

DOCUMENTO IDENTIFICANDO LA METODOLOGIA DEL SOFTWARE

Presentado por:

Ana Karina Yepes Gómez c.c. 43612987

Jheison Fernando Rojas Rendon c.c. 1053846295

Adan de Jesús Restrepo Zapata c.c. 71290603

Jesús Harvey Bernal López. c.c. 1069725992

Presentado a:

Carlos Alberto Fuel Tulcan

Ficha: 2627091

Tecnología en Análisis y Desarrollo de Software

Centro Agroindustrial

SENA Regional Quindío.

Contenido

1. INTRODUCCION.....	3
2. OBJETIVOS	6
2.1. Objetivo General	6
2.2. Objetivos Específicos	6
3. METODOLOGÍA SCRUM.....	7
3.1. Principios del Manifiesto Ágil.	7
3.2. Ventajas y desventajas de la Metodología Agile.....	10
3.3. Marco de Trabajo Scrum	13
3.3.1. Pilares fundamentales del Scrum.	13
3.3.2. Roles del Marco de Trabajo Scrum.	15
3.3.3. Eventos del marco de trabajo Scrum.....	15
3.3.4. Artefactos del Marco de Trabajo Scrum.	16
3.3.5. Ventajas y Desventajas del Marco de Trabajo Scrum.	17
4. DESARROLLO DEL PROYECTO	19
4.1. FASE INICIACIÓN:.....	20
4.1.1. Crear la visión del proyecto:	20
4.1.2. Identificar al scrum master e interesados del proyecto:	20
4.1.3. Desarrollo de épicas:	20
4.1.4. Elaboración de la lista de pendientes del Sprint:.....	20
4.2. FASE PLANIFICACION Y ESTIMACION.....	20
4.2.1. Elaborar historias de usuario:	20
4.2.2. Aprobar, estimar y asignar historias de usuarios.....	20
4.2.3. Elaboración de tareas:	20
4.2.4. Elaboración de la lista de pendientes	20
4.2.5. Crear entregables	20
4.2.6. Llevar a cabo el stand up diario:	20
4.2.7. Mantenimiento de la lista priorizada de pendientes del producto:	20
4.3. FASE REVISIÓN Y RETROSPECTIVA	20
4.3.1. Convocar Scrum de Scrums	20
4.3.2. Demostración y validación del Sprint:	20
4.3.3. Retrospectiva de Sprint:	20
5. CONCLUSIONES	21
6. REFERENCIAS BIBLIOGRAFICAS.....	22

1. INTRODUCCIÓN

El presente trabajo, pretende hacer mención a la metodología del software que implementaremos para el desarrollo de nuestro software, de acuerdo con las fuentes Identificadas en la documentación proveída por el instructor para el curso de Análisis y Desarrollo de Software ficha 2627091. En el desarrollo de la siguiente actividad se busca demostrar que la aplicación de la metodología SCRUM al Proyecto a desarrollar es la opción más acertada dado que nos permite fragmentar el Proyecto en procesos más pequeños para que al momento de su entrega satisfaga completamente las necesidades del cliente; involucrando al mismo en las reuniones programadas para mostrar los avances y problemas presentados en el desarrollo de este. Haciendo que el ritmo de trabajo del equipo sea sostenible dado que se adapta a los cambios presentados en el transcurso del desarrollo del proyecto consiguiendo un ritmo constante haciendo que se reduzcan los costos de recursos y se desarrolló el aplicativo de forma óptima. El proyecto a desarrollar se establece con el fin de satisfacer una necesidad del mercado que busca mejorar el servicio de vigilancia técnica vigitecol a los usuarios, prestando la función a través de un aplicativo que sirve como intermediario entre cliente y prestador de servicio, donde tanto el cliente como vigitecol podrán optimizar de manera satisfactoria la adquisición y prestación de productos y servicios que les brinden una mejora en los procesos donde el cliente tendrá la oportunidad de acceder al aplicativo para encontrar variedad de opciones de servicios de vigilancia que necesite para su domicilio o empresa que se ajusten a sus necesidades y al mismo tiempo vigitecol contara con la oportunidad de administrar la información necesaria para la comercialización de sus servicios.

Hasta el momento en el desarrollo de software se ha pasado por varios procesos, como identificar claramente las necesidades de nuestro cliente a través de entrevistas y de esta forma se pudo hacer el levantamiento de requerimientos de forma clara y concisa para entonces proceder con la elaboración de un prototipo y de casos de prueba que le permitirá al cliente evidenciar de forma básica a donde se pretende llegar con el desarrollo de este software; ahora con la aplicación de la metodología SCRUM se busca agilizar y hacer una planeación de cada uno de los procesos por los cuales pasará nuestro proyecto mejorando de esta forma las tareas y agilizando el resultado.

2. OBJETIVOS

2.1. Objetivo General

- Planificar de forma ágil el desarrollo del software a realizar por medio de la metodología aplicada SCRUM.

2.2. Objetivos Específicos

- Segmentar las tareas del desarrollo del proyecto para hacer el proceso más eficaz por medio de la metodología ágil.
- Agilizar los procesos a realizar dentro del proyecto para satisfacer las necesidades del cliente.

3. METODOLOGÍA SCRUM

Este enfoque nace como respuesta a los problemas que puedan ocasionar las metodologías tradicionales y se basa en dos aspectos fundamentales, retrasar las decisiones y la planificación adaptativa. Basan su fundamento en la adaptabilidad de los procesos de desarrollo.



Un modelo de desarrollo ágil, generalmente es un proceso incremental (entregas frecuentes con ciclos rápidos), también cooperativo (clientes y desarrolladores trabajan constantemente con una comunicación muy fina y constante), Sencillo (el método es fácil de aprender y modificar para el equipo) y finalmente Adaptativo (capaz de permitir cambios de último momento).

La metodología ágil consiste en un conjunto de técnicas que tienen como objetivo hacer que el proceso de entrega de un proyecto sea más rápido y eficiente. Sin embargo, esto no significa que el plazo de entrega sea más corto. Aunque, de hecho, la metodología ágil tiene como objetivo entregar valor al cliente de manera más rápida.

Creado en 2001 por un grupo de programadores de TI (Tecnología de la Información) a través del [“Manifiesto por el Desarrollo Ágil de Software”](#), la idea era que las actividades relevantes a un proceso se pudieran realizar de manera más dinámica y con mayores resultados. De su nombre en inglés “agile, software development” es que este proceso toma su nombre. Esto es posible porque sus ciclos son más cortos. Así, con cada etapa finalizada, ya se puede entregar algo, dejando de lado la necesidad de tener que esperar a la finalización total del proyecto.

3.1. Principios del Manifiesto Ágil.

La metodología Agile es un enfoque para la gestión de proyectos que utiliza cuatro pilares clave y 12 principios Agile para organizar los proyectos.

El Manifiesto Agile para el desarrollo de software describe los 12 principios de Agile que deben seguir todos los proyectos. Estos son:

1. **Nuestra máxima prioridad es satisfacer al cliente a través de la entrega anticipada y continua de software valioso.** El primer principio de la metodología Agile establece que los clientes deben recibir entregas o iteraciones del proyecto en intervalos regulares a lo largo del ciclo de vida de su proyecto, en lugar de solo una entrega del producto al final.

Principios del Manifiesto Ágil

Seguimos estos principios:

Nuestra mayor prioridad es satisfacer al cliente mediante la entrega temprana y continua de software con valor.

Aceptamos que los requisitos cambien, incluso en etapas tardías del desarrollo. Los procesos Ágiles aprovechan el cambio para proporcionar ventaja competitiva al cliente.

Entregamos software funcional frecuentemente, entre dos semanas y dos meses, con preferencia al periodo de tiempo más corto posible.

Los responsables de negocio y los desarrolladores trabajamos juntos de forma cotidiana durante todo el proyecto.

Los proyectos se desarrollan en torno a individuos motivados. Hay que darles el entorno y el apoyo que necesitan, y confiarles la ejecución del trabajo.

El método más eficiente y efectivo de comunicar información al equipo de desarrollo y entre sus miembros es la conversación cara a cara.

El software funcionando es la medida principal de progreso.

Los procesos Ágiles promueven el desarrollo sostenible. Los promotores, desarrolladores y usuarios debemos ser capaces de mantener un ritmo constante de forma indefinida.

La atención continua a la excelencia técnica y al buen diseño mejora la Agilidad.

La simplicidad, o el arte de maximizar la cantidad de trabajo no realizado, es esencial.

Las mejores arquitecturas, requisitos y diseños emergen de equipos auto-organizados.

A intervalos regulares el equipo reflexiona sobre cómo ser más efectivo para a continuación ajustar y perfeccionar su comportamiento en consecuencia.

[Volver al Manifiesto](#)

2. **Aprecia los requisitos cambiantes, incluso al final del desarrollo.** Los procesos Agile aprovechan el cambio para la ventaja competitiva del cliente. Uno de los problemas que los autores del Manifiesto encontraron con la gestión tradicional de proyectos es que era difícil complacer las solicitudes de cambio de última hora del cliente. Este principio garantiza que los proyectos Agile tengan la capacidad de adaptarse a cualquier cambio, sin importar en qué momento haya que hacerlo, con un retraso mínimo.

3. **Entrega de software funcional con frecuencia, desde un par de semanas hasta un par de meses, con preferencia por el plazo más corto.** Los proyectos Agile planean plazos de proyectos cortos y frecuentes que permiten una rápida entrega de productos funcionales. A menudo, los proyectos Agile se dividirán en sprints o intervalos de proyecto de 1 a 4 semanas y cada uno de ellos terminará con la entrega de un producto.

4. **Los empresarios y desarrolladores deben trabajar juntos todos los días durante todo el proyecto.** Este principio de Agile establece que la comunicación regular con todas las partes interesadas es fundamental para el éxito del proyecto. Normalmente, esto implica una breve reunión diaria tanto con el equipo del proyecto como con otras partes interesadas clave.

5. **Desarrolla proyectos en torno a personas motivadas.** Bríndales el entorno y el apoyo que necesitan y confía en ellas para que hagan el trabajo. Un concepto clave de la metodología de gestión de proyectos Agile es que hay que colocar a las personas adecuadas en los puestos idóneos y hay que darles la autonomía necesaria para hacer bien su trabajo. Es importante diseñar un equipo de proyecto basado en capacidades en lugar de en puestos de trabajo o cargos

dentro de la empresa. El enfoque del gestor de proyectos debe centrarse en motivar al equipo del proyecto y apoyarlo, en lugar de microdirigirlo.

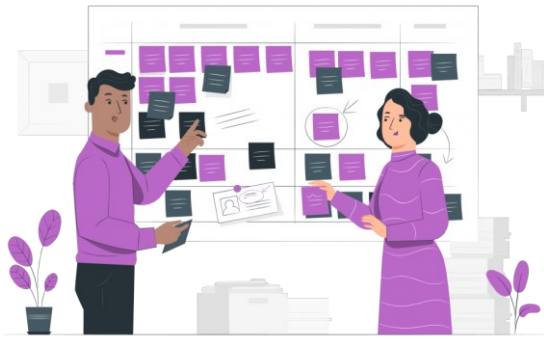
6. **El método más eficiente y efectivo para transmitir información a un equipo de desarrollo y dentro de él es la conversación cara a cara.** Los creadores del Manifiesto Agile creen firmemente en la

importancia de que los equipos y las partes interesadas se encuentren en el mismo lugar siempre que sea posible, ya que la comunicación cara a cara es más efectiva que otras opciones, como el correo electrónico o el teléfono. Si el equipo no puede estar en la misma ubicación, la videoconferencia es una opción que aún puede captar algunos de los mismos valores, como las señales no verbales.

7. **El software funcional es la medida principal del progreso. El énfasis dentro de la metodología Agile se pone en proporcionar resultados completos y funcionales.** Esto siempre debe tener prioridad sobre cualquier requisito adicional, como la documentación del proyecto. Además, otras métricas, como las horas dedicadas o el tiempo transcurrido, no se consideran tan importantes como entregar productos funcionales.
8. **Los procesos Agile promueven el desarrollo sostenible. Los patrocinadores, desarrolladores y usuarios deben poder mantener un ritmo constante indefinidamente.** De acuerdo con este principio, los proyectos Agile deben tener un ritmo constante para cada ciclo iterativo o sprint dentro del proyecto. Este desglose debería eliminar la necesidad de horas extra o plazos en caso de fallos, al tiempo que promueve la producción frecuente de productos funcionales. También debe crear un ciclo repetible que el equipo pueda seguir continuamente durante el tiempo que sea necesario.
9. **La atención continua a la excelencia técnica y al buen diseño mejora la agilidad. Un enfoque principal de un proyecto Agile debería ser mejorar el producto final y lograr avances sistemáticamente a lo largo del tiempo.** En otras palabras, cada iteración siempre debe ser una mejora con respecto a la anterior, y el equipo siempre debe buscar nuevas innovaciones.
10. **La simplicidad, el arte de maximizar la cantidad de trabajo no realizado, es esencial.** El objetivo de un proyecto Agile es hacer lo suficiente para completar con éxito el proyecto encargado y cumplir con las especificaciones solicitadas. Ha de evitarse o eliminarse cualquier documentación, pasos, procesos o trabajo adicionales que no añadan valor al cliente o mejoren los resultados del proyecto.
11. **Las mejores arquitecturas, requisitos y diseños surgen de equipos autoorganizados. Agile se basa en la creencia de que se requieren equipos motivados, autónomos y cualificados para ofrecer los mejores resultados y productos.** Los equipos deben estar capacitados para organizarse y estructurarse según sea necesario. Deben tener la libertad de colaborar e innovar como mejor les parezca, sin verse obstaculizados por demasiada supervisión.
12. **A intervalos regulares, el equipo reflexiona sobre cómo ser más efectivo, luego sintoniza y ajusta su comportamiento en consecuencia.** Un equipo exitoso y motivado requiere un fuerte enfoque en cómo puede potenciar sus habilidades y procesos para crecer y mejorar continuamente. El equipo debe tener revisiones periódicas sobre su rendimiento y resultados, incluida una conversación sobre cómo puede mejorar a medida que avanza.

Los cuatro pilares clave del libro Manifiesto Agile están diseñados para valorar:

1. Personas e interacciones por encima de herramientas y procesos
2. Software funcional por encima de documentación exhaustiva
3. Colaboración con el cliente por encima de negociación contractual
4. Respuesta ante el cambio por encima de seguir un plan



Para definir Agile, el método se basa en el concepto de olas continuas o sprints de planificación y ejecución de proyectos, lo que te permite adaptar y desarrollar continuamente tu plan, alcance y diseño a lo largo del proyecto.

De acuerdo con la definición de metodología Agile, los proyectos Agile requieren un enfoque iterativo, que admite entregas incrementales, frecuentes y coherentes de productos funcionales a tu cliente. Este

enfoque innovador garantiza que tu equipo de proyecto pueda entregar productos concretos de manera coherente sin retrasarse por los cambios y los requisitos cambiantes.

Agile tiene un alto nivel de participación del cliente e incluye revisiones frecuentes del progreso tanto con el equipo del proyecto como con el cliente.

3.2. Ventajas y desventajas de la Metodología Agile

Ventajas	Desventajas
Contacto continuo con el cliente En los métodos tradicionales de gestión de proyectos, el equipo del proyecto normalmente solo estaba en contacto con el cliente al principio y al final del proyecto, lo que implicaba que si los requisitos o expectativas del cliente no se captaron correctamente al principio, o cambiaron con el tiempo, el equipo del proyecto no se enteraba hasta que era demasiado tarde. Con Agile, hay un contacto continuo durante todo el proceso y entregas iterativas para garantizar que tu equipo va por buen camino, por lo que el producto final será exactamente lo que el cliente quiere.	El resultado del proyecto es estable y se comprende adecuadamente. Agile tiene por objeto ayudar a reducir el coste del cambio y la incertidumbre en un proyecto al dividirlo en etapas iterativas de gestión de proyectos. Sin embargo, si hay muy poca incertidumbre y una mínima posibilidad de cambios, puede que Agile no sea el enfoque más efectivo. Por ejemplo, si trabajas en una industria con fuertes regulaciones o donde ya se conocen muchos de los requisitos del proyecto, entonces no necesitas planificación iterativa y varios borradores.
Capacidad de adaptarse ¿Qué sucede si tu cliente te dijo a mitad de un proyecto que necesitaba un cambio de alcance? Utilizando un enfoque tradicional para la gestión de proyectos, no se pudo llevar a cabo o probablemente implicó aumentos significativos tanto de coste como de tiempo del proyecto. Con Agile, los cambios se pueden incorporar con un mínimo esfuerzo, sin importar lo avanzado que vaya tu equipo	Tu proyecto debe producir una entrega repetible. Por definición, un proyecto es «un esfuerzo temporal con un principio y un final, y debe usarse para crear un producto, servicio o resultado único». Pero, ¿qué sucede si un cliente te pide que construyas cinco casas idénticas y tú decides crear un proyecto independiente y formar un equipo de proyecto para cada una? Con Agile, esto podría dar lugar a cinco casas únicas en lugar

con el proyecto, ya que se pueden añadir fácilmente a la siguiente iteración.	de a cinco idénticas. Una de las desventajas de Agile es que no está diseñado para la reproducibilidad.
Entrega más rápida Agile incorpora un enfoque de desarrollo continuo que garantiza que tu equipo entregue constantemente productos funcionales. Esto significa que en lugar de esperar de 6 a 12 meses o más por un producto final, tu cliente obtiene una versión funcional del producto a intervalos mucho más cortos, generalmente cada 2 a 4 semanas.	Tu cliente no quiere Agile. Un proyecto Agile requiere un contacto continuo con el cliente final. Sin embargo, puede que algunos clientes no tengan el tiempo, la capacidad o el deseo de dedicarse a un proyecto. Si el proyecto se considera de bajo valor o de bajo riesgo, pueden preferir un enfoque más tradicional en el que solo se los involucre en las fases clave o la entrega final.
Menor riesgo del proyecto Dado que tu equipo desarrolla versiones del producto con frecuencia y recibe comentarios de los clientes desde el principio, se minimiza el riesgo de que falle un proyecto. Al dividir un proyecto grande en iteraciones, su riesgo también se reduce al fracaso de una iteración o borrador solamente. Es más probable que encuentres pronto problemas pequeños, que pueden abordarse fácilmente, en lugar de descubrir un problema importante ya en el momento de la última prueba antes de la entrega final. Esto significa que habrás invertido menos tiempo y dinero cuando se descubra un problema o el proyecto deba cancelarse.	Tu empresa no puede respaldar Agile. Si tu empresa o equipo de proyecto no está listo, intentar adoptar el ciclo de desarrollo Agile puede presentar riesgos para tu proyecto. Estos son cinco indicadores de que tu empresa no está lista para usar Agile: <ul style="list-style-type: none"> • Agile no se entiende bien. Si tu equipo y tu empresa no tienen formación en Agile o una comprensión sólida de sus principios, prácticas y marcos, no están listos para usar esta metodología. • Las partes interesadas clave son reticentes. Ya sea el patrocinador de tu proyecto o un miembro clave del equipo, si alguien se resiste a adoptar Agile, tendrás que resolver el problema antes de poder adoptar con éxito la metodología. • Tu organización no puede respaldar la colaboración diaria. Si existen barreras importantes para la comunicación diaria y la colaboración abierta entre los miembros del equipo, puede que Agile no sea el mejor enfoque. • La estructura de la empresa no puede respaldar equipos interfuncionales. En un proyecto Agile, las personas de muchas funciones diferentes necesitan reunirse, comunicarse y colaborar durante todo el ciclo de vida del proyecto. Si en tu

	<p>empresa las funciones están aisladas, puede que esto no sea realista.</p> <ul style="list-style-type: none"> • Tu organización requiere grandes cantidades de documentación. Si tu empresa requiere mucha documentación e informes de prueba, puede que adoptar Agile sea demasiado costoso.
<p>Innovación continua</p> <p>Agile apoya la colaboración y la mejora continua, lo que puede conducir a la innovación y al desarrollo de nuevos productos y funciones. Al encontrarse los equipos en la misma ubicación y tener reuniones diarias, se apoyan las sesiones de intercambio y generación de ideas. Agile respalda una «meritocracia de ideas» donde triunfa la mejor idea, sin importar quién la sugiera. El equipo del proyecto, otras partes interesadas y el cliente pueden determinar las funciones y las características juntos como un equipo.</p>	<p>El Informe del estado de Agile 2019 concluyó que los contratiempos más habituales a los que se enfrentan las empresas al implementar la metodología Agile son:</p> <ul style="list-style-type: none"> • La cultura de la organización no concuerda con los valores de Agile. • Existe una resistencia general al cambio en toda la organización. • El apoyo y el patrocinio de la dirección son inadecuados. • Hay una carencia de habilidades/experiencia en el uso de métodos Agile. • Los procesos y las prácticas son incoherentes entre los equipos. • La formación y la educación son insuficientes. • El cliente, la empresa o el propietario del producto no están lo suficientemente disponibles. • Los métodos de proyecto tradicionales están demasiado arraigados en la empresa. • Los datos y las métricas están fragmentados, lo que proporciona un «panorama general» incompleto. • No hay suficiente colaboración e intercambio de conocimientos entre los empleados. <p>Muchas de las dificultades a las que se enfrentan las empresas al implementar Agile son las mismas que las de cualquier iniciativa de cambio organizativo importante. En otras palabras, los contratiempos a menudo no son específicos de Agile, sino que son el resultado de una mala gestión del cambio organizativo. Al reflexionar acerca de cómo implementar la gestión de proyectos Agile, resulta imprescindible que incorpores no solo directrices Agile coherentes, sino también que</p>

	sigas las prácticas recomendadas de gestión de cambios, de las que hablaremos en la siguiente sección.
--	--

3.3. Marco de Trabajo Scrum

Se caracteriza por ser la «metodología del caos» que se basa en una estructura de desarrollo incremental, esto es, cualquier ciclo de desarrollo del producto y/o servicio se desgrana en «pequeños proyectos» divididos en distintas etapas: análisis, desarrollo y testing. En la etapa de desarrollo encontramos lo que se conoce como interacciones del proceso o Sprint, es decir, entregas regulares y parciales del producto final.

Esta metodología permite abordar proyectos complejos que exigen una flexibilidad y una rapidez esencial a la hora de ejecutar los resultados. La estrategia irá orientada a gestionar y normalizar los errores que se puedan producir en desarrollos demasiado largos, a través de, reuniones frecuentes para asegurar el cumplimiento de los objetivos establecidos.

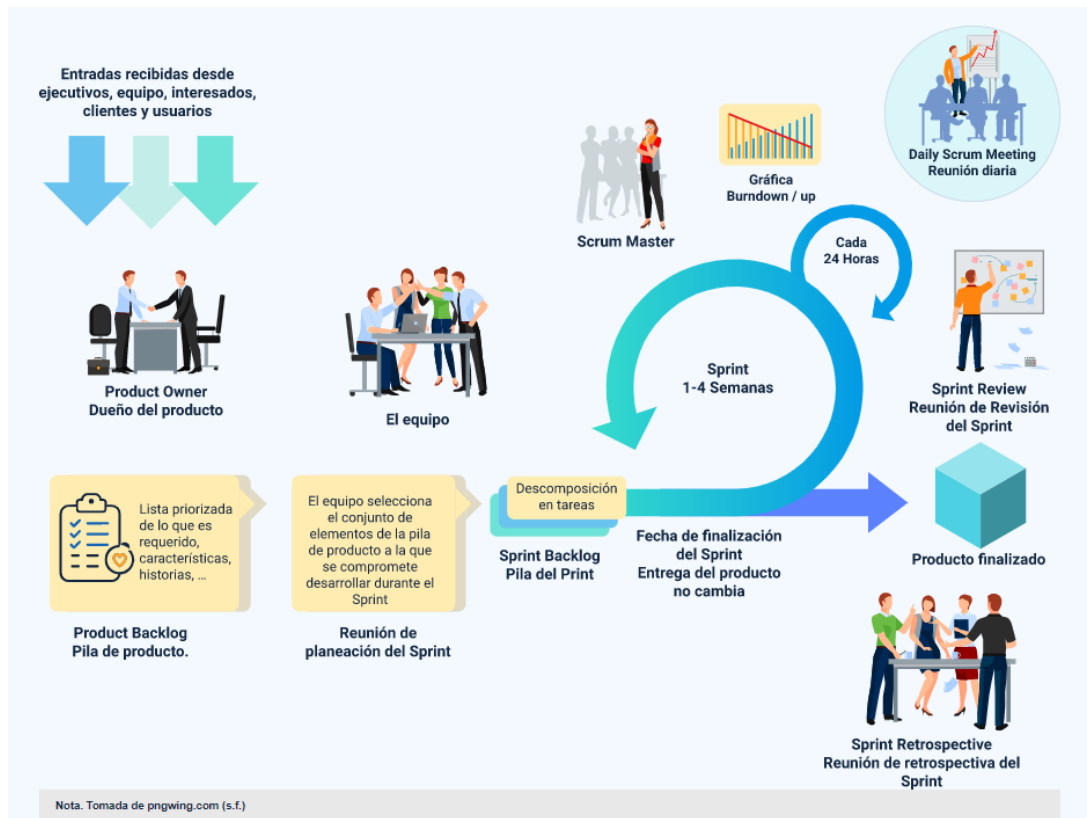
Las reuniones son el pilar fundamental de la metodología, donde diferenciamos entre: reuniones de planificación, diaria, de revisión y de retrospectiva, la más importante de todas ellas, ya que, se realiza después de terminar un sprint para reflexionar y proponer mejoras en los avances del proyecto. Los aspectos clave por los que se mueve el Scrum son: innovación, flexibilidad, competitividad y productividad.

3.3.1. Pilares fundamentales del Scrum.

Scrum es un marco de trabajo ágil de muy amplio uso en la industria del software que se fundamenta en los valores y principios ágiles definidos en (Manifiesto Ágil, 2001) y donde se definen tres pilares fundamentales según (SCRUMstudy, 2013) los cuales se describen a continuación:

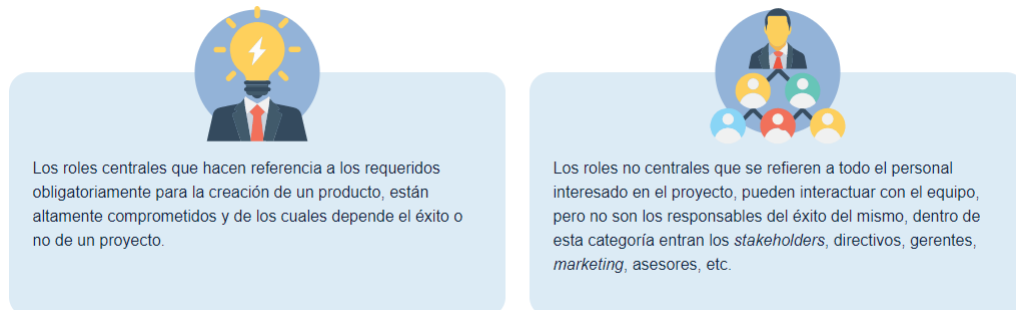
1. **Transparencia.** Hace referencia a que cualquier proceso de Scrum puede ser conocido por cualquiera. Esto es posible por medio de eventos como:
 - Las reuniones de revisión y reuniones diarias.
 - Artefactos como la pila de producto.
 - Cronogramas de lanzamiento.
 - Documentos de visión del proyecto.
 - Instrumentos de seguimiento, como: el burndown chart o el tablero de Scrum (Scrum board).
2. **Inspección.** Permite que cualquiera pueda estar enterado de las actividades realizadas por otros y en general conocer el estado actual de los procesos.
3. **Adaptación.** Por medio de la transparencia y la inspección es posible fijar actividades de mejoras que permitan modificar todo tipo de proceso en pro de lograr más altos estándares de calidad.

Adicionalmente, este marco de trabajo ágil está estructurado por un conjunto de roles, eventos y artefactos que pueden ser observados a continuación:



3.3.2. Roles del Marco de Trabajo Scrum.

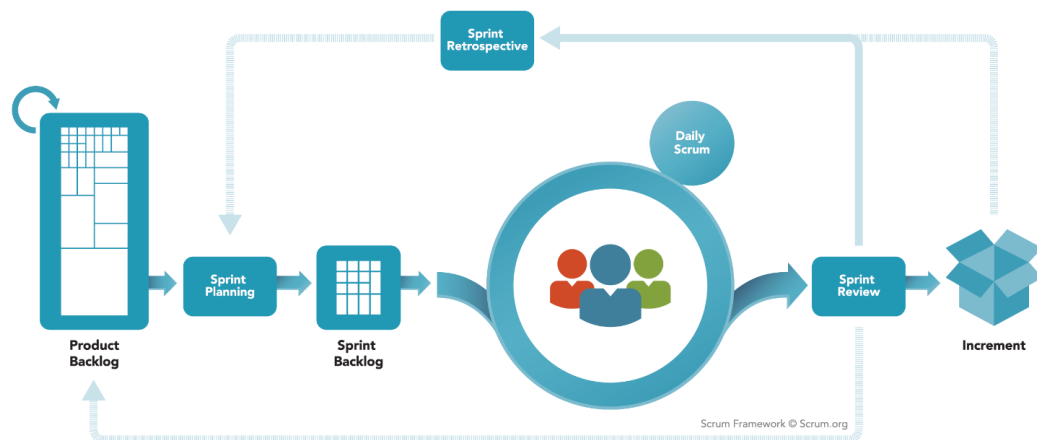
Por otro lado hay tres roles centrales dentro del marco de trabajo de Scrum (SCRUMstudy, 2013) que se describen a continuación:



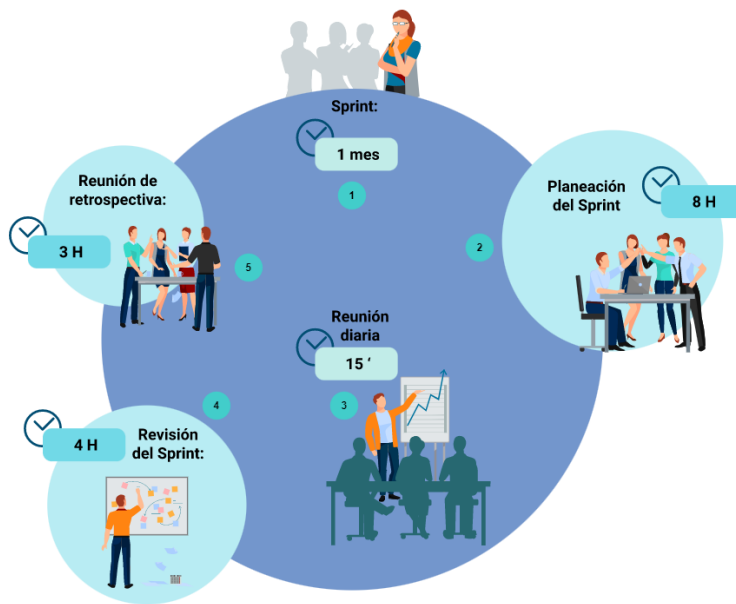
1. *Dueño del producto (Product Owner)*. persona con amplio conocimiento en el negocio del cliente, sus necesidades y las tendencias del mercado para el área específica. Este rol está encargado de maximizar el valor de negocio entregado al cliente y es el único responsable del control del Product Backlog (requerimientos) y su priorización. Este también representa al cliente en algunos procesos de demostración de avances y determina cuándo aprobar o no una entrega.
2. *Scrum Master*. Es un rol que se encarga de facilitar los procesos al interior del equipo de trabajo removiendo cualquier impedimento y apoyando procesos de empoderamiento personal, debe velar porque los elementos propios del marco de trabajo Scrum se apliquen de manera correcta.
3. *Equipo de desarrollo (Developer Team)*. Son los responsables de la transformación de los requerimientos en código ejecutable a ser usado por el cliente, pero también son responsables de la planificación de las iteraciones, establecimiento de características para tener en cuenta en la verificación de un requerimiento terminado y presentación de avances a los clientes. Generalmente es un equipo autoorganizado y autogestionado.

3.3.3. Eventos del marco de trabajo Scrum.

Además de los roles, Scrum define un conjunto de eventos con participantes y objetivos claros que se desarrollan en momentos particulares del flujo general de Scrum, a continuación, se detalla cada uno de estos:



1. *Sprint*: es el corazón de Scrum y se refiere a una iteración que está acotada generalmente por un lapso entre 2 y 4 semanas donde se realiza un ciclo completo de actividades de análisis, diseño, construcción y pruebas para desarrollar una versión del producto potencialmente entregable al cliente.
2. *Planeación del Sprint (Sprint planning)*: reunión realizada justo antes del inicio de un Sprint donde se definen el subconjunto de requerimientos (Sprint Backlog) a ser desarrollados en los siguientes y cómo será el proceso requerido para hacer la entrega, lo cual incluye detallar los requerimientos en tareas concretas, estimación de tiempos/esfuerzo y distribución inicial de responsabilidades. Dependiendo de la duración del Sprint este tiempo de planificación puede variar, pero la métrica establecida para un Sprint de 4 semanas corresponde a una planeación de Sprint de 8 horas.



3. *Reunión diaria (Daily Meeting)*: reunión realizada generalmente al inicio de cada día donde el equipo de trabajo informa en que ha venido trabajando, qué cosas realizará en el día y qué problemas se le han presentado. Esta es corta, se realiza de pie y debe tener una duración alrededor de los 15 minutos, por lo que, se alinea con los pilares de transparencia e inspección.

4. *Revisión del Sprint (Sprint Backlog)*: reunión realizada al finalizar el sprint donde el equipo de desarrollo muestra los resultados. Para un Sprint de 4 semanas se usan reuniones de revisiones 4 horas.

5. *Reunión de retrospectiva (Sprint Review)*: esta es la última, se realiza

luego de la revisión del Sprint y tiene como objetivo la autoevaluación personal y del grupo sobre el desempeño del Sprint que acaba de finalizar. En esta se identifican y documentan los aprendizajes por medio de diferentes técnicas en las que generalmente se busca dar respuesta a las siguientes preguntas:

- ¿Qué funcionó bien y se debe seguir haciendo?
- ¿Qué no funcionó bien y se debe dejar de realizar?
- ¿Qué debemos empezar a mejorar?

Para un Sprint de 4 semanas se utilizan 3 horas para esta reunión.

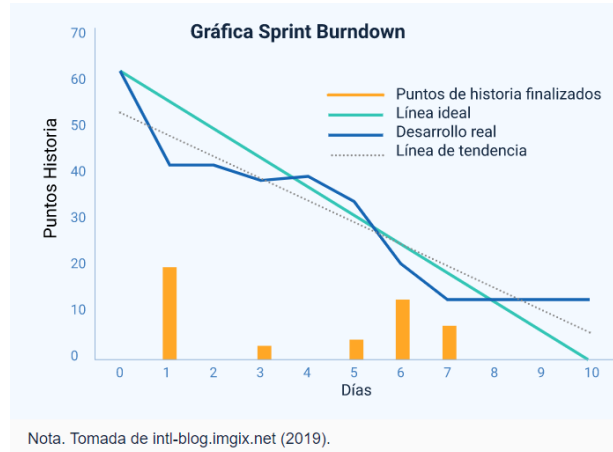
3.3.4. Artefactos del Marco de Trabajo Scrum.

Finalmente, el marco de trabajo Scrum define un conjunto de artefactos que permiten registrar y gestionar información clave para asegurar los tres pilares fundamentales y proveen información valiosa durante todo el proceso de desarrollo de software. Entre los artefactos representativos de este se encuentran los siguientes:

1. *Pila de producto (Product Backlog)*: lista priorizada de requerimientos generalmente descritos en formato de historias de usuarios que representa todas las características del sistema a construir.
2. *Pila del Sprint (Sprint Backlog)*: lista de requerimientos seleccionados desde el Product backlog por el equipo de trabajo para ser desarrollados durante un Sprint particular. Este es uno de los artefactos generados en la reunión de planeación del Sprint.

3. *Burndown Chart*: es un gráfico visual de dos ejes que muestra a los equipos la cantidad de trabajo pendiente por completar (eje Y) y el tiempo disponible para hacerlo (eje X). Este generalmente se realiza por cada Sprint ubicando la cantidad de trabajo a realizar del Sprint Backlog (usualmente medido por puntos de historia u de horas de trabajo) en un tiempo 0 y por cada día finalizado se resta la cantidad de puntos de historia u horas de cada tarea completada.

Por otro lado, también es posible usar este mismo gráfico para representar el avance general del proyecto ubicando en el eje Y la cantidad total de horas o esfuerzo del Product Backlog y en el eje X la cantidad de Sprint proyectados. Cada uno de estos puntos se une por medio de una línea y es posible determinar visualmente si el flujo de trabajo está en una situación óptima o no respecto al tiempo restante para completar el Sprint.



herramientas digitales que permiten la implementación de un tablero de Scrum.

4. *Tablero de Scrum (Scrumboard)*: es un elemento visual donde se integra la mayor parte de los elementos del marco de trabajo Scrum, en él se indica la carga de trabajo, el estado actual de cada una de las actividades y sus respectivos responsables. Este es un elemento que se sincroniza de manera permanente y facilita la implementación de los pilares de transparencia, inspección y adaptabilidad. Si bien se aconseja el uso de un tablero, existen diferentes tipos de

3.3.5. Ventajas y Desventajas del Marco de Trabajo Scrum.

Ventajas	Desventajas
<ul style="list-style-type: none"> • Ayuda a establecer metas cuantificables y, así, mantener al equipo ocupado todo el tiempo para garantizar la productividad. 	<ul style="list-style-type: none"> • Al momento de trabajar con Scrum, todo el equipo debe conocer a fondo sus principios y marco teórico. Tiene que haber conocimiento de los roles de Scrum o podría haber dispersión y choque de funciones.
<ul style="list-style-type: none"> • La flexibilidad del método Scrum te permite modificar el hilo conductor de los hechos en cualquier momento, lo que 	<ul style="list-style-type: none"> • En gran parte, el éxito de un proyecto conducido con el méEn gran parte, el éxito de un proyecto conducido con el método Scrum dependerá de la preparación y

favorece la resolución de conflictos u obstáculos sobre la marcha.	experiencia del Scrum Master (más que de la innovación, creatividad o calidad de los insumos).
<ul style="list-style-type: none"> • Muestra el progreso del sprint o iteración de requerimientos. 	<ul style="list-style-type: none"> • Si por algún motivo quedan tareas sin finalizar, el resto de pendientes comenzará a postergarse indefinidamente, pues a menudo hay una relación lógica y secuencial entre las actividades del tablero.
<ul style="list-style-type: none"> • Evita el perfeccionismo innecesario. 	<ul style="list-style-type: none"> • Puede ser difícil el manejo de Scrum en proyectos complejos.
<ul style="list-style-type: none"> • Se obtienen resultados rápidos y periodos de prueba muy cortos. Así, un producto puede estar listo para venta al público en poco tiempo. 	
<ul style="list-style-type: none"> • El método ofrece una mirada cercana y transparente del proyecto a todas las partes interesadas. Esto aporta confianza a los clientes. 	
<ul style="list-style-type: none"> • Motiva al cierre de los sprints (mejor tasa de cumplimiento). 	
<ul style="list-style-type: none"> • Te muestra una visión completa del proyecto. 	
<ul style="list-style-type: none"> • Usando un Software para Scrum, resulta muy fácil priorizar constantemente y saber distinguir entre los pendientes que pueden esperar y las tareas cruciales. 	
<ul style="list-style-type: none"> • Si analizas las ventajas de Scrum, notarás cómo se antepone la eficacia durante todo el ciclo de vida del proyecto, haciéndote menos propenso a caer en distracciones y rodeos. 	
<ul style="list-style-type: none"> • El método te empuja a realizar tareas oportunamente (con base en un sistema jerárquico de actividades más o menos importantes). 	
<ul style="list-style-type: none"> • El mejor de los beneficios de Scrum es, sin duda alguna, la forma como te ofrece una visión amplia y completa de todo el proyecto, desde las exigencias de los stakeholders hasta las fechas de culminación de cada fase. 	

4. DESARROLLO DEL PROYECTO

Para el desarrollo de nuestro proyecto (Software vigilancia técnica privada) decidimos aplicar la metodología ágil y el marco de trabajo SCRUM: Teniendo en cuenta nuestro proyecto y las características que éste posee, consideramos que esta metodología ágil se adapta perfectamente a la forma y el ritmo de trabajo que debemos aplicar. Esto con el fin de que el desarrollo del software sea lo más eficaz, ayudándonos a tener resultados anticipados, flexibilidad, adaptación, mitigación de riesgos, productividad, calidad y retorno de inversión, del servicio de vigilancia técnica vigitecol a los usuarios. Es así como, prestando la función de vigilancia a través de un aplicativo que sirve como intermediario entre cliente y prestador de servicio, tanto el cliente como vigitecol podrán optimizar de manera satisfactoria la adquisición y prestación de productos y servicios que les brinden una mejora en los procesos. Es así como el cliente tendrá la oportunidad de acceder al aplicativo para encontrar variedad de opciones de servicios de vigilancia que necesite para su domicilio o empresa, que se ajusten a sus necesidades y al mismo tiempo vigitecol contara con la oportunidad de administrar la información necesaria para la comercialización de sus servicios. Algunas de las funciones que se realizarán el desarrollo son: Gestionar las órdenes de trabajo de los empleados, manejar los inventarios de sus productos, además generar recordatorios a sus clientes por medio de medios electrónicos para la realización de sus mantenimientos; optimizando de esta forma el trabajo en la empresa y reduciendo gastos innecesarios. A través del desarrollo para vigitecol se espera dar solución o brindar una mejora tanto a la parte de gestión interna de las tareas de la empresa como mejorar la experiencia del cliente.

Teniendo en cuenta el contexto y las características generales de éste desarrollo para Vigitecol, podemos decir que la metodología ágil y el marco de trabajo Scrum se adapta a nuestro Proyecto, ya que este es un proyecto de tamaño pequeño, y sin embargo nos permitirá tener flexibilidad durante todo el proceso de desarrollo.

Esta metodología nos permitirá acelerar los procesos para satisfacer al cliente, actuar con rapidez ante los posibles cambios, y realizar entregas periódicas del trabajo. Esta metodología se ejecuta en ciclos temporales cortos y de duración fija (iteraciones que normalmente son de 2 semanas, aunque en algunos equipos son de 3 y hasta 4 semanas, límite máximo de feedback de producto real y reflexión).

Si bien nuestro equipo de trabajo está conformado por cuatro integrantes quienes debemos tener diariamente un encuentro en la fase de desarrollo del proyecto, estos encuentros no durarán más de 30 minutos y serán con el objetivo de obtener retroalimentación sobre las tareas de cada integrante o dificultades que se presenten en el desarrollo del producto. Se inicia con el desarrollo de la lista de requerimientos que son más importantes para el cliente en este caso sería (Registro de clientes , Ordenes de trabajo) lo cual nos permitirá las funcionalidades de mayor valor y conocer la velocidad a la que avanzamos en el Proyecto. Esto permitirá despejar riesgos efectivamente de manera anticipada.

Esta metodología también posee una gran ventaja y es que ya sea durante el desarrollo del proyecto o a futuro se contará con la gran capacidad de reacción ante los cambiantes requerimientos generados por las necesidades del cliente o la evolución del mercado. En cuanto a costos se contará con un equipo pequeño de (4 personas) para la creación de software solamente con las prestaciones que contribuyen a un mayor valor de negocio generando un mayor retorno de inversión con esta metodología aplicada.

Los procesos de esta metodología y la integración con nuestro proyecto se aplican de la siguiente manera según las fases de la metodología ágil y marco de trabajo SCRUM.

4.1. FASE INICIACIÓN:

4.1.1. Crear la visión del proyecto: Nuestro proyecto está pensado para satisfacer una necesidad del mercado que busca mejorar el servicio de vigilancia técnica prestando la función a través de un aplicativo que sirve como intermediario entre cliente y prestador de servicio.

4.1.2. Identificar al scrum master e interesados del proyecto: En este caso seríamos las cuatro integrantes del equipo de trabajo. Adan de Jesús Restrepo Zapata, Ana Karina Yepes Gómez, Jesús Harvey Bernal López Y Jheison Fernando Rojas Rendon.

4.1.3. Desarrollo de épicas: conjunto de historias de usuario, recogidas dentro de las iniciativas.

4.1.4. Elaboración de la lista de pendientes del Sprint: Se realizará un plan con un nivel de detalle suficiente como para que los cambios en el progreso se puedan entender en el Scrum Diario. Fase Planificación y Estimación

4.2. FASE PLANIFICACION Y ESTIMACION

4.2.1. Elaborar historias de usuario: Estas ya fueron realizadas en base a la solicitud de requerimientos específicos por parte del cliente aunque creemos que pueden ser mejoradas.

4.2.2. Aprobar, estimar y asignar historias de usuarios

4.2.3. Elaboración de tareas: listas necesarias para desarrollar los requisitos seleccionados.

4.2.4. Elaboración de la lista de pendientes del Sprint Fase implementación

4.2.5. Crear entregables

4.2.6. Llevar a cabo el stand up diario: Reunión diaria de todos los integrantes del equipo esta reunión no durará más de 15 minutos con el objetivo de obtener retroalimentación sobre las tareas de cada integrante o dificultades que se presenten en el desarrollo del producto.

4.2.7. Mantenimiento de la lista priorizada de pendientes del producto: Lista de requerimientos del cliente

4.3. FASE REVISIÓN Y RETROSPECTIVA

4.3.1. Convocar Scrum de Scrums

4.3.2. Demostración y validación del Sprint: Resultado de los entregables aceptados.

4.3.3. Retrospectiva de Sprint: Reuniones para analizar qué salió bien y qué podría mejorarse en el próximo sprint

5. CONCLUSIONES

Para el presente trabajo entonces podemos concluir que adoptamos la metodología ágil y el marco de trabajo Scrum, debido a las características del proyecto de software a construir, como Software Vigilancia Técnica Privada, ya que es una metodología que se adapta a la forma y ritmo de trabajo que debemos aplicar. Además, debido a que éste es un proyecto de tamaño pequeño, y sin embargo nos permitirá tener flexibilidad durante todo el proceso de desarrollo.

Una de las características del proyecto, que nos inclinó a escoger el método SCRUM es que el cliente requiere la creación rápida del producto, además de que necesita verificar constantemente la evolución de este.

El cliente puede obtener resultados importantes y utilizables desde el inicio, por eso escogimos esta metodología ágil y así llegar al objetivo.

Por otro lado, utilizamos la metodología ágil y el marco de trabajo scrum, con el fin de que el desarrollo del software sea lo más eficaz, ayudándonos a tener resultados anticipados, flexibilidad, adaptación, mitigación de riesgos, productividad, calidad y retorno de inversión, del servicio de vigilancia técnica vigitecol a los usuarios.

Esta metodología nos permitirá acelerar los procesos para satisfacer al cliente, actuar con rapidez ante los posibles cambios, y realizar entregas periódicas del trabajo, todo esto debido a que ésta metodología se ejecuta en ciclos temporales cortos y de duración fija (iteraciones que normalmente son de 2 semanas, aunque en algunos equipos son de 3 y hasta 4 semanas, límite máximo de feedback de producto real y reflexión).

Esta metodología también posee una gran ventaja y es que se contará con la gran capacidad de reacción ante los cambiantes requerimientos generados por las necesidades del cliente o la evolución del mercado. En cuanto a costos se contará con un equipo pequeño de (4 personas) para la creación de software solamente con las prestaciones que contribuyen a un mayor valor de negocio generando un mayor retorno de inversión con esta metodología aplicada.

En conclusión, la creación de nuestro proyecto de software requerido por el cliente , y la implementación de la metodología ágil SCRUM son la mezcla precisa para generar un proceso supervisado constantemente por nosotros y el cliente , además permite la organización del equipo de trabajo de forma ágil, determina los roles específicos y delega tareas puntuales, todo eso con el propósito de crear un producto ágil, intuitivo y fácil de manejar por nuestro cliente.

6. REFERENCIAS BIBLIOGRAFICAS

Boehm, B. W. (1979). A Spiral Model of Software Development and Enhancement. ACM Software Engineering Notes, 11(4), 22-42.

Durán, A., y Bernárdez, B. (2001). Metodología para el análisis de requisitos de sistemas software. https://www.infor.uva.es/~mlaguna/is1/materiales/metodologia_elicitacion.pdf

ISO/IEC 12207. (2008). Systems and software engineering - Software life cycle processes <https://www.iso.org/obp/ui/#iso:std:iso-iec:12207:ed-2:v1:en>

McCracken, D., y Jackson, M. A. (1981). "A Minority Dissenting Opinion". En W. W. Cotterman, J. D. Couger, N. L. Enger, F. Harold (Eds.). Systems Analysis and Design: A Foundation for the 1980s (pp. 551-553). Elsevier.

Pantaleo, G., y Rinaudo, L. (2018). Ingeniería de software. Alfaomega.

Penzenstadler, B. (s. f.). Requirements Engineering. CSU Long Beach. <https://bit.ly/3rtBKXN>

Pfleeger, Sh. (2002). Ingeniería del software. Teoría y práctica. Prentice Hall.

Porfirio, C. (s. f.). Técnicas de priorización: el desafío de conseguir un orden para las funcionalidades. atSistemas - Consultoría it blog. <https://bit.ly/3cvumqz>

Rivadeneira, M., S. (2014). Metodologías ágiles enfocadas al modelado de requerimientos. Informes Científicos Técnicos - UNPA, 5(1), 1-29 <https://doi.org/10.22305/ict-unpa.v5i1.66>

Sommerville I. (2011). Ingeniería del software. Addison-Wesley.

Tamayo, M.C. ¿Qué significa SCRUM como metodología de desarrollo Ágil? https://platzi.com/tutoriales/1750-scrum/7403-que-significa-scrum-como-metodologia-de-desarrollo-agil/?utm_source=google&utm_medium=cpc&utm_campaign=17739691128&utm_adgroup=&utm_content=&gclid=CjwKCAiAyfybBhBKEiwAgtB7fpDdm1hPrUuJ8_Gj615xNGCKKqGeJrDmn7EQ10Ac72JHuw6QWu292BoCtywQAvD_BwE&gclidsrc=aw.ds

Misla, C. 2020. 4 Razones para Utilizar scrum a la hora de desarrollar software. <https://invidgroup.com/es/4-razones-para-utilizar-scrum-la-hora-de-desarrollar-software/#:~:text=Fomenta%20el%20trabajo%20en%20equipo,y%20apoya%20en%20la%20misi%C3%B3n.>

Centro de eLearning UTN BA SCEU FRBA (2020). Introducción a metodologías y marcos de trabajo ágiles – Webinar UTN. [Video]. YouTube. <https://www.youtube.com/watch?v=uQZ0JQl49I>

Scrum Manager Gestión de Proyectos. Juan Palacio, Claudia Ruata

Manual imprescindible : métodos ágiles Scrum, Kanban, Lean / Carmen Lasa Gómez, Alonso Álvarez. 2017 https://biblioteca.sena.edu.co/F/DE8EEBCA651D9F4YSY35AA973QVRMJ7MF3Q1J1CF3692Q2DPR1-03731?func=full-set-set&set_number=005413&set_entry=000011&format=999

Adri Salazar . Oct 16, 2016. Academy, Proyectos, SCRUM, Uncategorized. <http://www.prozessgroup.com/procesos-de-scrum/>