final al terminar cada iteración.

Dentro de las características de Scrum, se relaciona que está definido en base a roles, reuniones y artefactos, los cuales se describen a continuación.

Roles

La figura del director del proyecto no está definida en Scrum, pero sus responsabilidades están divididas en tres roles fundamentales:

- El dueño de producto o product owner que administra el producto y la finalidad de este es quien conoce la finalidad del proyecto, es quien interactúa con el cliente y se encarga de recibir los requerimientos.
- El Scrummaster que gestiona los procesos y garantiza que el equipo cuente con todos los insumos necesarios para realizar su trabajo.
- El equipo que se encarga del desarrollo de las diferentes funcionalidades del sistema.

Artefactos

En Scrum no existen formatos que deban ser completados al pie de la letra, sino que se sugieren diferentes artefactos que permiten llevar la trazabilidad del proyecto a través del tiempo como se menciona a continuación:

- Pila del producto o product backlog: en este artefacto se relacionan las diferentes funcionalidades, cambio y errores del producto final, con tres características fundamentales, deben ser simples, para que puedan ser entendidos por todo los miembros del equipo, suelen usarse las historias de usuario; también deben estimarse, indicando que tan complejo puede llegar a ser su desarrollo, sin indicar su coste en tiempo y finalmente deben priorizarse para así poder desarrollar las funcionalidades de acuerdo a su importancia.

- Pila del sprint o sprint backlog: se presentan las diferentes tareas que deben realizarse para desarrollar una funcionalidad propuesta en el product backlog, en este artefacto se presenta una estimación en horas de cuánto puede tardar la realización de cada tarea, además se indica el miembro del equipo que se ofreció para realizarla y cuánto esfuerzo (tiempo) es necesario aún para terminar cada una de las tareas propuestas.

- Gráfico de avance o burndown chart: en este artefacto se puede apreciar el estado de avance del proyecto, permite revisar posibles desviaciones en la estimación realizada para el sprint, permitiendo tomar los correctivos necesarios para así no incumplir con los objetivos planteados, se grafica la cantidad de esfuerzo (horas) pendiente por emplear contra los días del proyecto. (págs. 32, 33)

Dentro de la metodología Ágil de Scrum se analiza el desempeño de las funciones de las características dentro de cada rol a desempeñar, es iterativa. El cliente es la cabeza primordial del proyecto porque es quien proporciona la información, En el Scrum se describe cada función específicamente desde lo más esencial hasta lo más complejo, se encarga de administrar los recursos de trabajo el Scrummaster provee las herramientas para garantizar el desempeño de desarrollo.

Los artefactos, no tienen un patrón de conducta establecido para su desarrollo, por lo que se establece de acuerdo a los beneficios que se añaden a las soluciones del software. Pila del producto o product backlog: es una herramienta que implica el desenvolvimiento del desarrollo del producto final de manera clara, sencilla y objetiva, Pila del sprint o sprint backlog: es la asignación de tareas y la administración de tiempo que se requiere para la solución del software.

Gráfico de avance o burndown chart: Analiza las condiciones de avances del desarrollo del software, permite el análisis de los errores cometidos en el proyecto para su debida corrección. Estos artefactos forman un desempeño muy importante en el desarrollo del software porque es un breve análisis de avance de nuestro caso de estudio proporcionan datos específicos para la solución.



Figura 2: Roles, artefactos y eventos principales de SCRUM
Fuente: Mariño S; Alonzo P.

Se pretende mostrar es la estructura de codificación de un proyecto y el análisis del mismo mediante su desarrollo de codificación se describe el proceso que se genera paso a paso en la aplicación de esta metodología ágil, dentro de las funcionalidades se determina el dueño del producto, las pautas, las herramientas y el equipo que contribuye en el proceso de codificación, además se define el que se encarga del sprint a ejecutar el avance del proyecto, el tiempo que se desarrolla el software es un mínimo de 4 semanas y un máximo de 2 meses donde se ejecuta las respectivas supervisiones diarias para analizar el avance del mismo.

KANBAN

(Bermejo, 2011) El Kanban es un sistema de gestión del trabajo en curso (WIP1), que sirve principalmente para asegurar una producción continua y sin sobrecargas en el equipo de producción multimedia. El Kanban es un sistema de gestión donde se produce exactamente aquella cantidad de trabajo que el sistema es capaz de asumir. El Kanban es un sistema de

trabajo just in time, lo que significa que evita sobrantes innecesarios de stock, que en la gestión de proyectos multimedia equivale a la inversión innecesaria de tiempo y esfuerzo en lo que no necesitaremos (o simplemente es menos prioritario) y evita sobrecargar al equipo. (pág. 8)

Kanban es una herramienta que permite visualizar el flujo de trabajo de manera óptica y el avance del mismo, realiza un análisis global del proyecto detectando los errores y captando las oportunidades de mejora, para evitar la sobrecarga de trabajo y detectar los denominados cuellos de trabajos además kanban es un administrador de los recursos innecesarios a utilizar porque precisa la tarea justa para evitar el congestionamiento de flujo de trabajo, dando a entender claramente en la pauta que es lo que se va a desarrollar.

Aplicación práctica del Kanban

(Bermejo, 2011, págs. 10-13) Para hacer más comprensible la aplicación práctica del Kanban, en este apartado vamos a imaginar un caso típico de la producción de proyectos multimedia, donde tenemos un equipo dedicado a dar servicio de esta clase de productos. En concreto, el servicio que proporciona este equipo se basa en el desarrollo de pequeñas tareas evolutivas de un producto ya instalado en casa del cliente. Nuestro equipo trabaja para diferentes clientes, que tienen contratado un servicio de mantenimiento donde desarrolla estos pequeños plug-ins a medida que los clientes van pidiendo. Entráis en este equipo y os piden gestionarlo. Por suerte, conocéis el Kanban y empezáis a aplicarlo.

El primer paso: Lo primero que tenemos que hacer en este equipo es representar su realidad actual de forma visual. La forma más habitual de aplicar el Kanban en la producción de productos multimedia es mediante el visual management, es decir, la representación visual

del flujo de trabajo mediante paneles que tienen que reflejar la realidad del equipo en cada instante.

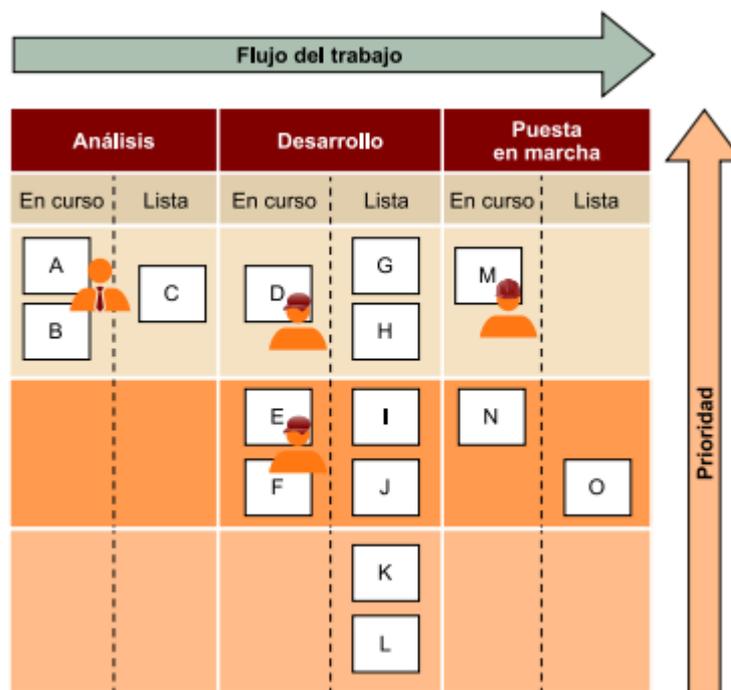


Figura 3: Panel Kanban
Autor: Bermejo M. (2011)

Un panel Kanban para gestionarlos a todos: Nuestro panel Kanban tiene que ser estudiado y juzgado en cada iteración para detectar fases que falten o sobren. Nunca hay que adoptar un panel como solución universal.

En el Kanban no existe un único modelo de panel adecuado para todos los equipos ni todas las situaciones. Simplemente representa de forma fiel la producción del equipo.

Leyendo un panel Kanban: Al acabar el primer mes como jefe de proyecto en el equipo del ejemplo presentado en la introducción de este apartado, tenemos un bonito panel Kanban en una de las paredes de la sala donde representada la realidad de nuestro equipo. El equipo lo mantiene actualizado de forma constante para que realmente represente el estado actual de los proyectos en curso.

El desarrollo de este esquema se muestra la representación gráfica multimedia que se realiza mediante la ejecución del proceso de desarrollo, los pasos a utilizar detallando específicamente paso a paso lo que haremos en el desarrollo empezamos con el análisis donde se estudian los métodos y caracteres a utilizar para desarrollo aplicaremos todo lo que habíamos planteado hacer, en la puesta en marcha pondremos a funcionar lo que hemos realizado, ejecutar la propuesta final.

El trabajo en curso (WIP)

(Bermejo, 2011) En el Kanban, hay que hacer que los trabajadores del equipo solo produzcan si el sistema acepta más cantidad de este trabajo producido. Una vez hemos representando la realidad actual del equipo, tenemos que actuar y aplicar la primera restricción del Kanban, que es limitar el WIP. Tenemos que hacer obligatorio que los programadores dejen de programar y se dediquen a acabar de terminar aquellas tareas que están bloqueadas. Hay que lograr que los trabajadores del equipo solo produzcan si el sistema acepta más cantidad de este trabajo producido. (pág. 14)

Cuando los desarrolladores de los software se encuentran ante un sistema de congestión es muy importante dejar de trabajar y dedicarse a realizar las tareas que están bloqueadas o en procesos de desarrollo, mientras esperamos que el sistema se regule para poder restablecer el trabajo por este motivo la metodología ágil kanban limita el WIP debido a que esta es la principal función que ayuda a que no se sature el sistema y se cree los cuellos de botellas, por esta razón no se permite la saturación de las tareas.

Los cuellos de botella

Una vez que la fase de desarrollo ha llegado al límite WIP, hemos llegado a tener un cuello de botella: la fase de puesta en marcha provoca un atasco.

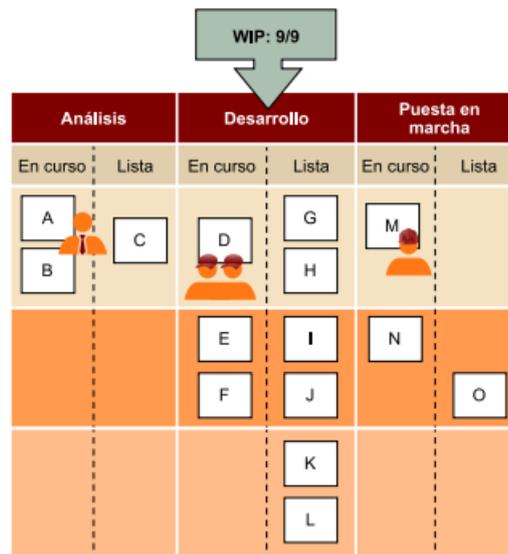


Figura 4: Panel Kanban
Autor: Bermejo M. (2011)

En esta representación gráfica nos muestra la congestión del WIP donde se crea el denominado cuello de botella debido a la cantidad de trabajos ingresados, por estas razones no se deben proporcionar más trabajos de los equipos de desarrollo de software de los que la máquina administre, podemos ver claramente en el desarrollo la saturación de información o trabajo recreando la parálisis del sistema.

Optimizando el equipo

(Bermejo, 2011) El Kanban es un espejo que refleja la realidad del trabajo que el equipo está desarrollando, refleja los flujos por donde pasan las tareas y el equipo en sí mismo. El Kanban nos hará aflorar ineficiencias en la gestión de los proyectos multimedia, pero también ineficiencias en el equipo.

La metodología ágil kanban es un sistema de autoanálisis de lo que estamos realizando, detalla cada instrucción dada dentro del sistema como la parte del desarrollador, que permite verificar tanto los errores internos como externos, tomando en cuenta las condiciones del sistema donde solo nos permite el ingreso de trabajo solo si el sistema lo admite, con la aplicación del WIP ayuda a mejorar la calidad de trabajo y el tiempo de entrega del trabajo.

Los beneficios de desarrollar en parejas: Una práctica muy extendida y probablemente la más conocida de desarrollo ágil de productos multimedia es el pair programming (programación por parejas). Esta práctica es muy recomendada en equipos que aplican el Kanban, puesto que ayuda a hacer posible este movimiento temporal de personas entre fases para desatascar el flujo de tareas.

Se destaca la importancia y el aporte para administrar el desarrollo software multimedia en equipos de trabajos de dos, este aporte es muy utilizado en las metodologías ágiles la colaboración de equipos de trabajo permite manejar y desarrollar las tareas con más prudencia ya que desempeñan un rol muy relevante en el des atascamiento de los flujos.

Kaizen: una cultura de mejora continúa: La palabra japonesa kaizen significa literalmente mejora en el cambio. La cultura en un puesto de trabajo donde la fuerza del trabajo está enfocada en la mejora de la calidad, la productividad y la satisfacción del cliente de forma continua durante todo el proceso de producción (no solo al principio o al final) se conoce como una cultura Kaizen.

Se describe el manejo del sistema de la cultura en el puesto de trabajo, donde el desarrollador tiene como objetivo satisfacer la necesidad del cliente aportando continuamente con la mejor productividad y el desempeño en realizar su trabajo, es la esencia que caracteriza

a esta cultura denominada Kaizen es una cultura que promueve la calidad de trabajo en todo el tiempo se involucra desde principio a fin del desarrollo.

Optimizando el flujo

Los tres pilares básicos del Kanban son los siguientes:

- 1) visualizar de forma continúa el estado del equipo de producción,
- 2) limitar el WIP para mejorar la calidad y el tiempo de entrega, y
- 3) potenciar el flujo.

Se plasma en tres pilares la optimización del flujo donde prioriza la visualización de manera continua el estado del equipo de producción también lo importante que es limitar el WIP para mejorar la calidad y el tiempo de entrega de los trabajos realizados donde su mayor logro es evitar la saturación de los sistemas de trabajo aplicando los recursos necesarios de entrada en el sistema de desarrollo.

Agile Inception

(BEAS, 2011) Agile Inception, también conocida como Inception Deck o simplemente Inception, es un conjunto de dinámicas orientadas a enfocar a todas las personas involucradas en un proyecto hacia un mismo objetivo, reduciendo muchas de las incertidumbres, ayudando a explicitar los riesgos más evidentes y poniendo en común las expectativas de todos.

Agile Inception no es otra cosa que el involucramiento de todo el equipo de trabajo y se realiza una retroalimentación para abolir las dudas sobre el tema a desarrollar y plasmar cual

es el objetivo que desean alcanzar de forma unánime. Que es lo que busca Agile Inception primero un equipo de trabajo totalmente capacitado que no tenga dudas en la ejecución del desarrollo del software, que todos trabajen en el mismo objetivo enfocados en la misma dirección.

(BEAS, 2011) La Agile Inception no es una garantía para conseguir el consenso pero ayuda mucho a conseguirlo. De hecho, uno de los resultados esperables de la misma es que el proyecto no es viable o demasiado arriesgado para iniciarlo.

Con lo estudiado sobre las Agile Inception en ocasiones no es garantía del consenso debido a la diversidad de criterios de los miembros del equipo de trabajo porque cada miembro tiene una óptica diferente sobre la posible solución del problema y sus opiniones aportadas crean un tema de debate para la construcción del caso, luego del planteamiento de la solución se enfocan en los criterios favorables que se aportan al desarrollo del software.

¿Cómo se hace?

(BEAS, 2011) Es una reunión muy cara porque requiere de la participación continuada de muchas personas durante mucho tiempo. Ejecutar todas las actividades propuestas requiere mucho tiempo porque muchas de ellas buscan la creatividad o la reflexión y puede llevar hasta dos días completos, tras los cuáles los participantes pueden estar exhaustos. La logística (comida, bebida, descanso, etc.) es muy importante para conseguir que todos los participantes puedan aportar en todas las actividades. Para ayudar a que las actividades fluyan es conveniente contar con un facilitador.

Las primeras actividades «¿Por qué estamos aquí?», «Crear un elevator pitch», «Diseñar una caja para el producto», «Crear una lista de lo que NO es» y «Conoce a tus vecinos» son

dinámicas que están todas orientadas a discutir de manera creativa sobre el producto que se quiere construir, centradas todas en el QUÉ y obviando siempre el CÓMO.

«Muestra la solución» se centra en el CÓMO y es una actividad previa a la discusión sobre los riesgos «Qué nos impide dormir». El resto tiene mucho que ver sobre cómo queremos priorizar nuestras expectativas, que es justamente lo que discutiremos en «Estimar el tamaño» y, sobre todo, «Compensar las decisiones».

La última actividad, «Cuánto va a durar esto», nos ayudará a responder a la pregunta que hace cualquier inversor (o gerente de un proyecto): «¿cuánto me va a costar esto?» y «¿cuándo lo voy a tener?». Idealmente también podremos responder a la pregunta: «¿y esto quién lo va a hacer?»

Esta receta, la Agile Inception Deck, se describe originalmente como un «deck» (baraja) porque no es obligatorio pasar por todas (aunque sí recomendable). Así, dependiendo del contexto, se pueden sustituir unas actividades por otras que persigan los mismos objetivos o incluso obviarlas si esos objetivos ya se han conseguido previamente.

El concepto de aplicación (trabajo) de Agile Inception requiere un análisis sistemático grupal que se lleva a cabo y esto puede durar muchos días dependiendo de las decisiones del caso, pero esta metodología es un poco extenuante para los participantes por lo que se debería ofrecer la comodidad para la estadía de los participantes, ese sistema requiere captar la mejor idea que aporte a la toma de decisiones para aplicar. Realizar un esquema con los parámetros de preguntas.

Es muy importante que se detalle los recursos de la creatividad, el énfasis de cooperar para administrar las funciones, en diferentes ocasiones no es necesario que se apliquen todos los pasos pero que se sustituyan por las medidas que soporten mejores ideas y con la cooperación de los procesos omitidos, los recursos económicos son muy elevados con la aplicación de la metodología Agile Inception Deck.

CONCLUSIONES

Los artículos, libros, autores y casos revisados se puede concluir que las metodologías tradicionales por su estructura compleja no son las adecuadas para el desarrollo de los software, las metodologías ágiles Scrum son una versión mejorada de las tradicionales porque forman un papel esencial en la nueva era tecnológica de desarrollo del software, hoy en día las empresa que no apliquen las metodologías ágiles no están aptas para realizar las funciones competitiva de desarrollo, es bien cierto que las adaptaciones de estas nuevas metodologías ágil Scrum en las empresas tradicionales no son las más sencillas, pero es lo mejor que se debe adoptar para continuar en el medio actual tecnológico.

Entre todas las metodologías de software se ha concluido que la metodología ágil Scrum, es la que se debe aplicar y recomendar para el uso en los desarrollos de los softwares en las empresas. Debido a su gran aportación de garantía en el desarrollo, es una estructura iterativa, que involucra directamente al cliente de manera permanente hasta que se culmine el proyecto constantemente aporta y asesora los cambios requeridos, los desarrolladores mantienen un ambiente laboral armonioso que es lo que toda empresa enfatiza para que todos los integrantes de cada equipo se vinculen con la mejor actitud en el proyecto.

Las empresas al momento de utilizar la metodología ágil Scrum tiene que utilizar diferentes artefactos (Product Backlog, Sprint Backlog y Burndown) para que los proyectos a realizarse no tengan ningún error, estos deben ser simples y entendibles, también con esta metodología se estimará el tiempo que le tomará a cada miembro del equipo de trabajo en el trabajo que está realizando y además se muestra el avance del mismo, todo esto es fundamental porque se va a saber de forma cronológica el proyecto que se está realizando y el plazo para concluir.

Como parte fundamental adaptarse a los cambios que se presentan he incluso cuando el desarrollo ha concluido, es aplicable y adaptable debido a que no tiene una estructura o patrón establecida, el trabajo se realiza en conjunto, ayuda a la optimización de los recursos tiempo y el recurso económico.

RECOMENDACIONES

Se recomienda utilizar las metodologías ágiles Scrum y no las tradicionales porque desempeñan un enfoque complejo en el desarrollo de software, por esta razón se deben de seguir las siguientes recomendaciones:

- Seleccionar el equipo de trabajo indicado y asignando las tareas correspondientes a desarrollar, prestos a colaborar con lo establecido por el cliente, todo esto las empresas lo podrán lograr bajo la supervisión de un Scrum Master que da apoyo y lidera al equipo de trabajo, este debe de conocer a la perfección esta metodología para así alcanzar el objetivo planteado en el proyecto.

- Elaborar una lista con los objetivos que se deben de alcanzar en el proyecto, priorizando las pautas a aplicar conforme al avance del desarrollo, realizando una estructura de acuerdo a la metodología ágil Scrum aplicando los tiempos establecidos y definir las actividades a seguir de todo el equipo de trabajo, sin desviarse de los objetivos concretados.

Realizar un panel de trabajo donde se visualice el proceso que se va a realizar en el equipo de trabajo de la empresa por medio de la metodología ágil Scrum, para la aplicación de los proyectos esto aportará una ventaja totalmente inversa con respecto a los sistemas tradicionales, esto también ayudará a las empresas para poder estudiar y analizar el avance concretado en el proyecto por medio de reuniones de forma periódica al equipo de trabajo, de esta forma se va a corregir el proceso determinado del proyecto.

BIBLIOGRAFÍA

- BEAS, J. M. (23 de enero de 2011). *Agile Inception*. Obtenido de jmbeas:
<https://jmbeas.es/guias/agile-inception/>
- Bermejo, M. (2011). *El Kanban*. Catalunya - España: UOC. Recuperado el 07 de agosto de 2019, de
https://s3.amazonaws.com/academia.edu.documents/37163648/El_Kanban.pdf?response-content-disposition=inline%3B%20filename%3DEl_acercamiento_Kanban_mejorado_con_tecn.pdf&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIWOWYYGZ2Y53UL3A%2F20190807%2Fus-
- Laínez Fuentes, J. R. (2015). *Desarrollo de software Ágil Extreme Programming y Scrum*. IT Campus Academy.
- Pantaleo, G., & Rinaudo, L. (2016). *Ingeniería de Software*. Argentina: Alfaomega.
- Salazar, J. C., Tovar, Á., Linares, J. C., Lozano, A., & Valbuena, L. (Diciembre de 2018). Scrum versus XP: similitudes y diferencias. *Tecnología, Investigación y Academia*, 32, 33.
- Tardivo, M. L., & Caymes-Scutari, P. (2016). *XXII CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN*. Argentina: CACIC.

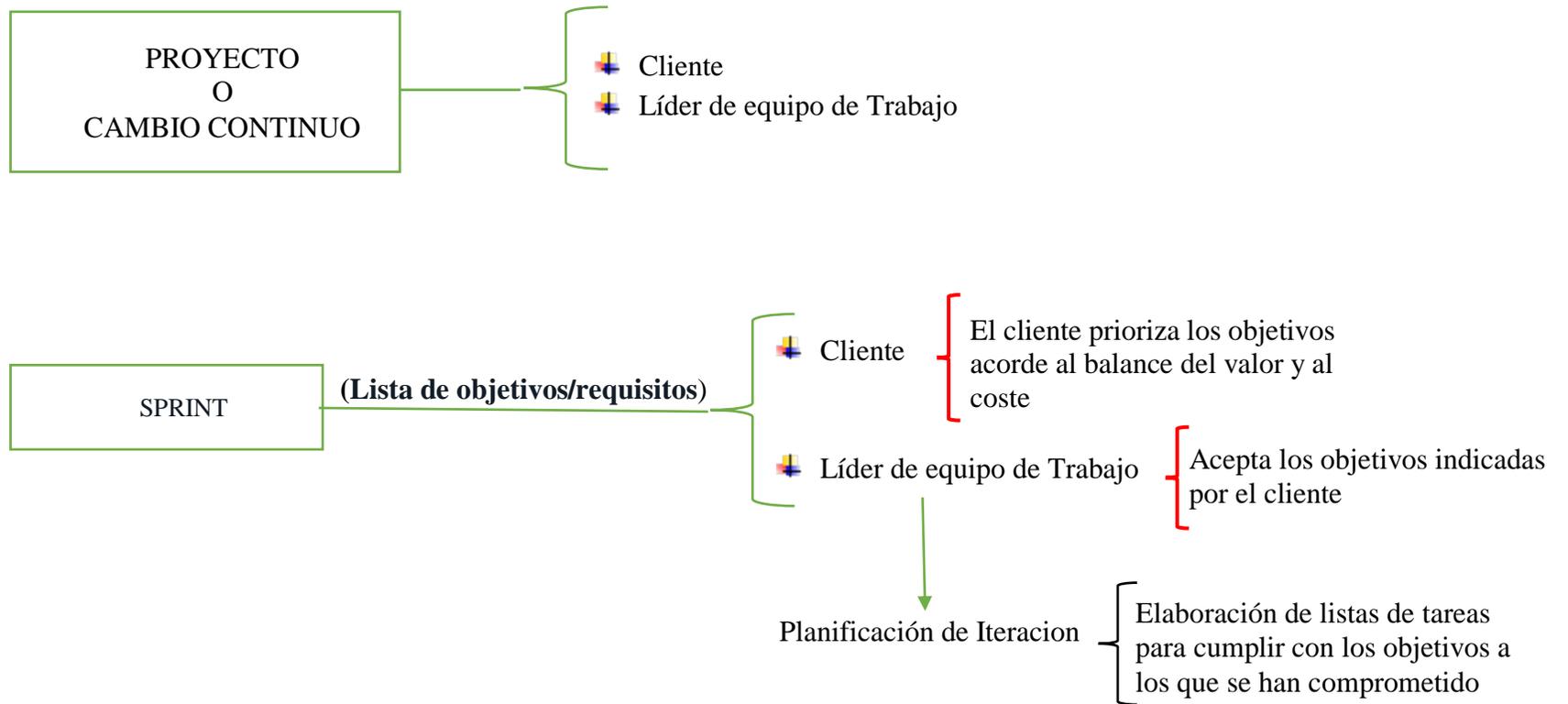
ANEXOS

Actualmente existen empresas que utilizan la metodología scrum como parte fundamental en el desarrollo de software y cambios emergentes, entre ellas tenemos la empresa Glosl Hitss una



empresa nacional e internacional.

PASOS PARA IMPLEMENTAR LA METODOLOGÍA SCRUM



SPRINT MEETING

Reuniones diarias

(15 minutos)



¿Qué he hecho ayer?

¿Qué voy a hacer hoy?

¿Qué impedimentos o contratiempo tengo para realizar mi tarea?

SCRUM MASTER



Proteger al equipo



Eliminar los obstáculos que impidan