



## PROCESO DE GESTIÓN DE FORMACIÓN PROFESIONAL INTEGRAL FORMATO GUÍA DE APRENDIZAJE

### IDENTIFICACIÓN DE LA GUIA DE APRENDIZAJE

- Denominación del Programa de Formación: Análisis y Desarrollo de Software
- Código del Programa de Formación: 228118
- Nombre del Proyecto construcción de software integrador de tecnologías orientadas a servicios.
- Fase del Proyecto Análisis
- Actividad de Proyecto Estructurar el plan de actividades de análisis partir de las características del proyecto y el modelo de desarrollo seleccionado. • Caracterizar la solución de software. • Identificar metodologías de desarrollo de software. • Seleccionar metodología de desarrollo de software.
- • Establecer las actividades de la metodología seleccionada
- Competencia o Técnica: Análisis de la especificación de requisitos del software.
- Resultados de Aprendizaje Alcanzar: Planear actividades de análisis de acuerdo con la metodología seleccionada.
- Claves
- Duración de la Guía: 30 horas

## 2. PRESENTACIÓN

Las metodologías de desarrollo de software proponen un conjunto de procesos y actividades que deben ser desarrolladas por el equipo de desarrollo de software para realizar un trabajo organizado que sea fácil hacerle seguimiento y de esta forma establecer planes de mejora en busca de una mejor calidad de los productos y servicios que se desarrollan. (Losavio, 2011). Existe un gran número de definiciones sobre lo que es una metodología, para evitar cualquier confusión en esta guía se utiliza la definición dada en Maida y Pacienza (2015), quienes indican que una metodología hace referencia a un conjunto de procedimientos genéricos y lógicos que se utilizan para alcanzar un objetivo particular usando un conjunto de habilidades y conocimientos. Mientras que un modelo para el desarrollo de software, es una representación abstracta de un proceso, desde una perspectiva particular para que proporcione información parcial sobre el proceso, son marcos de trabajo del proceso y que pueden ser adaptados para crear procesos más específicos.. (Dalvik, 2022).

Con el desarrollo de las actividades de aprendizaje que se plantearan, el aprendiz podrá evidenciar el conocimiento que adquirirá y reconocerá técnicas-herramientas que le permitan Planear actividades de análisis de acuerdo con la metodología seleccionada.



**Figura 1.**

*Metodologías de Desarrollo de Software*



(Banco Santander, 2022) <https://www.becas-santander.com/es/blog/metodologias-desarrollo-software.html>

### 3. FORMULACIÓN DE LAS ACTIVIDADES DE APRENDIZAJE

Durante las actividades a desarrollar los aprendices deben estar prestos a la socialización de la introducción que brinda el instructor para dar continuidad con el aprendizaje de las diferentes metodologías de software que existen en el mercado y su uso actualmente, seguidamente consultar en la web y en el material de las bases de datos de la biblioteca y académicas.

Las actividades se realizan de forma presencial-remota con la orientación del instructor y de manera autónoma con la revisión del material de apoyo, referencias bibliográficas y con la elaboración de talleres propuestos.

#### **Ambiente Requerido**

El ambiente de aprendizaje debe estar conformado por:

20 equipos con los requerimientos mínimos:

- ☐ Sistema operativo: Windows 10 Pro
- Disco Duro: 1 TERABYTE
- ☐ RAM: 32 GB
- ☐ Procesador: Intel XEON



- ☐ Mesas y sillas
- ☐ Cable HDMI
- ☐ Pantallas

1 servidor de aplicaciones

1 impresora mínimo

1 scanner

5 dispositivos móviles (Palm o Pocket PC)

Conexión a Internet permanente

Materiales

- Computadores de escritorio y portátiles con acceso a internet, software de aplicación para realizar
- informes y visualizar material digital.
- Marcadores

### **3.1 Actividades de reflexión inicial**

Realice la lectura del siguiente caso de estudio y brinde su apreciación.

La compañía Visión es una pequeña empresa dedicada al desarrollo de software, su sede es en el Parquesoft de Cali. Esta empresa tiene una participación en el mercado de desarrollo de software de gestión de un 45%, siendo líder en el desarrollo de diversos aplicativos relacionados con control y gestión de procesos para empresas importantes de la región. En la actualidad adelanta un proyecto dando la oportunidad a un nuevo equipo de empleados, recién egresados, que tiene el reto de desarrollar un aplicativo para la estructuración de los diferentes proyectos que llegan a la organización, el proyecto tiene una duración de 10 meses, iniciando el proceso, aún los implicados no tienen claros los requisitos del negocio, los modelos y metodologías que puede aplicarse de una forma eficiente a este proyecto, para que los tiempos proyectados se den y cumplan con los objetivos, dada la dinámica del mercado y la economía.

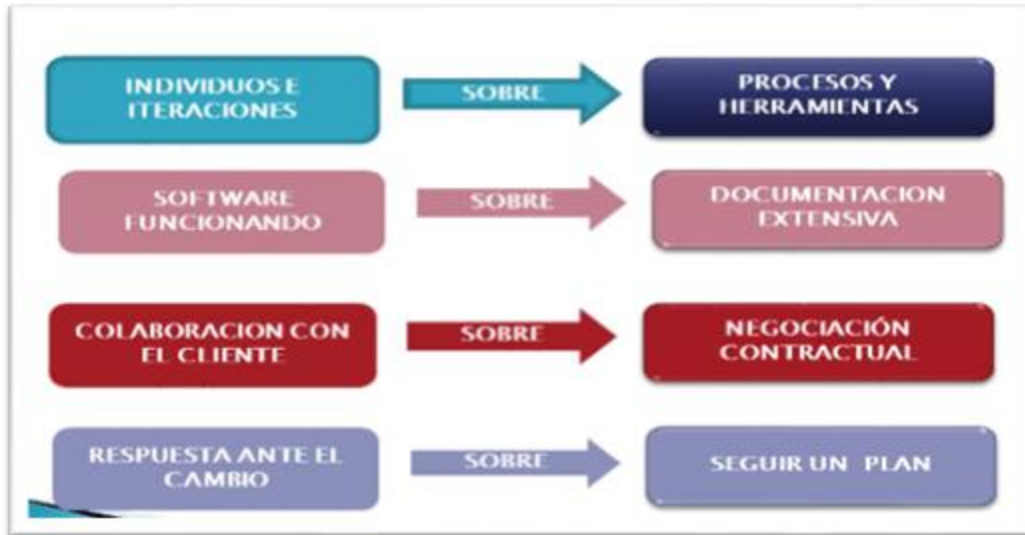
La empresa quiere saber cuál la opción para esperar o diferir el proyecto, basada en su experiencia con productos similares, la empresa sabe que puede esperar hasta un máximo de 2 meses antes de realizar dar un equipo de rescate al proyecto sin que esto genere una substancial pérdida en las ventas. Entrando la 3 mes no se identifica un cumplimiento de la planeación proyectada, funcionalidades que cumplan con la estimación de proyectos, se han hecho bastantes cambios en las reglas del negocio generando grandes brechas, lo cual se ha visto reflejado en la pérdida de un contrato muy importante.

- ¿Qué problemas se presentaron?
- ¿Qué pudo causar estos problemas?
- ¿Qué consecuencias trajo para la organización y para los clientes?



**Figura 2.**

*Características de Metodologías de Desarrollo de Software*



(Perez, 2015) <https://rei.iteso.mx/bitstream/handle/11117/2349/Modelado+de+Procesos.pdf?sessionid=8B712B70DDBC8C5A6F3AA221679BBED?sequence=2>

Las metodologías de desarrollo de software siempre parten de un componente teórico y cuando son usadas por los equipos de trabajo conllevan a la utilización de un conjunto de técnicas y métodos que al final determinarán las tareas generales y específicas que se deberían realizar para alcanzar un objetivo.

**Figura 2.**

*Ciclo de vida del Software.*



(Lucisdchart, s.f). <https://www.lucidchart.com/pages/es/que-es-la-notacion-de-modelado-de-procesos-de-negocio>



### 3.2 Actividades de contextualización e identificación de conocimientos necesarios para el aprendizaje.

Existen diferentes tipos de metodologías de desarrollo de software que fueron ideadas pensando en problemas particulares presentados en la industria en contextos específicos, por lo cual es importante conocer sus diferentes características y contrastarlas con las necesidades particulares a las que se enfrenta a la hora de desarrollar un producto y servicio, cada una tiene ventajas y enfoques que pueden ser reutilizados en diferentes momentos.

Existen dos grandes clasificaciones de metodologías de desarrollo de software que se agrupan generalmente como marcos de trabajo tradicionales o marcos de trabajo ágiles.

Para el desarrollo de un buen producto de software se debe iniciar por un excelente proceso de planificación y gestión de este durante todas las etapas y actividades que involucran transformar una idea o requerimiento en un producto o servicio que será usado por un cliente particular.

Los marcos de trabajo tradicionales o metodologías tradicionales se caracterizan por centrar la mayor parte de su esfuerzo en la planeación y control del proceso, lo que conlleva a una documentación exhaustiva y precisa de los artefactos que describen los requisitos y los modelos del sistema en las etapas iniciales del desarrollo del proyecto. (Maida y Pacienza, 2015).

#### Figura 3.

*Biblioteca Sena.*



Fuente: <https://biblioteca.sena.edu.co/>

Consultar en las bases de datos de la biblioteca y /o académicas para desarrollar un mapa conceptual.

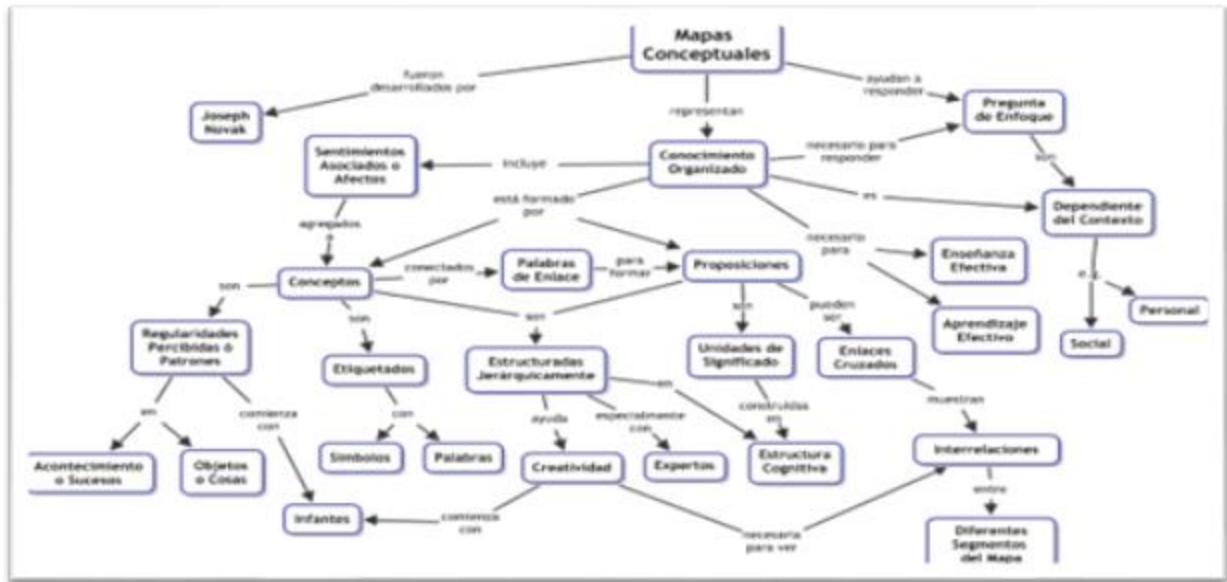
a. Marcos de trabajo tradicionales: RUP-MAF-METRICA3 identifique en cada uno sus características detalladamente.

b. Marcos de trabajo ágiles: XP-RAD-SCRUM- CRYSTALMETODOLOGIES-KANBAN identifique en cada uno sus características detalladamente.

El desarrollo de la investigación debe plasmarlos en un mapa conceptual el punto a y el punto b. de 4 niveles de acuerdo a los lineamientos del instructor.

**Figura 3.**

### Estructura Mapa Conceptual



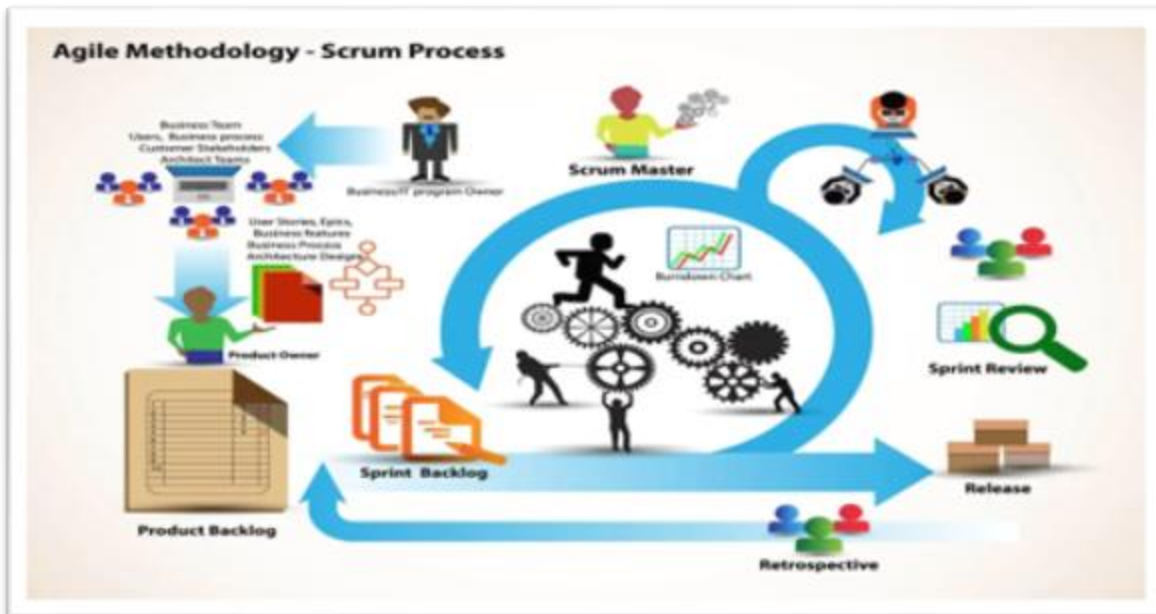
(Cañas & Novak, 2009) <https://cmap.ihmc.us/docs/mapaconceptual.php>

- a) ¿Cuál es el ciclo de vida del software detalle cada paso ( características)? Plasmarlo en un mapa conceptual.
- b) ¿Cuáles son los modelos de desarrollo de software, especifique prototipo, cascada, espiral ? plasmarlo en un cuadro comparativo.

La planificación de proyectos de software involucra un conjunto de actividades que buscan tener una imagen preliminar del software a construir a partir de la cual se realizan estimaciones que permiten determinar la viabilidad o no de un proyecto de acuerdo con la cantidad de recursos asignados .

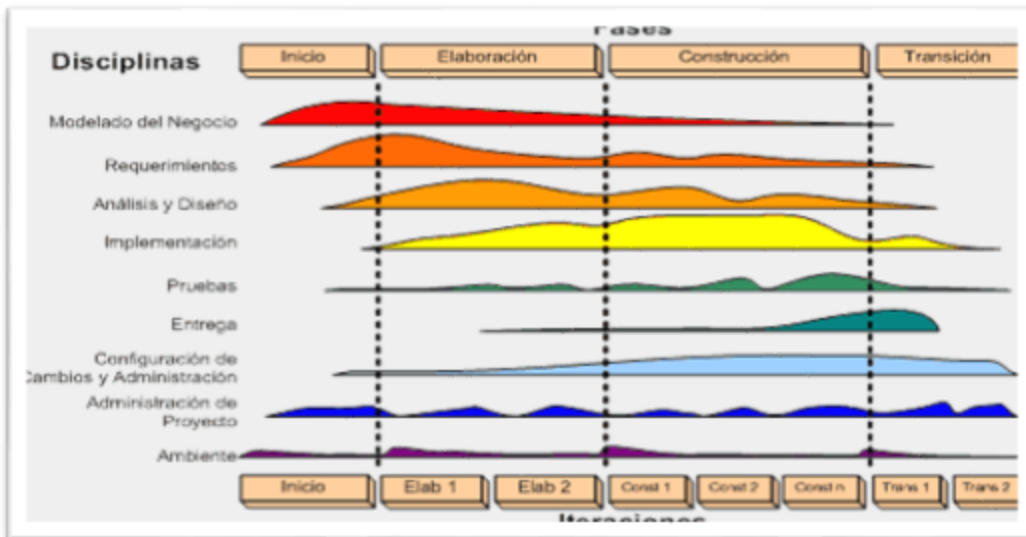
### Figura 4

## Metodologias Agiles



**Figura 4**

*Metodologías Robustas*



(Smartdraw, s.f) <https://www.smartdraw.com/flowchart/simbolos-de-diagramas-de-flujo.htm>

GFPI-F-135 V01

### 3.3 Actividades de Apropiación del conocimiento



RUP es una sigla en inglés equivalentes a Proceso Racional Unificado, el cual es un proceso de desarrollo de software tradicional basado en el modelo cascada y que fue desarrollado por la empresa Rational Software que es propiedad de IBM, esta metodología se centra en la arquitectura y es guía por casos de uso (requerimientos). RUP divide el proceso de desarrollo en cuatro grandes fases, dentro de las que se realizan algunas iteraciones donde se desglosan, en mayor o menor intensidad, un conjunto de disciplinas según la fase que se está abordando.

La primera gran fase definida por RUP es la de inicio, en la cual se abordan actividades principalmente enfocadas en la comprensión del problema y el tipo de tecnología a utilizar, por lo que hay una gran carga en actividades relacionadas con la disciplina de modelado del negocio, se identifican los riesgos asociados al proyecto y especificación de requisitos.

En la Fase de Elaboración: (Definición, análisis y diseño), se diseña la solución preliminar, se selecciona los casos de uso que permiten definir la arquitectura base del sistema y se desarrolla el primer análisis del dominio del problema. Dando continuidad la Fase de desarrollo o construcción: (implementación), se completa la funcionalidad del sistema, se clarifican los requisitos pendientes, se administran los cambios de acuerdo a las evaluaciones realizadas por los usuarios, y se realizan las mejoras para el proyecto.

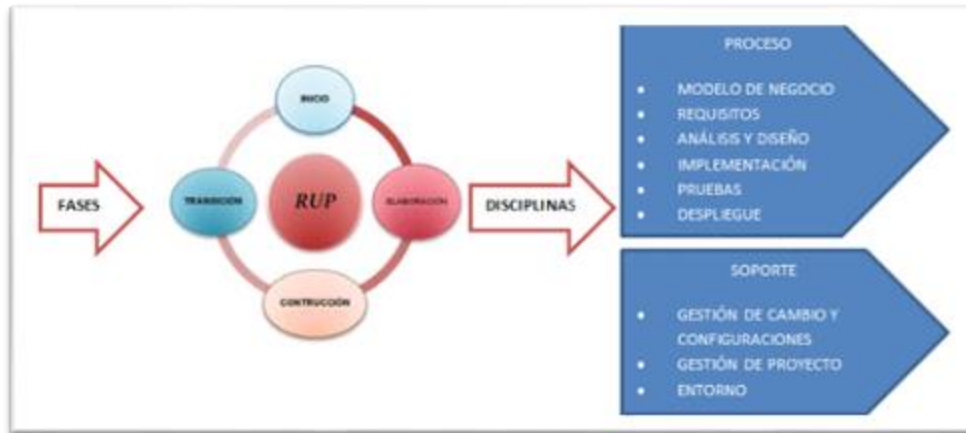
Seguidamente en la Fase de transición: (fin del proyecto y pruebas) o fase de cierre, el propósito es asegurar que el software esté disponible para los usuarios finales, se ajustan los errores y defectos encontrados en las pruebas de aceptación, se capacitan a los usuarios y se provee el soporte necesario.

Planear las 4 fases incluye: Asignación de tiempo, Hitos Principales, Iteraciones por Fases, Plan de proyecto. Cada fase puede descomponerse en iteraciones que son un ciclo de desarrollo completo dando como resultado una entrega de producto ejecutable (interna o externa). En el proceso se definen una serie de roles y los se distribuyen entre los miembros del proyecto y que definen las tareas de cada uno y el resultado (artefactos) que se espera de ellos. Todos los miembros del equipo deben compartir la base de conocimiento, el Proceso, una vista de cómo desarrollar software y un lenguaje de modelamiento (UML).

Los procesos siempre deberán adaptarse a las necesidades del cliente e interactuar con él, las características propias del proyecto, el tamaño del mismo, su tipo o las regulaciones que lo condicionen, influirán en su diseño específico. (Metodología grupo 23, 2014).

**Figura 5.**  
*Fases y Disciplinas RUP*

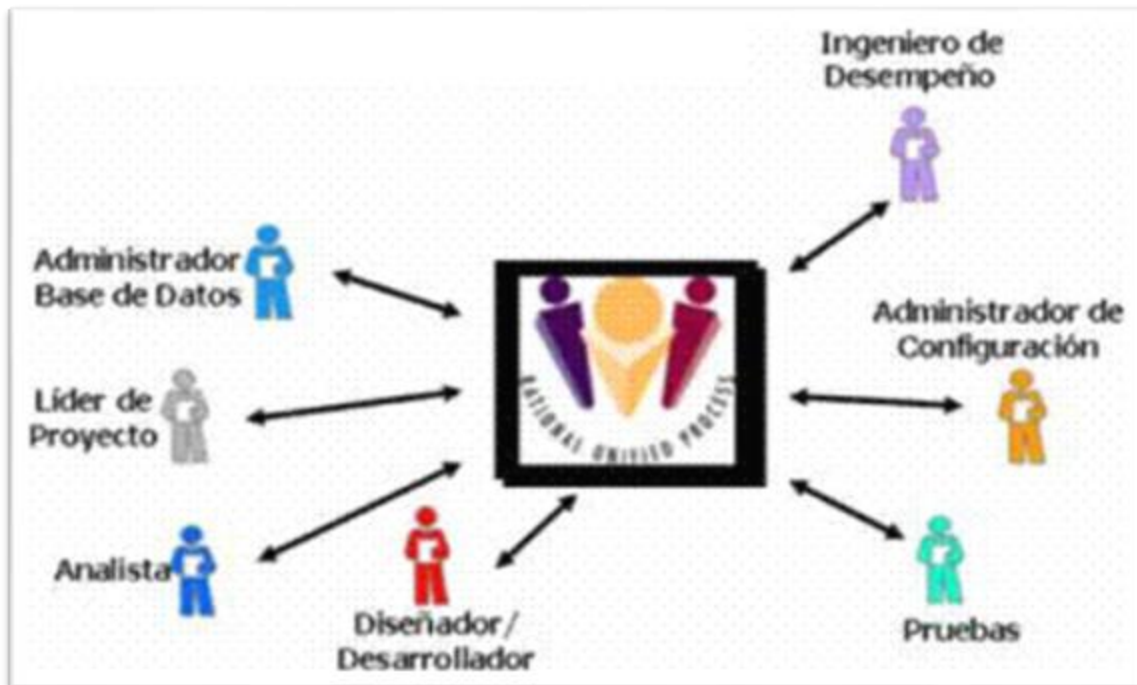




(Metodologíagrupo23,2014) <http://metodologiarupgrupo23.blogspot.com/2014/03/fases-metodologia-rup.html>

**Figura 6.**

*Roles metodología RUP*



(Ramírez, 2020) <https://eglisramirez6.wordpress.com/conociendo-tecnologias/>

### 3.4 Actividades de transferencia del conocimiento



Una de las metodologías Ágiles más reconocidas es Scrum, tiene sus principios fundamentales en la década de 1980, fue desarrollada por su necesidad en procesos de reingeniería por Goldratt (licenciado en Física de la Universidad de Tel Aviv, realizó su máster y doctorado en la Universidad de Bar-Ilan, creador de la Teoría de Restricciones), Takeuchi (físico y matemático japonés nacionalizado colombiano, reconocido docente e impulsor de la matemática del siglo XX en Colombia) y Nonaka (teórico organizacional japonés y profesor emérito de la Graduate School of International Corporate Strategy de la Universidad Hitotsubashi, más conocido por su estudio de la gestión del conocimiento), y nuevos procesos de desarrollo utilizados en productos exitosos en Japón y los Estados con requisitos muy generales para salir la mercado en poco tiempo, se comparaba la forma de trabajo de los equipos altamente productivos y multidisciplinarios con la colaboración entre los jugadores de Rugby y su formación de Scrum, de la cual se tomó su nombre.

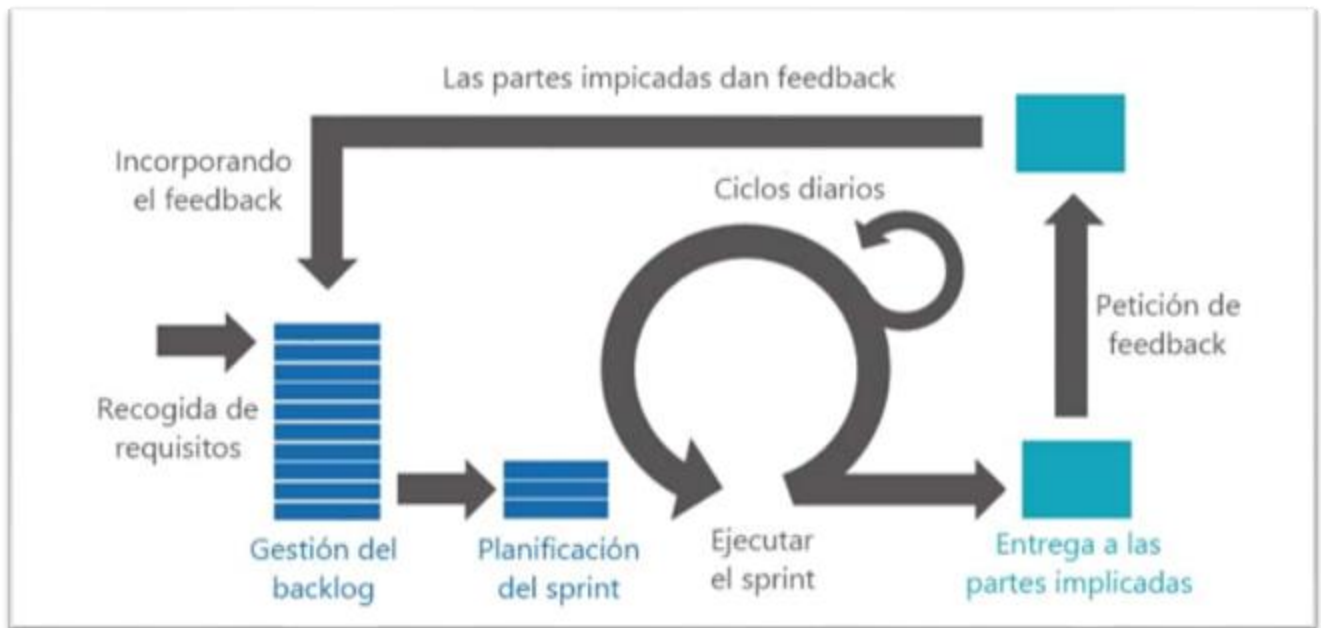
Scrum es un conjunto de buenas prácticas, para trabajar colaborativamente, en equipo, y obtener los mejores resultados posibles en un proyecto, las prácticas se debe apoyar unas a otras, su selección tiene origen en un estudio de la manera de trabajar de equipos altamente productivos. Se podría indicar que **se basa en cierto caos controlado, pero establece ciertos mecanismos para controlar esta indeterminación, manipular lo impredecible y controlar la flexibilidad.**

En Scrum se realizan entregas parciales y regulares del producto final, esta indicado para Proyectos en entornos complejos, donde se necesita obtener resultados pronto, donde los requisitos son cambiantes o poco definidos, donde la innovación, la competitividad, la flexibilidad, la productividad son fundamentales, también se utiliza para resolver situaciones en que no se está entregando al cliente lo que necesita, cuando las entregas se alargan demasiado, los costos se disparan o la calidad no es aceptable, cuando se necesita capacidad de reacción ante la competencia, cuando la moral de los equipos es baja y la rotación alta, cuando es necesario identificar y solucionar ineficiencias sistemáticamente o cuando se quiere trabajar utilizando un proceso especializado en el desarrollo de producto.

(Maida y Pacienza, 2015).

#### **Figura 6.**

*Proceso SCRUM*



(Maida y Pacienza, 2015). <https://repositorio.uca.edu.ar/bitstream/123456789/522/1/metodologias-desarrollo-software.pdf>

En Scrum un proyecto se ejecuta en bloques temporales cortos y fijos (iteraciones de 2 a 4 semanas como máximo), tiene que proporcionar un resultado completo, susceptible de ser entregado con el mínimo esfuerzo al cliente cuando lo solicite. El proceso parte de la lista de objetivos/requisitos priorizada del producto (Product Backlog), que actúa como plan del proyecto. En esta lista el cliente prioriza los objetivos balanceando el valor que le aportan respecto a su coste y quedan repartidos en iteraciones y entregas.

Los roles en Scrum son:

- Cliente (Product Owner): Definir los objetivos del producto o proyecto, dirigir los resultados del proyecto y maximizar su rol.
- Facilitador (Scrum Master): Se encarga de liderar al equipo, controlar que todos los participantes del proyecto sigan los valores y principios ágiles, las reglas, procesos, guiar la colaboración dentro del equipo y con el cliente de manera que las sinergias sean máximas, asegurar que exista una lista de requisitos priorizada y que esté preparada antes de la siguiente iteración.
- Equipo (Team): Es el grupo de personas que de manera conjunta desarrollan el producto del



proyecto, tienen un objetivo común y comparten la responsabilidad del trabajo que realizan.

(Maida y Pacienza, 2015).

Finalmente, y ya se tiene un conocimiento de la metodologías y modelos de software, se deberán desarrollar los estudios de caso planteados y atender las recomendaciones para aplicar los ajustes correspondientes antes de ser presentados en la presentación final del proyecto.

#### 4. ACTIVIDADES DE EVALUACIÓN

Evidencias de Aprendizaje	Criterios de Evaluación	Técnicas e Instrumentos de Evaluación
<b>Evidencias de Conocimiento :</b>  Prueba en línea sobre metodologías  <b>Evidencias de Desempeño:</b>  Desarrolla de talleres y actividades propuestas en la formación.  <b>Evidencias de Producto:</b>  Desarrollo de la guía de acuerdo a la normatividad.	Identifica procesos de la organización de acuerdo con la estructura organizacional de la empresa y los requerimientos del cliente.  Aplica técnicas de análisis de procesos, siguiendo la metodología establecida.  Aplica buenas prácticas de uso de la tecnología TIC, de acuerdo con los estándares y recomendaciones.  Participa de manera activa y coherente con las actividades propuestas.  Entrega actividades con pertinencia y calidad.  Maneja computadores, periféricos, tabletas y equipos celulares, de acuerdo con las funcionalidades y manuales de uso.  Aplica funcionalidades de sistema operativo, de acuerdo con las necesidades de administración de los recursos del equipo.	Cuestionario en línea.  Observación Directa  Lista de chequeo



	<p>Maneja procesador de texto, de acuerdo con las funcionalidades de los programas.</p> <p>Utiliza motores de búsqueda, programas de navegación, correo electrónico, transferencia de archivos, chat, programas de E-Learning y computación en la nube, de acuerdo con las necesidades de información y comunicación.</p>	
--	---	--

## 5. GLOSARIO DE TÉRMINOS

**Metodologías de desarrollo de Software:** Es un marco de trabajo usado para estructurar, planificar y controlar el proceso de desarrollo en sistemas de información.

**Modelos de desarrollo de Software:** Lo componen un conjunto de conceptos y prácticas con una estructuración definida que se usan como base para organizar y desarrollar un proyecto de software, o ciclo de vida del software.

**Metodología Agile:** Permite adaptar la forma del trabajo a las condiciones del proyecto, proporcionando flexibilidad e inmediatez en la comunicación con el cliente para amoldar el proyecto y su desarrollo a las circunstancias específicas del entorno.

**Metodologías robustas o tradicionales.** Fuerte planificación, centran su atención en llevar una documentación exhaustiva de todo el proceso de desarrollo y en cumplir con un plan de proyecto, definido en la fase inicial del mismo.

**Actividad:** Una actividad representa el trabajo realizado dentro de un proceso de negocio

## 6. REFERENTES BIBLIOGRÁFICOS

Abrahamsson P, S. Outi, J. Ronkainen, y J. Warsta, «Agile software development methods: review and analysis». 2017.

Betancur Cartagena L. F., «Propuesta estratégica de prácticas seguras para el desarrollo de software con metodologías ágiles», Tesis de Maestría, Universidad Nacional de Colombia - Sede Medellín, Colombia, 2016.



- Kumar G y P. Kumar Bhatia, «Comparative analysis of software engineering models from traditional to modern methodologies», presentado en Fourth International Conference on Advanced Computing & Communication Technologies, Rohtak, Haryana, India, 2014, pp. 189-196.
- Metodologiarupgrupo23(24 de marzo del 2014). Metodología desarrollo software – RUP[Blog]. Recuperado <http://metodologiarupgrupo23.blogspot.com/>.
- Montoya Suarez L. M., J. M. Sepúlveda Castaño, y L. M. Jiménez Ramos, «Análisis comparativo de las metodologías ágiles en el desarrollo de software aplicadas en Colombia», en Gestión del Talento Humano: Enfoques y Modelos, Corporación Centro Internacional de Marketing Territorial para la Educación y el Desarrollo - CIMTED, 2016, pp. 450-464.
- Navarro Cadavid, J. D. Fernández Martínez, y J. Morales Vélez, «Revisión de metodologías ágiles para el desarrollo de software», Prospectiva, vol. 11, n.o 2, pp. 30–39, 2013. Y. D. Amaya Balaguera, «Metodologías ágiles en el desarrollo de aplicaciones para dispositivos móviles», Revista de Tecnología, vol. 12, n.o 2, pp. 111–124, 2013.
- Navarro M. E., M. P. Moreno, J. Aranda, L. Parra, J. R. Rueda, y J. C. Pantano, «Selección de metodologías ágiles e integración de arquitecturas de software en el desarrollo de sistemas de información», presentado en XIX Workshop de Investigadores en Ciencias de la Computación - WICC 2017, Buenos Aires, Argentina, 2017, pp. 632–636.
- Orjuela Duarte A y M. Rojas C., «Las metodologías de desarrollo ágil como una oportunidad para la ingeniería del software educativo», Revista Avances en Sistemas e Informática, vol. 5, n.o 2, 2008.
- [15] R. G. Figueroa, C. J. Solís, y A. A. Cabrera, «Metodologías tradicionales vs. metodologías ágiles». Universidad Técnica Particular de Loja - Escuela de Ciencias en Computación, 2008.
- Peralta A y F. P. Romero, «Toma de Decisiones a partir de Conocimiento Extraído tras el Análisis de Comportamientos Previos. Aplicación Práctica para la Dirección de Proyectos de Desarrollo de Software», Rev. CINTEX, vol. 20, n.o 2, pp. 97-111, dic. 2015.
- Pérez A. O. A., «Cuatro enfoques metodológicos para el desarrollo de software RUP-MSF-XP-SCRUM», Inventum, vol. 10, pp. 64-78, 2011.
- Restrepo Ángel J. S., «Guía de buenas prácticas aplicable a la metodología de desarrollo ágil SCRUM para fortalecer la seguridad de la información», Trabajo de grado, Institución Universitaria Politécnico Grancolombiano, Bogotá, Colombia, 2017.
- Tiwari, «Software Engineering Issues in Development Models of Open Source Software», International Journal of Computer Science and Technology, vol. 2, n.o 2, pp. 38-44, 2011.
- Velásquez Restrepo Sandra Milena, Juan David Vahos-Montoya, Marta Ester Gómez-Adasme, Ana Alexandra Pino – Martínez, Erika Juliett Restrepo-Zapata & Sebastián Londoño-Marín. “Una revisión comparativa de la literatura acerca de metodologías tradicionales y modernas de desarrollo de software” Revista CINTEX, Vol. 24(2), pp. 13-23. 2019.
- Yepes González J. D., «AgileFM: Modelo de desarrollo ágil formal basado en la ISO/IEC 29110 para las micro, pequeñas y medianas empresas», Tesis de Maestría, Universidad EAFIT, Medellín, Colombia, 2016.



## 7. CONTROL DEL DOCUMENTO

	Nombre	Cargo	Dependencia	Fecha
Autor (es)	Paola Tovar Rugeles	Instructor	Teleinformática	noviembre 2022

## 8. CONTROL DE CAMBIOS (diligenciar únicamente si realiza ajustes a la guía)

	Nombre	Cargo	Dependencia	Fecha	Razón del Cambio
Autor (es)					