



UNIVERSIDAD CÉSAR VALLEJO

ESCUELA DE POSGRADO

**PROGRAMA ACADÉMICO DE MAESTRÍA EN INGENIERÍA
DE SISTEMAS CON MENCIÓN EN TECNOLOGÍAS DE LA
INFORMACIÓN**

**Metodologías ágiles para mejorar la calidad del software en una
entidad del estado, Lima 2021**

TESIS PARA OBTENER EL GRADO ACADÉMICO DE:

Maestro en Ingeniería de Sistemas con mención en Tecnologías de la Información

AUTOR:

Arenas Saldaña, Williams Jesus (ORCID: 0000-0002-2884-5465)

ASESOR:

Dr. Martínez López, Edwin Alberto (ORCID: 0000-0002-1769-1181)

LÍNEA DE INVESTIGACIÓN:

Sistema de Información y Comunicaciones

LIMA - PERÚ

2022

Dedicatoria

A mis padres Sonia y Jesus por inculcarme en mí el esfuerzo y valentía de no temer a las adversidades que se presentan en la vida.

A mi esposa e hijos por su comprensión, paciencia y apoyo incondicional durante todo este proceso que valió la pena.

Agradecimiento

Mi profundo agradecimiento para mis padres y a todos mis compañeros que me apoyaron en sus opiniones y consejos.

De igual manera, mis agradecimientos a la Universidad por permitir cumplir mis sueños.

Índice de Contenidos

Dedicatoria	ii
Agradecimiento	iii
Índice de Contenidos	iv
Índice de Tablas	v
Índice de Figuras	vi
RESUMEN	vii
ABSTRACT	viii
I. INTRODUCCIÓN	1
II. MARCO TEÓRICO	5
III. METODOLOGÍA	15
3.1. Tipo y diseño de investigación	15
3.2. Categorías, Subcategorías y matriz de categorización	16
3.3. Escenario de estudio	17
3.4. Participantes	18
3.5. Técnicas e instrumentos de recolección de datos	18
3.6. Procedimiento	19
3.7. Rigor científico	19
3.8. Método de análisis de datos	20
3.9. Aspectos éticos	20
IV RESULTADOS Y DISCUSIÓN	22
V. CONCLUSIONES	31
VI. RECOMENDACIONES	32
REFERENCIAS	33
ANEXOS	39

Índice de Tablas

Tabla 1: Categorías, Subcategorías y matriz de categorización.	16
Tabla 2: Criterios del rigor científico	19
Tabla 3: Categorías, Subcategorías emergentes de Planificación	28

Índice de Figuras

Figura 1: Organigrama de la entidad	17
Figura 2: Triangulación de Antecedentes, Marco Teórico, y Resultados	22
Figura 3: Triangulación de las técnicas utilizadas	25
Figura 4: Triangulación de Entrevista Semiestructurada, Observación y Marco Teórico	27
Figura 5: Triangulación de Entrevista Semiestructurada, Antecedentes, Marco Teórico	29

RESUMEN

Con el incremento en la demanda de los sistemas de información y la exigencia en el tiempo y la calidad del software, la presente investigación tuvo como objetivo principal proponer metodologías ágiles para mejorar la calidad del software en una entidad del Estado a través de resultados inmediatos que permitan cubrir las necesidades de la institución. La investigación fue de enfoque cualitativo, el tipo de investigación fue básica y se utilizó el diseño de investigación acción. Para ello, las técnicas que se emplearon fueron la entrevista semiestructurada que fueron realizadas a expertos en metodologías ágiles, además se aplicó la observación a la unidad de estudio y el análisis documental como parte de la investigación, posteriormente mediante la triangulación nos permitió contrastar enfoques a partir de los datos recolectados con el fin de obtener conclusiones de los objetivos.

Se concluye que para implementar las metodologías ágiles SCRUM y KANBAN es necesario considerar factores como la capacitación del personal respecto a los principios y procedimientos, la designación de los roles es fundamental para el desarrollo del proceso del software, respecto al equipo, estos deberán ser auto organizados y trabajar en espacios coubicados para mantener la comunicación y colaboración.

Palabras claves: Planificación, Mejora continua, Pila de producto, Desarrollo Iterativo.

ABSTRACT

With the increase in the demand for information systems and the demand on time and software quality, the main objective of this research was to propose agile methodologies to improve the quality of software in a State entity through immediate results that allow meeting the needs of the institution. The research was of a qualitative approach, the type of research was basic and the action research design was used. For this, the techniques that were used were the semi-structured interview that were carried out with experts in agile methodologies, in addition, observation was applied to the study unit and documentary analysis as part of the research and with triangulation it allowed us to contrast approaches from of the data collected in order to obtain conclusions of the objectives.

It is concluded that to implement the agile SCRUM and KANBAN methodologies it is necessary to consider factors such as the training of personnel regarding the principles and procedures, the designation of roles is essential for the development of the software process, regarding the team, these should be self organized and work in co-located spaces to maintain communication and collaboration.

Keywords: Planning, Continuous improvement, Product backlog, Iterative development.

I. INTRODUCCIÓN

En los últimos años se viene incrementando exponencialmente la demanda de software, tanto en instituciones públicas como privadas. Esto ha generado mayor exigencia en el desarrollo del software lo cual ahora implica otros factores que las metodologías tradicionales no contemplan, tales como entregas tempranas, mejora continua, flexibilidad y adaptabilidad a los cambios, medir las cantidades y el tiempo de cada proceso, entre otros. Es por ello que en los últimos años las organizaciones vienen adoptando las metodologías o marcos de trabajo ágiles con el objetivo de producir software con resultados inmediatos, mejor calidad y sobre todo que cumpla con las exigencias de los clientes, lo cual se ha convertido en un factor crítico de éxito para las organizaciones (Ver anexo 9).

A nivel internacional, las organizaciones dedicadas al desarrollo de software buscan constantemente mejorar la calidad del software mediante la adopción de nuevas metodologías de trabajo que permitan mejorar la eficiencia en el ciclo de vida del software y la entrega inmediata de valor al cliente. Al respecto Gaete, et al., (2020), sostiene que las metodologías ágiles vienen dando buenos resultados gracias a la flexibilidad que tienen estos ante los requisitos y soluciones que se presentan durante el proceso de desarrollo de software, puesto que pueden variar o evolucionar en el tiempo de acuerdo a las necesidades del proyecto, otra característica resaltante y que viene logrando buenos resultados en el desarrollo de software es la colaboración que existe dentro de los equipos de trabajo, ya que lo considera valioso para el cumplimiento de objetivos. El investigador Gaete et al., (2020), resalta a la metodología Scrum, Lean Software y Kanban como los que aportan mayor beneficio.

Por otro lado, Ozdenizci, (2020) sostiene que las organizaciones están adoptando enfoques ágiles en el proceso de desarrollo de software para asegurar la entrega incremental de la versión del software, entrega precisa de funcionalidades, alta satisfacción del cliente y menor fallas del proyecto. Las principales razones para implementar metodologías ágiles son acelerar la implementación de software, mejorar la capacidad de gestionar las prioridades cambiantes y aumentar la productividad.

Con el pasar de los años mayor son las organizaciones que vienen incorporando metodologías ágiles a los proyectos de desarrollo de software y

transformación digital. Al respecto, Marinho, et. al., (2019) afirma que los métodos ágiles se han vuelto comunes en las organizaciones que desarrollan software en todo el mundo. Según la encuesta de Agile Survey, el 86% de los encuestados tenía por lo menos un equipo incorporando el uso de prácticas ágiles. Entre las grandes organizaciones que vienen implementando metodologías ágiles para mejorar la calidad, entrega de productos y mejora continua son Spotify, el servicio Google Adwords de Google, Amazon y Apple.

Como se menciona, la empresa Spotify dedicada al servicio de música en Streaming ubicada en Suecia, implemento las metodologías Scrum y Kanban debido al rápido crecimiento de la empresa en el número de desarrolladores y en los procesos que fueron evolucionando. La aplicación de Scrum permitió el trabajo colaborativo en equipos pequeños y altamente productivos, pudiendo tomar mejores decisiones y controlar los equipos. Se cambió la estructura organizacional jerárquica a una que le permitía mayor libertad para experimentar ya que no dependía de un jefe, sino que la responsabilidad recaía en el equipo.

En Latinoamérica, países como Colombia, Brasil, Argentina, Chile, México, entre otros, están utilizando metodologías ágiles para mejorar los inconvenientes en la planificación y estimación de los requerimientos de software que según Pressman (2010), son algunas de las actividades que se requiere para poder construir software de alta calidad. Un estudio realizado por IDC para Everis, en la región, se tiene que, en el 2018 un 36% de las empresas ha logrado implementar metodologías ágiles en sus proyectos de software, lo cual representa una mejoría de 5% con respecto al 2017. Por otro lado, Quelal, et. al., (2018) sostiene que el 48% de encuestados en Ecuador indican haber utilizado metodologías ágiles en sus proyectos, sin embargo, se evidencia en los resultados un alto porcentaje de deserción por la falta de capacitación antes de la implementación.

La investigación confirma que la tasa de éxito del desarrollo de software ágil es de alrededor del 71,5%. Los métodos ágiles se están utilizando como un nuevo proceso en el desarrollo de software. (Neelu, et. al., 2021). Otras de las razones por las que las metodologías ágiles están teniendo éxito es por la flexibilidad que tienen para realizar cambios en los requisitos, aun cuando el proyecto está en proceso. En ese aspecto, Oluwaseun (2021) sostiene, que las metodologías ágiles tienen una ventaja en relación a las metodologías tradicionales o denominada también enfoque en cascada, al no seguir estrictamente la serie de pasos secuenciales que impone

las metodologías tradicionales. Por otro lado, Riesener (2021), afirma que para cumplir con las entregas constantes y a corto tiempo, las metodologías ágiles incorporan el trabajo iterativo que como resultado permite reducir el tiempo de entrega y mejora el manejo de los requisitos volátiles de los clientes.

Mientras tanto en Perú, cada vez más organizaciones están empleando metodologías ágiles en el desarrollo de software lo cual le permiten mejorar la producción de software, obteniendo el producto en menor tiempo. Referente a las organizaciones públicas, Farro (2018) sostiene que son algunas las que están implementándolas tales como el Minsa, Osinergmin y la Sunat, cada una de estas agrupaciones viene implementando estrategias ágiles para superar las barreras propias del sector público e identificando oportunidades de mejora en sus procesos para lograr una adopción exitosa.

En los últimos años las prácticas ágiles han tomado protagonismo sobre todo en el mundo financiero peruano, Castillo (2018) sostiene que BBVA continental viene implementándolo desde el 2014 en sus área de desarrollo y ya cuenta con más de 40 equipos scrum multidisciplinarios, BBVA Wallet es una billetera electrónica que permite realizar pagos desde el celular sin tener registrado la persona entre sus contactos, la cual fue desarrollado y puesto al servicio del público peruano en solo cuatro meses.

Al respecto el diario El Comercio (2021) sostiene que las metodologías top más usadas por organizaciones en el Perú son Scrum, Kanban y Lean y que llevan una gran ventaja sobre los métodos tradicionales ya que aplican técnicas de desarrollo colaborativo entre los actores que intervienen, y no se centran en una planificación estricta puesto que cuentan con flexibilidad para lograr productos adaptados a las necesidades y prioridades del cliente, siguiendo la misma línea, Mendoza (2019), afirma que en el Perú el 40% de las grandes empresas ya están implementando el uso de metodologías ágiles, así mismo el Perú se encuentra posicionado en un tercer lugar con respecto a la implementación de marcos ágiles, solo detrás de los colombianos (47%) y brasileños (45%).

Por tal motivo, en la SBN, en el área de Tecnología de información, mediante esta investigación se propone aplicar metodologías ágiles en el desarrollo de software, con el propósito de obtener mejores resultados en la calidad del software. Esta finalidad conlleva a planificar y estimar eficientemente los requerimientos, realizar entrega de producto constante en el menor tiempo, incentivar la

colaboración y autonomía entre los equipos, además de la participación constante de los Stakeholders.

En base a lo señalado anteriormente y para efecto de esta investigación, el enunciado del problema general es “¿De qué manera las metodologías ágiles mejorarían la calidad del software en una entidad del estado, Lima 2021?”, y como enunciado de los problemas específicos: (a) ¿Cómo es la planificación y estimación de las metodologías ágiles en la calidad de software en una entidad del Estado, Lima 2021?, (b) ¿Cómo aumentar la productividad del equipo para mejorar la calidad del software en una entidad del estado, Lima 2021?, (c) ¿Cuál es la satisfacción del cliente usando las metodologías ágiles para mejorar la calidad de software en una entidad del Estado, Lima 2021?.

Como justificación teórica, esta investigación acerca de las metodologías ágiles nos permitirá profundizar y distinguir con claridad los beneficios y ventajas que cada uno de estas metodologías pueden aportar en la resolución de la calidad de software en el área de tecnología de la información en una entidad del Estado. Como justificación práctica, se realiza porque urge la necesidad de mejorar la calidad de desarrollo del software a través de equipos auto gestionables y buenas prácticas en la gestión de requisitos en el área de tecnologías de la información con el uso de metodologías ágiles en una entidad del Estado.

Asimismo, se presenta una justificación económica, que permitirá reducir perdida en costos destinados a proyectos no culminados y sin ningún beneficio a la Entidad, mala imagen institucional de cara a los usuarios e instituciones públicas. Por otro lado, como justificación tecnológica, aportara en la innovación de nuevas técnicas y prácticas en la planificación de desarrollo de software en el área de T.I. para el desarrollo y bienestar de la entidad.

Respeto a la formulación del objetivo se plantea lo siguiente: Proponer metodologías ágiles para mejorar la calidad del software en una entidad del estado, Lima 2021; y como enunciado de los objetivos específicos: (a) Determinar la planificación y estimaciones de las metodologías ágiles para mejorar calidad de software en una entidad del Estado, Lima 2021; (b) Analizar la productividad del equipo en las metodologías ágiles para mejorar la calidad de software en una entidad del Estado, Lima 2021; (c) Analizar la satisfacción del cliente usando las metodologías ágiles para mejorar la calidad de software en una entidad del Estado, Lima 2021.

II. MARCO TEÓRICO

En este capítulo se ha considerado tanto antecedentes nacionales como internacionales, las cuales se presentan a continuación: Referente a lo primero, se menciona a Falen (2020) en su investigación referente a “Scrum en la gestión de proyectos de desarrollo de software en la empresa Innovatec, Magdalena del Mar”, considero que para mejorar la calidad del software se debía obtener mejores resultados con la entrega inmediata y a corto plazo de los productos de software y la colaboración del equipo de trabajo y comunicación constante del cliente. El enfoque de la investigación fue cualitativo, se basó en el paradigma interpretativo. Las conclusiones sostienen que para aplicar correctamente la metodología ágil se debe definir las herramientas, procedimientos y documentos necesarios con el propósito de amoldarlos a las necesidades del negocio, además se debe asignar los roles de scrum de acuerdo a las habilidades y conocimientos de cada integrante del equipo y establecer periódicamente las reuniones respetando los horarios dentro de la ejecución del proyecto.

Así mismo Valdivia (2021) en su investigación sobre “Modelo de Gestión de Servicios de Red LTE Privada, basado en Metodologías Ágiles para Big Mining Compañías”, tuvo por finalidad proponer el modelo de gestión de Servicios LTE privado basado ITIL y con el apoyo de metodologías ágiles de acuerdo a las necesidades de una empresa minera. Se trató de una investigación exploratoria - descriptiva, porque analiza e investiga aspectos específicos de la realidad que aún no han sido analizados en profundidad y establece una descripción de un contexto específico. Para la validación, se tomaron las opiniones de 9 expertos, los cuales fueron seleccionados en base a su nivel de coeficiente de competencia. Se concluye que, Scrum es una metodología de gestión de proyectos ágil orientada a alcances menos conocidos.

Por otra parte, Trigoso (2021) en su investigación “Metodologías ágiles en la mejora de la gestión de proyectos en la empresa inmobiliaria Dean Valdivia Inversiones SAC, Lima – 2020”, tuvo como prioridad investigar sobre mejorar la gestión de proyectos en una empresa con metodologías ágiles, la investigación fue aplicada y de diseño pre-experimental. Para la recolección de datos se utilizó la técnica de observación y como instrumento la ficha de observación. Para la validación de los instrumentos se utilizó el juicio de expertos. Concluye enfatizando que la implementación de metodologías ágiles logra mejorar significativamente la

gestión de proyectos de software en relación a los costos ya que mejoro en un 18.16% en promedio y en el cronograma se visualiza una mejora de 16.05% en promedio.

Así como, Ganoza, et. al., (2020) en su investigación “Metodologías ágiles: Percepción de los profesionales de la Ciudad de Lima”, tuvo como finalidad identificar los requisitos, beneficios y limitaciones para adoptar metodologías ágiles en empresas privadas de la ciudad de Lima, para ello empleo el enfoque cuantitativo y como método de recolección de datos se utilizó una encuesta online a través de la herramienta Google Forms validada por siete expertos en temas de agilidad. Se concluyó que, los requisitos para adoptar agile son el cambio de mindset en la organización, capacitación mediante coachs ágiles, reestructuración en el modelo de trabajo tanto en los empleados como los lideres a través de la autonomía y retroalimentación constante y liderazgo compartido y colaborativo, en cuanto a los beneficios obtenidos se menciona a mejora en la productividad, adaptación al cambio al responder de forma ágil y flexible ante los cambios constantes, y el acercamiento del cliente al equipo de desarrollo y el aumento en la satisfacción al cliente mediante el feedback. Mientras que la resistencia al cambio y las diferencias entre la cultura de la empresa y los principios ágiles pueden generar limitaciones.

Por otro lado, como antecedentes internacionales se menciona a AbdulHafeez, et. al., (2021), en su estudio “Investigar los factores cruciales del desarrollo de software ágil a través de Enfoque compuesto”, realizado en Pakistan, esta investigación tuvo como objetivo principal, determinar los factores críticos de éxito que se debe considerar para obtener resultados exitosos en la implementación de metodologías ágiles. El trabajo tuvo un enfoque cuantitativo y el instrumento utilizado fue la encuesta que a su vez fue validada por expertos en ingeniería de software tales como profesores universitarios e investigadores de posgrado. Se concluye que los factores de dimensión más importantes son la capacidad del equipo, comunicación, e involucramiento del cliente. Además, que el factor de éxito, más importante es la capacidad del equipo mientras que el factor menos significativo es la cultura de organización del desarrollo, sin embargo, se sugiere tomarlo en cuenta.

Mientras que, Jovanović, et. al., (2016) en su investigación “Juegos retrospectivos ágiles para diferentes equipos Fases de desarrollo” recomienda incorporar el uso de juegos en las retrospectivas ágiles y en todos los eventos de

reuniones con el fin de mejorar los procesos, o evitar otras casuísticas que podrían afectar la calidad del software. La aplicación de los juegos es bastante simple, pero y a su vez aportan dinamismo al proceso, logrando mayor interés entre los participantes. Consiste en una serie de votaciones y clasificaciones lo que permite calentar al equipo y promover la interacción grupal.

En otras palabras, Prakash, et. al., (2017) en su investigación referente a “Una encuesta sobre técnicas de estimación de software en Modelos de desarrollo tradicionales y ágiles” tuvo como prioridad investigar sobre las técnicas de estimación más usadas en las metodologías ágiles. Para determinar las técnicas utilizó el enfoque de juicio de expertos siendo el póker la técnica más utilizada, story point los puntos de historia se basa en la secuencia Fibonacci o con el tamaño de una tarea modelo que todo el equipo pueda estimar y a partir de ella poder clasificar las demás tareas como pequeño, mediano y grande.

En otro orden de ideas, Carandang (2018), en su trabajo sobre “investigación de la colaboración del cliente en el desarrollo un sistema de ventas e inventario a través de metodología ágil”, utilizó el enfoque cualitativo y tuvo como finalidad determinar los efectos de framework agile en la motivación del cliente en la participación del desarrollo. Como método de recolección de datos se utilizó la entrevista y la grabación de voz. Concluye que la participación directa de los clientes en el proceso de desarrollo de software permitió un mejor entendimiento de las especificaciones y con el apoyo de los métodos y técnicas ágiles empleadas permitió aumentar la eficiencia y productividad de los involucrados, además de la detección de problemas.

Mientras que, Montoya, et. al., (2018) en su investigación sobre “Metodologías Ágiles de desarrollo, un caso de aplicación en Medellín, Colombia 2016-2017”, fue de enfoque cualitativo y la metodología aplicada consistió en una encuesta realizada a 12 empresas privadas y públicas. La investigación concluye que las metodologías ágiles más usadas y aplicadas son Scrum y Kanban y que tienen un grado de satisfacción alta si se aplican de manera correcta, además sostiene que entre las ventajas que se ha observado son mayor control en los proyectos, desarrollo y requisitos, flexibilidad en los alcances y cambios en los requisitos, entrega continua y tempranas basado en la priorización de las necesidades, validación de los clientes sobre los productos ya construidos y no sobre documentación exhaustiva.

De hecho, Neelu y Kavitha (2020) en su investigación sobre “Software Development Technique for the Betterment of End User Satisfaction using Agile Methodology”, indico que la tasa de éxito de los proyectos que implementan metodologías ágiles es de 71%, como conclusión sostiene que utilizar metodologías ágiles influye en predecir los requisitos de mano de obra en función de la complejidad del producto y el tiempo de entrega, además asegura una alta tasa de éxito de los proyectos de software.

En otras palabras, Alsaber (2021) en su investigación sobre la “Perspectives on the adherence to scrum rules in software project management”, tuvo como objetivo determinar si los desarrolladores de software comprenden las reglas de scrum y cómo afecta esto al éxito de los proyectos de software, el método que se empleo fue mixto para recopilar datos cuantitativos y cualitativos a través de la observación y posteriormente una entrevista con el equipo involucrado en el proceso de trabajo. La investigación concluye que el enfoque ágil proporciona un enfoque eficaz, Este estudio de las perspectivas sobre la adherencia a las reglas de scrum en SPM examinó con éxito las perspectivas de la industria del desarrollo de software sobre el uso apropiado de scrum en la aplicación de principios ágiles en proyectos.

Por lo mismo, Sáenz (2017) en su investigación sobre “Establishment of agile teams for software development: review of literatura”, tuvo como finalidad investigar sobre el aporte de metodologías ágiles en la conformación de equipos auto gestionados y la planificación, control y desarrollo de proyectos de software. Concluye que las metodologías ágiles son beneficiosas si el desarrollo de software pasa por el proceso de mejoramiento continuo y cooperación entre el cliente y el equipo de desarrollo, además de una persona que haga cumplir las reglas de los principios ágiles dentro de la organización, establecer tareas en cada iteración y la adopción de una cultura enfocada en generar valor a los stakeholders.

Además, Alqudah (2018), en su investigación “Un estudio empírico de la formación Scrumban basado en la selección de las prácticas de Scrum y Kanban” afirmo que cada desarrollo de software son diferentes y que la elección de la metodología ágil o la combinación de ellas dependerá del grado de dificultad, el tamaño del equipo y tiempo estimado del proyecto para lograr la calidad deseada. Mientras Scrum realiza entrega y mejora continua mediante iteraciones en el menor tiempo, Kanban proporciona la visualización del flujo de trabajo y al mismo tiempo

controla las iteraciones y restringe los trabajos en curso a través de la disminución de las ocurrencias. Se concluye que se debe establecer una serie de criterios para la elección de las metodologías adecuadas.

Finalmente, Aziz (2020), en su investigación “Incrementar la productividad de los miembros del equipo en un proyecto Scrum”, Se basó en el método de la observación el cual consistió en evidenciar el trabajo realizado en el sprint 1 y sprint 2, para el sprint 2 incorporó buenas prácticas tales como, la incorporación de reuniones cortas, se utilizó la comunicación frontal para solucionar inmediatamente los problemas, se incorporó herramientas visuales para solucionar problemas, se involucró al cliente para que pueda brindar retroalimentación al equipo cuando sea necesario, la programación de trabajos fueron en pareja y bien distribuidas; y finalmente se logró desglosar correctamente las tareas de usuarios en tareas únicas para facilitar la estimación del tiempo. Con ello se logró mejorar notablemente el proceso de desarrollo con respecto al primer sprint.

Referente a las categorías de la presente investigación, las metodologías ágiles, cuentan con un conjunto de buenas prácticas que permiten agilizar y entregar productos de calidad. Al respecto, (Sandst, et. al., 2021) sostiene que las practicas más habituales en el desarrollo ágil son mantener los requisitos ordenados, tener al equipo coubicado, contar con la participación constante del cliente, mantener equipos auto organizados, tener un panorama del progreso de los avances, entregas incrementales y elaborar documentos necesarios. Respecto a los beneficios más relevantes son la atención inmediata a las necesidades del cliente, generar retroalimentación en cada iteracción, establecer comunicación fluida entre el equipo y el cliente y adaptabilidad a los cambios.

Por otro lado, Alsari, et. al., (2020) sostiene que las metodologías agiles tienen su mayor potencial en la colaboración entre diferentes equipos multifuncionales y la iteración que se genera con la colaboración de los clientes. Las metodologías están conformadas por un conjunto de procedimientos que son utilizados para alcanzar los objetivos de un determinado trabajo orientado al desarrollo del software. Luego, Domagala (2021) sostiene que las metodologías agiles son secuenciales y, en muchas etapas, se basan en estándares aplicables en la gestión de proyectos, de igual modo, Maida (2015), menciona que las metodologías agiles están conformadas por un conjunto de técnicas, métodos y prácticas que guían en la planificación, desarrollo y optimización del proceso del

desarrollo de software. En conclusión, una metodología define una estrategia para saber el que, el cómo y cuándo hacer en el desarrollo y mantenimiento de un proyecto.

Existen diversas metodologías ágiles y aunque todos cumplen con los principios del manifiesto ágil, cada una de ellas tienen características propias. Según AbdulHafeez, et. al., (2020) que para el desarrollo ágil se puede seleccionar una, un par o una combinación de más de una metodología ágil para lograr la máxima calidad y productividad del software. La elección de las metodologías dependerá de la complejidad del software y lo que se quiere conseguir con ello. Para Vries (2020) el uso de la metodología Scrum en una empresa de tecnología financiera ha dado buenos resultados en muchos años, los beneficios más destacados fueron en el buen trabajo en equipo que lograba lanzamientos incrementales regulares. Además, Alsaqqa, et. al., (2020), indica que Scrum es bueno para el trabajo en equipos pequeños y cubricados, la responsabilidad recae en todo el equipo, proporciona herramientas de autoevaluación de los objetivos alcanzados frente a los requeridos.

De acuerdo con Kniberg y Skarin, (2010) menciona que mientras que SCRUM es una metodología que prescribe (mas reglas a seguir) pero que a diferencia de las metodologías tradicionales, herramientas RUP o inclusive que la metodología ágil XP tiene menor restricciones y Kanban es una metodología adaptativa (menos reglas a seguir) puede ser complemento de SCRUM porque brinda a los miembros del equipo y a los stakeholders visibilidad sobre los efectos de sus acciones, estudios sostienen que está demostrado que Kanban cambia el comportamiento de los participantes y motiva la colaboración en el centro de labores. Además, permite la visualizar los cuellos de botella, desperdicios y variabilidad. El impacto que genera también fomenta la discusión sobre posibles mejoras.

Planificación, consiste en definir en función de los objetivos del negocio en lugar de tareas, priorizando lo que aporte mayor valor al negocio. La planificación ágil tiene como base el control empírico. Para Torrecilla, et. al., (2015) la planificación en agile no es una fase, sino más bien un proceso, por lo que este plan inicial necesita ser revisado y actualizado constantemente a través de las iteraciones. Es decir, que en el camino del desarrollo se puede realizar ajustes al plan inicial las veces que sea necesario. Por otro lado, Oluwaseun, et. al., (2021) sostiene que la planificación se inicia con una visión del sistema a desarrollar, luego

se desarrolla una lista de todos los requisitos llamada Product Backlog, la cual es priorizada de acuerdo a las prioridades del cliente, luego es dividido en bloques llamados Sprint entre 2 y 4 semanas. Los Sprint son creados en una reunión de planificación que dura 8 horas para un Sprint de 4 semanas, y es generado por el Product Onwner y el equipo.

Gestión de requerimientos, es el proceso por el cual se define y controla todo lo que un sistema debe hacer para evitar problemas y falta de confianza entre todas las partes. Para Miranda, et. al., (2016) la gestión de requerimientos es el proceso en donde se especifican y revisan los servicios que debe proporcionar el sistema, así como las limitaciones sobre las que se deberá operar. Gestionar correctamente los requerimientos permitirá entender de manera completa y clara las necesidades del cliente y se podrá proporcionar la funcionalidad esperada.

Técnicas de estimaciones, se trata de un conjunto de procedimientos que se deben ejecutar para predecir el tiempo, costo y recursos que se empleara para llevar a cabo una tarea. Para (Vyas, et. al., 2018) la técnica de estimación lo define como una metodología para anticipar las medidas prácticas que son el esfuerzo y el costo medidos en horas individuales y capital en el contexto de la estimación de software. Introducir técnicas de estimación en las tareas de desarrollo permite planificar los trabajos, tiempo y esfuerzo del equipo la cual se basan en evidencias empíricas de proyectos pasados. Algunas de las técnicas manuales más usadas son Delphi, Poker, Story points, técnica de las camisetitas y como herramientas de software Monday, GanttPro o Wrike.

El desarrollo iterativo e incremental consiste en dividir un proceso en bloques pequeños con un tiempo de finalización establecido y que cada uno deberá hacer entrega de un producto funcional con la intención de proporcionar retroalimentación para el siguiente bloque. De mismo modo Solano, et. al., (2020) sostiene que el desarrollo iterativo e incremental construye una implementación parcial de la aplicación y que cada secuencia lineal o iteración produce un incremento funcional en el desarrollo del producto final. Por lo tanto, se puede concluir que, el desarrollo iterativo e incremental hace entrega de avances del producto final y que estas a su vez están conformadas y seleccionadas por requerimientos de mayor importancia o que tengan mayor impacto en las necesidades del cliente y que posteriormente se incorporaran al producto los demás requerimientos.

Productividad de equipo ágil, es la capacidad que tiene un grupo de personas multidisciplinares, auto-gestionados y enfocados en el cliente para generar valor incremental y funcional. Iqbal, et. al., (2019) sostiene que la productividad en equipos ágiles determina el desempeño general del proyecto en un proceso de desarrollo de software ágil y que para llevar a cabo se requiere de ciertos factores tales como la coordinación, la comunicación, la confianza mutua y la selección de los miembros del equipo. Para la productividad en un equipo ágil será necesario que tengan combinación de diversas habilidades, tales como la comunicación constante y que puedan auto-gestionarse eficientemente.

Por otro lado, Hernández, et. al., (2019) sostiene que para ser más productivo las organizaciones han incorporado métodos que se basan en los valores y principios propuestos en el manifiesto ágil. Estos métodos básicamente buscan desarrolladores de software motivados y altamente capacitados, además de cumplir prácticas establecidas como equipos auto organizados que trabajen a un ritmo donde se mantenga la creatividad y productividad, proporcionando productos rápidos y que solucionen problemas al cliente, pero que sean abiertos a cambios frecuentes en las especificaciones. Para ello se hace uso de métricas que en su mayoría apuntan a medir la entrega temprana y frecuente del software y la importancia o valor que agregan las tareas al producto. En su mayoría son de escala, numéricas y comparativas.

El manifiesto ágil es un conjunto de valores y principios que todo método que quiera llamarse ágil debe cumplir. Naicker, et. al., (2020) señala que el Manifiesto Ágil creado en el año 2001 proporcionó una serie de pautas firmes para el uso de los desarrolladores de software con el propósito de mejorar las prácticas tradicionales. En conclusión, el manifiesto ágil es un documento que contiene buenas prácticas basado en complacer las necesidades de los clientes y cambios improvisados en el desarrollo del software que un modelo tradicional no contempla.

Auto organización, es la capacidad que tiene un grupo de personas para dirigirse y organizarse por ellos mismos sin la necesidad de que una persona les diga lo que tienen que hacer. Para Barke, et. al., (2019) sostiene que tener autonomía, una buena comunicación y la colaboración son componentes principales para construir equipos auto organizados. En resumen, la auto organización es la en un equipo ágil es la clasificación y selección de requisitos de las iteraciones, estimar el esfuerzo para finalmente cada integrante del equipo auto

asignarse las tareas.

Colaborativo, se trata de compartir conocimiento, habilidades e ideas hacia los demás para lograr un objetivo. Venatesh et. al., (2019) sostiene que, en una cultura colaborativa, los miembros del equipo están listos para apoyarse entre sí y ofrecer su experiencia de manera proactiva. En las metodologías ágiles la colaboración es un punto importante para fortalecer los equipos de trabajo para incrementar su nivel de respuesta.

Flujo de trabajo, viene hacer la secuencia de procesos por los que pasa un trabajo desde que es concebido hasta su término y permite visualizar el comportamiento de cada actividad del proceso. Según Alaidaros et. al., (2018) sostiene que el flujo de trabajo es un método que permite visualizar el flujo de trabajo, medir y administrar el flujo de trabajo. Para ello se hace uso de la metodología Kanban la cual proporciona la comprensión, la visibilidad y el control del flujo de trabajo. Por lo general para la implementación en proyectos de software se hace uso de herramientas de gestión basado en Kanban que permiten establecer objetivos, asignar tiempos y prioridades y visualizar el avance de los procesos. Algunos de las herramientas de gestión son Trello, Jira, Asana, Azure Boards.

Satisfacción del cliente, se refiere al cumplimiento de expectativas que tienen los clientes sobre el resultado de un productos o servicio. Para Jigeesh et. al., (2015) la satisfacción del cliente es una de las características del enfoque ágil y que a su vez están orientadas a lograr el objetivo principal del enfoque ágil, además sostiene que es uno de las causas para lograr el éxito en un proyecto de desarrollo de software ágiles. La máxima prioridad de los métodos ágiles consiste en satisfacer al cliente a través de la entrega temprana y continua de software valioso y todo ello se consigue mediante la retroalimentación constante de nuestros clientes.

Entrega continua, es una de las características de las metodologías ágiles, la cual consiste en entregar nuevas versiones del producto con funcionalidad mejorando constantemente el software gracias a la retroalimentación del cliente que se lleva a cabo en cada entrega. Para Dingsyr, et. al., (2016), la entrega continua tiene como objetivo mantener constantemente el software en un estado liberable. Los beneficios propuestos de la entrega continua incluyen una mayor visibilidad, una retroalimentación más rápida y el empoderamiento de las partes interesadas.

El aseguramiento de la calidad consiste en el seguimiento o supervisión de las tareas planificadas en un proyecto, con el fin de demostrar que el proceso se

viene cumpliendo de acuerdo a lo planificado. Al respecto Mkoba, et. al., (2020) menciona que el aseguramiento de la calidad es un enfoque planificado y sistemático para la evaluación de la calidad y el cumplimiento de los estándares, procesos y procedimientos de los productos de software. Además, sostiene que algunas cualidades de la calidad del producto del software son la funcionalidad idónea, rendimiento, eficiencia, usabilidad, confiabilidad, mantenibilidad entre otros. De tal manera que podemos concluir que el aseguramiento de la calidad debe estar presente en todas las fases del proceso de desarrollo de software para lograr el nivel de calidad deseado en el producto y proceso.

Las actividades referentes al control y aseguramiento de la calidad durante el ciclo de vida del software permiten mayor beneficio con la finalidad de ofrecer optimización, eficiencia y satisfacción de los clientes. Callejas (2017). En ingeniería del software se han diseñado diversas normas, modelos y estándares relacionados con las buenas prácticas, criterios entre otros aspectos que se debe considerar para que los softwares cuenten con la calidad demandada. Jiménez (2020). La industria del software requiere productos y servicios de alta calidad lo cual se emplea modelos, estándares y metodologías reconocidas internacionalmente. Ortiz (2016).

Las pruebas tienen costos muy altos por lo que lo dejan al final del proyecto y no consideran todas las pruebas recomendadas. Por otro lado, las pruebas son consideradas importantes para la evaluación de la calidad de un software. Vásquez (2018), sin embargo, calidad y pruebas no es lo mismo. La calidad debe estar presente en todo el proceso del desarrollo del software, si no se incluye desde el principio, es muy probable que no esté para las etapas finales. Por esta razón deben centrarse en prevenir y en las actividades de control del ciclo de vida del software. Flores (2019).

Por otro lado, Carrizo, et. al., (2018) sostiene que la calidad se obtiene mejorando día a día los procesos y aplicando procedimientos para la aplicación eficaz de métodos y herramientas, supervisar las revisiones técnicas, las pruebas de software y controlar los cambios. Para Akbar, et. al, (2018), la calidad consta de tres factores que son: El tiempo que demora el producto, el presupuesto financiero que involucra el proyecto y la expectativa o la satisfacción del cliente. Para Mercado, et. al., (2015) indica que la calidad del proyecto concierne a todos, desde el cliente, el arquitecto y los desarrolladores respecto a las historias de usuarios.

III. METODOLOGÍA

El enfoque de la investigación fue cualitativo, debido a lo que se pretendió es comprender y descubrir la calidad de software con el uso de las metodologías ágiles para posteriormente interpretarlos y confrontarlos en base a las experiencias y no ha mediciones. Para Hernández & Mendoza (2018) en un enfoque cualitativo las interpretaciones de los participantes y del investigador se convergen para crear una realidad y que además sobresale la lógica inductiva empezando desde lo particular hasta lo más general. Por otro lado, el método de investigación fue el paradigma interpretativo porque lo que se pretende es profundizar el conocimiento de las metodologías ágiles para mejorar la calidad en el software determinando la planificación y analizando la productividad del equipo a través del pensamiento, ideas, creencias de los seres humanos. Según Hernández & Mendoza (2018) el paradigma interpretativo se presenta cuando se busca encontrar un sentido a los fenómenos en base a los significados obtenidos por las personas.

3.1. Tipo y diseño de investigación

Tipo de investigación

La presente investigación fue de tipo básica, porque se basó en un marco teórico y su objetivo principal fue la de incrementar los conocimientos científicos sin someterla a ningún aspecto práctico. Siguiendo la misma línea, Carrasco (2019) sostiene que la investigación básica solo pretende ampliar y profundizar los conocimientos científicos sobre un tema y no aplicarla inmediatamente. A su vez Hernández (2018), sostiene que el tipo de investigación básica es cuando está orientado a conseguir nuevos conocimientos de modo sistemático con el objetivo de incrementar el conocimiento de una realidad.

Diseño de investigación

Para la presente investigación se empleó el diseño de investigación acción, debido a que se pretende inicialmente observar el escenario en donde acontece el problema para identificar los posibles problemas, luego recolectar información de diversas fuentes confiables, y posteriormente analizar e interpretar para finalmente realizar una propuesta para mejorar la calidad del software utilizando metodologías ágiles en la oficina de tecnología de la información de una entidad del Estado.

Para Hernández y Mendoza (2018), la investigación acción permite primero

comprender y luego resolver problemas vinculadas a un grupo de un determinado ambiente a través del aprendizaje de teorías y prácticas que luego serán escritas en un plan de acción. Para Bausela (2004), constituye una vía para reflexionar sistemáticamente sobre la práctica y que pretende mejorarla a través de la transformación, la colaboración de los sujetos en todas las fases del proceso de investigación y la realización del análisis crítico de la situación.

3.2. Categorías, Subcategorías y matriz de categorización

Las metodologías ágiles aportan agilidad en el desarrollo de software y en la entrega de valor a los clientes en corto tiempo. Además, reducen la documentación y ponen en énfasis la práctica. Para Sharma (2016) el desarrollo de software ágil es un conjunto de métodos basado principalmente en el desarrollo incremental e iterativo, donde los requerimientos y las soluciones se desarrollan mediante la colaboración de equipos con características multifuncionales y auto organizados. Para comprender con facilidad el tema ha sido dividido en categorías y subcategorías tal como se puede apreciar en la Tabla 1.

Tabla 1

Categorías, Subcategorías y matriz de categorización

Categorías	Sub categorías
Planificación y estimaciones	Gestión de requerimientos Técnicas de estimaciones Desarrollo iterativo e incremental
Productividad de equipo	Manifiesto ágil Auto organización Colaboración Flujo de trabajo
Satisfacción del cliente	Entrega continua Aseguramiento de calidad

Fuente: Sandst (2021)

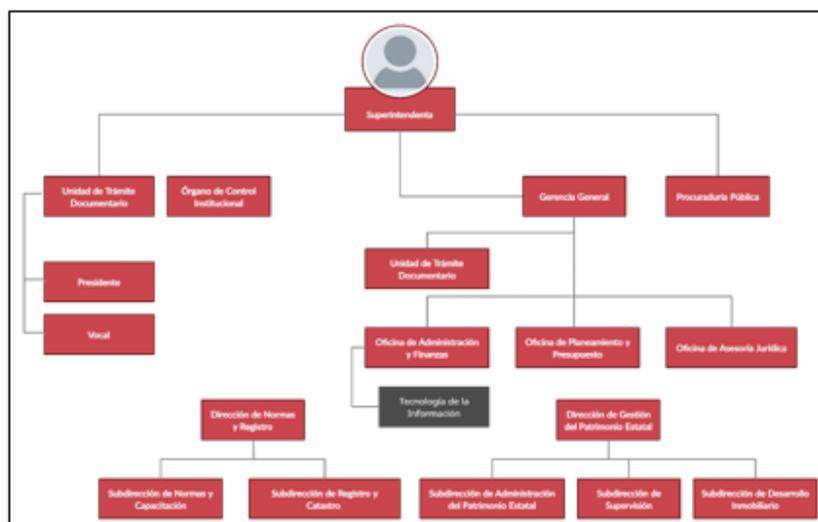
3.3. Escenario de estudio

El lugar de estudio es la oficina de Tecnología de la información (TI) de la Superintendencia Nacional del Bienes Estatales (SBN) ubicado en el distrito de San Isidro, Lima. De acuerdo al organigrama de la entidad la oficina de TI se encuentra ubicado jerárquicamente debajo de la Gerencia General (GG) y la Oficina de administración y finanzas (OAF). Para el estudio se solicitó la autorización del Supervisor de TI. El tamaño del área es de aproximadamente 130 mts² y está separada por cuatro ambientes uno para los desarrolladores de aplicaciones y administrador de base de datos; uno para infraestructura, soporte técnico y secretaria; un despacho para el supervisor responsable del área; y uno para el centro de datos.

Respecto al equipo de TI, está conformado por especialistas y alta experiencia en desarrollo de aplicaciones web, móvil, sistemas de información geográficos por el lado de desarrollo; por el lado de infraestructura se cuenta con personal calificado en la administración de servidores, redes y conectividad. El desarrollo del software se realiza mediante metodologías tradicionales y los requisitos son canalizados por una sola persona (coordinador) que posteriormente informa a los desarrolladores que estarán a cargo del proyecto y la entrega del software se entrega al final del proyecto. Las actividades realizadas en la semana son informadas por los coordinadores al supervisor de TI quien a su vez informa al área de recursos humanos (SDAPE) del trabajo del área. (Figura 1).

Figura 1

Organigrama de la entidad



3.4. Participantes

Para esta investigación se ha visto conveniente la colaboración de tres personas dentro de la oficina de TI de la entidad del Estado. Para seleccionar a las personas idóneas se ha considerado la experiencia que tienen en los procesos del desarrollo del software, el grado de responsabilidad del puesto y conocimientos en las metodologías ágiles. Las personas internas son el Coordinador de proyectos, analista programador y el coordinador de pases a producción. Todos ellos tienen la ventaja de conocer de cerca el proceso del desarrollo de software y los problemas que se presentan en cada proyecto, desde la toma de los requisitos, acuerdos con los clientes, la planificación y cronograma de las actividades y recursos hasta la capacitación, la implementación y puesta a producción de los productos de software.

3.5. Técnicas e instrumentos de recolección de datos

Una de las técnicas usada en esta investigación es la entrevista la cual consiste en entablar una conversación entre el entrevistador y puede ser a un entrevistado o a un grupo de entrevistados. El tipo de entrevista que se utilizó fue la entrevista semiestructurada que según Hernández & Mendoza (2018) el entrevistador se apoya en una guía de preguntas como base, pudiendo formular preguntas adicionales a la guía para esclarecer conceptos o ampliar la información. Otra de las técnicas usadas fue la Observación, que fue usada para recopilar datos referentes a personas, procesos y comportamientos, referente a ello Supo (2020) indica que la observación es científica solamente si el observador es consciente y que no es igual que mirar, sino más bien de evidenciar los hechos directamente o a través de medios de observación.

También se usó el análisis documental como parte de la investigación la cual consistió en interpretar y extraer parte de información de documentos utilizados en la investigación. Según Sierra (1991) sostiene que es importante observar y reconocer el significado de los elementos que conforman los documentos para su interpretación. Para Hernández & Mendoza (2018) consiste en información obtenida bajo otro contexto previamente revisado y seleccionado.

3.6. Procedimiento

Para entender mejor la investigación fue dividida en categorías y subcategorías, para ello, se tomó las mejores referencias en revistas indexadas y repositorios de tesis de posgrado, como instrumentos de recolección de datos se llevó a cabo la entrevista semiestructurada la cual fue realizada a expertos en metodologías ágiles, lo cual permitieron profundizar más en las preguntas, además se aplicó la técnica de la observación en el escenario de estudio para evidenciar como se lleva a cabo el desarrollo de software en la oficina de T.I. en todas las etapas por las cuales pasa la construcción del software. El Análisis documental fue otra técnica utilizada en la investigación. Finalmente, la información recolectada fue validada a través de la triangulación para concluir con los resultados.

3.7. Rigor científico

En la presente investigación se ha tomado en cuenta el cumplimiento del rigor científico lo cual implica una serie de requisitos que debe considerarse en la investigación cualitativa. La información obtenida con los instrumentos de recolección de datos debe pasar por el rigor científico o también llamado validez y confiabilidad según Hernández & Mendoza (2018). Si bien es cierto existe diversos requisitos que debe cumplirse, para Hernández & Mendoza (2018) la dependencia, credibilidad, transferencia y la confirmación son las más usadas.

Tabla 2

Criterios del rigor científico

Paradigma racionalista (cuantitativo) (cualitativa)	Paradigma naturalista
Validez interna	Credibilidad
Validez externa, generalización	Transferibilidad, Intercambiabilidad
Confiabilidad	Dependencia
Objetividad	Imparcialidad, Honradez Confirmabilidad, Neutralidad

La credibilidad, se obtiene cuando los hallazgos del estudio son reconocidos como verdaderos por los participantes del estudio y por aquellos que han experimentado o tuvieron contacto con el fenómeno investigado. Para Hernández & Mendoza (2018), es llamado también “máxima validez” y corresponde a que el investigador haya captado el significado de las experiencias de los participantes, específicamente con los vinculados con el planteamiento del problema.

La transferencia, según Hernández & Mendoza (2018), se refiere a que una parte o esencia de la investigación puede aplicarse en otro contexto. Confirmación, este criterio según Hernández & Mendoza (2018), está vinculado a la credibilidad de los datos y la claridad con la que se interpretó las fuentes. Dependencia, es la consistencia de los resultados y consiste en que los datos deben ser revisados e interpretados por distintos investigadores.

3.8. Método de análisis de datos

Para el análisis de esta investigación se tomó la información recolectada de la entrevista semiestructurada, las evidencias tomadas de la observación realizada a la unidad de estudio y del análisis documental y se aplicó la triangulación de las técnicas de recolección de datos con el fin de validar el estudio realizado. Según Hernández & Mendoza (2018) es conveniente recoger información de diversas fuentes y métodos de recolección de datos siempre que el tiempo y los recursos lo permitan.

Para Okuda et. al., (2005) la triangulación es una herramienta que le concede a un estudio mayor rigor, profundidad y consistencia a la información hallada. Además, que permite minimizar sesgos y aumentar la comprensión de la investigación. Para llevarlo a cabo primero se realizó la matriz de desgravación de la entrevista que consiste en listar las preguntas de los especialistas, posteriormente se realizó la matriz de codificación de la entrevista que tiene una columna de información adicional con respecto a la anterior.

3.9. Aspectos éticos

La presente investigación es un aporte al conocimiento científico y a la sociedad en general, por tanto, la investigación fue llevado a cabo de forma transparente y honesta de acuerdo a lo establecido en el código de ética aprobado en la resolución 262-2020 de la universidad, además la información recogida a través de los

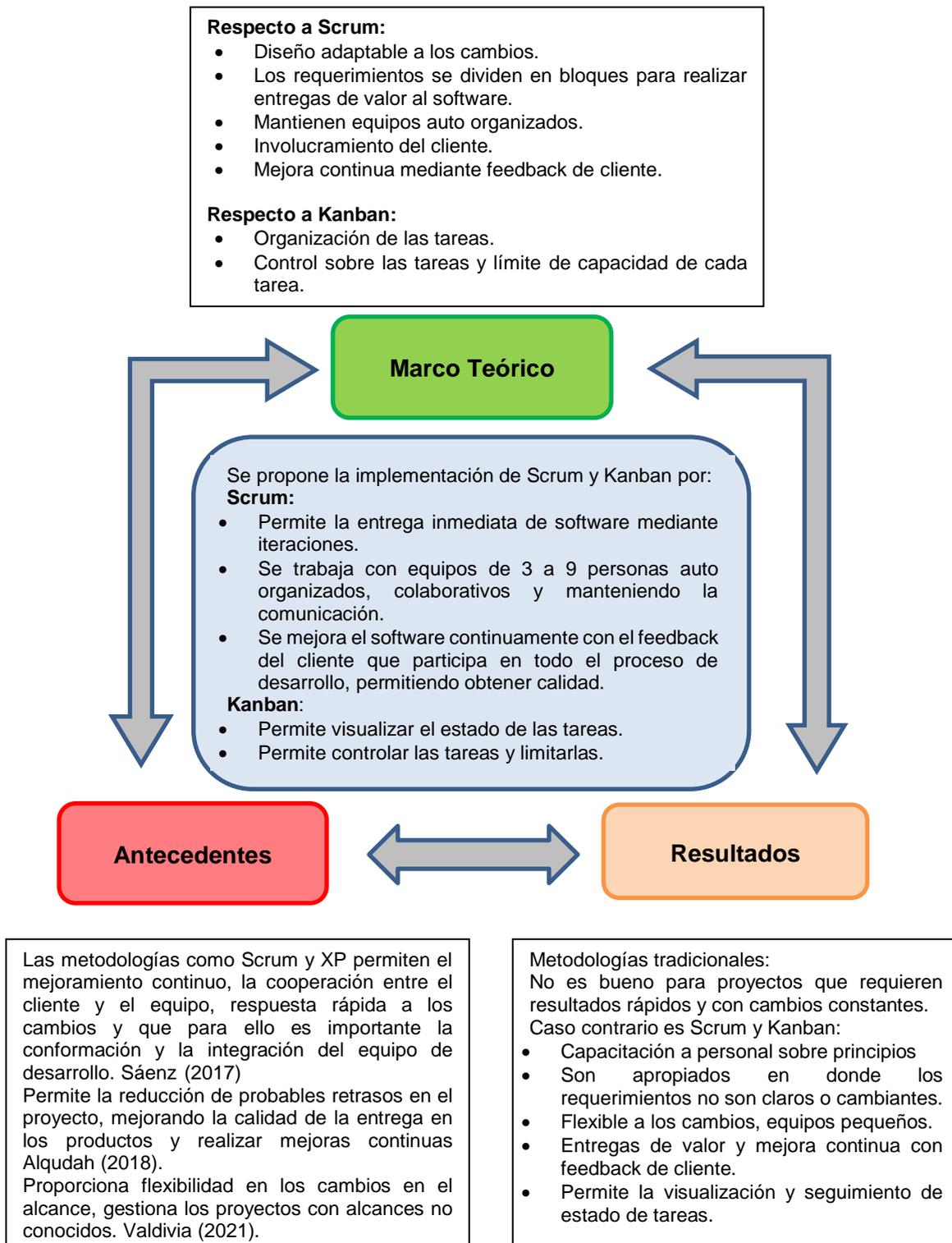
instrumentos de recolección de datos proviene de profesionales confiables y con amplia experiencia en el campo de desarrollo del software con metodologías ágiles. La recopilación de información proviene de diferentes fuentes que a su vez han sido citados debidamente siguiendo los estándares de publicación internacional APA y para validar la originalidad de la investigación se hizo el uso de la herramienta turnitin.

IV. RESULTADOS Y DISCUSIÓN

Los resultados de la investigación se obtuvieron con las siguientes triangulaciones.

Figura 2

Triangulación de Antecedentes, Marco Teórico y Resultados



De acuerdo a la triangulación de la Figura 2 se puede concluir que las metodologías ágiles resultan idóneas para mejorar la calidad del software. Para ello, se propone el uso de las mejores prácticas de las metodologías SCRUM y Kanban porque permiten agilizar la entrega de software funcional en tiempos cortos y mejorar la calidad con la mejora continua a través del feedback proporcionado por los clientes. En cuanto al monitoreo y control de los avances de las tareas, el uso de la metodología Kanban nos permitirá tener un mejor panorama de los avances de las tareas en el desarrollo del software a través de un tablero que estará constantemente actualizado. Al tener la participación del cliente en el desarrollo de software y la entrega inmediata de software funcional aumentara la satisfacción del cliente.

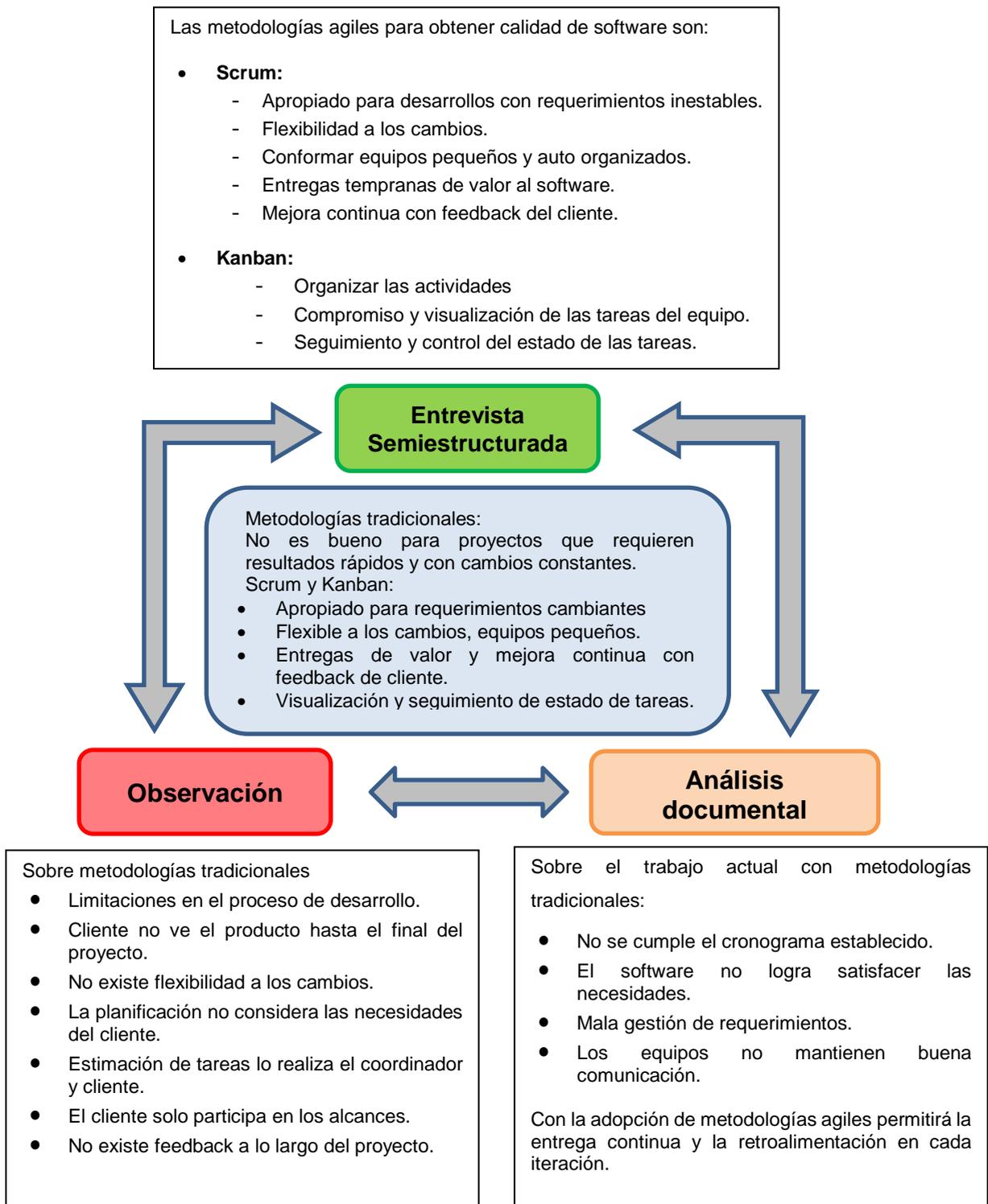
La implementación de SCRUM permitirá que los cambios de los alcances del software sean flexibles, rápidos y evitar cualquier conflicto. Al respecto (Valdivia, et. al., 2021) sostiene que la flexibilidad en la implementación aumentara al contar con un Backlog dinámico, es decir, cada vez que alguien quiera provocar un cambio en el alcance del proyecto se agregara un nuevo elemento a la cartera de pedidos sin tener que realizar una reunión para aprobarlo ya que SCRUM gira de acuerdo a las necesidades reales y cambiantes del negocio. Otro punto importante a mencionar es que el uso de ambas metodologías se complementa en varias tareas semejantes. Según Alqudah (2018), sostiene para utilizar las metodologías SCRUM y Kanban, el equipo deben considerar algunos factores tales como el tamaño del equipo, la priorización de los requisitos, el tiempo de entrega y la calidad del software.

Respecto a los beneficios que aportara ambas metodologías, el primer entrevistado sostiene que el equipo de desarrollo mejoraría su desempeño en la entrega de software con calidad gracias a la comunicación constante con el cliente. A su vez Vries (2020) concuerda con ello, ya que al aplicar metodologías ágiles destacaron la mejoría de trabajo en equipo logrando lanzamientos en tiempos regularmente cortos. Además, el primer entrevistado también sostiene que permitirá estandarizar las tareas, las entregas de software maduraran con el feedback del cliente y permitirá controlar la saturación del trabajo de cada actividad al incorporar tiempos límites. En esa misma línea Alqudah (2018), concuerda que la entrega inmediata mediante iteraciones lograra la satisfacción del cliente junto con mejora la continua que se llevara a cabo con la presencia del cliente, y (Kniberg & Skarin) que contribuye a detectar cuellos de botella, colas y desperdicios.

Respecto a los factores que deben considerarse para la implementación tanto el entrevistado 1 y 2 coinciden en que se debe realizar previamente una capacitación al personal para comprender la importancia de los principios y prácticas de las metodologías ágiles. Para (Ganoza, et. al., 2020 y AbdulHafeez, et. al., 2021), sostienen y concuerdan en que la falta de capacitación en metodologías ágiles es un obstáculo para las empresas. Tanto los miembros del equipo y los gerentes deben conocer el proceso ágil y buena relación con el cliente.

Figura 3

Triangulación de las técnicas utilizadas



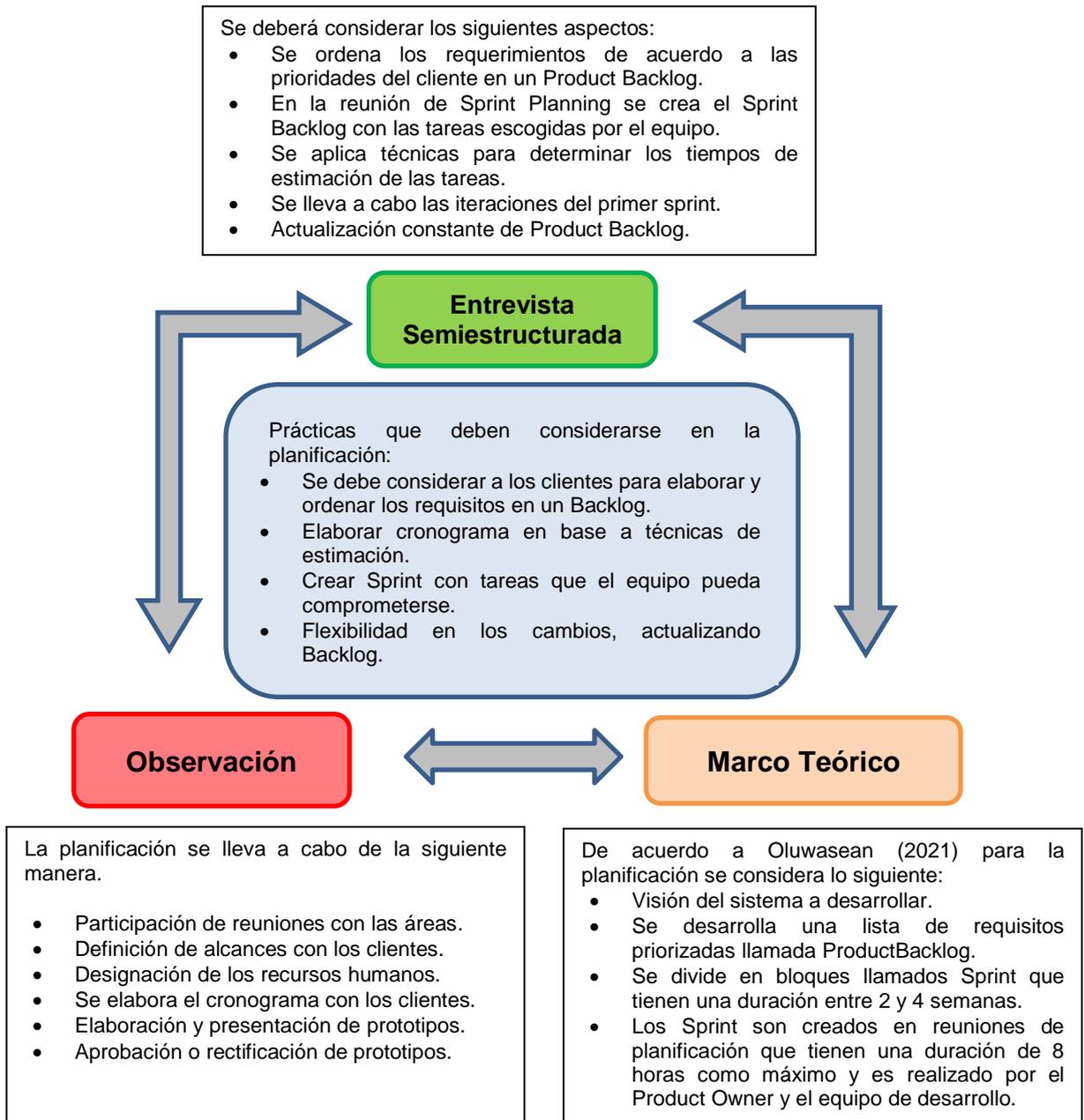
De acuerdo a la triangulación de la Figura 3 se puede concluir que existen factores que influyen en la satisfacción del cliente, las cuales son entregas tempranas de valor al software, la implementación de mejora continua y la flexibilidad ante cualquier cambio en el alcance, lo cual implica adicionar la necesidad del cliente inmediatamente después del término de la iteración actual.

Respecto a la satisfacción del cliente, Callejas (2017) sostiene que, para llegar a ello, es importante llevar un control en todos los procesos que involucre el ciclo de vida del software, para Ortiz (2016) y Jiménez (2017), sostienen en que esos controles deben llevarse a cabo con normas, modelos y estándares reconocidas internacionalmente. En desacuerdo a ello, Carrizo, et. al., (2018) sostiene que las normas, estándares y metodologías como CMMI, ISO, entre otras, no son tan sencillas aplicarla porque requiere de alto presupuesto, personal especializado, infraestructura adecuada, y tiempo. Para contrarrestar ello, se debe aplicar una propuesta personalizada de aseguramiento de la calidad que puede adaptarse a cualquier metodología de desarrollo y que consiste en utilizar tareas específicas, prácticas propias de ingeniería de software (métodos y herramientas), control de todos los trabajos que involucre el software y de los cambios que sufre, procedimientos para garantizar el cumplimiento de los estándares y mecanismos de medición y reporte.

Para Mercado, et. al., (2015), la satisfacción del cliente a través de metodologías ágiles se logra obtener con las buenas prácticas que estas proporcionan tales como el desarrollo iterativo y evolutivo que consiste en la entrega constante y mejorar continua con el feedback del cliente, las diferentes pruebas (funcionales y no funcionales) que se realizan para garantizar la calidad, también debe considerarse la definición clara de las historias de usuarios y la solución debe estar suficientemente desacoplada que permita la evolución en el tiempo.

Figura 4

Triangulación de Entrevista Semiestructurada, Observación y Marco Teórico



De acuerdo a la triangulación de la Figura 4 se concluye que la planificación y las estimaciones se inician con los alcances o requerimientos que son presentados por el representante de los clientes (Product Owner) en formato claro y preciso (Historias de usuarios) para que el equipo de desarrollo pueda entenderlo. Posteriormente en una reunión de planificación (Sprint Planning) el equipo de desarrollo deberá elegir las tareas que serán capaces de terminar (primer Sprint). Luego, se deberá estimar los tiempos que demoraran en terminarlo para crear una

hoja de ruta (plan de acción), para ello desglosan el trabajo en acciones más asequibles para tener una visión más detallada de todos los pasos requeridos. La estimación de tiempo debe apoyarse con técnicas de estimación como Scrum Poker, Story Points, votación por puntos, entre otras.

En la entrevista semiestructurada de acuerdo a las respuestas de la entrevista 1 y 2, coinciden en que para llevar una planificación y estimación se debe ordenar los requerimientos de acuerdo a las necesidades del negocio, para ello, se debe definir un rol para que una persona pueda definir el backlog junto con el cliente. Además, el equipo de desarrollo será el que defina con que tareas se va comprometer para la entrega del primer sprint.

La cual coincide con Torrecilla, et. al., (2015) que sostiene que la planificación se debe llevar a cabo en función de los objetivos del negocio, es decir se debe ordenar las prioridades en un backlog para que el equipo pueda seleccionar de ahí las tareas que se desarrollaran inicialmente. Sostiene además que el plan inicial puede ser revisado y actualizado constantemente a través de las iteraciones. Caso contrario a lo observado en el área de estudio, lo cual se pudo observar la poca participación del equipo de desarrollo en las reuniones de alcance y que la definición del cronograma lo realizaba los propios clientes con el coordinador.

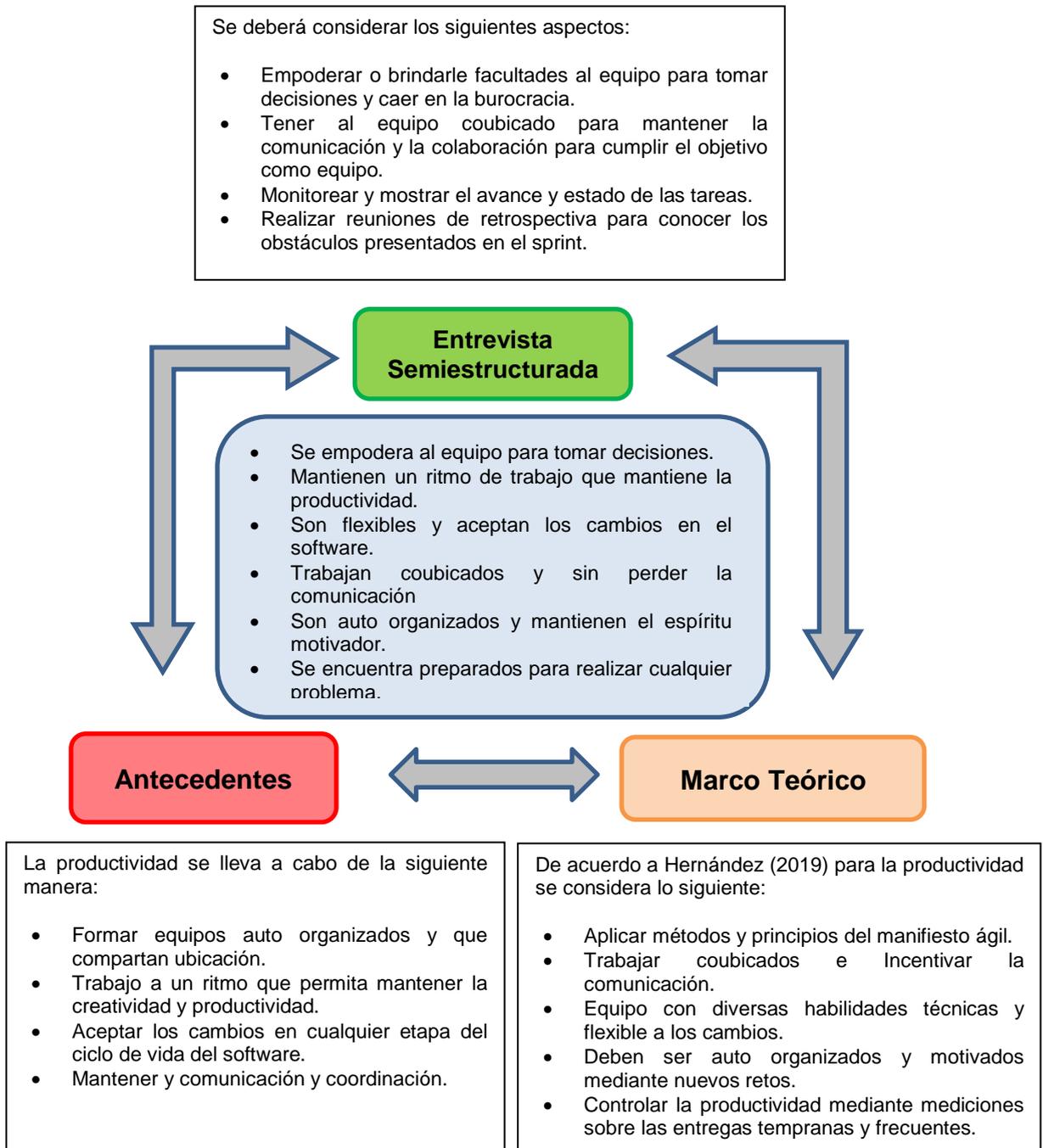
Tabla 3

Categorías, Subcategorías emergentes de Planificación y Estimaciones

Categorías	Sub categorías
Técnicas de estimación	Swim Lane Bulk Scrum Pocker Pivote 5

Figura 5

Triangulación de Entrevista Semiestructurada, Antecedentes, Marco Teórico



De acuerdo a la triangulación de la figura 5, se concluye que para conseguir buenos resultados a través de los equipos es necesario que mantengan un ritmo sostenido, que la flexibilidad ante los cambios en cualquier situación sea aceptada, además deben ser personas auto organizadas y la distribución sea cubicado para mantener una estrecha comunicación. Sobre ello el entrevistado 2 y 3 sostienen que los

equipos funcionan mejor cuando son empoderados ya que les atribuye ciertas tareas de decisión que anteriormente solo competía al jefe o coordinador. También concuerdan que para lograr la productividad del equipo es importante que el equipo se encuentre coubicados para mantener una comunicación fluida entre los equipos y clientes para sostener a un mismo ritmo el trabajo y la colaboración, evitando posibles malos entendidos.

Respecto a los antecedentes, Aziz (2020) sostiene que en el análisis que se realizó a un equipo SCRUM determinaron que la buena comunicación mejoro el desempeño y la comprensión de las tareas, pudiendo solicitar ayuda y realizar consultas recibiendo el apoyo inmediato. Para ello es necesario que la ubicación es la adecuada y que tengan las herramientas necesarias para discutir y explicar problemas. Finalmente sostiene que la reunión SCRUM y el involucramiento del cliente permitían resolver dudas rápidamente sobre las tareas del desarrollo. Además, sostuvo que para la productividad no sea afectada por las diferencias culturales, será necesario generar las buenas relaciones personales, una cultura de quipo abierta mediante la comunicación y una cultura empresarial abierta con un sistema de valores iguales. En esa misma línea, Sáenz (2017), concuerda que el nivel de comunicación del equipo es vital para obtener la mejor productividad del equipo.

Respecto al marco teórico, Iqbal, et. al., (2019), concuerda que la productividad de los equipos de desarrollo juega un papel importante dentro de un proyecto de software y que para llegar a obtener buenos resultados se debe considerar la coordinación, la comunicación, la confianza y además de la selección o conformación de los miembros del equipo en base a sus destrezas. Otras características tales como el empoderamiento del equipo, la coordinación los requisitos en forma de historias de usuarios para un mejor entendimiento pueden ayudar al mejor rendimiento.

V. CONCLUSIONES

Primera: En relación al objetivo general, se concluye que, las metodologías ágiles SCRUM y KANBAN son las idóneas para integrarlos en el proceso de desarrollo de software ya que resultan fácil de implementarlas en conjunto, pero para ello, debe considerarse algunos factores para el éxito en la implementación tales como la capacitación del personal respecto a los principios y procedimientos que emplean dichas metodologías, los roles deberán ser establecidos adecuadamente y de acuerdo a la capacidad y destreza del personal, además se debe formar equipos en donde prevalezca la comunicación, la colaboración, la auto organización para ello, será fundamental generar un espacio de trabajo coubicado donde el equipo de desarrollo se sienta cómodo con autoridad a tomar decisiones.

Segunda: Para el primer objetivo específico, se concluye que, los alcances o requerimientos deben ser entendibles y claros por el equipo de desarrollo para ello, el Product Ownner (PO) debe presentarlo a través de técnicas (historias de usuarios-HU) donde se presenta de forma clara y precisa. A su vez el PO deberá ordenar los requerimientos en base a las prioridades del negocio, el equipo de desarrollo deberá planificar los entregables (Sprint) en base a las cantidades de HU que el equipo se comprometa. Deberán desglosar lo más detallado posible para entender bien la tarea y estimarlos en base a técnicas especializadas.

Tercera: Para el segundo objetivo específico, se concluye que, el equipo produce más cuando son considerados en nuevos retos y objetivos, cuando tienen mayor libertad a tomar decisiones, y soluciona inconvenientes rápidamente al mantener la comunicación y la colaboración. Mantienen un ritmo de trabajo sostenido gracias a las reuniones sostenidas y al feedback del cliente en cada entregable de sprint. Además, son flexibles porque aceptan los cambios en cualquier momento de los alcances.

Cuarta: Para el tercer objetivo específico, se concluye que, los clientes obtienen mayor satisfacción cuando la entrega de software se realiza inmediatamente y con la aportación de ellos mismos. Además, se notó mayor grado de satisfacción con la mejora continua efectuada después de cada iteración. Otro factor importante que se observó fue la flexibilidad de que los clientes puedan incorporar nuevas funcionalidades en el proceso a comparación de la metodología tradicional que solo se hacía al término del proyecto.

VI. RECOMENDACIONES

Primera:

Se recomienda capacitación de todo el personal involucrado en el desarrollo de software del uso de las metodologías ágiles, de sus principios y prácticas, también deberá estar considerado alta dirección y los representantes de cada área. Respecto a los roles, estos deberán estar asignados adecuadamente a personas de acuerdo a su perfil y desenvolvimiento. Respecto al equipo de desarrollo, todos deberán tener conocimiento de los procesos del negocio y de T.I para desenvolverse en cualquier situación que lo amerite, en cuanto a la ubicación de trabajo deberán estar coubicados para no perder la comunicación, colaboración y resolver dudas o problemas inmediatamente. El equipo debe tener libertad de tomar decisiones creativas que conlleven a mejorar el desarrollo del software.

Segunda:

Se recomienda la participación y comunicación constante entre el cliente y el equipo de desarrollo para crear la pila del producto, para mejorar constantemente con la retroalimentación hasta lograr la calidad del software. Las estimaciones de las tareas deben realizarse con técnicas que permitan acercarse más a la realidad.

Tercera:

Se recomienda, formar equipos entre 3 a 9 personas por temas de espacios, no perder la fluidez de la comunicación y la colaboración. El equipo debe ser capacitado para conocer los principios del manifiesto ágil y empoderarlos para poder tomar decisiones libremente sin encontrar obstáculos, es decir las funciones deben ser horizontales.

Cuarta:

Se recomienda, que las tareas comprometidas sean finalizadas para que la entrega sea completa y que los cambios en los alcances sean incluidos en el product backlog para el siguiente entregable y no afectar la iteración en curso.

Otro aspecto importante a considerar es que el cliente participe en todo el proceso del software para contribuir en la madurez del software.

REFERENCIAS

- AbdulHafeez, M., Ansar, S., Quadri, N., Usman, S., Mohd, H. & Basit S. (2021). Investigating Crucial Factors of Agile Software Development through Composite Approach. Tech Science Press. DOI: 10.32604/iasc.2021.014427
- Alsaqqa, S., Sawalha, S., & Abdel-Nabi, H. (2020). Agile Software Development: Methodologies and Trends. *International Journal of Interactive Mobile Technologies (iJIM)*, 14(11), 246. doi:10.3991/ijim.v14i11.13269
- Akbar, M. A., Sang, J., Khan, A. A., Fazal-E-Amin, Nasrullah, Shafiq, M., Xiang, H. (2018). Improving the Quality of Software Development Process by Introducing a New Methodology–AZ-Model. *IEEE Access*, 6, 4811–4823. doi:10.1109/access.2017.2787981
- Alsaber, L. (2021). “Perspectives on the adherence to scrum rules in software project management”, *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 21, No. 1, pp. 360-366, 2021 DOI: 10.11591/ijeecs.v21.i1.pp360-366
- Aziz MN (2020). Increase productivity of team members in a Scrum project. *International Journal of Natural and Social Sciences*, 7(2): 36-39. <https://doi.org/10.5281/zenodo.3957840>
- Alqudah, M., Razali, R. (2018). An Empirical Study of Scrumban Formation based on the Selection of Scrum and Kanban Practices. *Malaysia International Journal on Advanced Science, Engineering and Information Technology*. DOI: 10.18517/ijaseit.8.6.6566
- Alsari, A., Qureshi, R., & Algarni, A. (2020). Agile Framework To Transform Traditional Team. 2020 *IEEE Frontiers in Education Conference (FIE)*. doi:10.1109/fie44824.2020.9274240
- Alaidaros, H., Omar, M. y Romli, R. (2018). Identificación de criterios que afectan la tarea de seguimiento de proyectos de software del método Agile Kanban. doi: 10.1063 / 1.5055423
- Barke, H., Prechelt, L. (2019). Role clarity deficiencies can wreck agile teams. *PeerJ Computer Science*, 1-20. doi: 10.7717 / PEERJ-CS.241
- Callejas, C., Alarcon, A., Alvarez, A. (2017). Modelos de calidad del software, un estado del arte. v 13 No. 1.

- Carandang, R., Cervales, G., Rashinkar, N. (2018). investigation of client collaboration in developing a sales and inventory system through agile methodology. *innovatus*. doi: 10.11591/ijeecs.v7.i3.pp867-876
- Carrizo, D., & Alfaro, A. (2018). Método de aseguramiento de la calidad en una metodología de desarrollo de software: un enfoque práctico. *Ingeniare. Revista Chilena de Ingeniería*, 26(1), 114–129. doi:10.4067/s0718-33052018000100114
- Domagala, A., Grobler-Dębska, K., Waś, J. y Kucharska, E. (2021). Desarrollo de software ERP posterior a la implementación: actualización o reimplementación. *Ciencias Aplicadas*, 11 (11), 4937. doi: 10.3390 / app11114937
- Dingsyr, T. y Lassenius, C. (2016). Temas emergentes en el desarrollo ágil de software: Introducción a la sección especial sobre entrega continua de valor. *Tecnología de la información y el software*, 77, 56–60. doi: 10.1016 / j.infsof.2016.04.018
- Falen, R. (2020). “Scrum en la gestión de proyectos de desarrollo de software en la empresa Innovatec, Magdalena del Mar”, Universidad Cesar Vallejo.
- Farro, I. (2018). La urgencia de estrategias ágiles en las instituciones públicas. *ESAN Graduate School of Business*. Recuperado en [https://n9.cl/icl2bi]
- Flores, D., Tovar, P. (2019). Creación de una metodología óptima y eficiente para la implementación de pruebas de software. *Instituto Politécnico Nacional de México*. (c7.1726)
- Gaete, J y Villarroel, R. (2020). Enfoque de aplicación ágil con Scrum, Lean Kanban. *Revista chilena de ingeniería*. doi: 10.4067 / S0718-33052021000100141, pp. 141-157
- Ganoza, L., Godo, A., Montoya, A., Tovar, A. (2020). “Metodologías ágiles: Percepción de los Profesionales de la Ciudad de Lima”, Pontificia Universidad Católica del Perú. <https://n9.cl/7n55d>
- Hernández, G., Martínez Á., Jiménez R., Jiménez F. (2019) “Métricas de productividad para equipo de trabajo de desarrollo ágil de software: una revisión sistemática”, *Tecnológicas*, vol. 22, pp.63-81, 2019.<https://doi.org/10.22430/22565337.1510>
- H. D. Ortiz Alzate, L. G. M. M., J. Cardeño Espinosa, Y N. C. Alzate Osorno (2016.).

- Impacto del uso de objetos interactivos de aprendizaje en la apropiación de conocimiento y su contribución en el desarrollo de competencias matemáticas: un resultado de experiencia de investigación. *Rev. CINTEX*, vol. 21, n.o 1, pp. 7188, vol., n.o 1,, pp. 7188.
- Kniberg, H., & Skarin, M. (2010). *Kanban and Scrum - getting the best of both of them*. 2010. Librería del Congreso. ISBN: 978-0-557-13832-6
- Iqbal, J., Omar, M., & Yasin, A. (2019). An Empirical Analysis of the Effect of Agile Teams on Software Productivity. 2019 2nd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET). doi:10.1109/icomet.2019.8673413
- Jiménez Bibián, O. P. (2017). Pruebas de calidad aplicadas al sitio web Allison. Instituto Tecnológico de Colima.
- Jigeesh, N., Chakraborty, S., Chakravorty, T. (2015). An Empirical Study of Agile Testing Attributes for Higher Customer Satisfaction in IT Projects in India. *International Journal of Business & Information*. Sep2015, Vol. 10 Issue 3, p365-386. 22p.
- Jovanović, M., Mesquida, A., Radaković, N., Mas, A. (2016). Agile Retrospective Games for Different Team Development Phases. *Journal of Universal Computer Science*, vol. 22, no. 12 (2016), 1489-1508
- Maida, E., Pacienza, J. (2015). "Metodologías de desarrollo de software", Pontificia Universidad Católica Argentina Santa María de los Buenos Aires.
- Marinho, M., Noll, J., Richardson, I. y Beecham, S. (2019). Los enfoques basados en planes están vigentes en el desarrollo de software global ágil. Simposio internacional ACM / IEEE 2019 sobre ingeniería y medición de software empírico (ESEM). doi: 10.1109 / esem.2019.8870168
- Mendoza M. (2019). Empresas peruanas: terceras en la región en uso de metodologías ágiles. *El Comercio*. Recuperado en [<https://n9.cl/jukka>]
- Mercado, V., Zapata, J., Ceballos, Y. (2015). "Tools and good practices for software quality insurance under skilful methodologies", *Revista de Investigación Desarrollo e Innovación*, DOI: 10.19053/20278306.3277
- Miranda, J. (2016). La Gestión de requerimientos en la Ingeniería Web. *Sathiri*, 10, <http://dx.doi.org/10.32645/13906925.180>
- Montoya, L., Sepulveda, J., García, M. (2018). "Metodología Ágiles de Desarrollo, un Caso de Aplicación en Medellín, Colombia 2016 – 2017", Lámpsakos,

- no. 18, pp 58-63, 2017 DOI: 10.21501/21454086.2521
- Mkoba, E. y Marnewick, C. (2020). Marco conceptual para la auditoría de proyectos ágiles. *IEEE Access*, 8, 126460–126476. doi: 10.1109 / access.2020.3007874
- Mtsh, R., Vijayakumar, T. (2019). Effects of Agile adoption on Trust, Knowledge Sharing and Collaboration in IT Organizations. *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, 12, 2652–2655. doi: 10.35940/ijitee.K2219.1081219
- Naicker, N., & Maharaj, M. S. (2020). Investigating agile requirements engineering practices in the south african software development market. *Journal of Computing and Information Technology*, 28(1), 33-58. doi:10.20532/cit.2020.1004868
- Neelu, L., & Kavitha, D. (2021). Estimation of software quality parameters for hybrid agile process model. *SN Applied Sciences*, 3(3). doi:10.1007/s42452-021-04305-0
- Neelu, L., Kavitha, D. (2020). “Software Development Technique for the Betterment of End User Satisfaction using Agile Methodology”, *UIKTEN - Association for Information Communication Technology Education and Science*, pp 992-1002 DOI: 10.18421/TEM93-22
- Oluwaseun Alexander Dada & Ismaila Temitayo Sanusi (2021): The adoption of Software Engineering practices in a Scrum environment, *African Journal of Science, Technology, Innovation and Development*, DOI: 10.1080/20421338.2021.1955431
- Ozdenizci Kose, B. (2020). Business process management approach for improving agile software process and agile maturity. *Journal of Software: Evolution and Process*. doi:10.1002/smr.2331
- Prakash, B., Viswanathan, V. (2017). A Survey on Software Estimation Techniques in Traditional and Agile Development Models. *Indonesian Journal of Electrical Engineering and Computer Science*. DOI: 10.11591/ijeecs.v7.i3.pp867-876
- Quelal, R. E., Villavicencio, M., & Mendoza, L. E. (2018). A survey of agile software development methodologies in Ecuador. *2018 13th Iberian Conference on Information Systems and Technologies (CISTI)*. doi:10.23919/cisti.2018.8399186

- Riesener, M., Doelle, C. y Perau, S. (2021). Metodología para el modelado de sistemas iterativos en el desarrollo ágil de productos. Laboratorio de máquinas herramienta e ingeniería de producción WZL en la Universidad RWTH de Aachen
- Sáenz, F., Gutiérrez, F., & Ramos, J. (2017). Establishment of agile teams for software development: review of literature. *Dimensión Empresarial*, 16(2), 39-54. DOI: <http://dx.doi.org/10.15665/dem.v16i2.1901>
- Sandst, R., Reme-Ness, C. (2021). Agile Practices and Impacts on Project Success. *Journal of Engineering, Project, and Production Management*, Volume 11, Issue 3, Pages 255 – 262. 10.2478/jepm-2021-0024
- ScrumMstudy (2017). La Guía para el Cuerpo de Conocimiento de Scrum (Guía SBOK™). <https://cutt.ly/EYV6Wag>
- Sharma, K. (2016). a proposed model of agile methodology in software development. *international journal of engineering sciences & research technology*, 5(7), 531–537. <https://doi.org/10.5281/zenodo.57031>
- Solano, E., & Porras, D. (2020) El modelo iterativo e incremental para el desarrollo de la aplicación de realidad aumentada Amón_RA. *Tecnología en Marcha*. Edición especial. Escuela de Arquitectura y Urbanismo, Diciembre 2020. Pág 165-177 <https://doi.org/10.18845/tm.v33i8>
- Torrecilla-Salinas, C. J., Sedeño, J., Escalona, M. J., & Mejías, M. (2015). Estimating, planning and managing Agile Web development projects under a value-based perspective. *Information and Software Technology*, 61, 124–144. doi:10.1016/j.infsof.2015.01.006
- Trigoso, C. (2021). “Metodologías ágiles en la mejora de la gestión de proyectos en la empresa inmobiliaria Dean Valdivia Inversiones SAC, Lima – 2020”, Universidad Cesar Vallejo.
- Valdivia, J., Bedregal, N., Castañeda, E. (2021). “Private LTE Network Service Management Model, based on Agile Methodologies for Big Mining Companies”, *International Journal of Advanced Computer Science and Applications*, Vol. 12, No. 4, 2021
- Vries, C. (2020). Selecting a scaled agile approach for a fin-tech company. *South African Journal of Industrial Engineering* November 2020 Vol 31(3) Special Edition, pp 196-208. doi: 10.7166/31-3-2432
- Yauri, H. (2021). Metodologías ágiles: ¿Cuál es el reto de su implementación en

empresas grandes del Perú? El Comercio. Recuperado en
[<https://n9.cl/lvsb0>]

ANEXOS

Anexo 1:

Matriz de categorización

Título: Metodologías ágiles para mejorar la calidad del software en una entidad del estado, Lima 2021

Autor: Williams Jesus Arenas Saldaña

Problema General	Objetivo General	Categorías	Subcategorías	Técnicas	Instrumentos
¿De qué manera las metodologías ágiles mejorarían la calidad del software en una entidad del estado, Lima 2021?	Proponer metodologías ágiles para mejorar la calidad del software en una entidad del estado, Lima 2021				
Problemas Específicos	Objetivos Específicos:				
¿Cómo es la planificación y estimación de las metodologías ágiles en la calidad de software en una entidad del Estado, Lima 2021?	Determinar la planificación y estimaciones de las metodologías ágiles para mejorar calidad de software en una entidad del Estado, Lima 2021	<ul style="list-style-type: none"> Planificación y estimaciones 	<ul style="list-style-type: none"> Gestión de requerimientos Técnicas de estimaciones Desarrollo iterativo e incremental 	<ul style="list-style-type: none"> Entrevista Semi estructurada Observación 	<ul style="list-style-type: none"> Guía de entrevista Guía de observación Ficha de análisis documental
¿Cómo aumentar la productividad del equipo para mejorar la calidad del software en una entidad del estado, Lima 2021	Analizar la productividad del equipo en las metodologías ágiles para mejorar la calidad de software en una entidad del Estado, Lima 2021	<ul style="list-style-type: none"> Productividad de equipo 	<ul style="list-style-type: none"> Manifiesto ágil Auto organización Colaboración Flujo de trabajo 	<ul style="list-style-type: none"> Análisis documental 	
¿Cuál es la satisfacción del cliente usando las metodologías ágiles para mejorar la calidad de software en una entidad del Estado, Lima 2021?	Analizar la satisfacción del cliente usando las metodologías ágiles para mejorar la calidad de software en una entidad del Estado, Lima 2021	<ul style="list-style-type: none"> Satisfacción del cliente 	<ul style="list-style-type: none"> Entrega continua Aseguramiento de calidad 		

Fuente: Sandst (2021)

Anexo 2:

Preguntas para las entrevistas a profundidad

1. ¿De qué manera las metodologías ágiles mejorarían la calidad del software en una Entidad del Estado?
2. ¿Cuáles son los beneficios de implementar las metodologías ágiles en el desarrollo del software en comparación con las metodologías tradicionales en una Entidad del Estado?
3. ¿Qué metodologías ágiles propondría para mejorar la calidad del software en una Entidad del Estado?
4. ¿Cómo se llevaría a cabo la planificación y estimaciones en las metodologías ágiles en una Entidad del Estado?
 - a. ¿Qué implica llevar a cabo una buena gestión de requerimientos en las metodologías ágiles?
 - b. ¿Qué técnicas de estimaciones deberían considerarse en las metodologías ágiles?
 - c. ¿Cuál es la importancia del desarrollo iterativo e incremental en las metodologías ágiles?
5. ¿Cómo podemos aumentar la productividad del equipo para mejorar la calidad de software en una Entidad del Estado?
 - a. ¿Cuáles son los principios más importantes que aporta el manifiesto ágil para el equipo de desarrollo?
 - b. ¿Cómo se podría construir equipos con auto organización con las metodologías ágiles?
 - c. ¿En qué se debe mejorar para que existe mayor colaboración en equipo?
 - d. ¿Qué herramientas se podría utilizar para monitorear y controlar el flujo de trabajo de los equipos utilizando metodologías ágiles?
6. ¿Qué debe tomarse en cuenta en el proceso de desarrollo para obtener la satisfacción del cliente usando metodologías ágiles?
 - a. ¿Cómo se realiza la entrega continua para mejorar la calidad de software en los métodos ágiles?
 - b. ¿Cómo se puede asegurar la calidad del software mediante las metodologías ágiles?

Anexo 3:

Matriz de desgravación de entrevistas

N°	Preguntas	Entrevistado 1 – Scrum Master
1	¿Qué factores deberían tomarse en cuenta para que la implementación de metodologías ágiles proporcionen calidad al desarrollo de software en una Entidad del Estado?	<p>Las personas que estén involucradas en el desarrollo del software tales como los supervisores, coordinadores, analistas, desarrolladores y clientes deberán conocer los principios y prácticas esenciales del manifiesto ágil, para ello deberán ser capacitados por una persona certificada, además de ser abiertos al cambio de nuevas tecnologías (uso de nuevas herramientas).</p> <p>Conocer el grado de madurez en el que se encuentra el desarrollo del software.</p> <p>Tener claro las funciones que propone las metodologías ágiles y posteriormente asignar las personas idóneas en cada cargo.</p> <p>La adaptabilidad del equipo de desarrollo deberá realizarse gradualmente a la nueva forma de trabajo.</p> <p>Establecer correctamente los sprints y el desglose de las actividades y/o tareas para facilitar el desarrollo y seguimiento de las tareas con el fin de evidenciar de forma temprana cualquier error en el software y mejorarlo antes que finalice el proyecto completo.</p> <p>Participación constante del cliente en todo el proceso de desarrollo del software para perfeccionar el software con su feedback.</p> <p>Por otro lado, las pruebas de automatización pueden elevar aún más la calidad del software.</p>

2	<p>¿Cuáles son los beneficios de implementar las metodologías ágiles en el desarrollo del software en comparación con las metodologías tradicionales en una Entidad del Estado?</p>	<p>Desde mi punto de vista los beneficios más importantes que aporta las metodologías ágiles son:</p> <p>Permite estandarizar las tareas del desarrollo de software al área de T.I.</p> <p>Permite realizar entregas de software al cliente en intervalos cortos y con rápida respuesta a los cambios, ya que es flexible ante cualquier modificación en los requerimientos.</p> <p>Nos permite tener mayor feedback del cliente para aportar mayor valor al software.</p> <p>Prueba de funcionalidades (MVP) antes de lanzar algo más grande.</p> <p>Por otro lado, permiten conocer el estado o situación de las tareas para hacerle seguimiento o poder conocer cuando un equipo llega al WIP.</p>
3	<p>¿Qué metodologías ágiles propondría para mejorar la calidad del software en una Entidad del Estado?</p>	<p>Existen varias metodologías ágiles buenas y que podrían adaptarse, sin embargo, recomiendo Scrum porque ayuda a mejorar el trabajo en equipo y mejora la productividad y realiza entregas inmediatas de software, Kanban porque permite monitorear y ver el estado de las tareas en curso y otros como Extreme Programming (XP) y Lean que también son usadas por algunas empresas.</p>
4	<p>¿Cómo se llevaría a cabo la planificación y estimaciones en las metodologías ágiles en una Entidad del Estado?</p>	<p>La planificación en metodologías ágiles como Scrum se realiza mediante el Sprint Planning que es una reunión en la cual consiste en definir las tareas que conformaran un sprint y para ello se utilizan herramientas de estimación para puntuar las historias o tareas que se trabajaran en cada sprint. En la gestión de requerimiento, se debe contar con un rol encargado que realice la priorización del product backlog teniendo en cuenta las</p>

		<p>necesidades del cliente y el feedback que den sobre las entregas incrementales. Para estimar el tiempo de cada tarea se usa técnicas de estimaciones, entre las más usadas son Planning poker y Swim Lane Bulk. Luego de haber planificado y establecido las tareas de cada sprint estas son ejecutadas mediante el desarrollo iterativo e incremental, que es una de las principales diferencias con las metodologías tradicionales, ya que los ciclos iterativos e incrementales permiten tener un menor Time to market y tener mayor contacto e interacción con los usuarios finales.</p>
5	<p>¿Cómo podemos aumentar la productividad del equipo para mejorar la calidad de software en una Entidad del Estado?</p>	<p>Se requiere de empoderarlos, es decir darle la libertad de tomar decisiones y permitiendo que sean auto organizados. Sobre manifiesto ágil, considero que todos los principios del manifiesto ágil aportan al equipo de desarrollo, pero si tuviera que destacar algunos serían el satisfacer al cliente mediante la entrega temprana y continua de software con valor y la mejora continua a la excelencia técnica, respecto a la auto organización, es fundamental para mantener la productividad, pero para esto es básico considerar que las personas sean receptivas a este nuevo tipo de gestión que está compuesta de autonomía y sobre todo responsabilidad. Las personas y el equipo deben tener un mismo objetivo y ser claros en lo que quieren alcanzar, además deben tener diversas habilidades técnicas para poder afrontar cualquier reto o problema que se presente en el proyecto. Para que exista una mejor comunicación y colaboración es necesario que el equipo se encuentre en un mismo espacio para garantizar que la comunicación sea directa. La colaboración se mantiene activa cuando se integra a todo el equipo y a los clientes en las reuniones de definición de</p>

		<p>objetivos, planificación del producto, reuniones diarias, de retrospectiva donde pueden aportar ideas e innovación para el desarrollo del software. Sobre flujo de trabajo, los tableros kanban implementados en alguna plataforma (estilo Jira, Microsoft Planner, u otro) permiten mayor visibilidad y transparencia en el trabajo que realiza cada miembro del equipo.</p>
6	<p>¿Qué debe tomarse en cuenta en el proceso de desarrollo para obtener la satisfacción del cliente usando metodologías ágiles?</p>	<p>Debemos recibir feedback del cliente mediante las ceremonias que se realiza, Sobre entrega continua, teniendo sprint cortos que permitan tener ciclos iterativos, además, debemos acondicionar personas, herramientas y procesos. Las personas se forman cambiando y madurando el mindset cultural, las herramientas para la integración, entrega y despliegue continuo ayudan para automatizar tareas que antes consumían tiempo y tenían mucho riesgo. Por último, los procesos deben estar adaptados para soportar estas metodologías. Sobre Asegurar calidad del software, se debe inspeccionar y adaptar el product backlog, y realizar pruebas de test antes de cada entrega.</p>

N°	Preguntas	Entrevistado 2 – Scrum Master
1	¿Qué factores deberían tomarse en cuenta para que la implementación de metodologías ágiles proporcionen calidad al desarrollo de software en una Entidad del Estado?	<p>Se debe capacitar a las personas sobre el uso de las metodologías ágiles, se deberá gestionar conocimiento a todo el equipo de desarrollo respecto al proceso completo del desarrollo de software para que cualquiera pueda responder ante un problema y no depender de una persona. Las reuniones de sprintReview y la sprintRetrospective luego de cada sprint son de importancia para recopilar información de cliente y considerarlo en próximas entregas y a la mejora continua. La organización de los equipos es fundamental, para ello los equipos deben formarse en grupo no menor a 3 y no mayor a 9 personas porque al tener equipos con menos personas es más fácil realizar seguimiento de las tareas del equipo y la interacción entre ellos.</p>
2	¿Cuáles son los beneficios de implementar las metodologías ágiles en el desarrollo del software en comparación con las metodologías tradicionales en una Entidad del Estado?	<p>Los beneficios que aporta la implementación de metodologías ágiles son:</p> <p>Permite adaptar el trabajo en función de las necesidades del usuario, esto implica responder al cambio, frente a seguir un plan como es el caso de la metodología waterfall,</p> <p>La entrega del software se realiza de manera iterativa e incremental, se puede ver el software funcionando y no esperar hasta que el proyecto termine como sucede en las metodologías tradicionales.</p> <p>Se divide el trabajo por sprints, esto permite que el equipo se enfoque en lo que va a desarrollar durante ese periodo, con el fin de cumplir el objetivo.</p> <p>El feedback de los usuarios o stakeholders es más frecuente, genera más transparencia sobre el trabajo realizado, porque todos pueden verlo en cualquier momento y no esperar hasta el final.</p>

		El trabajo es colaborativo y colectivo porque todos persiguen un mismo objetivo y no un objetivo por área o función como se puede dar en un desarrollo waterfall.
3	¿Qué metodologías ágiles propondría para mejorar la calidad del software en una Entidad del Estado?	En mi apreciación y experiencia podría recomendar las siguientes metodologías en ese orden: Scrum para la gestión de equipos y entregas constantes, XP para la comunicación entre programadores e iteraciones, y Kanban para planificación y visualización de tareas.
4	¿Cómo se llevaría a cabo la planificación y estimaciones en las metodologías ágiles en una Entidad del Estado?	La planificación en metodologías ágiles está basado en inspección y adaptación (control empírico), donde se prioriza las necesidades en función de los beneficios del cliente en iteraciones de 1 a 4 semanas. Sobre Gestión de requerimiento , es ordenar los requerimientos por prioridad en un backlog claro para enriquecer lo que agrega valor al cliente, además debemos tener claro cuáles son las dependencias, riesgos y esfuerzo requerido para poder satisfacer cada requerimiento. Para eso se utiliza técnicas de estimaciones , las cuales pueden ser cualquiera que se adecue mejor al equipo, generalmente usamos estimaciones relativas, aunque el pivote 5 es una buena herramienta para dimensionar complejidades. Luego de planificar y estimar las tareas se deben agrupar en sprint e iniciar los trabajos con el desarrollo iterativo e incremental , que es una de las características más importantes la cual consiste en ejecutar tareas de formas repetidas logrando sumar nuevas funcionalidades al software base.

5	<p>¿Cómo podemos aumentar la productividad del equipo para mejorar la calidad de software en una Entidad del Estado?</p>	<p>Para ello se debe fomentar la colaboración y comunicación constante entre los integrantes de los equipos ágiles, la planificación de sprints deben estar diseñados en función de la capacidad de los equipos. En el manifiesto ágil, se indica que responder al cambio es más importante con seguir un plan estricto, esto se logra de la mano con la colaboración con el cliente, además, se prefiere un software funcionando sobre documentación exhaustiva. Sobre auto organización, primero es importante que todo el equipo se conozca para ir construyendo confianza y cada uno confíe en el trabajo del resto de sus compañeros. Una comunicación abierta también es importante porque permitirá que el equipo tome decisiones. Sobre colaboración, consiste en ayudarse mutuamente para conseguir el objetivo como equipo y no de forma individual, para ello, se debe potenciar las habilidades de cada persona para construir un equipo multifuncional. Sobre flujo de trabajo, creo que más que monitorear y controlar el trabajo realizado, diría que las herramientas permiten tener visibilidad y transparencia al trabajo realizado para revisar el progreso y a partir de ahí generar una conversación sobre impedimentos o riesgos que pueda tener el equipo, para ello se puede utilizar un tablero Kanban o Jira como alternativa.</p>
6	<p>¿Qué debe tomarse en cuenta en el proceso de desarrollo para obtener la satisfacción del cliente usando metodologías</p>	<p>Los SprintReview permite revisar junto con el cliente y evaluar el producto para realizar ajustes en base al feedback si es necesario y respecto a las SprintRetrospective consiste en inspeccionar y adaptar sus métodos de trabajo para aumentar la calidad y efectividad. Sobre entrega continua, se realiza en una sesión llamada SprintReview el cual está enfocado en mostrar el avance del software, de ser posible una demo para una mejor explicación y que el</p>

	ágiles?	<p>público objetivo pueda dar feedback. La entrega continua hace que el proceso de software sea más repetitivo y predecible, mejorando notablemente la calidad y frecuencia de las entregas. Sobre Asegurar la calidad del software, hay herramientas que te hacen validación de código y otras auditorias que te ayudan a validar, por ejemplo, en casos de SW expuestas a internet pasar por un ethical hacking ayuda muchísimo.</p>
--	---------	---

N°	Preguntas	Entrevistado 3 – Scrum Master
1	<p>¿Qué factores deberían tomarse en cuenta para que la implementación de metodologías ágiles proporcionen calidad al desarrollo de software en una Entidad del Estado?</p>	<p>El equipo de desarrollo debe ser altamente competitivo, para ello, deben ser capacitados de acuerdo a sus funciones, deben saber adaptarse a las exigencias de la situación y contar con alto grado de autonomía. Es determinante gestionar eficientemente los requisitos, para eso se deberá identificar y si es necesario desglosar lo necesario en tareas para agruparlas de forma correcta para crear sprint en base a tareas que el equipo de verdad pueda realizarlo en el tiempo que se ha comprometido. La conformación de los equipos debe ser balanceado en cuanto a habilidades y en cuanto al número de personas se recomienda no exceder de 9 personas. Por otro lado, incorporar un testing ágil, con pruebas continuas y tempranas, además de pruebas automatizadas ayudan a maximizar la probabilidad de la calidad lo que se está entregando.</p>
2	<p>¿Cuáles son los beneficios de implementar las metodologías ágiles en el desarrollo del software en comparación con las metodologías tradicionales en una Entidad del Estado?</p>	<p>Uno de los beneficios que te da las metodologías ágiles es que minimiza el riesgo a través de la revisión continua y la adaptación al cambio, por ejemplo, cuando se presente cambios repentinos en los requerimientos del product backlog estos se consideran en el siguiente sprint evitando mayores costos en el proyecto. Otro beneficio es la mejora de la experiencia del cliente ya que se trabaja en conjunto con el cliente lo cual se reciben el feedback rápidos y precisos que permitan llevar a cabo la mejora continua y ahorro de tiempo, otro beneficio es que se enfoca en las personas para dar importancia al equipo de trabajo y puedan trabajar de manera conjunta, productiva y de manera responsable y autónoma permitiendo un trabajo colaborativo, fluidez en la comunicación y eficacia en las entregas.</p>

3	<p>¿Qué metodologías ágiles propondría para mejorar la calidad del software en una Entidad del Estado?</p>	<p>Las metodologías ágiles que han tenido mayor impacto y que por consiguiente se usan con más frecuencia son Scrum, XP, Kanban. En scrum hace énfasis en las personas y en reducir la complejidad en el desarrollo del software, en XP son similares a diferencia que sus iteraciones son más cortas que van desde 1 a 2 semanas, además incorpora buenas prácticas para el desarrollo como test-driven development, pair programming y en cuanto a Kanban permite visualizar gráficamente el flujo de trabajo de los equipos y tener un mejor panorama de estado de cada trabajo, además que permite delimitar la capacidad de atención de tareas de un determinado equipo o área.</p>
4	<p>¿Cómo se llevaría a cabo la planificación y estimaciones en las metodologías ágiles en una Entidad del Estado?</p>	<p>La planificación y estimaciones las llevamos por mediciones y estimaciones relativas, creando un punto de referencia como unidad de medida única para cada equipo. La planificación se realiza a través del SprintPlannig antes de cada sprint, en ella el equipo se compromete con las tareas que podrá desarrollar en un sprint. Sobre gestión de requerimiento, Implica que estos requerimientos tienen que venir ya estructurados además priorizados de acuerdo a la necesidad del cliente y ver si realmente calza con el tiempo que deseamos entregar. Sobre Técnicas de estimaciones, una buena técnica es la estimación por puntos considerando la escala de Fibonacci: 1, 2, 3, 5, 8, 13 y 21 y para ello se puede utilizar herramientas como Scrum Pocker. También se puede usar la técnica de las camisetitas: XS, S, M, L, XL en función a la complejidad de la tarea. Swim Lane, planning póker, Sobre desarrollo iterativo e incremental, es importante porque permite entregar algo de valor que puede ser potencialmente desplegado al cliente para ir validando hipótesis y no esperar hasta terminar todo el proyecto para recién desplegar el producto.</p>

5	<p>¿Cómo podemos aumentar la productividad del equipo para mejorar la calidad de software en una Entidad del Estado?</p>	<p>Es importante mantener al equipo altamente motivado, se debe tener un balance de las HU y tareas que tenga cada equipo, y debe haber mucha comunicación con el equipo, además utilizar el jira para ver que estamos realizando. Sobre manifiesto ágil, las personas del negocio y los desarrolladores deben trabajar juntos de forma cotidiana a través del proyecto; construcción de proyectos en torno a individuos motivados, dándoles la oportunidad y el respaldo que necesitan y procurándoles confianza para que realicen la tarea; la forma más eficiente y efectiva de comunicar información de ida y vuelta dentro de un equipo de desarrollo es mediante la conversación cara a cara; los procesos ágiles promueven el desarrollo sostenido. Los patrocinadores, desarrolladores y usuarios deben mantener un ritmo constante de forma indefinida; la atención continua a la excelencia técnica enaltece la agilidad. Sobre auto organización, mediante la identificación de líderes internos y la motivación a adquirir retos en cada sprint, además, una de los principales desafíos que tenemos actualmente es hacer que los equipos piensen colectivamente, ya que esto conlleva a una responsabilidad compartida donde hay un solo objetivo compartido y las acciones para alcanzarlo son libres y coordinadas en el equipo. Sobre colaboración, mediante el trabajo en equipo y constante retroalimentación, además, la confianza, la integración en un equipo son cruciales para poder ayudar a llegar a una colaboración fluida. Sobre flujo de trabajo, tableros Kanban (Jira como herramienta, pero pueden ser otras como trello, asana, etc).</p>

6	¿Qué debe tomarse en cuenta en el proceso de desarrollo para obtener la satisfacción del cliente usando metodologías ágiles?	Es fundamental entender las necesidades del cliente y la buena gestión de los requerimientos y una constante comunicación bidireccional con el cliente durante todo el proceso de desarrollo. Sobre entrega continua , desplegando pequeños entregables, como MVPs que permitan ir validando al cliente su solicitud de lo que se ha desarrollado. Para asegurar la calidad , se debe realizar pruebas en cada una de las fases del desarrollo y estableciendo criterios o valores numéricos para que estas pruebas sean aceptadas. Como, por ejemplo, pruebas unitarias, pruebas de integración, pruebas de aceptación de usuario, pruebas de regresión. Usamos herramientas que dan transparencia a diferentes frentes como SonarQube, esto es gestionado por personas, Cross QA, Chapter leaders, desarrolladores del squad, objetivos de tribus y squads. Brindando esto el equipo puede mantener la calidad del sw en un tiempo y espacio requerido.
---	---	---

Anexo 4:

Matriz de codificación de la entrevista

N°	Preguntas	Entrevistado 1 – Especialista en agile	Entrevista 1 codificada
1	¿Qué factores considerar para aplicar las metodologías agiles en el desarrollo de software?	<p>Las personas que estén involucradas en el desarrollo del software tales como los supervisores, coordinadores, analistas, desarrolladores y clientes deberán conocer los principios y prácticas esenciales del manifiesto ágil, para ello deberán ser capacitados por una persona certificada, y ser abiertos al cambio de nuevas tecnologías (uso de nuevas herramientas).</p> <p>Conocer el grado de madurez en el que se encuentra el desarrollo del software.</p> <p>Tener claro las funciones que propone las metodologías agiles y posteriormente asignar las personas idóneas en cada cargo.</p> <p>La adaptabilidad del equipo de desarrollo deberá realizarse gradualmente a la nueva forma de trabajo.</p> <p>Establecer correctamente los sprints y el desglose de las actividades y/o tareas para facilitar el desarrollo y seguimiento de las tareas con el fin de evidenciar de forma temprana cualquier error en el software y mejorarlo antes que finalice el</p>	<ul style="list-style-type: none"> • Principios y practicas • Capacitación de personal • Grado de madurez • Adaptabilidad del equipo • Establecer sprints • Desglose de actividades • Participación constante del cliente • Pruebas de automatización

		<p>proyecto completo.</p> <p>Participación constante del cliente en todo el proceso de desarrollo del software para perfeccionar el software con su feedback.</p> <p>Por otro lado, las pruebas de automatización pueden elevar aún más la calidad del software.</p>	
2	<p>¿Qué beneficios se tiene al implementar las metodologías ágiles en el desarrollo del software?</p>	<p>Desde mi punto de vista los beneficios más importantes que aporta las metodologías ágiles son:</p> <p>Permite estandarizar las tareas del desarrollo de software al área de T.I.</p> <p>Permite realizar entregas de software al cliente en intervalos cortos y con rápida respuesta a los cambios, ya que es flexible ante cualquier cambio en los requerimientos.</p> <p>Nos permite tener mayor feedback del cliente para aportar mayor valor al software.</p> <p>Prueba de funcionalidades (MVP) antes de lanzar algo más grande. Por otro lado, permiten conocer el estado o situación de las tareas para hacerle seguimiento o poder conocer cuando un equipo llega al WIP.</p>	<ul style="list-style-type: none"> • Estandarizar tareas • Entregas de software • Intervalos cortos • Flexible de los requerimientos • Feedback del cliente • Mayor valor al software • Situación actual de las tareas • WIP

3	<p>¿Qué metodologías ágiles propondría para mejorar la calidad del software?</p>	<p>Existen varias metodologías ágiles buenas y que podrían adaptarse, sin embargo, recomiendo Scrum porque ayuda a mejorar el trabajo en equipo, mejora la productividad y realiza entregas inmediatas de software, Kanban porque permite monitorear y ver el estado de las tareas en curso y otros como Extreme Programming (XP) y Lean que también son usadas por algunas empresas.</p>	<ul style="list-style-type: none"> • Scrum • La productividad • Entregas inmediatas • Kanban • Monitorear • Tareas en curso • Extreme Programming (XP) • Lean
4	<p>¿Cómo se llevaría a cabo la planificación y estimaciones en las metodologías ágiles?</p>	<p>La planificación en metodologías ágiles como Scrum se realiza mediante el Sprint Planning que es una reunión en la cual consiste en definir las tareas que conformaran un sprint y para ello se utilizan herramientas de estimación para puntuar las historias o tareas que se trabajarán en cada sprint. En la gestión de requerimiento, se debe contar con un rol encargado que realice la priorización del product backlog teniendo en cuenta las necesidades del cliente y el feedback que den sobre las entregas incrementales. Para estimar el tiempo de cada tarea se usa técnicas de estimaciones, las cuales las más usadas son Planning poker y Swim Lane Bulk. Luego de haber planificado y establecido las tareas de cada</p>	<ul style="list-style-type: none"> • Sprint Planning • Herramientas de estimación • Puntuar historias • Product backlog • Feedback • Entregas incrementales • Estimar tiempo • Planning póker • Swim Lane Bulk • Time to market

		<p>sprint estas son ejecutadas mediante el desarrollo iterativo e incremental, que es una de las principales diferencias con las metodologías tradicionales, ya que los ciclos iterativos e incrementales permiten tener un menor Time to market y tener mayor contacto e interacción con los usuarios finales.</p>	
5	<p>¿Cómo podemos aumentar la productividad del equipo en metodologías ágiles?</p>	<p>Se requiere de empoderarlos, es decir darle la libertad de tomar decisiones y permitiendo que sean auto organizados. Sobre manifiesto ágil, considero que todos los principios del manifiesto ágil aportan al equipo de desarrollo, pero si tuviera que destacar algunos serían el satisfacer al cliente mediante la entrega temprana y continua de software con valor y la mejora continua a la excelencia técnica. La auto organización, es fundamental para mantener la productividad, pero para esto es básico considerar que las personas sean receptivas a este nuevo tipo de gestión que está compuesta de autonomía y sobre todo responsabilidad. Las personas y el equipo deben tener un mismo objetivo y ser claros en lo que quieren alcanzar, además deben tener diversas habilidades técnicas para poder afrontar cualquier reto o problema que se presente en el proyecto. Para que</p>	<ul style="list-style-type: none"> • Empoderarlos • Tomar decisiones • Entrega temprana • Mejora continua • Habilidades técnicas • Retrospectiva • Tableros kanban • Jira • Microsoft Planner

		<p>exista una mejor comunicación y colaboración es necesario que el equipo se encuentre en un mismo espacio para garantizar que la comunicación sea directa. La colaboración se mantiene activa cuando se integra a todo el equipo y a los clientes en las reuniones de definición de objetivos, planificación del producto, reuniones diarias, de retrospectiva donde pueden aportar ideas e innovación para el desarrollo del software. Sobre flujo de trabajo, los tableros kanban implementados en alguna plataforma (estilo Jira, Microsoft Planner, u otro) permiten mayor visibilidad y transparencia en el trabajo que realiza cada miembro del equipo.</p>	
6	<p>¿Qué se debe considerar para mejorar la satisfacción del cliente?</p>	<p>Debemos recibir feedback del cliente mediante las ceremonias que se realiza, Sobre entrega continua, teniendo sprint cortos que permitan tener ciclos iterativos, además, debemos acondicionar personas, herramientas y procesos. Las personas se forman cambiando y madurando el mindset cultural, las herramientas para la integración, entrega y despliegue continuo ayudan para automatizar tareas que antes consumían tiempo y tenían mucho riesgo. Por último, los procesos deben estar adaptados para soportar estas metodologías. Sobre Asegurar calidad del software,</p>	<ul style="list-style-type: none"> • Feedback • Ceremonias • Sprint • Ciclos iterativos • Mindset cultural • Product backlog

		inspeccionando y adaptando el product backlog .	
--	--	--	--

N°	Preguntas	Entrevistado 2 – Especialista en agile	Entrevista 2 codificada
1	¿Qué factores considerar para aplicar las metodologías ágiles en el desarrollo de software?	Se debe capacitar a las personas sobre el uso de las metodologías ágiles, se deberá gestionar conocimiento a todo el equipo de desarrollo respecto al proceso completo del desarrollo de software para que cualquiera pueda responder ante un problema y no depender de una persona. Las reuniones de sprintReview y la sprintRetrospective luego de cada sprint son de importancia para recopilar información de cliente y considerarlo en próximas entregas y a la mejora continua . La organización de los equipos es fundamental, para ello los equipos deben formarse en grupo no menor a 3 y no mayor a 9 personas porque al tener equipos con menos personas es más fácil realizar seguimiento de las tareas del equipo y la interacción entre ellos.	<ul style="list-style-type: none"> • Capacitar a las personas • Gestionar conocimiento • Reuniones de SprintReview • SprintRetrospective • Recopilar información • Mejora continua • Organización de los equipos
2	¿Qué beneficios se tiene al implementar las metodologías ágiles en el desarrollo del software?	Los beneficios que aporta la implementación de metodologías ágiles son: Permite adaptar el trabajo en función de las necesidades del usuario , esto implica responder al cambio, frente a seguir un plan como es el caso de la metodología waterfall, La entrega del software se realiza de manera iterativa e	<ul style="list-style-type: none"> • Adaptar el trabajo • Necesidades del usuario • Iterativa e incremental • Feedback de los usuarios • Trabajo colaborativo

		<p>incremental, se puede ver el software funcionando y no esperar hasta que el proyecto termine como sucede en las metodologías tradicionales.</p> <p>Se divide el trabajo por sprints, esto permite que el equipo se enfoque en lo que va a desarrollar durante ese periodo, con el fin de cumplir el objetivo.</p> <p>El feedback de los usuarios y stakeholders es más frecuente, genera más transparencia sobre el trabajo realizado, porque todos pueden verlo en cualquier momento y no esperar hasta el final.</p> <p>El trabajo es colaborativo y colectivo porque todos persiguen un mismo objetivo y no un objetivo por área o función como se puede dar en un desarrollo waterfall.</p>	
3	<p>¿Qué metodologías ágiles propondría para mejorar la calidad del software?</p>	<p>En mi apreciación y experiencia podría recomendar las siguientes metodologías en ese orden: Scrum para la gestión de equipos y entregas constantes, Extreme Programming para la comunicación entre programadores e iteraciones, y Kanban para planificación y visualización de tareas.</p>	<ul style="list-style-type: none"> • Scrum • Extreme programming • Kanban • Planificación y visualización de tareas

4	<p>¿Cómo se llevaría a cabo la planificación y estimaciones en las metodologías ágiles?</p>	<p>La planificación en metodologías ágiles está basado en inspección y adaptación (control empírico), donde se prioriza las necesidades en función de los beneficios del cliente en iteraciones de 1 a 4 semanas. Sobre Gestión de requerimiento, es ordenar los requerimientos por prioridad en un backlog claro para enriquecer lo que agrega valor al cliente, además debemos tener claro cuáles son las dependencias, riesgos y esfuerzo requerido para poder satisfacer cada requerimiento. Sobre técnicas de estimaciones, cualquiera que se adecue mejor al equipo, generalmente usamos estimaciones relativas, aunque el pivote 5 es una buena herramienta para dimensionar complejidades. Sobre desarrollo iterativo e incremental, es una de las características más importantes la cual consiste en ejecutar tareas de formas repetidas logrando sumar nuevas funcionalidades al software base.</p>	<ul style="list-style-type: none"> • Control empírico • Beneficios del cliente • Iteraciones • Prioridad en un backlog • Valor al cliente • Pivote 5 • Software base
5	<p>¿Cómo podemos aumentar la productividad del</p>	<p>Para ello se debe fomentar la colaboración y comunicación constante entre los integrantes de los equipos ágiles, la planificación de sprints deben estar</p>	<ul style="list-style-type: none"> • Colaboración y comunicación • Planificación de sprints • Capacidad de los equipos

	<p>equipo en metodologías ágiles?</p>	<p>diseñados en función de la capacidad de los equipos. Sobre manifiesto ágil, desde mi punto de vista es responder al cambio antes de seguir un plan, pero esto va de la mano con la colaboración con el cliente, además, se prefiere un software funcionando sobre documentación exhaustiva y responder al cambio sobre seguir el plan. Sobre auto organización, primero es importante que todo el equipo se conozca para ir construyendo confianza y cada uno confíe en el trabajo del resto de sus compañeros. Una comunicación abierta también es importante porque permitirá que el equipo tome decisiones. Sobre colaboración, consiste en ayudarse mutuamente para conseguir el objetivo como equipo y no de forma individual, para ello, se debe potenciar las habilidades de cada persona para construir un equipo multifuncional. Sobre flujo de trabajo, creo que más que monitorear y controlar el trabajo realizado, diría que las herramientas permiten tener visibilidad y transparencia al trabajo realizado para revisar el progreso y a partir de ahí generar una conversación sobre impedimentos o riesgos que pueda tener el equipo, para ello se puede utilizar un</p>	<ul style="list-style-type: none">• Colaboración con el cliente• Software funcionando• Responder al cambio• Equipo multifuncional• Monitorear y controlar el trabajo• Tablero kanban• Jira
--	--	--	--

		<p>tablero Kanban o Jira como alternativa.</p>	
6	<p>¿Qué se debe considerar para mejorar la satisfacción del cliente?</p>	<p>Los SprintReview permite revisar junto con el cliente y evaluar el producto para realizar ajustes en base al feedback si es necesario y respecto a las SprintRetrospective consiste en inspeccionar y adaptar sus métodos de trabajo para aumentar la calidad y efectividad. Sobre entrega continua, se realiza en una sesión llamada SprintReview el cual está enfocado en mostrar el avance del software, de ser posible una demo para una mejor explicación y que el público objetivo pueda dar feedback. La entrega continua hace que el proceso de software sea más repetitivo y predecible, mejorando notablemente la calidad y frecuencia de las entregas. Sobre Asegurar la calidad del software, hay herramientas que te hacen validación de código y otras auditorias que te ayudan a validar, por ejemplo, en casos de SW expuestas a internet pasar por un ethical hacking ayuda muchísimo.</p>	<ul style="list-style-type: none"> • SprintReview • Feedback • SprintRetrospective • Entrega continua

N°	Preguntas	Entrevistado 3 – Especialista en agile	Entrevista 3 codificada
1	¿Qué factores considerar para aplicar las metodologías ágiles en el desarrollo de software?	<p>El equipo de desarrollo debe ser altamente competitivo, para ello, deben ser capacitados de acuerdo a sus funciones, deben saber adaptarse a las exigencias de la situación y contar con alto grado de autonomía. Es determinante gestionar eficientemente los requisitos, para eso se deberá identificar y si es necesario desglosar lo necesario en tareas para agruparlas de forma correcta para crear sprint en base a tareas que el equipo de verdad pueda realizarlo en el tiempo que se ha comprometido. La conformación de los equipos debe ser balanceado en cuanto a habilidades y en cuanto al número de personas se recomienda no exceder de 9 personas. Por otro lado, incorporar un testing ágil, con pruebas continuas y tempranas, además de pruebas automatizadas ayudan a maximizar la probabilidad de la calidad lo que se está entregando.</p>	<ul style="list-style-type: none"> • Capacitación • Adaptarse a las exigencias • Autonomía • Gestionar eficientemente los requisitos • Conformación de los equipos • Habilidades técnicas • Personas • Testing ágil • Pruebas continuas • Pruebas automatizadas
2	¿Qué beneficios se tiene al implementar las metodologías ágiles en el	<p>Uno de los beneficios que te da las metodologías ágiles es que minimiza el riesgo a través de la revisión continua y la adaptación al cambio, por ejemplo, cuando se presente cambios repentinos en los requerimientos del</p>	<ul style="list-style-type: none"> • Minimiza el riesgo • Adaptación al cambio • Experiencia del cliente • Mejora continua

	<p>desarrollo del software?</p>	<p>product backlog estos se consideran en el siguiente sprint evitando mayores costos en el proyecto. Otro beneficio es la mejora de la experiencia del cliente ya que se trabaja en conjunto con el cliente lo cual se reciben el feedback rápidos y precisos que permitan llevar a cabo la mejora continua y ahorro de tiempo, otro beneficio es que se enfoca en las personas para dar importancia al equipo de trabajo y puedan trabajar de manera conjunta, productiva y de manera responsable y autónoma permitiendo un trabajo colaborativo, fluidez en la comunicación y eficacia en las entregas.</p>	<ul style="list-style-type: none"> • Ahorro de tiempo • Enfoque en las personas • Trabajo colaborativo
3	<p>¿Qué metodologías ágiles propondría para mejorar la calidad del software?</p>	<p>Las metodologías ágiles que han tenido mayor impacto y que por consiguiente se usan con más frecuencia son Scrum, Extreme Programing, Kanban. En scrum hace énfasis en las personas y en reducir la complejidad en el desarrollo del software, en XP son similares a diferencia que sus iteraciones son más cortas que van desde 1 a 2 semanas, además incorpora buenas prácticas para el desarrollo como test-driven development, pair programing y en cuanto a Kanban permite visualizar gráficamente el flujo de trabajo de los equipos y tener un mejor panorama</p>	<ul style="list-style-type: none"> • Scrum • Entreme programing • Kanban • Buenas practicas • Test-driven development • Pair programing • Flujo de trabajo en equipo

		de estado de cada trabajo, además que permite delimitar la capacidad de atención de tareas de un determinado equipo o área.	
4	¿Cómo se llevaría a cabo la planificación y estimaciones en las metodologías ágiles?	La planificación y estimaciones las llevamos por mediciones y estimaciones relativas, creando un punto de referencia como unidad de medida única para cada equipo. La planificación se realiza a través del SprintPlannig antes de cada sprint, en ella el equipo se compromete con las tareas que podrá desarrollar en un sprint. Sobre gestión de requerimiento , Implica que estos requerimientos tienen que venir ya estructurados además priorizados de acuerdo a la necesidad del cliente y ver si realmente calza con el tiempo que deseamos entregar. Sobre Técnicas de estimaciones , una buena técnica es la estimación por puntos considerando la escala de Fibonacci : 1, 2, 3, 5, 8, 13 y 21 y para ello se puede utilizar herramientas como Scrum Pocker . También se puede usar la técnica de las camisetas : XS, S, M, L, XL en función a la complejidad de la tarea. Swim Lane, planning póker , Sobre desarrollo iterativo e incremental , es importante porque permite entregar algo	<ul style="list-style-type: none"> • SprintPlanning • Priorización de necesidades • Estimación por puntos • Escala de Fibonacci • Scrum Pocker • Técnica de las camisetas • Planning póker

		de valor que puede ser potencialmente desplegado al cliente para ir validando hipótesis y no esperar hasta terminar todo el proyecto para recién desplegar el producto.	
5	¿Cómo podemos aumentar la productividad del equipo en metodologías ágiles?	<p>Es importante mantener al equipo altamente motivado, se debe tener un balance de las HU y tareas que tenga cada equipo, y debe haber mucha comunicación con el equipo, además utilizar el jira para ver que estamos realizando. Sobre manifiesto ágil, las personas del negocio y los desarrolladores deben trabajar juntos de forma cotidiana a través del proyecto; construcción de proyectos en torno a individuos motivados, dándoles la oportunidad y el respaldo que necesitan y procurándoles confianza para que realicen la tarea; la forma más eficiente y efectiva de comunicar información de ida y vuelta dentro de un equipo de desarrollo es mediante la conversación cara a cara; los procesos ágiles promueven el desarrollo sostenido. Los patrocinadores, desarrolladores y usuarios deben mantener un ritmo constante de forma indefinida; la atención continua Pruebas continuas</p>	<ul style="list-style-type: none"> • Balance de las HU • comunicación con el equipo • Jira • Conversación cara a cara • Desarrollo sostenido • Ritmo constante • Trabajo en equipo • Colaboración fluida • Tableros kanban

		<p>a la excelencia técnica enaltece la agilidad. Sobre auto organización, mediante la identificación de líderes internos y la motivación a adquirir retos en cada sprint, además, una de los principales desafíos que tenemos actualmente es hacer que los equipos piensen colectivamente, ya que esto conlleva a una responsabilidad compartida donde hay un solo objetivo compartido y las acciones para alcanzarlo son libres y coordinadas en el equipo. Sobre colaboración, mediante el trabajo en equipo y constante retroalimentación, además, la confianza, la integración en un equipo son cruciales para poder ayudar a llegar a una colaboración fluida. Sobre flujo de trabajo, tableros Kanban (Jira como herramienta, pero pueden ser otras como trello, asana, etc).</p>	
6	<p>¿Qué se debe considerar para mejorar la satisfacción del cliente?</p>	<p>Es fundamental entender las necesidades del cliente y la buena gestión de los requerimientos y una constante comunicación bidireccional con el cliente durante todo el proceso de desarrollo. Sobre entrega continua, desplegando pequeños entregables, como MVPs que permitan ir validando al cliente su solicitud de lo que se ha</p>	<ul style="list-style-type: none"> • Gestión de los requerimientos • Comunicación bidireccional • MVP(Producto viable mínimo) • Pruebas unitarias • Pruebas de integración • Pruebas de aceptación de

		<p>desarrollado. Para asegurar la calidad, se debe realizar pruebas en cada una de las fases del desarrollo y estableciendo criterios o valores numéricos para que estas pruebas sean aceptadas. Como, por ejemplo, pruebas unitarias, pruebas de integración, pruebas de aceptación de usuario, pruebas de regresión. Usamos herramientas que dan transparencia a diferentes frentes como SonarQube, esto es gestionado por personas, Cross QA, Chapter leaders, desarrolladores del squad, objetivos de tribus y squads. Brindando esto el equipo puede mantener la calidad del sw en un tiempo y espacio requerido.</p>	<p>usuarios</p> <ul style="list-style-type: none">• SonarQube• Cross QA
--	--	---	--

Anexo 5:

Matriz de entrevistados y conclusiones

N°	Pregunta	E1 – Cargo	E2 – Cargo	E3 – Cargo	Similitud	Diferencias	Conclusión
1	¿Qué factores considerar para aplicar las metodologías ágiles en el desarrollo de software?	<ul style="list-style-type: none"> • Principios y practicas • Capacitación de personal • Grado de madurez • Adaptabilidad del equipo • Establecer sprints • Desglose adecuado de actividades • Participación constante del cliente • Pruebas de automatización 	<ul style="list-style-type: none"> • Capacitar a las personas • Gestionar conocimiento • Reuniones de SprintReview • Organización de los equipos 	<ul style="list-style-type: none"> • Capacitación • Adaptabilidad del equipo • Gestionar eficientemente los requisitos • Conformación de los equipos • Habilidades técnicas • Pruebas automatizadas 	E1 = E3	E2	<ul style="list-style-type: none"> • Aplicar los principios y prácticas de las metodologías ágiles. • Los equipos deben ser debidamente capacitados y contar con habilidades técnicas suficientes para resolver problemas. • Involucrar al cliente en el proceso de desarrollo del software. • Considerar las pruebas de automatización en todos los procesos. • Gestionar correctamente los requisitos para desglosar en tareas que puedan ser terminadas. <p>Categorías y Subcategorías emergentes</p> <ul style="list-style-type: none"> • Gestionar conocimiento • Pruebas de automatización

2	<p>¿Qué beneficios se tiene al implementar las metodologías ágiles en el desarrollo del software?</p>	<ul style="list-style-type: none"> • Estandarizar tareas • Entrega constante de software • Intervalos cortos de trabajo • Flexibilidad en los requerimientos • Feedback del cliente • Visualización de la situación actual de las tareas • WIP 	<ul style="list-style-type: none"> • Trabajo iterativa e incremental • Feedback de los usuarios • Trabajo colaborativo 	<ul style="list-style-type: none"> • Minimiza el riesgo • Adaptación al cambio • Experiencia del cliente • Mejora continua • Ahorro de tiempo • Enfoque en las personas • Trabajo colaborativo 		E1, E2, E3	<ul style="list-style-type: none"> • La estandarización de tareas permitirá identificar y eliminar actividades innecesarias. • La entrega constante permitirá la satisfacción del cliente y la mejora continua se podrá realizar con el feedback del cliente. • Fomentara el trabajo colaborativo entre todos los equipos del área. • Permitirá conocer la situación actual de las tareas y asignarle un WIP • Reducirá el riesgo de fracaso siendo flexible en la atención de los requerimientos. <p>Categorías y Subcategorías emergentes</p> <ul style="list-style-type: none"> • Visualización de la situación actual de las tareas <ul style="list-style-type: none"> - WIP
---	---	---	---	---	--	------------	---

3	¿Qué metodologías ágiles propondría para mejorar la calidad del software?	<ul style="list-style-type: none"> • Scrum • La productividad • Kanban • Extreme Programing (XP) • Lean 	<ul style="list-style-type: none"> • Scrum • Extreme programming • Kanban • Planificación y visualización de tareas 	<ul style="list-style-type: none"> • Scrum • Entreme programing • Kanban • Buenas practicas • Test-driven development • Flujo de trabajo en equipo 	E1 = E2 = E3		<ul style="list-style-type: none"> • Las metodologías más usadas son Scrum, Kanban, Extreme Programing (XP) y Lean, sin embargo, se sugiere utilizar Scrum para realizar entregas o avances en tiempos cortos y para conformar equipos pequeños para controlar fácilmente las tareas y Kanban para organizar y monitorear los avances y la situación actual de las tareas. <p>Categorías y Subcategorías emergentes</p> <ul style="list-style-type: none"> • Lean • FDD • TDD • XP
4	¿Cómo se llevaría a cabo la planificación y estimaciones en las metodologías ágiles?	<ul style="list-style-type: none"> • Sprint Planning • Herramientas de estimación • Puntuar historias • Product backlog • Feedback • Entregas incrementales 	<ul style="list-style-type: none"> • Control empírico • Iteraciones • Prioridad en un backlog • Valor al cliente • Pivote 5 • Software base 	<ul style="list-style-type: none"> • SprintPlanning • Priorización de necesidades • Estimación por puntos • Escala de Fibonacci • Scrum Pocker 	E1 = E3	E2	<ul style="list-style-type: none"> • En el Sprint Planning se ordena los requerimientos priorizando las necesidades del cliente en el backlog • El equipo escogerá las tareas que crear posible desarrollar en el primer

		<ul style="list-style-type: none"> • Estimar tiempo • Planning póker • Swim Lane Bulk 		<ul style="list-style-type: none"> • Técnica de las camisetas • Planning póker 			<p>Sprint.</p> <ul style="list-style-type: none"> • Aplicar técnicas para estimar los tiempos de cada tarea. <p>Categorías y Subcategorías emergentes</p> <ul style="list-style-type: none"> • Técnicas de estimación <ul style="list-style-type: none"> - Swim Lane Bulk - Scrum Pocker - Pivote 5
5	¿Cómo podemos aumentar la productividad del equipo en metodologías ágiles?	<ul style="list-style-type: none"> • Empoderarlos • Tomar decisiones • Entrega temprana • Mejora continua • Habilidades técnicas • Retrospectiva • Tableros kanban 	<ul style="list-style-type: none"> • Colaboración y comunicación • Planificación de sprints • Colaboración con el cliente • Monitorear y controlar el trabajo • Tablero kanban 	<ul style="list-style-type: none"> • Balance de las HU • Comunicación con el equipo • Conversación cara a cara • Ritmo constante • Trabajo en equipo • Colaboración fluida • Tableros kanban 	E2 = E3	E1	<ul style="list-style-type: none"> • Empoderara al equipo para que puedan tomar decisiones y evitar la burocracia. • Practicar constantemente la colaboración y comunicación en el área. • Monitorear las tareas mediante el uso del tablero kanban. • Detectar los problemas y considerar las mejoras a través de las reuniones de retrospectiva.
6	¿Qué se debe considerar para mejorar la satisfacción	<ul style="list-style-type: none"> • Feedback • Ceremonias 	<ul style="list-style-type: none"> • SprintReview • Feedback 	<ul style="list-style-type: none"> • Gestión de los requerimientos 		E1, E2, E3	<ul style="list-style-type: none"> • Se debe aplicar la comunicación

	del cliente?	<ul style="list-style-type: none"> • Sprint • Ciclos iterativos • Mindset cultural • Product backlog 	<ul style="list-style-type: none"> • SprintRetrospective • Entrega continua 	<ul style="list-style-type: none"> • Comunicación bidireccional • Pruebas unitarias • Pruebas de integración • Pruebas de aceptación de usuarios 			<p>bidireccional en las ceremonias realizadas en cada parte del proceso para generar feedback y aplicarlo en las siguientes entregas.</p> <ul style="list-style-type: none"> • Aplicar ciclos iterativos en cada sprint para generar valor en cada entrega. • La gestión de los requerimientos es importante para obtener el mejor producto y la satisfacción del cliente.
--	--------------	--	---	--	--	--	--

Conclusiones de las entrevistas semiestructuradas

Se puede concluir que para aplicar metodologías ágiles es importante la capacitación del personal involucrado en el desarrollo de software, incluido el cliente y gerente general para entender en que consiste los principios y prácticas ágiles, además es primordial que el equipo de desarrollo sea auto gestionable y tenga diversas habilidades técnicas que permitan desenvolverse en cualquier parte del proceso de desarrollo del software, se debe mantener una comunicación constante entre los equipos de desarrollo e incluir al cliente en todo el proceso de desarrollo para obtener feedback en cada entregable y utilizarlo en mejorar continuamente la calidad del software en los próximos sprints. En cuanto a los requerimientos, estos deberán ser definidos y ordenados de acuerdo a las necesidades del cliente en una lista llamada backlog y posteriormente debe ser desglosados en tareas con el fin de que el equipo de TI pueda entenderlo para estimar correctamente los tiempos. Al utilizar metodologías ágiles permitirá estandarizar las tareas, la entrega constante de software usable permitirá la satisfacción del cliente, y se logrará la colaboración entre todos, respecto al avance de las tareas, se podrá visualizar el estado de las tareas a través de un cuadro kanban estableciendo un límite en la capacidad de atención.

Anexo 6:

Guía de Observación	
Empresa:	Entidad del Estado
Ubicación:	San Isidro
Área:	Tecnología de la Información
Observador:	Williams Jesus Arenas Saldaña
<p>Redacción del trabajo realizado por las personas que laboran dentro de la unidad de estudio, siendo T1: Coordinador de proyectos, T2: Analista programador, T3: Coordinador de pases a producción</p> <p>T1: Gestiona y coordina los nuevos proyectos y mantenimiento de softwares, participa en reuniones con las áreas para definir los alcances de los cambios y/o, mejoras o desarrollo de nuevos aplicativos, posteriormente, designa los recursos necesarios que participaran en el desarrollo del proyecto, estimación de los tiempos que requiera en todas las etapas de análisis, diseño, desarrollo y pruebas, genera el cronograma de actividades en el software Microsoft Project y luego es presentado a las áreas. Se reúne con el equipo de desarrollo para revisar los avances de los trabajos y los obstáculos presentados, informa sobre los avances al jefe del área de T.I. Se encarga de absolver las dudas del equipo referente al alcance de todos los proyectos de software. Sostiene que los aplicativos se ejecutan mediante metodologías tradicionales. Se observó que paraba ocupado en temas diferentes a proyectos de software tales como coordinación de migración de aplicativos a otros servidores y la administración de la base de datos al no contar con un DBA. Respecto a las metodologías agiles, indico que nunca ha aplicado pero que estaría dispuesto a aplicarlo en el proceso del área de T.I.</p> <p>T2: Analiza los requerimientos del proyecto de software a cargo, diseña el prototipo del software que es presentado al área usuaria, en el desarrollo del software, escribe código y genera los objetos de base de datos en el ambiente de desarrollo, realiza pruebas básicas de acuerdo al tiempo establecido. Coordina el pase a producción con el encargado. Respecto a las incidencias diarias, están son revisadas y posteriormente corregidas. Se observó malestar en el analista programador por las constantes horas extras que debía quedarse y porque no</p>	

podía cumplir con el cronograma del desarrollo del software por diversos motivos tales como diversas incidencias que requieren previo análisis para poder resolverlos y a las diversas reuniones extensas que es convocado. Además, los tiempos de desarrollo de software no son establecidos por el equipo de desarrollo y eso genera quedarse más del horario y que los softwares no salgan con las pruebas necesarias para evitar correcciones que quitan tiempo para avanzar nuevos proyectos de software. La documentación también es un factor que interfiere en el desarrollo del software.

T3: Ejecuta los pases a producción mediante los documentos de pase a producción generados por el equipo de desarrollo. Se observó malestar ya que los pases a producción no tienen horarios establecidos y se realizan constantemente y en cualquier momento del día. Considera que esto se debe a la falta de pruebas o coordinación con el cliente respecto a los alcances.

Conclusión de la Guía de Observación

De las observaciones se concluye que, se hace uso de las metodologías tradicionales para el desarrollo de software lo cual es una limitante para el proceso de desarrollo de software y por tanto en los clientes ya que el producto es entregado al final de cada proyecto y no son flexibles en cuanto a cambios en el camino. No existe una buena planificación puesto que no se realiza un orden de prioridad de las necesidades del cliente. Además, se ha notado que la estimación de tiempo de los desarrollos de software es establecida entre el coordinador y el cliente y no consideran al equipo de desarrollo que son los que realizan el trabajo. Se ha notado incomodidad en los analistas programadores por la forma de gestionar los trabajos y los recursos. Por el lado del cliente se puede apreciar que no se le involucra en el desarrollo del software y que la retroalimentación llega en la parte final del proyecto cuando se realiza la entrega del producto, lo cual impide la retroalimentación temprana en cada pasaje del proyecto, la mejora continua y la calidad del software. Referente a las metodologías ágiles, indicaron que si conocían pero que nunca lo habían aplicado en un proyecto pero que estarían de acuerdo con la implementación si se mejora todos los aspectos indicados por el investigador.

Anexo 7:

Ficha de Análisis Documental	
Ubicación:	San Isidro
Área:	Tecnología de la Información
Observador:	Williams Jesus Arenas Saldaña
El área de Tecnología de la información en estos últimos meses tuvo como proyectos el desarrollo de softwares para la automatización de los procesos de la entidad, en el reporte de los proyectos se logra visualizar que ningún proyecto cumplió con el tiempo establecido por el cronograma por diversos motivos tales como mala estimación de tiempos, mala definición y cambios en los requerimientos, mala organización del equipo de desarrollo, entre otros. El tiempo extra que demora para concluir los softwares supera las 3 semanas. Además de ello, algunas áreas presentaron un MEMO expresando su disconformidad por el resultado del producto y que requerían una pronta solución lo cual consistió que se re programe el cronograma generando mayor tiempo y personal a cargo.	

Conclusión del Análisis Documental:

Como conclusión, se puede apreciar que los proyectos de desarrollo de software no vienen cumpliendo con el cronograma establecido y al concluir el software no se está logrando satisfacer las necesidades de los usuarios por deficiencias en la gestión de los requerimientos, mala organización de los equipos de desarrollo, además la retroalimentación del cliente se obtiene al final del desarrollo del software lo cual impide mejorar constantemente la calidad del software impactando en la labor de los clientes y la imagen de la institución. Con la adopción de metodologías ágiles permitirá la entrega continua de software, permitirá obtener retroalimentación en cada iteración para mejorar continuamente, permitirá manejar de forma adecuada los requerimientos al ser flexible en los cambios y se priorizará las personas trabajando de forma grupal e integral enfatizando en la comunicación, autogestión y la colaboración entre ellos.

Tabla 4: Tiempo de retraso de los Proyectos de software

Área de Tecnología de la Información						
Nro	Proyectos	Fecha de inicio	Fecha Final	Tiempo estimado (Meses)	Fecha de Entrega	Retraso de Entrega (Meses)
1	Sistema de registro de predios del Estado	23/03/2020	25/07/2020	4	17/10/2020	3
2	Notificaciones Electronicas	08/06/2020	08/09/2020	3	12/12/2020	3
3	Mesa de partes virtual	04/04/2020	04/06/2020	2	04/08/2020	2
4	Tramite transparente	05/05/2020	05/07/2020	2	05/09/2020	2
5	Reunete Virtual y Presencial	09/10/2020	09/12/2020	2	10/02/2021	2
6	Subastas publicas	05/02/2021	05/05/2021	3	05/08/2021	3
7	Firma digital de documentos	06/03/2021	06/06/2021	3	06/08/2021	2

Anexo 8:

Carta de Solicitud de la Universidad a la Entidad del Estado



UNIVERSIDAD CÉSAR VALLEJO



"Decenio de la Igualdad de Oportunidades para mujeres y hombres"
"Año del Bicentenario del Perú: 200 años de Independencia"

Lima, 1 de octubre de 2021
Carta P. 0995-2021-UCV-VA-EPG-F01/J

Ing.
Jorge Luis Rondón Chávez
SUPERVISOR
SUPERINTENDENCIA NACIONAL DE BIENES ESTATALES

De mi mayor consideración:

Es grato dirigirme a usted, para presentar a ARENAS SALDAÑA, WILLIAMS JESUS; identificado con DNI N° 42147852 y con código de matrícula N° 6700284500; estudiante del programa de MAESTRÍA EN INGENIERÍA DE SISTEMAS CON MENCIÓN EN TECNOLOGÍAS DE LA INFORMACIÓN quien, en el marco de su tesis conducente a la obtención de su grado de MAESTRO, se encuentra desarrollando el trabajo de investigación titulado:

Metodologías ágiles para mejorar la calidad del software en una entidad del estado, Lima 2021

Con fines de investigación académica, solicito a su digna persona otorgar el permiso a nuestro estudiante, a fin de que pueda obtener información, en la institución que usted representa, que le permita desarrollar su trabajo de investigación. Nuestro estudiante investigador ARENAS SALDAÑA, WILLIAMS JESUS asume el compromiso de alcanzar a su despacho los resultados de este estudio, luego de haber finalizado el mismo con la asesoría de nuestros docentes.

Agradeciendo la gentileza de su atención al presente, hago propicia la oportunidad para expresarle los sentimientos de mi mayor consideración.

Atentamente,



Dr. Carlos Ventura Orbegoso
Jefe
ESCUELA DE POSGRADO
UCV FILIAL LIMA
CAMPUS LIMA NORTE

Autorizado
Jorge Luis Rondón Chávez
DNI 06157504

Anexo 9:

Proyectos de software tradicionales y ágiles

Figura 6

Tasas de fracaso de proyectos de software según dimensiones (Standish Group Report, 2020)

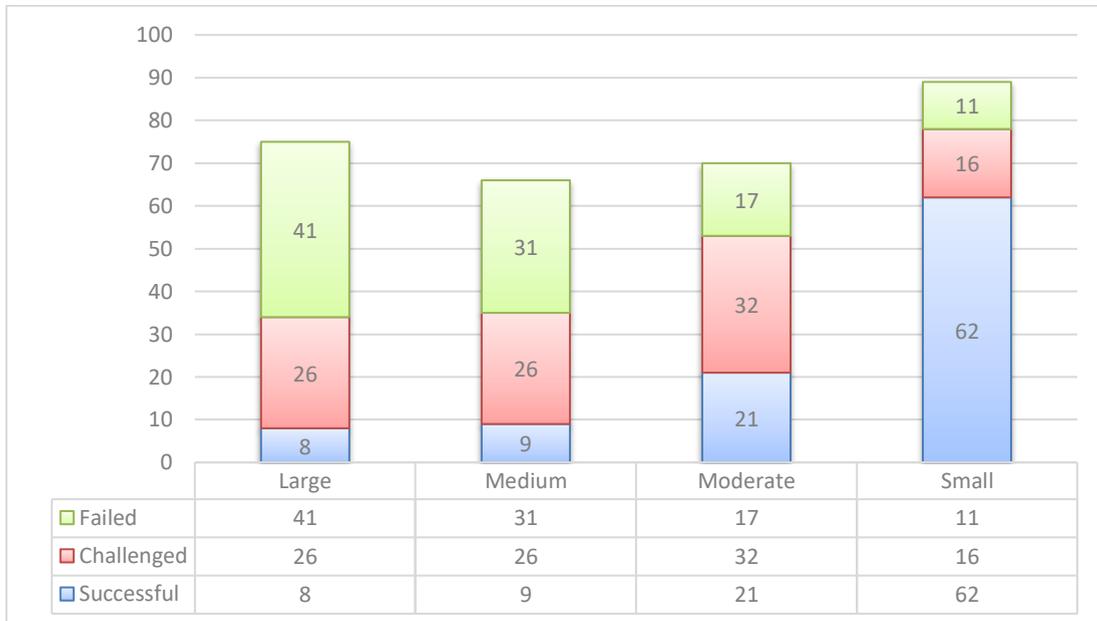
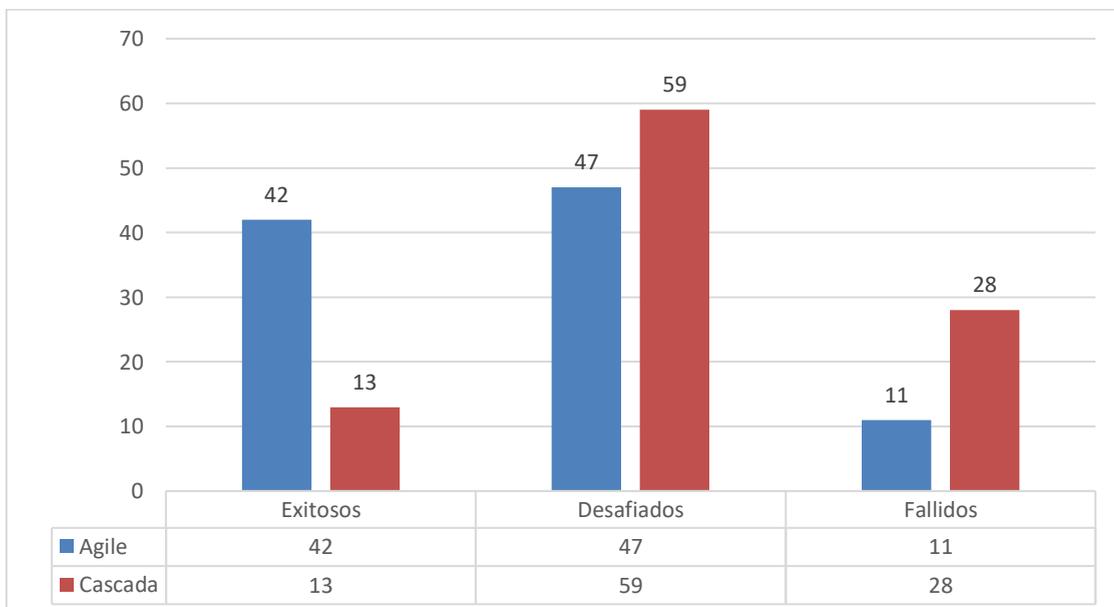


Figura 7

Comparativo entre Proyectos ágiles vs cascada (Standish Group Report, 2020)



Anexo 10:

ENTIDAD DEL ESTADO

**GUÍA PARA LA APLICACIÓN DE METODOLOGÍAS
ÁGILES EN EL PROCESO DE DESARROLLO DE
SOFTWARE**

Propuesta de investigación presentada al área de Tecnología de la Información de
una Institución Estatal

Perú, diciembre de 2021

Índice

1. El problema
 - 1.1. Formulación del problema
 - 1.2. Objetivos de la investigación propuesta
2. Plan de trabajo
 - 2.1. Etapas del trabajo, principales actividades de cada etapa
 - 2.2. Cronograma y control
3. Recursos
 - 3.1. Humanos
 - 3.2. Materiales y equipos

1. El problema

1.1. Formulación del problema

Mejorar las actividades del desarrollo de software para mejorar la calidad del software utilizando las metodologías ágiles SCRUM y KANBAN.

1.2. Objetivos de la investigación propuesta

Reducir los tiempos de entrega de software y mejorar la calidad del software continuamente con la retroalimentación de los clientes.

2. Plan de trabajo

2.1. Etapas del trabajo, principales actividades de cada etapa

- Mapeo o reconocimiento de las actividades y/o tareas del desarrollo de software.
- Capacitación del personal involucrado en el desarrollo del software.
- Establecer las funciones y personas a cargo de acuerdo al manifiesto ágil.
- Establecer niveles de dificultad en el desarrollo del software (mínimo 3).
- Establecer los alcances y criterios de aceptación con el cliente.
- Establecer equipos de desarrollo entre 3 a 9 personas
- Determinar los tipos y los tiempos de las reuniones.
- Determinar el monitoreo y control de las actividades.
- Determinar el tamaño de las iteraciones.

2.2. Cronograma y control

Plan de actividades		
N°	ACTIVIDAD	DURACION
1	Mapeo o reconocimiento de las actividades y/o tareas del desarrollo de software	2 semanas
2	Capacitación del personal involucrado en el desarrollo del software	2 semanas
3	Establecer las funciones y personas a cargo de acuerdo al manifiesto ágil	4 días
4	Establecer niveles de dificultad en el desarrollo del software	4 días
5	Establecer los alcances y criterios de aceptación con el cliente	2 días
6	Establecer equipos de desarrollo entre 3 a 9 personas	3 días
7	Determinar los tipos y los tiempos de las reuniones	4 horas
8	Determinar el monitoreo y control de las actividades	3 horas
9	Determinar el tamaño de las iteraciones	2 horas

3. Recursos

En esta sección se detalla los recursos humanos, materiales y financieros que se requiere para poder implementar la propuesta.

3.1. Humanos:

Para llevar a cabo adecuadamente la implementación de las metodologías SCRUM y KANBAN es necesario contar principalmente con 3 funciones.

Product Owner:

Representa a los usuarios o clientes y es responsable de trabajar con ellos para determinar qué características deben priorizarse del software. El PO también interactúa con el equipo de desarrollo, así como con otras partes interesadas (stakeholders), para crear y gestionar el trabajo acumulado. También motiva al equipo para que se alinee con la meta y la visión del proyecto.

Principales responsabilidades:

- Desarrolla la dirección y estrategia para los productos y servicios.
- Proporciona conocimiento sobre el producto o servicio.
- Comprender y explicar claramente las necesidades del cliente al equipo de desarrollo.
- Reunir, priorizar y gestionar los requisitos del producto.

- Asume la responsabilidad del presupuesto del producto.
- Mostrar las principales realizaciones del equipo de desarrollo al final de cada Sprint.

Scrum Master:

Es el facilitador de un equipo de desarrollo Agile. El SM es responsable de organizar reuniones, lidiar con problemas que surjan en el entorno del equipo de desarrollo. Asimismo, un SM interactúa con el propietario del producto para garantizar que la acumulación de productos esté lista para el próximo sprint. Además, garantiza que el equipo cumpla con los principios de la metodología.

Principales responsabilidades:

- Actúa como Coach del equipo a seguir los valores y prácticas de ágil.
- Elimina impedimentos y protege al equipo de interferencias externas.
- Promueve la buena relación entre el equipo y las partes interesadas.
- Protege al equipo de las distracciones de la organización.

Equipo Scrum:

El equipo Scrum es un conjunto de personas expertas en diversas tecnologías y pueden desenvolverse en diversas actividades como programador, diseñador, probador. Son los responsables de determinar la cantidad de trabajo que pueden completar en cada iteración.

El equipo Scrum debe estar compuesto de mínimo 3 personas y un máximo de 9 personas.

Principales características:

- Deben ser auto organizados, cada uno administra su propio esfuerzo y no existe un líder en el equipo.
- Deben estar comprometidos con el trabajo para cumplir con sus actividades y contribución al equipo.
- Deben ser multifuncionales, debe poseer todas los conocimientos y

habilidades para proporcionar un buen servicio o producto.

3.2. Materiales y equipos

El ambiente debe ser uno solo, en donde todo el equipo se encuentre cerca y pueda interactuar fácilmente entre ellos a través de la comunicación cara a cara y la colaboración.

Cada integrante del equipo deberá contar con una laptop para trasladarse fácilmente y consultar cualquier inquietud y adicionalmente un monitor para tener mayor visibilidad del trabajo que realiza. Será necesario una pizarra acrílica para mantener informado sobre los avances de las tareas.