

**Análisis Comparativo entre Scrum y Waterfall como Guía para la Gestión de Proyectos de  
Desarrollo de Software**

Autores:

Allison Vanessa Diaz Guerrero

Xiomara Dussan Cubi

Fabian Steven Correa Triana

Universidad EAN

Facultad de Ingeniería

Proyecto Integrador

Ingeniera: Diana Carolina Beltrán Peña

Bogotá, Colombia

24 de Mayo de 2025

## **1. Resumen ejecutivo**

La gestión de proyectos de desarrollo de software ha evolucionado significativamente con la aparición de metodologías ágiles, entre las cuales Scrum se ha consolidado como una de las más utilizadas en la industria tecnológica. Tradicionalmente, los proyectos se gestionaban bajo el modelo en cascada (Waterfall), caracterizado por una estructura secuencial y una planificación rigurosa desde las etapas iniciales. No obstante, la necesidad de mayor adaptabilidad y capacidad de respuesta al cambio ha impulsado la adopción de enfoques ágiles.

Este proyecto tiene como propósito realizar un análisis comparativo entre las metodologías Scrum y Waterfall, con el fin de explorar sus características, beneficios y limitaciones desde una perspectiva teórica y académica. La metodología empleada es de tipo documental, basada en la revisión de literatura científica y fuentes especializadas, lo cual permite contrastar ambos enfoques en términos de eficiencia, adaptabilidad, comunicación y resultados en equipos de desarrollo.

Cabe destacar que este estudio no pretende establecer una metodología superior, sino ofrecer una guía de referencia que oriente a profesionales, equipos y organizaciones en la elección del enfoque más adecuado, según el contexto, los objetivos y las condiciones particulares de cada proyecto. A partir del análisis, se presentarán conclusiones que sirvan de apoyo para una toma de decisiones fundamentada y contextualizada.

Palabras clave: metodologías de desarrollo de software, gestión de proyectos, Scrum, Waterfall, enfoques ágiles, análisis comparativo, toma de decisiones, eficiencia, adaptabilidad.

## TABLA DE CONTENIDO

1.	Resumen ejecutivo .....	2
2.	Introducción.....	6
3.	Antecedentes .....	7
4.	Planteamiento del problema .....	8
5.	Objetivos .....	9
	Objetivo general:.....	9
	Objetivos específicos: .....	9
6.	Justificación.....	10
7.	Análisis de requisitos.....	11
8.	Marco de referencia.....	13
	Metodología Scrum.....	13
	Características principales de Scrum incluyen: .....	14
	Metodología en Cascada (Waterfall) .....	14
	Entre las características principales del modelo Waterfall se encuentran: .....	15
	Comparaciones entre Scrum y Waterfall .....	15
	Normas en Ingeniería de Software.....	16
9.	Análisis de Restricciones.....	18

1. Restricciones Ambientales.....	18
2. Restricciones Económicas .....	18
3. Restricciones Legales .....	19
4. Restricciones de Salud y Seguridad.....	19
5. Restricciones Socioculturales .....	20
10. Diseño metodológico.....	21
1. Enfoque de la investigación.....	21
2. Tipo y nivel de investigación.....	21
3. Método de recolección de información.....	21
4. Técnica e instrumentos de análisis.....	22
5. Población y muestra.....	22
6. Alcance de la investigación .....	23
7. Limitaciones del estudio .....	23
11. Solución de Ingeniería.....	24
Criterios de Evaluación y Asignación de Puntajes .....	29
Indicadores para Metodología Waterfall .....	29
Indicadores para Metodología Scrum .....	29
Tabla de Asignación de Puntajes .....	30
Interpretación y Recomendación .....	30
12. Alternativa de Solución.....	32

Justificación de la alternativa.....	33
Propuesta específica.....	33
1. Etapa de planificación y análisis de requisitos: .....	33
2. Etapa de diseño y desarrollo: .....	33
3. Etapa de pruebas y aseguramiento de calidad:.....	34
4. Entrega, documentación y cierre: .....	34
13. Análisis de costos .....	35
14. Plan de implementación .....	37
El desarrollo del proyecto se llevará a cabo en seis fases principales: .....	37
1. Planificación y Análisis Inicial: .....	37
2. Investigación y Marco Teórico: .....	37
3. Diseño Metodológico:.....	37
4. Desarrollo de la Alternativa de Solución: .....	37
5. Análisis de Costos y Horas-Hombre (HH): .....	37
6. Redacción del Informe Final:.....	38
Restricciones Legales .....	38
15. Matriz de riesgos .....	39
16. Conclusiones .....	41
17. Referencias .....	46

## 2. Introducción

La gestión de proyectos de desarrollo de software ha experimentado una transformación significativa con el paso del tiempo, adoptando diversas metodologías que buscan optimizar los procesos, mejorar la calidad del producto final y garantizar el cumplimiento de los resultados esperados. Entre estas metodologías, **Waterfall** y **Scrum** representan dos enfoques ampliamente conocidos y utilizados, cada uno con fundamentos, estructuras y dinámicas propias que responden a diferentes necesidades organizacionales y contextos de proyecto.

El modelo **Waterfall** se caracteriza por su enfoque lineal y secuencial, en el que cada fase del proyecto debe completarse antes de avanzar a la siguiente. Este modelo resulta especialmente útil en proyectos con requerimientos estables y claramente definidos desde el inicio. Sin embargo, su rigidez puede convertirse en una limitación en contextos donde los cambios son frecuentes o donde se requiere mayor flexibilidad para adaptarse a nuevas condiciones.

En contraste, **Scrum**, como marco de trabajo ágil, se orienta hacia la adaptabilidad, la entrega continua de valor y la colaboración constante entre los miembros del equipo. Su enfoque iterativo e incremental permite responder de forma más efectiva a la incertidumbre y a la evolución de los requerimientos a lo largo del desarrollo. No obstante, su implementación también implica desafíos, como la necesidad de una transformación cultural y organizacional, así como la formación adecuada de los equipos involucrados.

A partir de este panorama, se hace necesario comprender de manera profunda las particularidades de ambas metodologías, evaluar sus beneficios y limitaciones en la práctica, y establecer criterios que orienten su aplicación adecuada según las características del proyecto, del entorno y de los objetivos estratégicos de cada organización. Esta reflexión permite no solo comparar dos enfoques de gestión, sino también aportar a una toma de decisiones más informada y contextualizada en el ámbito del desarrollo de software

### 3. Antecedentes

En las últimas décadas, la gestión de proyectos de desarrollo de software ha experimentado una transformación significativa, impulsada por la evolución de las metodologías utilizadas para planificar, ejecutar y controlar los procesos. Tradicionalmente, el modelo en cascada (Waterfall) ha sido ampliamente adoptado por su enfoque estructurado, donde las fases del proyecto se desarrollan de manera secuencial y con escasa flexibilidad para cambios posteriores. Esta metodología fue una de las primeras formalizadas en la ingeniería de software y ha sido documentada ampliamente desde la década de 1970 (Royce, 1970), siendo valorada por su claridad y control.

Sin embargo, con la creciente demanda de soluciones más rápidas y adaptables, surgieron nuevas formas de gestionar proyectos que pudieran responder de manera más eficiente a la incertidumbre y a los cambios frecuentes en los requerimientos. En este contexto, las metodologías ágiles, y en particular Scrum, ganaron protagonismo. Propuesto formalmente en la década de los noventa por Schwaber y Sutherland, Scrum ha sido adoptado por organizaciones de distintos tamaños por su capacidad para mejorar la comunicación, la productividad y la entrega continua de valor.

Diversos estudios han abordado la comparación entre metodologías ágiles y tradicionales. Por ejemplo, Boehm y Turner (2004) proponen un enfoque equilibrado para determinar cuándo conviene aplicar métodos ágiles o tradicionales, dependiendo de factores como la complejidad del proyecto, la estabilidad de los requerimientos o la experiencia del equipo. Otros trabajos, como el de Petersen y Wohlin (2010), destacan cómo la agilidad puede mejorar la eficiencia, pero también reconocen los desafíos que implica su implementación, especialmente en entornos corporativos con estructuras jerárquicas rígidas.

#### 4. Planteamiento del problema

En el ámbito del desarrollo de software, la elección de una metodología de gestión de proyectos adecuada constituye un factor crítico que puede influir directamente en el éxito o fracaso de estos. Las organizaciones actuales enfrentan entornos cada vez más dinámicos, en los que deben gestionar recursos de manera eficiente, adaptarse a cambios constantes y cumplir con los plazos y requisitos establecidos. En este escenario, dos enfoques metodológicos han cobrado especial relevancia: el modelo tradicional en cascada (Waterfall) y las metodologías ágiles, entre las cuales destaca Scrum por su enfoque iterativo y su creciente adopción a nivel global.

Sin embargo, la amplia disponibilidad de enfoques metodológicos y la diversidad de características propias de cada organización y proyecto generan una dificultad recurrente: **¿cómo seleccionar la metodología más adecuada según el contexto?** Esta decisión, que debería estar fundamentada en criterios técnicos y estratégicos, suele tomarse sin una evaluación profunda, lo que puede comprometer la eficiencia del proceso, la calidad del producto final y la satisfacción de los clientes o usuarios. Aunque existen diversos estudios que abordan por separado las características de Scrum y Waterfall, se evidencia una **falta de análisis comparativos actualizados** que no solo describan sus diferencias, sino que también integren **criterios orientativos claros y aplicables** para apoyar la toma de decisiones metodológicas. Esta necesidad es especialmente relevante en organizaciones que se encuentran en transición hacia modelos más ágiles, pero que aún conservan estructuras jerárquicas o procesos formales arraigados. En este sentido, el presente estudio se orienta a resolver la siguiente pregunta: **¿cómo se pueden comparar de manera objetiva y fundamentada las metodologías Scrum y Waterfall para orientar su aplicación en distintos contextos organizacionales del desarrollo de software?** Dar respuesta a este interrogante permitirá construir una base de conocimiento útil para profesionales y organizaciones, promoviendo decisiones metodológicas más conscientes, contextualizadas y alineadas con los objetivos estratégicos de cada entorno.

## 5. Objetivos

### **Objetivo general:**

Realizar un análisis comparativo entre las metodologías Scrum y Waterfall en la gestión de proyectos de desarrollo de software que orienten su adecuada aplicación en diversos contextos organizacionales.

### **Objetivos específicos:**

- Describir los fundamentos conceptuales y principios operativos de las metodologías Scrum y Waterfall.
- Identificar y contrastar las ventajas, limitaciones y desafíos asociados a la implementación de Scrum y Waterfall en la gestión de proyectos de software.
- Proponer criterios orientativos para la selección de una metodología según el tipo de proyecto, el entorno organizacional y los objetivos estratégicos.

## 6. Justificación

En el ámbito del desarrollo de software, la elección de una metodología adecuada para la gestión de proyectos es un factor decisivo que incide directamente en la eficiencia del proceso, la calidad del producto entregado y el cumplimiento de los objetivos establecidos. Si bien el modelo en cascada (Waterfall) ha sido tradicionalmente valorado por su estructura secuencial y ordenada, el crecimiento de entornos dinámicos y exigentes ha favorecido la adopción de enfoques más flexibles como Scrum, una de las metodologías ágiles más difundidas en la actualidad.

No obstante, esta dualidad metodológica plantea un desafío común en muchas organizaciones: ¿cuándo y por qué aplicar uno u otro enfoque? La elección entre Scrum y Waterfall no siempre es evidente, ya que depende de factores como la complejidad del proyecto, la estabilidad de los requerimientos, la cultura organizacional, la composición del equipo y las expectativas del cliente. Por ello, es fundamental contar con análisis comparativos que no solo describan las metodologías, sino que también proporcionen criterios prácticos de selección que orienten su aplicación en diversos escenarios.

La relevancia de este estudio radica en su aporte teórico y práctico. En lo académico, permite consolidar y actualizar el conocimiento en torno a Scrum y Waterfall, contrastando sus fundamentos conceptuales, principios operativos, ventajas y limitaciones a partir de literatura especializada. Esto contribuye al fortalecimiento del debate sobre su aplicabilidad en diferentes tipos de organizaciones y proyectos, y establece una base para futuras investigaciones sobre metodologías de gestión en contextos específicos. Desde el plano práctico, el análisis y los criterios propuestos buscan servir como una herramienta de consulta para estudiantes, investigadores y profesionales, ayudando a tomar decisiones fundamentadas en función de las necesidades, capacidades y objetivos estratégicos de cada entorno. En última instancia, este estudio pretende aportar a la mejora de las prácticas de gestión de proyectos de software, facilitando procesos de planificación, ejecución y entrega más eficientes y adaptados a la realidad de cada organización.

## 7. Análisis de requisitos

Para llevar a cabo un análisis comparativo riguroso entre las metodologías Scrum y Waterfall en el contexto de la gestión de proyectos de desarrollo de software, es fundamental establecer requisitos que estructuren y orienten metodológicamente el estudio. Dado que esta investigación adopta un enfoque teórico-documental, los requisitos se organizan en tres categorías principales: teóricos, metodológicos y documentales. Se excluye la recolección de datos empíricos, ya que el propósito no es validar las metodologías desde la práctica, sino construir un marco comparativo orientativo a partir de fuentes confiables y pertinentes.

### 1. Requisitos teóricos

- Definir con claridad los **conceptos fundamentales** de la gestión de proyectos, incluyendo la diferenciación entre enfoques tradicionales y ágiles.
- Profundizar en los **principios, roles, procesos y estructuras operativas** que caracterizan a las metodologías **Waterfall y Scrum**.
- Establecer **criterios conceptuales comparativos** que permitan analizar aspectos clave como la eficiencia, adaptabilidad, gestión del cambio, control de calidad y comunicación dentro de cada enfoque.

### 2. Requisitos metodológicos

- Adoptar una estrategia de investigación de tipo **cualitativa y documental**, centrada en el **análisis comparativo** de fuentes especializadas.
- Establecer una **estructura comparativa clara** que permita evaluar ambas metodologías en función de dimensiones como **flexibilidad, alineación estratégica, contexto organizacional y tipo de proyecto**.

- Seleccionar fuentes confiables y académicas que aseguren la **validez y relevancia del análisis**.
- Determinar los **criterios de inclusión y exclusión de literatura** para garantizar un corpus riguroso y enfocado en estudios aplicables.

### 3. Requisitos documentales

- Recopilar y examinar **artículos académicos, libros especializados, informes técnicos y estudios de caso** que exploren la implementación de Scrum y Waterfall en proyectos de software.
- Incluir investigaciones que presenten **comparaciones entre ambas metodologías** en distintos sectores y tipos de organización.
- Priorizar fuentes **actualizadas** que aborden las **tendencias recientes, ventajas competitivas y desafíos prácticos** en la gestión de proyectos con enfoques ágiles y tradicionales.

## 8. Marco de referencia

La gestión de proyectos ha experimentado una profunda transformación con la incorporación de metodologías ágiles, entre las que destaca **Scrum**, en contraste con los enfoques tradicionales como el modelo en cascada (**Waterfall**). De acuerdo con **Schwaber y Sutherland (2017)**, Scrum se caracteriza por su enfoque en la colaboración constante, la adaptabilidad frente al cambio y la entrega continua de valor, elementos cruciales en proyectos de alta incertidumbre como el desarrollo de software. Ellos destacan que el trabajo en equipo y la entrega incremental permiten responder de forma rápida a las necesidades cambiantes del cliente.

Por otro lado, **Pressman (2014)** señala que el modelo en cascada continúa siendo válido, especialmente en proyectos donde los requisitos son claros desde el principio. Según **el PMBOK (Project Management Institute, 2017)**, el enfoque Waterfall permite una planificación detallada y estructurada, ideal para entornos donde el cambio es mínimo. Sin embargo, su rigidez puede ser un obstáculo en ambientes volátiles, donde se requiere mayor flexibilidad.

### Metodología Scrum

Scrum se basa en principios iterativos e incrementales, promoviendo una gestión flexible que se adapta a las prioridades cambiantes del negocio. Según **Takeuchi y Nonaka (1986)**, los proyectos exitosos imitan al rugby, donde todo el equipo avanza como una unidad hacia la meta. Bajo esta analogía, Scrum facilita que los equipos trabajen colaborativamente a través de ciclos cortos llamados sprints.

**Schwaber (2004)** enfatiza que la autogestión de los equipos es uno de los pilares fundamentales de Scrum, permitiendo mayor motivación y responsabilidad compartida. Además, **Sutherland (2014)** recalca que la práctica de las reuniones diarias o "daily stand-ups" ayuda a identificar problemas rápidamente, manteniendo la fluidez del proyecto.

**Highsmith (2002)** también señala que las metodologías ágiles, y en especial Scrum, promueven una gestión basada en el valor, donde las necesidades del cliente son atendidas progresivamente, priorizando los requisitos más importantes en cada ciclo.

#### **Características principales de Scrum incluyen:**

- **Gestión de expectativas del cliente:** Priorizando las entregas de mayor valor primero.
- **Planificación flexible:** Que permite la redefinición del trabajo a medida que evoluciona el proyecto.
- **Retroalimentación continua:** Mejorando la calidad del producto final y la satisfacción del cliente.
- **Transparencia y comunicación constante:** Facilitada por eventos como las revisiones de sprint y las retrospectivas.

Larman (2004) añade que esta estructura iterativa no solo favorece la calidad técnica, sino que también incrementa el aprendizaje organizacional y la innovación.

#### **Metodología en Cascada (Waterfall)**

A diferencia de Scrum, el modelo Waterfall sigue una secuencia estricta de fases, desde el análisis de requisitos hasta el mantenimiento. **Royce (1970)** fue uno de los primeros en describir este enfoque, destacando que su mayor ventaja era la claridad de cada etapa y la documentación exhaustiva.

**Hughes y Cotterell (2009)** sostienen que, en proyectos donde los cambios son mínimos, Waterfall permite optimizar los recursos y garantizar la calidad mediante una validación exhaustiva en cada fase. No obstante, **Fairley (2009)** advierte que la principal desventaja del modelo es la dificultad para responder a cambios una vez que el proyecto ha avanzado.

### **Entre las características principales del modelo Waterfall se encuentran:**

- **Estructura rígida:** Cada fase debe completarse antes de pasar a la siguiente.
- **Planificación previa detallada:** Incluyendo especificaciones técnicas, cronogramas y asignación de recursos.
- **Documentación extensa:** Que asegura trazabilidad y cumplimiento de los requisitos establecidos.
- **Difícil adaptación al cambio:** Lo que puede representar un riesgo en entornos dinámicos.

Sommerville (2011) comenta que, si bien la cascada parece anticuada frente a los métodos ágiles, en proyectos de defensa, aeronáutica y grandes infraestructuras sigue siendo el modelo preferido por su control riguroso de calidad.

### **Comparaciones entre Scrum y Waterfall**

A lo largo del tiempo, diversos estudios han comparado ambas metodologías, resaltando sus ventajas y desventajas según el contexto. **Boehm (1988)**, a través de su "balanza de riesgos", mostró que enfoques ágiles como Scrum son más eficaces en proyectos de alta incertidumbre, mientras que Waterfall resulta más efectivo en proyectos predecibles.

**Cockburn (2002)** refuerza la idea de que la comunicación frecuente y la colaboración son claves del éxito en entornos cambiantes, aspectos que Scrum favorece notablemente frente a la rigidez de Waterfall.

**Ghosh (2012)** realizó comparaciones entre el PMBOK, PRINCE2, Scrum e ICB, concluyendo que la elección de la metodología depende en gran medida de la complejidad y naturaleza del proyecto. Proyectos altamente innovadores se benefician más de enfoques ágiles, mientras que proyectos regulatorios o de cumplimiento encuentran en Waterfall una estructura más adecuada.

**Rigby, Sutherland y Takeuchi (2016)** demostraron que la adopción de Scrum no solo acelera los tiempos de entrega, sino que también mejora la calidad del producto mediante entregas continuas y feedback constante.

De forma complementaria, **Abrahamsson et al. (2002)** encontraron que, aunque Scrum puede parecer menos formal que Waterfall, su disciplina diaria y los roles definidos aseguran un control riguroso del proyecto.

**Chin (2004)** y **Fowler (2001)** también destacan que la agilidad no significa ausencia de control, sino adaptación continua basada en principios sólidos de ingeniería y gestión.

**Poppendieck y Poppendieck (2003)**, inspirados en el pensamiento Lean, promueven en Scrum la eliminación del desperdicio y la maximización del valor entregado al cliente, diferenciándolo claramente de Waterfall, donde frecuentemente se invierte mucho esfuerzo en tareas que no siempre agregan valor inmediato.

## **Normas en Ingeniería de Software**

El uso de normas y buenas prácticas es fundamental en cualquier metodología de gestión de proyectos para asegurar la calidad del producto final. Entre las más relevantes encontramos:

- **IEEE 829:** Estándar para la documentación de pruebas de software, crucial para asegurar la calidad tanto en entornos ágiles como tradicionales.
- **Guía Scrum (Schwaber y Sutherland, 2020):** Documento oficial que describe los principios y prácticas fundamentales de Scrum.
- **ISO/IEC 15504 (SPICE):** Norma que proporciona un marco para la evaluación de procesos de software.

- **PMBOK Guide (PMI, 2017)**: Compendio de buenas prácticas en la dirección de proyectos, especialmente asociado a métodos tradicionales como Waterfall.
- **ISO/IEC 12207:2017**: Norma que define los procesos del ciclo de vida del software.
- **ISO/IEC 27001**: Norma para la gestión de la seguridad de la información, cada vez más relevante en proyectos de software.

Además, **VersionOne (2019)** en su "State of Agile Report" muestra que el 58% de las organizaciones combinan prácticas ágiles con estándares tradicionales para optimizar su desempeño en proyectos diversos.

**Larman y Basili (2003)** también señalan que la integración de prácticas disciplinadas en ambientes ágiles es un factor clave para el éxito, evitando caer en la improvisación.

Finalmente, **Beck et al. (2001)** en el "Manifiesto Ágil" resumieron la filosofía detrás de metodologías como Scrum, priorizando a las personas y su interacción, el software funcional, la colaboración con el cliente y la capacidad de respuesta ante el cambio.

## **9. Análisis de Restricciones**

En el marco de este análisis comparativo entre Scrum y Waterfall, se deben considerar diversas restricciones que pueden influir en el desarrollo del estudio y en la aplicabilidad de las metodologías en distintos entornos. Estas restricciones se analizan desde perspectivas ambientales, económicas, legales, de salud y seguridad, y socioculturales, teniendo en cuenta los factores internos y externos que podrían afectar la viabilidad del estudio y la implementación de las metodologías en proyectos reales.

### **1. Restricciones Ambientales**

El impacto ambiental de la gestión de proyectos de software es indirecto, ya que no implica procesos físicos con residuos o contaminación. Sin embargo, el uso de metodologías como Scrum y Waterfall puede influir en aspectos como el consumo energético de los servidores y centros de datos debido a la gestión continua de documentación, reuniones y herramientas tecnológicas utilizadas. Además, el uso de modelos ágiles como Scrum puede reducir el desperdicio de recursos al enfocarse en entregas incrementales y evitar el desarrollo de funcionalidades innecesarias. Así mismo, en la metodología Scrum Debido a su enfoque iterativo y adaptativo, puede facilitar ajustes rápidos en proyectos que buscan reducir su huella ecológica. Sin embargo, su dinamismo también puede implicar cambios frecuentes de recursos o herramientas, lo cual podría aumentar el consumo si no se gestiona adecuadamente.

Mientras que, en la metodología waterfall debido a su enfoque iterativo y adaptativo, puede facilitar ajustes rápidos en proyectos que buscan reducir su huella ecológica. Sin embargo, su dinamismo también puede implicar cambios frecuentes de recursos o herramientas, lo cual podría aumentar el consumo si no se gestiona adecuadamente. Como, por ejemplo, en proyectos de construcción sostenible, Waterfall permite una planificación cuidadosa de materiales, pero Scrum se adapta mejor a innovaciones ecológicas en el camino.

### **2. Restricciones Económicas**

La adopción de cada metodología implica costos asociados, como la capacitación del personal, la implementación de herramientas de gestión y la inversión en infraestructura tecnológica. En entornos donde el presupuesto es limitado, la adopción de Scrum podría ser más viable debido a su enfoque en equipos pequeños y autogestionados, mientras que Waterfall podría generar costos mayores por requerir una planificación detallada desde el inicio. Además, cambios en el contexto macroeconómico, subsidios gubernamentales o barreras arancelarias en la importación de software pueden afectar la disponibilidad y costos de las herramientas utilizadas en la gestión de proyectos. Por ejemplo, en sectores con presupuestos fijos (como obras públicas), Waterfall suele ser preferido. En cambio, startups tecnológicas prefieren Scrum por su flexibilidad financiera.

### **3. Restricciones Legales**

En el ámbito legal, la gestión de proyectos de software debe cumplir con normativas relacionadas con protección de datos, seguridad informática y cumplimiento de estándares internacionales. Normas como ISO/IEC 12207 para el ciclo de vida del software y PMBOK para la gestión de proyectos establecen lineamientos que pueden influir en la selección de la metodología. Adicionalmente, en algunos sectores regulados, como el financiero o el de salud, el uso de Waterfall puede ser más adecuado debido a su enfoque estructurado y controlado, mientras que Scrum podría requerir adaptaciones para cumplir con regulaciones específicas.

### **4. Restricciones de Salud y Seguridad**

Si bien el desarrollo de software no representa riesgos físicos directos, el uso de metodologías de trabajo puede afectar la salud mental y la seguridad de los equipos de trabajo. Scrum, con sus ciclos iterativos y alta interacción, puede generar estrés si no se gestiona adecuadamente, mientras que Waterfall, al concentrar la carga de trabajo en ciertas fases, puede aumentar la presión en momentos clave del proyecto. Se debe garantizar que la metodología seleccionada no afecte negativamente la salud de los trabajadores ni genere condiciones laborales desfavorables.

## **5. Restricciones Socioculturales**

Las metodologías de gestión de proyectos deben adaptarse a las dinámicas culturales y organizacionales de las empresas y equipos de trabajo. En organizaciones con una estructura jerárquica rígida, Waterfall puede ser más aceptado, mientras que, en empresas con una cultura de innovación y flexibilidad, Scrum puede ser más efectivo. Además, factores como la disponibilidad de talento especializado, la resistencia al cambio y la aceptación de metodologías ágiles pueden influir en la viabilidad de implementación de cada enfoque.

De esta manera, la elección entre Scrum y Waterfall no puede hacerse en abstracto; debe basarse en un análisis detallado de las restricciones específicas del entorno, el tipo de proyecto, los actores involucrados y el contexto sociocultural. En algunos casos, un enfoque híbrido puede ser la mejor opción, combinando la planificación rigurosa de Waterfall con la flexibilidad de Scrum.

## **10. Diseño metodológico**

### **1. Enfoque de la investigación**

La investigación se aborda desde un enfoque cualitativo, dado que busca comprender e interpretar fenómenos complejos como la gestión de proyectos a través de metodologías Scrum y Waterfall, a partir del análisis de documentos y literatura especializada. Este enfoque permite un estudio profundo de conceptos, principios, ventajas, desventajas y contextos de aplicación de ambas metodologías, priorizando la comprensión sobre la medición cuantitativa.

### **2. Tipo y nivel de investigación**

El estudio es de tipo documental y de nivel descriptivo-comparativo:

- Documental, porque se fundamenta en fuentes secundarias previamente publicadas, tales como libros especializados, artículos científicos, normas internacionales y guías metodológicas reconocidas.
- Descriptivo-comparativo, ya que describe de manera detallada cada metodología y establece una comparación entre ambas, identificando puntos convergentes y divergentes que permitan determinar en qué contextos resulta más adecuada cada una.

### **3. Método de recolección de información**

Se utilizó la técnica de revisión bibliográfica y documental. La selección de las fuentes se hizo bajo criterios de:

- Relevancia: Se priorizaron textos clave en gestión de proyectos y desarrollo de software.
- Actualidad: Se incorporaron fuentes recientes, especialmente para Scrum y marcos ágiles.
- Credibilidad: Se emplearon autores reconocidos como Schwaber, Sutherland, Pressman, Sommerville, Boehm y documentos oficiales como el PMBOK y la Guía Scrum.

Entre las fuentes consultadas destacan:

- Libros: Pressman (2014), Sommerville (2011), Hughes & Cotterell (2009).
- Normas: ISO/IEC 12207, ISO/IEC 15504, IEEE 829, ISO/IEC 27001.
- Guías y documentos clave: PMBOK (2017), Guía Scrum (2020), Manifiesto Ágil (2001).
- Estudios de caso e informes industriales: VersionOne (2019), Rigby et al. (2016).

#### **4. Técnica e instrumentos de análisis**

Se empleó la técnica de análisis de contenido, estructurada en una matriz comparativa en la que se organizaron y clasificaron los hallazgos con base en categorías temáticas:

- Fundamentos teóricos.
- Ciclo de vida del proyecto.
- Grado de flexibilidad y adaptabilidad.
- Participación del cliente.
- Dinámica de comunicación.
- Gestión del cambio.
- Rol del equipo de trabajo.
- Calidad del producto y documentación.
- Normas técnicas aplicables.

Estas categorías permitieron estructurar un análisis sistemático que facilitó la comparación directa entre Scrum y Waterfall.

#### **5. Población y muestra**

Dado que se trata de una investigación documental, la población corresponde al conjunto de fuentes bibliográficas relacionadas con metodologías de gestión de proyectos. La muestra fue no

probabilística y por conveniencia, seleccionando aproximadamente 20 fuentes clave con base en su autoridad, aplicabilidad y aporte teórico-práctico al tema.

## **6. Alcance de la investigación**

El alcance de este trabajo es descriptivo, analítico y propositivo:

- Descriptivo, porque caracteriza ambas metodologías con sus elementos constitutivos.
- Analítico, al establecer relaciones y contrastes entre ellas según criterios definidos.

Propositivo, al brindar una guía sobre cuándo es más conveniente aplicar Scrum o Waterfall, dependiendo del tipo de proyecto y su entorno.

## **7. Limitaciones del estudio**

Entre las principales limitaciones se encuentran:

- La falta de aplicación empírica directa: el estudio se basa únicamente en fuentes documentales y no incluye validación práctica en proyectos reales.
- Posibles sesgos de interpretación: al basarse en análisis cualitativos de autores, algunas afirmaciones pueden variar según el contexto particular de cada organización
- Actualización constante del campo ágil: las metodologías ágiles, incluyendo Scrum, están en constante evolución, por lo que algunos elementos pueden quedar obsoletos rápidamente si no se actualizan las fuentes.

## **11. Solución de Ingeniería**

Con base en la encuesta unificada de gestión de proyectos diseñada para recopilar información sobre el contexto, características del proyecto y metodología actual utilizada, se propone un sistema de evaluación cuantitativa que permita, a partir de las respuestas obtenidas, recomendar la metodología de gestión más adecuada: Scrum, Waterfall o un enfoque Híbrido.

Este sistema considera preguntas clave que reflejan las características del proyecto, la participación del cliente, la flexibilidad requerida y el nivel de incertidumbre, asignando puntajes que se interpretan para definir la recomendación metodológica.

### **Objetivo de la encuesta**

El objetivo de esta encuesta es brindar a las organizaciones una herramienta práctica que les permita:

- Evaluar las condiciones internas y externas del proyecto.
- Identificar el grado de adecuación entre el proyecto y las metodologías ágiles, tradicionales o híbridas.
- Obtener una base empírica para tomar decisiones sobre el enfoque de gestión más pertinente.
- Reconocer oportunidades de mejora en su enfoque actual de gestión de proyectos.

### **Diseño de la encuesta**

La Encuesta Unificada de Gestión de Proyectos se estructura en seis secciones, cada una orientada a captar dimensiones fundamentales del contexto organizacional y del proyecto en curso:

#### **1. Información General**

Permite caracterizar a la organización y al proyecto. Incluye:

1. ¿A qué sector pertenece su empresa?
2. ¿Cuántos empleados tiene el equipo de desarrollo?
3. ¿Qué tipo de proyecto está desarrollando actualmente?
  - a. Requisitos claros y bien definidos desde el inicio (ej. migración de datos, construcción).
  - b. Alta incertidumbre o requisitos que pueden cambiar (ej. desarrollo de software, innovación).
  - c. Aún no está completamente definido.

## **2. Metodología Actual**

Permite conocer el enfoque actual de gestión de proyectos:

4. ¿Qué metodología de gestión de proyectos utilizan actualmente?
  - a. Scrum
  - b. Waterfall
  - c. Híbrido
  - d. Otra (especifique): \_\_\_\_\_
5. ¿Cuál fue el motivo de la selección de esa metodología?
6. ¿Han cambiado de metodología en los últimos 5 años? ¿Por qué?

## **3. Evaluación del Proyecto y su Entorno**

Utiliza una escala Likert del 1 al 5 (1 = Totalmente en desacuerdo, 5 = Totalmente de acuerdo) para evaluar:

#### **A. Requisitos y Alcance**

7. Los requisitos del proyecto están claramente definidos desde el inicio.
8. Es poco probable que los requisitos cambien durante el desarrollo.
9. El cliente desea ver el producto final solo al finalizar el desarrollo.

#### **B. Participación del Cliente**

10. El cliente estará disponible y comprometido durante todo el proceso.
11. Se requieren entregas frecuentes para revisión del cliente.
12. El cliente desea involucrarse en la priorización de funcionalidades.

#### **C. Complejidad y Riesgo**

13. El proyecto es complejo y podría cambiar con el tiempo.
14. Se desconoce cómo se comportará el sistema hasta que se empiece a construir.
15. Es necesario validar hipótesis o tecnologías durante el desarrollo.

#### **D. Entorno del Proyecto**

16. El equipo está familiarizado con metodologías ágiles.
17. Se trabaja en un entorno cambiante o dinámico.
18. Es posible trabajar en iteraciones con entregas parciales funcionales.

### **4. Evaluación de la Metodología Actual**

Permite conocer la experiencia real con el enfoque actual:

19. ¿Qué beneficios han identificado con la metodología que utilizan?

20. ¿Qué desafíos han enfrentado?

21. En una escala del 1 al 5, ¿qué tan satisfechos están con su metodología actual?

22. ¿Están considerando adoptar otra metodología en el futuro?

## **5. Necesidades y Preferencias**

Permite explorar expectativas y cultura organizacional:

23. ¿Qué importancia tiene la flexibilidad para modificar requisitos durante el proyecto?

- Alta
- Media
- Baja

24. ¿Qué tan involucrado estará el cliente o usuario final durante el desarrollo?

- Alta participación
- Moderada
- Baja

25. ¿Qué tan importante es entregar el producto en fases o versiones parciales?

- Muy importante
- Algo importante
- No es importante

26. ¿Cómo describirías la experiencia y habilidades del equipo?

- Autónomo, colaborativo y acostumbrado a iteraciones
- Técnico con experiencia en planificación estructurada
- Mixto o aún sin definir

27. ¿Cuál es el enfoque principal del proyecto?

- Innovación, mejora continua o adaptación al cambio
- Precisión, cumplimiento de especificaciones y control
- Ambos

28. ¿Cuál de las siguientes opciones representa mejor tu necesidad?

- Necesito una metodología flexible y colaborativa.
- Necesito una metodología estructurada y predecible.

## **6. Opinión**

Finalmente, se abre un espacio para la reflexión:

29. ¿Qué aspectos mejorarías en tu actual enfoque de gestión de proyectos?

### **5.4 Aplicación sugerida**

Esta encuesta puede ser aplicada:

- Antes del inicio de un nuevo proyecto, como parte de la planificación.
- Como parte de una auditoría o mejora continua en la oficina de gestión de proyectos (PMO).

- En procesos de transición entre metodologías.
- Como instrumento de análisis en estudios de caso académicos o empresariales.

### **Criterios de Evaluación y Asignación de Puntajes**

Se seleccionaron 12 preguntas relevantes de la encuesta, agrupadas en dos categorías principales según su afinidad con metodologías tradicionales o ágiles.

### **Indicadores para Metodología Waterfall**

Se asigna 1 punto por cada afirmación favorable a características propias de un enfoque tradicional y estructurado:

- Requisitos claros y estables desde el inicio.
- Baja probabilidad de cambios en requisitos.
- Entrega final sin revisiones intermedias.
- Preferencia por control y cumplimiento de especificaciones.
- Necesidad de predictibilidad y planificación rígida.

### **Indicadores para Metodología Scrum**

Se asigna 1 punto por cada afirmación que refleja afinidad con metodologías ágiles:

- Alta participación y compromiso del cliente durante el proceso.
- Entregas frecuentes y revisiones continuas.
- Proyectos con alta incertidumbre o complejidad.
- Necesidad de validación continua de hipótesis.

- Entornos de trabajo dinámicos y adaptativos.
- Preferencia por flexibilidad y colaboración.

### Tabla de Asignación de Puntajes

Pregunta	Característica	Puntaje Waterfall	Puntaje Scrum
7	Requisitos claros y definidos	1	0
8	Poca probabilidad de cambios en requisitos	1	0
9	Producto final entregado al término	1	0
10	Cliente disponible y comprometido durante el proceso	0	1
11	Entregas frecuentes para revisión	0	1
13	Proyecto complejo y con cambios posibles	0	1
15	Necesidad de validar hipótesis o tecnologías	0	1
16	Equipo familiarizado con metodologías ágiles	0	1
27	Enfoque de precisión y control	1	0
28	Necesidad de método estructurado y predecible	1	0
28	Necesidad de método flexible y colaborativo	0	1

*Nota: Algunas preguntas como la 28 tienen opciones mutuamente excluyentes.*

### Interpretación y Recomendación

La puntuación total obtenida en cada grupo indica la afinidad del proyecto con cada metodología. La interpretación es la siguiente:

Puntaje Waterfall	Puntaje Scrum	Recomendación
0 – 2	4 – 6	Recomendación: <b>Scrum</b>
4 – 6	0 – 2	Recomendación: <b>Waterfall</b>
3	3	Recomendación: <b>Híbrido</b>
2 – 3	2 – 3	Evaluar contexto y considerar enfoque híbrido

Esta interpretación permite una guía clara para seleccionar la metodología más adecuada, considerando las particularidades del proyecto y el equipo.

### Ejemplo de Aplicación

Para un proyecto en el que el equipo respondió afirmativamente a preguntas relacionadas con requisitos definidos, poca incertidumbre y entrega final, obteniendo un puntaje Waterfall de 5 y Scrum de 1, la recomendación será emplear la metodología Waterfall.

En cambio, para un proyecto con alta incertidumbre, entregas frecuentes y compromiso continuo del cliente, con un puntaje Scrum de 5 y Waterfall de 1, se recomendará la adopción de Scrum.

## 12. Alternativa de Solución

Tras la aplicación del proceso metodológico anterior, se concluye que no existe una única metodología universalmente superior. La elección entre Scrum y Waterfall depende fundamentalmente del tipo de proyecto, el entorno organizacional, la claridad de los requisitos, y la capacidad del equipo para adaptarse a cambios.

Scrum se presenta como la solución más adecuada para proyectos con un alto grado de incertidumbre, donde los requisitos pueden evolucionar con el tiempo y donde la retroalimentación del cliente es constante. Su estructura basada en sprints, retrospectivas y entregas incrementales permite una mejora continua, gestión activa del riesgo, y una mayor satisfacción del cliente a través de resultados visibles en plazos cortos.

Por otro lado, Waterfall es una alternativa válida cuando el proyecto tiene una planificación estable, requisitos bien definidos desde el inicio, y cuando es necesario cumplir con regulaciones estrictas o mantener una trazabilidad completa. Esta metodología ofrece una ejecución más controlada, útil para sectores como manufactura, infraestructura, o proyectos con un enfoque documental riguroso.

Finalmente, en lugar de imponer una única metodología, se propone una guía de selección contextual que permita a cada organización o equipo de trabajo evaluar las características de su proyecto y adoptar el enfoque más conveniente. Además, se contempla la posibilidad de combinar elementos de ambas metodologías en un modelo híbrido, adaptando la flexibilidad de Scrum a la estructura formal de Waterfall cuando el entorno lo permite. De esta forma, se maximiza el valor generado por el proyecto y se optimiza el uso de recursos.

Teniendo en cuenta las características, ventajas y limitaciones de las metodologías Waterfall y Scrum, se propone como alternativa de solución la adopción de un enfoque híbrido, también conocido

como modelo "Water-Scrum-Fall" o Scrum combinado con gestión tradicional, que integre los beneficios de ambos marcos metodológicos según las necesidades del proyecto.

### **Justificación de la alternativa**

Ambas metodologías han demostrado ser efectivas, pero bajo condiciones diferentes:

- Scrum es ideal para entornos de alta incertidumbre, proyectos innovadores o de desarrollo incremental, con una comunicación constante con el cliente.
- Waterfall es más útil cuando los requisitos son estables, las etapas están bien definidas y se requiere una fuerte trazabilidad y documentación.
- En contextos reales, muchos proyectos no encajan del todo en un solo paradigma, por lo que adoptar un modelo híbrido puede proporcionar:
- Flexibilidad ágil en el desarrollo, mediante sprints y entregas incrementales (Scrum).
- Estructura sólida en fases iniciales y finales, con documentación detallada, control de calidad y gestión formal del cierre (Waterfall).

### **Propuesta específica**

#### **1. Etapa de planificación y análisis de requisitos:**

- Utilizar técnicas tradicionales (Waterfall) para definir objetivos, cronogramas generales y presupuestos.

Reunir requisitos funcionales y no funcionales mediante entrevistas, workshops y análisis documental.

#### **2. Etapa de diseño y desarrollo:**

- Implementar Scrum con equipos multidisciplinarios, planificación de sprints, reuniones diarias y entregas incrementales.
- Priorización del backlog según valor de negocio.

### **3. Etapa de pruebas y aseguramiento de calidad:**

- Aplicar estrategias de validación formales (de tipo Waterfall) junto con pruebas continuas del enfoque ágil.

### **4. Entrega, documentación y cierre:**

- Generar documentación formal y entregar el producto final mediante procesos clásicos de cierre de proyecto.
- Beneficios esperados
- Reducción de riesgos por mala gestión del cambio.
- Mejor adaptación del producto a las necesidades del cliente.
- Aumento en la productividad y motivación del equipo.
- Mejora en la calidad del software mediante ciclos de retroalimentación cortos.
- Cumplimiento de estándares técnicos y regulatorios mediante fases formales de documentación.

### 13. Análisis de costos

Perfil Profesional	Nº de Personas	Valor Hora (COP)	Total de HH por Persona	Total de HH por Perfil	Costo por Perfil (COP)
Ingeniería de Sistemas	2	\$ 25.000	28 HH	56 HH (28 HH × 2)	\$ 1.400.000
Ingeniería Industrial	4	\$ 22.000	26 HH	104 HH (26 HH × 4)	\$ 2.288.000
<b>Total Proyecto</b>				<b>160 HH</b>	<b>\$3.688.000 COP</b>

Tabla 1

Costos estimados por perfil profesional para el desarrollo del proyecto. *Elaboración propia.*

Fase del Proyecto	HH Ingeniería de Sistemas	HH Ingeniería Industrial (por persona)	HH Ingeniería Industrial (x2 personas x2)
Planeación y definición de objetivos	10 HH	10 HH	20 HH
Recolección de información	8 HH	12 HH	24 HH
Análisis comparativo (Scrum vs Waterfall)	12 HH	8 HH	16 HH
Desarrollo del diseño metodológico	8 HH	6 HH	12 HH
Redacción del marco teórico y solución	10 HH	8 HH	16 HH
Preparación del informe final	8 HH	8 HH	16 HH
<b>Total de HH por Perfil</b>	<b>56 HH</b>	<b>52 HH por persona</b>	<b>104 HH total Ingeniería Industrial</b>

Tabla 2

Distribución de horas por fase del proyecto según perfil profesional. *Elaboración propia.*

- **Distribución del Trabajo:**

El proyecto fue desarrollado por tres personas: un ingeniero de sistemas y dos ingenieros industriales. La distribución horaria se realizó según las fases del proyecto, asegurando que cada perfil aportara desde su especialidad.

- **Valor Hora Diferenciado:**

El valor hora estimado varía por perfil profesional, siendo más alto en ingeniería de sistemas debido a la especialización en metodologías de desarrollo de software, lo cual justifica su tarifa superior.

- **Costos Proporcionales:**

Aunque los ingenieros industriales participaron en mayor número, su costo unitario por hora es más bajo. Aun así, su participación acumulada (104 HH) genera el mayor impacto en el presupuesto.

- **Costo Total Estimado:**

La suma total de los costos por perfil asciende a \$3.688.000 COP, lo que representa una inversión razonable para un análisis metodológico completo considerando criterios técnicos, económicos y organizacionales.

- **Eficiencia Horaria:**

La planificación detallada por fases muestra una distribución eficiente del tiempo, evitando redundancias y permitiendo que cada integrante contribuya según su área.

- **Sostenibilidad del Proyecto:**

La inversión estimada es viable desde una perspectiva académica y empresarial para proyectos de análisis metodológico y puede escalarse o adaptarse en función de la complejidad.

## **14. Plan de implementación**

El desarrollo del proyecto se llevará a cabo en seis fases principales:

### **1. Planificación y Análisis Inicial:**

Durante esta primera semana, se delimita el problema, se formulan los objetivos del proyecto y se asignan los roles según los perfiles profesionales de cada integrante (Ingeniería de Sistemas e Ingeniería Industrial). Se revisa el contexto académico y se define el alcance del trabajo, estableciendo la hoja de ruta para las siguientes etapas.

### **2. Investigación y Marco Teórico:**

En la segunda semana, se realiza la recopilación de información sobre las metodologías de gestión de proyectos más relevantes (Scrum y Waterfall), así como sobre marcos normativos y estándares técnicos como IEEE 829, PMBOK e ISO/IEC 15504. Esta fase culmina con la elaboración del marco de referencia que sustenta teóricamente la propuesta.

### **3. Diseño Metodológico:**

Durante tres días se desarrolla el diseño metodológico, definiendo el tipo de investigación, el enfoque cualitativo, las técnicas de recolección de información y los criterios de análisis. También se elabora una herramienta que permite la comparación objetiva entre las metodologías seleccionadas.

### **4. Desarrollo de la Alternativa de Solución:**

A lo largo de una semana se construye la propuesta de un modelo híbrido de gestión de proyectos. Esta etapa implica el diseño técnico del modelo, la justificación de su viabilidad y la validación interna por parte del equipo, considerando la aplicabilidad académica de la solución.

### **5. Análisis de Costos y Horas-Hombre (HH):**

Durante dos días, las integrantes de Ingeniería Industrial realizan una estimación detallada de las Horas-Hombre invertidas en el proyecto según cada perfil profesional. Posteriormente, se calculan los costos asociados al tiempo dedicado por cada integrante, utilizando valores promedio del mercado para perfilar el costo humano del proyecto.

## **6. Redacción del Informe Final:**

En los últimos cuatro días, el equipo liderado por la estudiante de Ingeniería de Sistemas, integra cada uno de los capítulos desarrollados, revisa el contenido, corrige el estilo y da formato final al documento. Se incluye bibliografía en formato APA, gráficos, tablas y anexos correspondientes.

## **Restricciones Legales**

El proyecto se encuentra limitado por los siguientes aspectos legales:

- **Propiedad Intelectual:** Todo el contenido desarrollado debe respetar los derechos de autor, citando correctamente cada fuente conforme a las normas APA (séptima edición).
- **Uso Académico Exclusivo:** La propuesta de solución es exclusivamente de carácter académico y no debe ser implementada en entornos empresariales reales sin previa validación legal y técnica.
- **Normativas Técnicas:** Se garantiza la referencia y alineación con estándares técnicos internacionales reconocidos, como IEEE 829 (documentación de pruebas), ISO/IEC 25010 (calidad del producto software) e ISO/IEC 12207 (procesos del ciclo de vida del software).
- **Protección de Datos:** No se hace uso de información personal, sensible o confidencial durante el desarrollo del proyecto, cumpliendo con principios de ética profesional y académica.

## 15. Matriz de riesgos

	PROBABILIDAD	IMPACTO	NIVEL DE RIESGO	COMO PODER MITIGARLO
DIFICULTAD PARA ACCEDER A FUENTES CONFIABLES	MEDIA	ALTA	ALTO	PLANIFICAR TIEMPO SUFICIENTE PARA LA REVISIÓN
RETRASOS EN LA RECOLECCIÓN Y ANÁLISIS DE INFORMACIÓN	ALTA	MEDIA	ALTO	DIVIR TAREAS EN EQUIPO Y UTILIZAR HERRAMIENTAS O INDICADORES DE SEGUIMIENTO
LIMITADA DISPONIBILIDAD DE EXPERTOS PARA VALIDACIÓN	MEDIA	MEDIA	MEDIO	TENER MAS DE UNA OPCIÓN Y CONTACTAR CON ANTICIPACIÓN
AMBIGÜEDAD EN LOS CRITERIOS DE COMPARACIÓN	MEDIA	ALTA	ALTO	DEFINIR CRITERIOS CLAROS DESDE EL INICIO
CAMBIOS EN EL ALCANCE DEL PROYECTO DOCUMENTAL	BAJA	ALTA	MEDIO	REVISAR EL ALCANCE EN CADA FASE

Matriz de riesgos del análisis comparativo de metodologías Scrum y Waterfall. *Elaboración propia.*

La matriz de riesgos presentada corresponde a un análisis comparativo entre las metodologías Scrum y Waterfall, enfocándose en la identificación, evaluación y mitigación de riesgos comunes en proyectos que utilizan estas metodologías. Esta matriz clasifica los riesgos según tres criterios fundamentales: la probabilidad, el impacto y el nivel de riesgo, además de proponer estrategias concretas para mitigar cada uno. El primer riesgo identificado es la dificultad para acceder a fuentes confiables, que tiene una probabilidad media pero un impacto alto, resultando en un nivel de riesgo alto; para mitigar este riesgo, se recomienda planificar con suficiente tiempo la revisión de fuentes.

El segundo riesgo corresponde a los retrasos en la recolección y análisis de información, cuya probabilidad es alta y el impacto medio, lo que también da como resultado un riesgo alto; para mitigarlo se sugiere dividir las tareas entre los miembros del equipo y utilizar herramientas o indicadores de seguimiento que permitan controlar el avance de forma eficiente. En tercer lugar, se considera la limitada

disponibilidad de expertos para la validación, riesgo con probabilidad e impacto medios, por lo tanto de nivel medio, que puede ser mitigado teniendo más de una opción de experto disponible y contactándolos con anticipación. Otro riesgo identificado es la ambigüedad en los criterios de comparación, que presenta una probabilidad media y un impacto alto, generando un riesgo alto que se puede controlar definiendo desde el inicio criterios claros y consensuados entre todos los participantes del proyecto.

Finalmente, se analizan los cambios en el alcance del proyecto documental, que presentan una probabilidad baja pero un impacto alto, resultando en un riesgo medio; para mitigar esta situación, se recomienda revisar el alcance del proyecto en cada fase. En conjunto, esta matriz permite visualizar de forma clara y estratégica los riesgos potenciales que podrían afectar el desarrollo de un proyecto y plantea medidas proactivas y prácticas para reducir su impacto, fomentando así una gestión de proyectos más eficiente, anticipada y controlada tanto en entornos ágiles como tradicionales.

## 16. Conclusiones

1. La elección metodológica debe contextualizarse. No existe una metodología universalmente superior. Tanto Scrum como Waterfall presentan ventajas y limitaciones que dependen del tipo de proyecto, el entorno organizacional, los actores involucrados, las regulaciones aplicables y el nivel de incertidumbre presente. En algunos casos, un enfoque híbrido puede resultar más apropiado, combinando la planificación estructurada de Waterfall con la flexibilidad iterativa de Scrum.
2. Scrum es altamente efectivo en entornos cambiantes e innovadores. Su estructura basada en ciclos cortos (sprints), reuniones frecuentes (como dailies y retrospectivas), y la autogestión del equipo permite detectar errores tempranamente, adaptarse rápidamente a los cambios y entregar valor continuo al cliente. Esto lo convierte en una opción ideal para proyectos dinámicos donde los requisitos evolucionan constantemente.
3. Waterfall es útil en contextos predecibles y regulados. Su enfoque lineal, con fases secuenciales claramente definidas y una fuerte orientación documental, ofrece una mayor trazabilidad y control. Resulta especialmente valioso en sectores como la ingeniería, manufactura, construcción o gobierno, donde los procesos normativos requieren una planificación rigurosa y documentación formal.
4. La gestión del cambio es un factor clave de diferenciación. Mientras Scrum integra el cambio como parte del proceso natural de desarrollo mediante inspección y adaptación continua, Waterfall lo ve como una interrupción al plan inicial, lo que puede dificultar la incorporación de nuevas necesidades durante la ejecución.
5. La colaboración y comunicación constante en Scrum fortalecen la cohesión del equipo. Esta metodología promueve la participación activa de los stakeholders y fomenta la responsabilidad compartida, lo que facilita la alineación de objetivos y la toma de decisiones. Waterfall, al espaciar los puntos de control, puede dificultar la detección temprana de problemas y ralentizar la respuesta ante imprevistos.

6. Scrum optimiza recursos a largo plazo. Aunque puede parecer menos predecible en términos de cronograma y costos al inicio, su capacidad de reducir retrabajo y enfocar el esfuerzo en lo que realmente aporta valor puede resultar más eficiente en proyectos complejos. En contraste, Waterfall puede generar sobrecostos si se presentan cambios imprevistos, debido a su rigidez estructural.
7. La madurez del equipo influye en el éxito de Scrum. Esta metodología requiere equipos disciplinados, autónomos y con buenas habilidades de comunicación. Sin estos elementos, pueden surgir confusiones en los roles o pérdida de enfoque. Waterfall, al tener una estructura jerárquica y fases bien delimitadas, es más tolerante a equipos con menor experiencia en metodologías ágiles.
8. La documentación tiene un rol diferente en cada enfoque. Waterfall promueve una documentación extensa desde las primeras etapas del proyecto, lo cual es útil para auditorías, mantenimientos posteriores o validación formal. Scrum, por su parte, prioriza la comunicación directa y la entrega funcional, reduciendo la carga documental a lo esencial.
9. El tipo de cliente también condiciona la elección metodológica. Clientes que disponen de tiempo para participar activamente suelen adaptarse mejor a Scrum, al permitir iteraciones frecuentes y ajustes sobre la marcha. En cambio, clientes con disponibilidad limitada o que prefieren una planificación cerrada se sienten más cómodos con el enfoque Waterfall.
10. La industria en la que se ejecuta el proyecto también es determinante. Sectores tradicionales con alta regulación y bajo margen de cambio tienden a favorecer Waterfall. En contraste, industrias más dinámicas como la tecnológica, el desarrollo de software, el marketing digital o la educación optan por Scrum para responder con agilidad a las exigencias del entorno.
11. En proyectos interdisciplinarios, Scrum facilita la integración de equipos diversos. Al permitir entregas incrementales y constantes revisiones, se favorece la colaboración simultánea entre áreas. Waterfall, al requerir la finalización de una fase para iniciar otra, puede generar cuellos de botella si uno de los equipos se retrasa.

12. Scrum promueve una cultura organizacional centrada en la mejora continua. Su estructura iterativa fomenta la reflexión constante sobre los procesos, el aprendizaje colectivo y la entrega de valor. Esto lo convierte en una herramienta poderosa no solo para gestionar proyectos, sino también para transformar la forma de trabajar en las organizaciones.
13. El desarrollo de un sistema de puntuación basado en las respuestas a la encuesta unificada permite identificar de manera objetiva la metodología de gestión de proyectos más adecuada para cada contexto, ya sea Scrum, Waterfall o un enfoque híbrido (Autor, Año).
14. Las preguntas seleccionadas reflejan características esenciales que diferencian los enfoques tradicionales y ágiles, tales como el grado de definición y estabilidad de los requisitos, la participación del cliente, la flexibilidad ante cambios y la complejidad del proyecto (Autor, Año).
15. El uso de una matriz de puntajes facilita la interpretación de los resultados, simplificando la toma de decisiones para la selección metodológica y disminuyendo la subjetividad en el proceso (Autor, Año).
16. El sistema reconoce que no todos los proyectos encajan estrictamente en un único modelo, permitiendo una recomendación híbrida en casos donde las características del proyecto y del equipo no se alinean completamente con un solo enfoque (Autor, Año).
17. La aplicación práctica de esta solución mejora la alineación entre la metodología de gestión y las necesidades específicas del proyecto, incrementando la probabilidad de éxito en la entrega de productos de software (Autor, Año).
18. Como limitación, esta herramienta se basa en información autoinformada mediante la encuesta, por lo que su efectividad depende de la precisión y honestidad en las respuestas proporcionadas por los participantes (Autor, Año).
19. Finalmente, la herramienta propuesta es escalable y puede adaptarse o actualizarse con nuevas preguntas y criterios para responder a la evolución constante de las metodologías de gestión de proyectos (Autor, Año).

20. Scrum, al permitir iteraciones cortas y entregas frecuentes, reduce la exposición al riesgo al facilitar ajustes tempranos. Waterfall, en cambio, concentra los riesgos en fases críticas como la planificación inicial o la entrega final, donde los errores pueden ser más costosos. Un enfoque híbrido permite una mejor distribución de los riesgos a lo largo del ciclo de vida del proyecto.
21. Las organizaciones con estructuras horizontales, alta participación colaborativa y apertura al cambio suelen adaptarse mejor a Scrum. En cambio, las organizaciones con jerarquías marcadas, procesos formales y aversión al cambio tienden a encajar mejor con Waterfall. Evaluar la cultura interna es clave para evitar resistencias y fricciones en la ejecución.
22. Combinar lo mejor de ambos mundos permite ajustar la metodología a los distintos momentos del proyecto o a diferentes equipos involucrados. Por ejemplo, se puede aplicar Waterfall para la planificación contractual y Scrum para el desarrollo técnico, optimizando tiempos, costos y calidad según el contexto.
23. Independientemente del enfoque adoptado, el éxito depende en gran medida de la comprensión y aplicación adecuada por parte del equipo. La formación en principios ágiles, gestión del cambio y liderazgo colaborativo incrementa significativamente la efectividad de Scrum, mientras que el dominio de herramientas de gestión tradicional potencia el control en proyectos Waterfall.
24. El tipo de cliente también influye en la elección  
Clientes que requieren visibilidad temprana, flexibilidad y entregas parciales valorarán Scrum. Aquellos que prefieren planes definidos y entregables finales cerrados se alinean mejor con Waterfall
25. La alineación con los objetivos del negocio debe guiar la elección metodológica  
No se trata solo de gestionar tareas, sino de elegir la metodología que mejor apoye la entrega de valor al cliente y el cumplimiento de metas estratégicas.
26. La correcta selección metodológica mejora los resultados del proyecto  
Elegir entre Scrum y Waterfall de manera informada permite optimizar tiempos, costos y la satisfacción del cliente, minimizando retrabajos y fallos de planificación.

27. La cultura de la empresa, el nivel de madurez en gestión de proyectos, y la experiencia del equipo en metodologías ágiles o tradicionales son elementos clave a considerar antes de seleccionar una metodología.
28. Proporcionar una guía que evalúe variables como complejidad, tiempo, recursos, flexibilidad, y nivel de incertidumbre, permite a las organizaciones tomar decisiones informadas, reducir riesgos y mejorar la eficiencia de ejecución.
29. El enfoque híbrido es una alternativa creciente, muchas organizaciones que cuentan con desarrollo de software pueden optar por un enfoque híbrido combinado lo mejor de ambas metodologías para adaptarse a proyectos complejos o con múltiples componentes.

## 17. Referencias

1. Abrahamsson, P., Salo, O., Ronkainen, J., & Warsta, J. (2002). *Agile software development methods: Review and analysis*. VTT Publications.
2. Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., ... & Thomas, D. (2001). *Manifesto for Agile Software Development*. <http://agilemanifesto.org/>
3. Boehm, B. (1988). *A spiral model of software development and enhancement*. ACM SIGSOFT Software Engineering Notes, 11(4), 14-24.
4. Chin, G. (2004). *Agile project management: How to succeed in the face of changing project requirements*. AMACOM.
5. Cockburn, A. (2002). *Agile Software Development: Principles, Patterns, and Practices*. Addison-Wesley.
6. Fairley, R. (2009). *Managing and leading software projects*. Wiley-IEEE Computer Society Press.
7. Fowler, M. (2001). *The New Methodology*.  
<https://martinfowler.com/articles/newMethodology.html>
8. Ghosh, S. (2012). *Comparative study of traditional and agile software development methodologies*. International Journal of Computer Science and Information Technologies, 3(3), 4117-4120.
9. Highsmith, J. (2002). *Agile software development ecosystems*. Addison-Wesley.
10. Hughes, B., & Cotterell, M. (2009). *Software Project Management* (5th ed.). McGraw-Hill Education.
11. Larman, C. (2004). *Agile and Iterative Development: A Manager's Guide*. Addison-Wesley Professional.
12. Larman, C., & Basili, V. R. (2003). *Iterative and incremental development: A brief history*. Computer, 36(6), 47-56.

13. Poppendieck, M., & Poppendieck, T. (2003). *Lean Software Development: An Agile Toolkit*. Addison-Wesley.
14. Pressman, R. S. (2014). *Ingeniería de Software: Un enfoque práctico* (7ª ed.). McGraw-Hill Education.
15. Project Management Institute (PMI). (2017). *A Guide to the Project Management Body of Knowledge (PMBOK® Guide)* (6th ed.). PMI.
16. Rigby, D. K., Sutherland, J., & Takeuchi, H. (2016). *Embracing Agile*. Harvard Business Review, 94(5), 40-50.
17. Royce, W. W. (1970). *Managing the Development of Large Software Systems*. Proceedings of IEEE WESCON.
18. Schwaber, K. (2004). *Agile Project Management with Scrum*. Microsoft Press.
19. Schwaber, K., & Sutherland, J. (2017). *The Scrum Guide*. Scrum.org.
20. Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide: The Definitive Guide to Scrum*. Scrum.org.
21. Sommerville, I. (2011). *Software Engineering* (9th ed.). Addison-Wesley.
22. Takeuchi, H., & Nonaka, I. (1986). *The new new product development game*. Harvard Business Review, 64(1), 137-146.
23. VersionOne. (2019). *13th Annual State of Agile Report*. <https://stateofagile.com/>
24. Abrahamsson, P., Salo, O., Ronkainen, J., & Warsta, J. (2002). Agile software development methods: Review and analysis. VTT Publications.
25. Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., ... & Thomas, D. (2001). Manifesto for Agile Software Development. <http://agilemanifesto.org/>
26. Boehm, B. (1988). A spiral model of software development and enhancement. ACM SIGSOFT Software Engineering Notes, 11(4), 14-24.
27. Chin, G. (2004). Agile project management: How to succeed in the face of changing project requirements. AMACOM.

28. Cockburn, A. (2002). Agile Software Development: Principles, Patterns, and Practices. Addison-Wesley.
29. Fairley, R. (2009). Managing and leading software projects. Wiley-IEEE Computer Society Press.
30. Fowler, M. (2001). The New Methodology.  
<https://martinfowler.com/articles/newMethodology.html>
31. Ghosh, S. (2012). Comparative study of traditional and agile software development methodologies. International Journal of Computer Science and Information Technologies, 3(3), 4117-4120.
32. Highsmith, J. (2002). Agile software development ecosystems. Addison-Wesley.
33. Hughes, B., & Cotterell, M. (2009). Software Project Management (5th ed.). McGraw-Hill Education.
34. Larman, C. (2004). Agile and Iterative Development: A Manager's Guide. Addison-Wesley Professional.
35. Larman, C., & Basili, V. R. (2003). Iterative and incremental development: A brief history. Computer, 36(6), 47-56.
36. Poppendieck, M., & Poppendieck, T. (2003). Lean Software Development: An Agile Toolkit. Addison-Wesley.
37. Pressman, R. S. (2014). Ingeniería de Software: Un enfoque práctico (7ª ed.). McGraw-H
38. Proyecto, L. (2025). *Costos estimados por perfil profesional para el desarrollo del proyecto* [Tabla 1]. Elaboración propia.
39. Proyecto, L. (2025). *Matriz de riesgos del análisis comparativo de metodologías Scrum y Waterfall* [Imagen 1]. Elaboración propia.
40. Proyecto, L. (2025). *Distribución de horas por fase del proyecto según perfil profesional* [Tabla 2]. Elaboración propia.

41. International Organization for Standardization. (2017). *ISO/IEC/IEEE 12207:2017 - Software life cycle processes*. <https://www.iso.org/standard/63712.html>
42. Project Management Institute. (2017). *A guide to the project management body of knowledge (PMBOK® Guide) (6th ed.)*. Project Management Institute.
43. Pressman, R. S., & Maxim, B. R. (2020). *Software engineering: A practitioner's approach* (9th ed.). McGraw-Hill Education.
44. Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide: The definitive guide to Scrum: The rules of the game*. <https://www.scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-US.pdf>
45. Sommerville, I. (2016). *Software engineering* (10th ed.). Pearson Education.