PROCESO DE GESTIÓN DE FORMACIÓN PROFESIONAL INTEGRAL
FORMATO GUÍA DE APRENDIZAJE

1. IDENTIFICACIÓN DE LA GUÍA 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:** especificar el modelo conceptual y estructurar la propuesta técnica del proyecto de software.
- **Competencias:**
  Técnicas:
  220501093 - Evaluar requisitos de la solución de software de acuerdo con metodologías de análisis y estándares.
  220501094 - Estructurar propuesta técnica de servicio de tecnología de la información según requisitos técnicos y normativa.

  Claves:
  240201528 - Razonar cuantitativamente frente a situaciones susceptibles de ser abordadas de manera matemática en contextos laborales, sociales y personales.
  240202501 - Interactuar en lengua inglesa de forma oral y escrita dentro de contextos sociales y laborales según los criterios establecidos por el Marco Común Europeo de Referencia para las Lenguas.

- **Resultados de aprendizaje a alcanzar:**
  Técnicos:
  220501093-02 - Modelar las funciones del software de acuerdo con el informe de requisitos.
  220501093-04 - Verificar los modelos realizados en la fase de análisis de acuerdo con lo establecido en el informe de requisitos.

  220501094-01 - Definir especificaciones técnicas de acuerdo con las características del software a construir.
  220501094-02 - Elaborar propuesta técnica del software de acuerdo con las especificaciones técnicas definidas.
  220501094-03 - Validar las condiciones de la propuesta técnica del software de acuerdo con los intereses de las partes.

  Claves:
  240201528-01 - Identificar modelos matemáticos de acuerdo con los requerimientos del problema planteado en contextos sociales y productivo.
  240201528-02 - Plantear problemas matemáticos a partir de situaciones generadas en el contexto social y productivo.
  240201528-03 - Resolver problemas matemáticos a partir de situaciones generadas en el contexto social y productivo.
  240201528-04 - Proponer acciones de mejora frente a los resultados de los procedimientos matemáticos de acuerdo con el problema planteado.

240202501-02 - Intercambiar opiniones sobre situaciones cotidianas y laborales actuales, pasadas y futuras en contextos sociales orales y escritos.

- **Duración de la guía:** 384 horas
  - Técnico: 240 horas.
  - Claves: 144 horas.

2. PRESENTACIÓN

Estimado aprendiz, el SENA extiende una cordial bienvenida a la segunda guía de aprendizaje que comprende las competencias técnicas de: evaluar requisitos de la solución de software de acuerdo con metodologías de análisis y estándares, y la competencia de estructurar propuesta técnica de servicio de tecnología de la información según requisitos técnicos y normativa.

Por otra parte, la sociedad actual demanda que todos los ciudadanos tengan conocimientos básicos de matemáticas en cualquier ámbito, no soglo el profesional o académico; muchos problemas de la vida diaria requieren un conocimiento mínimo sobre operaciones básicas como la suma, la resta, la multiplicación o la división; por ejemplo: revisar el saldo de una tarjeta de crédito, solicitar un préstamo, comprar víveres o realizar un presupuesto mensual. Y es que, es casi imposible encontrar una profesión o carrera que no lleve implícito algún componente lógico– matemático, o que incluya alguna rama de las matemáticas.

Finalmente, encontrarán la competencia clave de inglés como una de las habilidades contemporáneas más importantes y de mayor impacto en el mundo laboral y social, teniendo en cuenta el aprendizaje articulado de las cuatro habilidades de la lengua (leer, escribir, hablar y escuchar) alineado al MCERL (Marco Común Europeo de Referencia para Lenguas) como estándar del nivel de conocimiento de lengua y categorización de los conocimientos en niveles de competencia en un idioma.

Para el desarrollo de las actividades planteadas en esta guía, contará con el acompañamiento de los instructores asignados al programa, los cuales de forma continua y permanente lo orientarán con las pautas necesarias para el logro de las actividades de aprendizaje, brindando herramientas básicas de tipo conceptual y metodológico. Los instructores programaran encuentros de asesoría virtual, para brindar orientaciones específicas relacionadas con las temáticas a desarrollar en las actividades. Es importante que organice su tiempo, dada la exigencia que demanda la realización de esta guía de aprendizaje. No olvide revisar y explorar los materiales de estudio del programa.

Por consiguiente, se presentan cada una de las acciones de aprendizaje que le permitirán desarrollar lo anteriormente mencionado.

3. FORMULACIÓN DE LAS ACTIVIDADES DE APRENDIZAJE

En este apartado se describirán las actividades de aprendizaje para cada una de las competencias que plantea la fase de planeación del proyecto formativo: determinar las especificaciones funcionales del software y metodología a utilizar.

De este programa de formación, en las actividades se detallan cada una de las evidencias a entregar durante las diez (10) semanas planeadas para esta guía. En el siguiente cuadro se observa para cada una de las competencias, el momento y las evidencias a entregar según las actividades de aprendizaje asociadas a cada competencia:

Competencia/Semana	Semana 1	Semana 2	Semana 3	Semana 4	Semana 5	Semana 6	Semana 7	Semana 8	Semana 9	Semana 10
220501093	Evidencia de conocimiento: Foro temático: Lenguaje Unificado de Modelado (UML). GA2-220501093-AA1-EV01.	Evidencia de desempeño: Elaboración de diagramas y plantillas para casos de uso del proyecto. GA2-220501093-AA1-EV02.	Evidencia de desempeño: Elaboración de historias de usuario del proyecto. GA2-220501093-AA1-EV03.	Evidencia de producto: Diagramas y documentación de actividades del proyecto. GA2-220501093-AA1-EV04.	Evidencia de desempeño: Elaboración de los diagramas del modelo de dominio del proyecto. GA2-220501093-AA2-EV01.		Evidencias de conocimiento: Mapa conceptual sobre validación de documentos. GA2-220501093-AA3-EV01.		Evidencia de desempeño: Elaboración del informe de análisis con listas de chequeo para la validación de artefactos. GA2-220501093-AA3-EV02.	
	12	12	12	12	12	12	6	6	6	6

220501094	Evidencia de conocimiento: Foro temático: Licenciamiento de software. GA2-220501094-AA1-EV01.	Evidencia de desempeño: Diseño de las fichas técnicas para la recolección de la información. GA2-220501094-AA1-EV02.		Evidencia de producto: Especificación de los referentes técnicos del hardware - software y estimación de las condiciones económicas. GA2-220501094-AA1-EV03.		Evidencia de producto: Ficha técnica de los productos requeridos, los términos de referencia para el contrato y lista de chequeo para evaluación de proveedores. GA2-220501094-AA2-EV01	Evidencia de desempeño: Diseño de tablas comparativas sobre presupuestos de hardware y software. GA2-220501094-AA3-EV01.	Evidencia de producto: Propuesta técnica y económica para la implementación del proyecto. GA2-220501094-AA3-EV02.	Evidencia de conocimiento: Taller de especificación del modelo conceptual y estructurar la propuesta técnica del proyecto de software. GA2-220501094-AA4-EV01.	Evidencia de producto: Propuestas técnica y económica con ajustes de acuerdo a la negociación tecnológica. GA2-220501094-AA4-EV02.
	12	16	16	16	12	12	12	12	18	18
240201528	Evidencia de conocimiento: Cuestionario procedimientos aritméticos. GA2-240201528-AA1-EV01.	Evidencia de desempeño: Planteamiento de ecuación. GA2-240201528-AA2-EV01.	Evidencia de producto: Video sustentación. GA2-240201528-AA3-EV01.	Evidencia de desempeño: Algoritmo para el cálculo de áreas y volúmenes. GA2-240201528-AA4-EV01.						
	12	12	12	12						
240202501					Evidencia de conocimiento: Cuestionario. GA2-240202501-AA1-EV01.	Evidencia de desempeño: Video entrevista virtual. GA2-240202501-AA1-EV02.	Evidencia de producto: Crónica. GA2-240202501-AA1-EV03.	Evidencia de conocimiento: Cuestionario. GA2-240202501-AA2-EV01.	Evidencia de desempeño: presentación. GA2-240202501-AA2-EV02.	Evidencia de producto: documento escrito. GA2-240202501-AA2-EV03.

					16	16	16	16	16	16
Total hrs semana	36	40	40	40	40	40	34	34	40	40

3.1 Actividades de aprendizaje de la competencia evaluar requisitos de la solución de software de acuerdo con metodologías de análisis y estándares-220501093.

Con base en los requisitos de *software* especificados con esta competencia se empieza construir los artefactos del modelo necesarios para representar la solución, por medio de los siguientes documentos: diagramas de clase, documentos de casos de uso o historias de usuario en plantillas, modelo de base de datos, modelo del dominio, diagramas de actividades y también un documento informe con los resultados obtenidos.

3.1.1 Actividad de aprendizaje GA2-220501093-AA1: elaborar diagrama y documentación de casos de uso / historias de usuario de acuerdo con el refinamiento de requisitos.

Esta actividad se centra en el estudio de la elaboración de los artefactos de casos de uso e historias de usuario acordes a los requerimientos del sistema a implementar.

Duración: 48 horas.

Materiales de formación: para el desarrollo de esta actividad es importante la lectura y análisis del material de formación: "**Diagramas para la especificación y análisis de requisitos".**

Evidencias:

A continuación, se describen las acciones y las correspondientes evidencias que conforma la actividad de aprendizaje:

- **Evidencia GA2-220501093-AA1-EV01: foro temático: Lenguaje Unificado de Modelado (UML).**

Comentado [U1]: No hay ninguna descripción de lo que se debe hacer en el foro. Sobre que gira el foro, para hacer las participaciones. Falta todo de la evidencia.

El lenguaje UML es fundamental para representar el proceso de analisis de requisitos de software por tanto el estudiante debe proponer su punto de vista acerca de la importancia del lenguaje de modelado en el proceso de desarrollo de software asi como tambien identificar y describir los tipos de diagramas que existen y el conjunto de simbolos que se usan para definirlos y debatir con los demas compañeros acerca de sus experiencias con las mejores herramientas de software que permiten de una manera agil construir los diagramas.

Responder las siguientes preguntas:

¿ Cual es la importancia de UML ?
¿ Cuales son los diagramas que se pueden reprentar con UML?
¿Que herramientas de software existen en el mercado para UML?

Elementos a tener en cuenta:

- Indicar los tipos de símbolos existentes en UML.
- Dar ejemplos claros de tipos de diagramas que se pueden crear con UML
- Dar una opinión satisfactoria a las preguntas de 3 compañeros acerca de UML.
- Cumplir con todas las normas de ortografía al contestar el foro temático.
-

Lineamientos generales para la entrega de la evidencia:

- **Producto para entregar:** participación en el foro.
- **Formato:** LMS.
- **Extensión:** libre.
- Para hacer el envío de la evidencia remítase al área de la actividad correspondiente y acceda al espacio: **Foro temático: Lenguaje Unificado de Modelado (UML). GA2-220501093-AA1-EV01.**

- **Evidencia GA2-220501093-AA1-EV02: elaboración de diagramas y plantillas para casos de uso del proyecto.**

Comentado [U2]: Igual que el comentario anterior. No dice que se debe hacer en la evidencia. Se sugiere que revise otras guías ya finalizadas y se dé cuenta cómo se construye una evidencia. Esto aplica para todas las evidencias técnicas.

Con base en el estudio de la evidencia anterior acerca de UML se pretende construir los diagramas que aplican en este caso los de casos de uso bajo un formato estandar de facil nomenclatura y de reconocimiento que pertmita llevar un control ordenado de la actviddad , registrar la evolucion, responsables y trazabilidad de esos documentos de casos de uso.

Elementos a tener en cuenta:

- Estudiar detenidamente los conceptos y características definidas en el componente formativo.
- Revisar el video sugerido como material complementario del componente formativo.
- Revisar acerca de como se construye el documento de casos de uso de una funcionalidad del sistemas.
- Tener presente los requisitos del software.
- Indicar los tipos de diagramas que existen para modelar con base en los requisitos del software.
- Diagramar con UML los artefactos del sistema.
- Manejar herramientas de software para apoyar la elaboración de los diagramas.
- Elaborar documentos plantilla de casos de uso con base en estándares de documentación.

Lineamientos generales para la entrega de la evidencia:

- **Producto para entregar:** diagramas, documento de casos de uso e historias de usuario.
- **Formato:** PDF o Word**.**
- **Extensión:** libre.
- Para hacer el envío de la evidencia remítase al área de la actividad correspondiente y acceda al espacio: **Elaboración de diagramas y plantillas para casos de uso del proyecto. GA2-220501093-AA1-EV02.**

- **Evidencia GA2-220501093-AA1-EV03: elaboración de historias de usuario del proyecto.**

El Concepto de historias de usuario viene asociado a la implementación de metodologias ágiles de desarrollo de software en donde en un doumento se describe de manera corta lo mas simple y clara posible la necesidad del

usuario o el punto a resolver de tal manera que no se generen procesos tan largos para llegar a un resultado se supone que maximo puede ser de 1 semana.

Aspectos a tener en cuenta:

- Se deben seguir las normas básicas de presentación de un documento escrito; es decir, el documento debe tener como mínimo una portada, introducción, alcance, lista de requerimientos y versión del documento. Los requerimientos serán redactados usando el modelo IEEE830 y también el modelo de descripción de requisitos por medio de historias de usuario.
- Identificar la metodología de desarrollo a seguir.
- Identificar los requerimientos del sistema.
- Manejar herramientas de software para apoyar la elaboración de los documentos.
- Elaborar documento de caso de uso a desarrollar.
- Respecto a lista de requerimientos el aprendiz deberá agregar una sección donde se describa cada requisito usando los siguientes elementos del estándar IEEE830:
  - Perspectiva del producto.
  - Funciones del producto.
  - Características de los usuarios.
  - Restricciones.
  - Requisitos funcionales (formato de casos de uso).
  - Requisitos no funcionales.
- Respecto a la lista de requerimientos el aprendiz deberá agregar una sección donde se describa cada requisito usando la estructura de historias de usuario con los siguientes elementos por historia:
  - Número de historia (priorizada).
  - Nombre de la historia.
  - Usuario.
  - Puntos estimados de esfuerzo.
  - Descripción de la historia de usuario.
  - Observaciones.
  - Criterios de aceptación.

Lineamientos generales para la entrega de la evidencia:

- **Producto para entregar:** documento de historias de usuario.
- **Formato:** PDF.
- **Extensión:** libre.
- Para hacer el envío de la evidencia remítase al área de la actividad correspondiente y acceda al espacio: **Elaboración de historias de usuario del proyecto. GA2-220501093-AA1-EV03.**

- **Evidencia GA2-220501093-AA1-EV04: diagramas y documentación de actividades del proyecto.**

Comentado [U3]: Incompleta.

Con base en los requisitos del sistema ya especificados se empiezan a construir los artefactos del modelo necesarios para representar la solución de software por medio de los siguientes documentos: diagramas de clase, documentos de casos de uso o historias de usuario en plantillas, modelo de base de datos, modelo del dominio, diagramas de actividades y tambien un documentos informe con los resultados obtenidos.

Aspectos a tener en cuenta:

- Estudiar detenidamente los conceptos y características definidas en el componente formativo.
- Revisar el video sugerido como material complementario del componente formativo.
- Manejar el lenguaje de modelado UML.

- Manejar Metodologias Tipo Agil.
- Identificar la metodología de desarrollo a seguir.
- Identificar el conjunto de actividades a cumplir.
- Manejar herramientas de software para construir los diagramas.
- Elaborar el documento de cronograma de trabajo del proyecto.

Lineamientos generales para la entrega de la evidencia:

- **Producto para entregar:** diagramas, documento de casos de uso e historias de usuario.
- **Formato:** PDF.
- **Extensión:** libre.
- Para hacer el envío de la evidencia remítase al área de la actividad correspondiente y acceda al espacio: **Diagramas y documentación de actividades del proyecto. GA2-220501093-AA1-EV04.**

3.1.2 Actividad de aprendizaje GA2-220501093-AA2: representar el contexto del negocio a través del diagrama de dominio y actividades.

Esta actividad se centra en el estudio de los componentes necesarios para diagramar el contexto del negocio y representar su dominio, así como también el cómo construir un diagrama de actividades.

Duración: 24 horas.

Comentado [U4]: La planeación dice otra cantidad horaria.

Materiales de formación: para el desarrollo de esta actividad es importante la lectura y análisis del material de formación: "**Construcción de requerimientos.".**

Evidencias:

A continuación, se describen las acciones y las correspondientes evidencias que conforma la actividad de aprendizaje:

- **Evidencia GA2-220501093-AA2-EV01: elaboración de los diagramas del modelo de dominio del proyecto.**

Comentado [U5]: Incompleta.

Esta actividad se centra en la elaboracion del diagrama de clases del proyecto acerca de como identificar las categorias que intervienen como estas se relacionan teniendo en cuenta su cardinalidad, dependencias y herencias llevando esto a la organización de objetos por medio de segmentacion de copmponentes en paquetes claramente identificables.

Aspectos a tener en cuenta:

- Se deben seguir las características e instrucciones para la elaboración de un diagrama de dominio del proyecto.
- El modelo conceptual resultante debe guardar las proporciones necesarias de modo que se vea ordenado y sea fácil su lectura.
- Para complementar el modelo del dominio se puede representar por medio de otra vista que correponde al diagrama de paquetes del proyecto.
- Tener en cuenta los requisitos del software.
- Identificar el tipo de diagrama apropiado para modelar el dominio.

- Diagramar con UML los artefactos del sistema diagrama de clases y de paquetes.
- Manejar herramientas de software para apoyar la elaboración de los diagramas.
- Elaborar documento plantilla de casos de uso con base en estándares de documentación.

Lineamientos generales para la entrega de la evidencia:

- **Producto para entregar:** documento del dominio del proyecto.
- **Formato:** PDF.
- **Extensión:** libre.
- Para hacer el envío de la evidencia remítase al área de la actividad correspondiente y acceda al espacio: **Elaboración de los diagramas del modelo de dominio del proyecto. GA2-220501093-AA2-EV01.**

Comentado [U6]: Elija uno, un archivo de imagen nos e que tan compatible sea con Word, debido a la calidad con que quede.

Comentado [7]: *Respuesta a Usuario (17/11/2021, 18:21): "..."*
Se selecciona archivos PDF debido a su compatibilidad para diferentes plataformas.

3.1.3 Actividad de aprendizaje GA2-220501093-AA3: validar artefactos de análisis generados.

Esta actividad se centra en realizar la verificación de los documentos generados.

Duración: 24 horas.

Materiales de formación: para el desarrollo de esta actividad es importante la lectura y análisis del material de formación: "**Validación de documentos".**

Evidencias:

A continuación, se describen las acciones y las correspondientes evidencias que conforma la actividad de aprendizaje:

- **Evidencia GA2-220501093-AA3-EV01: mapa conceptual sobre validación de documentos.**

Comentado [U8]: Incompleta.

Por medio de esta evidencia se prente establecer una secuecia lógica de dependencias para llevar a acabo la validacion de los documentos identificados en el proceso de desarrollo de software.

Elementos a tener en cuenta en el mapa conceptual:

- Se deben seguir las características e instrucciones para la elaboración del mapa conceptual identificando categorias y sus relaciones desde un orden lógico.
- El modelo conceptual resultante debe guardar las proporciones necesarias de modo que se vea ordenado y sea fácil su lectura.
- Manejar herramientas de software que faciliten la elaboracion de los diagrama y sus fácil entendimiento.
- Identificar la metodología de desarrollo a seguir.
- Identificar los conceptos para relacionar los documentos a validar.
- Construir lo formatos de aceptación.
- Elaborar el documento con diagrama conceptual.

Lineamientos para la entrega del producto:

- **Producto para entregar:** Mapas conceptuales con la validación de documentos.
- **Formato:** PDF, Word o JPG.
- **Extensión:** libre.

- Para hacer el envío de la evidencia remítase al área de la actividad correspondiente y acceda al espacio: **Mapa conceptual sobre validación de documentos. GA2-220501093-AA3-EV01.**

- **Evidencia GA2-220501093-AA3-EV02: elaboración del informe de análisis con listas de chequeo para la validación de artefactos.**

Comentado [U9]: Incompleta.

Esta evidencia esta orientada hacia la elaboración de un documento informe para validar los artefactos del software diseñados y que estén acorde son lo requerido por el cliente.

Elementos a tener en cuenta:

- Se deben seguir las normas básicas de presentación de un documento escrito, es decir el documento debe tener como mínimo una portada, introducción, alcance, lista de requerimientos y versión del documento.
- El informe debe evidenciar la implementación de características paso a paso de cumplimiento.
- El informe debe evidenciar la implementación de estrategias de cumplimiento.
- Construir el documento de informe de artefactos entregados.
- Identificar los procesos y usuarios interesados.
- Validar con los usuarios los requerimientos encontrados.
- Elaborar el documento de lista de chequeo paso a paso.

Lineamientos generales para la entrega de la evidencia:

- **Producto para entregar:** informe técnico.
- **Formato:** PDF**.**
- **Extensión:** libre
- Para hacer el envío de la evidencia remítase al área de la actividad correspondiente y acceda al espacio: **Elaboración del informe de análisis con listas de chequeo para la validación de artefactos. GA2-220501093-AA3-EV02.**

3.2. Actividades de aprendizaje de la competencia estructurar propuesta técnica de servicio de tecnología de la información según requisitos técnicos y normativa-220501094.

En esta competencia se detalla la forma correcta de preparar, elaborar y presentar una propuesta técnica de *software* de manera adecuada, siguiendo los parámetros y especificaciones técnicas establecidas, teniendo en cuenta como insumo principal las necesidades de los *stakeholders* o partes interesadas.

3.2.1 Actividad de aprendizaje GA2-220501094-AA1: Determinar las especificaciones técnicas a partir del diagnóstico de necesidades informáticas

Esta actividad se centra en determinar especificaciones técnicas de una solución de *software*, como son el diseño y elaboración de fichas técnicas, licencias de *software* y las condiciones económicas del producto.

Duración: 72 horas.

Materiales de formación: Para el desarrollo de esta actividad es importante la lectura y análisis del material de formación: componente formativo 10 especificaciones de software.

Evidencias:

A continuación, se describen las acciones y las correspondientes evidencias que conforma la actividad de aprendizaje:

- **Evidencia de conocimiento GA2-220501094-AA1-EV01: Foro Temático: Licenciamiento de software**

Lineamientos para la entrega del producto:

Teniendo en cuenta que los licenciamientos de software son contratos que detallan los derechos legales y el uso de una solución de software, analice la realidad actual de este tema y exprese su concepto generando aportes significativos sobre los tipos de licencia existentes y cuáles son los más utilizados.

Como preguntas de orientación para realizar la participación en el foro, se proponen las siguientes:

- ✓ ¿Qué es software?
- ✓ ¿Qué es una licencia de software?
- ✓ ¿Tipos de licencia de software?
- ✓ ¿Cuáles son las más adecuadas y por qué ?

Realice seguimiento al foro y responda a un comentario crítico-reflexivo de acuerdo con la intervención que hayan realizado sus compañeros demostrando construcción de conocimiento frente a los planteamientos que cada uno proponga.

Elementos a tener para la participación en el foro:

- Estudiar detenidamente los conceptos y características definidas en el componente formativo de especificaciones de software.
- Revisar los temas sugeridos como material complementario del componente formativo.

Lineamientos generales para la entrega de la evidencia:

- **Productos para entregar:** participación en el foro.
- **Formato:** LMS**.**
- **Extensión:** libre.
- Para hacer el envío de la evidencia remítase al área de la actividad correspondiente y acceda al espacio: **Foro temático: Licenciamiento de software. GA2-220501094-AA1-EV01.**

- **Evidencia GA2-220501094-AA1-EV02: diseño de las fichas técnicas para la recolección de la información.**

  las fichas técnicas es un documento que reúne las características de los productos de software desarrollados y permiten el registro e identificación de las fuentes de información, hay que diseñarlas teniendo en cuenta los estándares definidos en el componente formativo 10 especificaciones de software.

Elementos a tener en cuenta:

Comentado [U10]: Es un cuestionario o un foro. Revisar y ajustar.

Comentado [MFMC11]: Listo

Comentado [U12]: Cuáles estándares, cómo se hace. Está incompleta.

- Estudiar detenidamente los conceptos y características definidas en el componente formativo de especificaciones de software.
- Revisar los temas sugeridos como material complementario del componente formativo.
- Para la elaboración de la ficha técnica se debe apoyar en las normatividades en este caso se realizará la ficha técnica tomando como referencia la Norma NTC ISO 9000 Sistema de Gestión de Calidad
- Realizar el diseño de las fichas técnicas de acuerdo a la información que se requiera recolectar, se debe teniendo en cuenta las características de las fuentes consultadas.

Lineamientos generales para la entrega de la evidencia:

- **Productos a entregar:** Fichas técnicas para la recolección de la información.
- **Formato:** PDF, Word o Excel.
- **Extensión:** formatos de fichas diligenciado.
- Para hacer el envío de la evidencia remítase al área de la actividad correspondiente y acceda al espacio: **Planograma de exhibición. GA2-220501094-AA1-EV02.**

- **Evidencia GA2-220501094-AA1-EV03: especificación de los referentes técnicos del hardware - software y estimación de las condiciones económicas.**

Las especificaciones de los referentes técnicos son muy importantes en la elaboración de una solución de software respecto a las necesidades de la empresa, por otra parte, la estimación de las condiciones económicas permite tener una proyección de lo que se puede desarrollar.

Elementos a tener en cuenta en el documento:

- Para elaborar el documento se deben seguir las condiciones básicas estipuladas en el componente formativo 10 especificaciones de software.
- Se debe estimar las condiciones técnicas y económicas de acuerdo con las características de la solución de software y las necesidades de la empresa.
- Presentación de un documento escrito; es decir, el documento debe tener como mínimo una portada, introducción, contenido y conclusión

Lineamientos generales para la entrega de la evidencia:

- **Producto para entregar:** documento informe.
- **Formato:** PDF.
- **Extensión:** mínimo 3 hojas máximo 10 de contenido.
- Para hacer el envío de la evidencia remítase al área de la actividad correspondiente y acceda al espacio: **Especificación de los referentes técnicos del hardware - software y estimación de las condiciones económicas. GA2-220501094-AA1-EV03.**

3.2.2 Actividad de aprendizaje GA2-220501094-AA2: realizar el análisis de proveedores a partir de las fichas técnicas.

Esta actividad se centra en el estudio del análisis comparativo de proveedores considerando costos, cantidad y especificaciones técnicas para seleccionar las mejores opciones.

Duración: 12 horas.

Materiales de formación: Para el desarrollo de esta actividad es importante la lectura y análisis del componente formativo 11 Propuesta técnica.

Evidencias:

A continuación, se describen las acciones y las correspondientes evidencias que conforma la actividad de aprendizaje:

- **Evidencia GA2-220501094-AA2-EV01: ficha técnica de los productos requeridos, los términos de referencia para el contrato y lista de chequeo para evaluación de proveedores.**

La ficha técnica es un documento que contiene la descripción, características y funciones de un determinado producto.

Elementos a tener en cuenta en el documento:

- Para elaborar el documento se deben seguir las condiciones básicas estipuladas en el componente formativo 11 propuesta técnica.
- Elaborar la ficha técnica de los productos requeridos y los términos de referencia para el contrato de manera adecuada
- Las fichas técnicas deben contener las descripciones, características y funciones del producto
- Presentación de un documento escrito, es decir el documento debe tener como mínimo una portada, introducción, contenido y conclusión

Lineamientos generales para la entrega de la evidencia:

Teniendo en cuenta los términos de referencia, establecidos en el componente formativo 11 “propuesta técnica”, elaborar la ficha técnica, teniendo en cuenta los siguientes conceptos.

- ✓ Descripciones
- ✓ Características
- ✓ funciones:
- ✓ El formato se debe ser manera horizontal
- ✓ Se puede utilizar tablas de Word o Excel
- ✓ Se debe enriquecer con un diseño o logo, como si fuera la papelería de una empresa real

- **Producto para entregar:** Ficha técnica
- **Formato:** Excel o Word**.**
- **Extensión:** Libre.

3.2.3 Actividad de aprendizaje GA2-220501094-AA3: elaborar la propuesta técnica y financiera a partir de los referentes establecidos.

Esta actividad se centra en que el aprendiz obtenga las competencias necesarias en los temas de estimación de costos de *software* y *hardware* de acuerdo con las características del negocio y la correcta elaboración de una propuesta técnica en el estudio del análisis comparativo de proveedores considerando costos, cantidad y especificaciones técnicas para seleccionar las mejores opciones.

Duración: 24 horas.

Materiales de formación: Para el desarrollo de esta actividad es importante la lectura y análisis del componente formativo 11propuesta

Evidencias:

A continuación, se describen las acciones y las correspondientes evidencias que conforma la actividad de aprendizaje:

- **Evidencia GA2-220501094-AA3-EV01: diseño de tablas comparativas sobre presupuestos de hardware y software.**

Las tablas comparativas son utilizadas para estimar costos en el de presupuesto de *software* y *hardware*.

Elementos a tener en cuenta en el documento:

- Para elaborar el documento se deben seguir las condiciones básicas estipuladas en el componente formativo número 11 de propuesta técnica.
- Elaborar el Diseño de tablas comparativas sobre presupuestos de hardware y software, realizando una correcta, estimación costos
- Presentación de un documento escrito, es decir el documento debe tener como mínimo una portada, introducción, contenido y conclusión

Lineamientos generales para la entrega de la evidencia:

- **Producto para entregar:** diseño de tablas comparativas sobre presupuestos de *hardware* y *software*.
- **Formato:** PDF, Word o Excel.
- **Extensión:** Libre.
- Para hacer el envío de la evidencia remítase al área de la actividad correspondiente y acceda al espacio: **Diseño de tablas comparativas sobre presupuestos de hardware y software. GA2-220501094-AA3-EV01.**

Comentado [U13]: De esta AA faltó la evidencia:
Evidencia de producto:
Propuesta técnica y económica para la implementación del proyecto
GA2-220501094-AA3-EV02. OJO.

Comentado [MFMC14]: Corregido

- **Evidencias GA2-220501094-AA3-EV02: Propuesta técnica y económica para la implementación del proyecto**

La propuesta técnica es el documento que se entrega al posible cliente, en la cual se detalla la estrategia a desarrollar para darle solución a un problema existente por medio de una solución de software
Elementos a tener en cuenta en el documento:

- Para elaborar el documento se deben seguir las condiciones básicas estipuladas en el componente formativo 11 propuesta técnica.
- Elaborar la propuesta técnica y económica para la implementación del proyecto de manera adecuada
- Presentación de un documento escrito, es decir el documento debe tener como mínimo una portada, introducción, contenido y conclusión

Lineamientos generales para la entrega de la evidencia:

Teniendo en cuenta los términos de referencia, establecidos en el componente formativo 11 “propuesta técnica”, elaborar la propuesta técnica la cual debe tener el siguiente contenido

- ✓ Presentación y saludos
- ✓ Alcance de la Propuesta
- ✓ Valoración de la situación
- ✓ Opciones de Propuesta
- ✓ Estimar costos
- ✓ Cotización
- ✓ Términos y condiciones
- ✓ La propuesta se debe enriquecer con un diseño o logo, como si fuera la papelería de una empresa real

- **Producto para entregar:** Propuesta técnica
- **Formato:** PDF o Word**.**
- **Extensión:** mínimo dos (2) opciones de propuestas técnicas con su respectiva cotización.
- Para hacer el envío de la evidencia remítase al área de la actividad correspondiente y acceda al espacio: **Ficha técnica de los productos requeridos, los términos de referencia para el contrato y lista de chequeo para evaluación de proveedores. GA2-220501094-AA2-EV01.**

3.2.4 Actividad de aprendizaje GA2-220501094-AA4: presentar la propuesta técnica y económica de acuerdo con las condiciones del contrato a desarrollar.

Esta actividad se centra en la presentación de la propuesta técnica y económica, con los debidos ajustes obtenidos como resultado de las sugerencias y la negociación tecnológica.

Duración: 36 horas.

Materiales de formación: Para el desarrollo de esta actividad es importante la lectura y análisis del material de formación: "**Propuesta técnica".**

Evidencias:

A continuación, se describen las acciones y las correspondientes evidencias que conforma la actividad de aprendizaje:

- **Evidencia GA2-220501094-AA4-EV01: taller de especificación del modelo conceptual y estructurar la propuesta técnica del proyecto de software.**

Una propuesta técnica se centra en proporcionar información sobre una solución de software para resolver un problema de un negocio, automatizando un proceso, estas pueden ser presentadas en diferentes formatos, dependiendo del proyecto y objetivo previsto.

En este taller se realizará una representación de una propuesta técnica, definiendo los conceptos básicos y la estructura

Elementos a tener en cuenta en el documento del informe técnico:

- Leer detenidamente los conceptos básicos definidos en el componente formativo 11 Propuesta técnica.
- Se deben seguir las normas básicas de presentación de un documento escrito, es decir el documento debe tener como mínimo una portada, introducción, y conclusiones.
- Documentar las sugerencias del cliente de acuerdo a las necesidades del negocio
- Ajustas parámetros técnicos según resultados de la negociación
- Realizar una propuesta alternativa con otras especificaciones y costos
- Consultar material de referente al tema para enriquecer el producto.

Lineamientos generales para la entrega de la evidencia:

- **Producto para entregar:** informe técnico.
- **Formato:** PDF.
- **Extensión:** libre.
- Para hacer el envío de la evidencia remítase al área de la actividad correspondiente y acceda al espacio: **Taller de especificación del modelo conceptual y estructurar la propuesta técnica del proyecto de software. GA2-220501094-AA4-EV01.**

- **Evidencia GA2-220501094-AA4-EV02: propuestas técnica y económica con ajustes de acuerdo a la negociación tecnológica.**

Con base en la propuesta técnica desarrollada, realizar los respectivos ajustes técnicos, de acuerdo con los resultados de la negociación y las sugerencias del cliente.

Elementos a tener en cuenta en el documento:

- Para elaborar el documento se deben seguir las condiciones básicas estipuladas en el componente formativo 11 propuesta técnica.
- Presentación de un documento escrito, es decir el documento debe tener como mínimo una portada, introducción, contenido y conclusión

Lineamientos generales para la entrega de la evidencia:

Teniendo en cuenta los términos de referencia, establecidos en el componente formativo 11 “propuesta técnica”, realizar los ajustes a la propuesta técnica presentada, esta debe tener:

- ✓ Presentación y saludos
- ✓ Alcance de la Propuesta
- ✓ Valoración de la situación
- ✓ Opciones de Propuesta
- ✓ Estimar costos
- ✓ Cotización
- ✓ Términos y condiciones
- ✓ La propuesta se debe enriquecer con un diseño o logo, como si fuera la papelería de una empresa real

- **Producto para entregar:** Propuesta técnica con las correcciones sugeridas
- **Formato:** PDF**.**
- **Extensión:** libre.
- Para hacer el envío de la evidencia remítase al área de la actividad correspondiente y acceda al espacio: **Ficha técnica de los productos requeridos, los términos de referencia para el contrato y lista de chequeo para evaluación de proveedores. GA2-220501094-AA2-EV01.**

3.3 Actividades de aprendizaje de la competencia razonar cuantitativamente frente a situaciones susceptibles de ser abordadas de manera matemática en contextos laborales, sociales y personales-240201528.

Las actividades propuestas para el desarrollo de esta competencia, están diseñadas para potenciar el trabajo autónomo con las labores de investigación y generación de documentos sobre las temáticas planteadas, y el trabajo colaborativo, a partir de la interacción de aprendices e instructores mediante encuentros sincrónicos y asincrónicos.

3.3.1 Actividad de aprendizaje GA2-240201528-AA1: desarrollar procedimientos aritméticos para resolver problemáticas de la vida cotidiana.

Comentado [U15]: Falta la descripción de la AA.

Duración: 12 horas.

Materiales de formación: para el desarrollo adecuado de las evidencias el aprendiz debe apropiar los contenidos del componente formativo “**Básico**”.

Evidencias:

A continuación, se describen las acciones y las correspondientes evidencias que conforman la actividad de aprendizaje:

- **Evidencia GA2-240201528-AA1-EV01: cuestionario procedimientos aritméticos.**

Para el desarrollo de esta evidencia, en primera instancia consulte el anexo "**Preguntas_conjuntos-240201528-AA1-EV01",** en el que se presentan una serie de preguntas similares a las que encontrará en la evaluación, pruebe resolverlas y comente cualquier inquietud al instructor. Luego debe responder la evaluación disponible en plataforma.

Antes de responder las preguntas es importante tener en cuenta lo siguiente:

- Revisar el material de formación (conjuntos numéricos, operaciones aritméticas, razones y proporciones) seguramente le permitirá comprender con detalle los conceptos.
- Debe responder la totalidad de las preguntas formuladas en la evaluación en línea

Lineamientos generales para la entrega de la evidencia:

- **Producto para entregar:** evaluación.
- **Formato:** LMS**.**
- **Duración:** 50 minutos.
- Para hacer el envío de la evidencia remítase al área de la actividad correspondiente y acceda al espacio: **Cuestionario procedimientos aritméticos. GA2-240201528-AA1-EV01.**

3.3.2 Actividad de aprendizaje GA2-240201528-AA2: resolver problemas de aplicación de situaciones de los contextos productivo y social a partir del uso de herramientas matemáticas.

A continuación, se describen las acciones y las correspondientes evidencias que conforman la actividad de aprendizaje:

Comentado [U16]: Esto hace parte del espacio **Evidencias:** acomodarla allá y acá hacer la descripción de la AA

Duración: 12 horas

Materiales de formación: para el desarrollo adecuado de las evidencias el aprendiz debe apropiar los contenidos del componente formativo **"Medio"**.

Evidencias:

- **Evidencia GA2-240201528-AA2-EV01: planteamiento de ecuación.**

Para esta evidencia se toma como estrategia el aprendizaje basado en problemas, en el cual usted resolverá un problema de aplicación utilizando las herramientas matemáticas propuestas en el material de formación del componente medio.

Problema de aplicación:
Una firma de arquitectos en una estrategia de mercadeo muy innovadora busca entregar a cada uno de sus clientes una casa en escala de chocolate, como la que se ve en la figura:



Figura 1: Prototipo de casa de chocolate.
(imagen tomada de Getty Images/iStockphoto)

La repostería que contrataron para llevar a cabo dicho proyecto tiene dos inconvenientes. El primero es el uso óptimo de la materia prima en el diseño de las casas y el segundo es encontrar una opción económicamente viable para el empaque de la casa pues al ser comestible debe estar protegida con una vitrina de metacrilato.

Se solicita que para aportar a la solución de esta situación realice lo siguiente:

a. Plantee una ecuación que represente el área total de la casa de chocolate.
b. Busque una función que represente el costo total de una casa de chocolate vs cantidad de casas de chocolate. Para esto debe tener en cuenta que hay unos gastos fijos como el costo de la materia prima, el salario de los reposteros, costo del material de la vitrina en la que se entregará la casa entre otros.
c. Proponga una solución más rentable para la entrega de casas de chocolate.
d. Plasmar estos resultados en un documento donde justifique la solución que le dio al problema.

Informe planteamiento de ecuación:

Para el desarrollo de esta evidencia deberá elaborar un informe en el cuál aborde los siguientes puntos:

- Portada.
- La ecuación que define el área total de la casa y su respectiva explicación.
- La función que relaciona el costo de producción de las casas vs cantidad de casas de chocolate, con su respectiva justificación.
- Propuesta de mejora de proceso.
- Conclusiones.
- Bibliografía (Se puede apoyar en la biblioteca virtual del Sena http://biblioteca.sena.edu.co/)

Lineamientos generales para la entrega de la evidencia:

- **Producto para entregar:** informe del planteamiento de la ecuación.
- **Formato:** PDF.

- Para hacer el envío de la evidencia remítase al área de la actividad correspondiente y acceda al espacio: **Planteamiento de ecuación. GA2-240201528-AA2-EV01.**

3.3.3 Actividad de aprendizaje GA2-240201528-AA3: realizar un muestreo estadístico acerca de una situación contextualizada en la vida diaria.

A continuación, se describen las acciones y las correspondientes evidencias que conforman la actividad de aprendizaje:

Comentado [U17]: Esto hace parte del espacio **Evidencias:** acomodarla allá y acá hacer la descripción de la AA

Duración: 12 horas.

Materiales de formación: para el desarrollo adecuado de las evidencias el aprendiz debe apropiar los contenidos del componente formativo "**Intermedio**".

Evidencias:

- **Evidencia GA2-240201528-AA3-EV01: video sustentación.**

Para esta actividad se toma como estrategia el aprendizaje estudio de caso, en el cual usted debe realizar el análisis de la información presentada utilizando las herramientas matemáticas propuestas en el **componente intermedio.**

Caso de estudio: Censo de población y vivienda en Colombia.
Busque información en la página oficial sobre el último Censo realizado en Colombia, el cual da cuenta de la población y vivienda, a partir de un conteo y caracterización de los habitantes colombianos, así como de sus viviendas y hogares constituidos en el país.
Con esta información obtenida se pretende generar información estadística relevante, la cual es un referente para la toma de decisiones en el orden político, económico y bienestar social, entre otros factores.

La información se presenta distribuida en tres categorías (https://www.dane.gov.co/index.php/estadisticas-por-tema/demografia-y-poblacion/censo-nacional-de-poblacion-y-vivienda-2018):

- ¿Cuántos somos? Presenta la distribución de la población por sexo, grupos de edad (estructura de la población) y el porcentaje de personas (proporción) que tienen dificultades para realizar actividades cotidianas, así como las que saben leer y escribir, las que asisten a alguna institución educativa en los departamentos y municipios de la región, nivel educativo alcanzado, fecundidad por grupos de edad, promedio de hijos por mujer. También se presenta información de migración interdepartamental en el último año, características de los extranjeros residentes en Colombia (sexo, edad, ocupación, nivel educativo), inmigración internacional en los últimos 5 años y el último año.

- ¿Dónde estamos? Presenta la distribución por ubicación geográfica de la población, los hogares y las viviendas en el territorio nacional (cabeceras municipales, rural disperso, centros poblados), así como el lugar de nacimiento de los residentes en los departamentos y municipios de la región. También se presentan los indicadores demográficos por departamento, la tasa de

alfabetismo por departamento, el porcentaje (proporción) de migración interna por departamento.

- ¿Cómo vivimos? Presenta el total de hogares y viviendas, su uso (residencial o mixto), su distribución por tipo (casa, apartamento, cuarto, étnica), el acceso a servicios públicos, de dónde obtienen el agua los hogares para preparar los alimentos; además, el porcentaje de personas (proporción) por hogar en los departamentos y municipios de la región, los tipos de hogar (unipersonal, nuclear, monoparental, extensos), y porcentaje de mujeres y hombres que son jefes de hogar.

Realice un análisis detallado de la información, revise si existen gráficos estadísticos que soporten dicha información, en caso de no encontrarlos, elabórelos. (Por ejemplo, histogramas, diagramas de barras, diagramas circulares, entre otros, dependiendo de las variables analizadas).

A partir de la información consultada, determine las variables que fueron objeto de medición en este censo, y precise a partir de ello, cómo estos datos son esenciales para determinar el desarrollo de las variables demográficas. (Por ejemplo, el tamaño de los hogares, el índice de envejecimiento, el índice de juventud, los fenómenos migratorios dentro del país y, desde y hacia el exterior, por mencionar algunos).

Con base en los resultados obtenidos, por medio de un video promocional exponga las conclusiones extraídas de la consulta del caso. Debe incluir en el video:

- Una presentación inicial.
- La explicación del instrumento usado para la recolección de datos y la caracterización de los mismos.
- Tablas y gráficos (utilice colores adecuados para la presentación de gráficos).
- Análisis de la información tabulada o graficada.
- Conclusiones con base en el análisis de la información.
- Procure utilizar colores adecuados en toda la presentación, si va a hacer uso de imágenes, estas deben ser acordes al tema tratado y al público al que se va a presentar.
- Pruebe ayudarse con el siguiente texto: https://rua.ua.es/dspace/bitstream/10045/4412/2/02-Consejos%20buena%20presentaci%C3%83%C2%B3n.pdf

Tenga en cuenta los siguientes términos de referencia de forma:

- Grabar un video de la presentación de su propuesta de lanzamiento. Esta debe tener una duración de máximo 5 minutos.
- El Video debe estar editado y poseer buena calidad de audio.
- Se debe expresar el proceso y la solución a este (idea propuesta).
- El aprendiz debe ser el presentador del video.
- Se pueden utilizar ayudas audiovisuales que permitan comprender mucho más la propuesta.
- No hay límite para la creatividad.
- Al final del video deben aparecer los créditos con los datos del aprendiz.

Especificaciones técnicas del video:

- Formato de grabación: MP4.
- Resolución: Mínimo 720p
- Equipos recomendados para hacer la grabación: teléfono celular, cámaras digitales, cámaras de computador, cámaras de acción, entre otras.
- Programas recomendados para hacer la edición del video: *Windows Movie Maker, Adobe Premiere, IMovie, Adobe Spark*, entre otros.

Lineamientos generales para la entrega de la evidencia:

- **Producto a entregar:** video sustentación.
- **Formato de entrega:** MP4. Resolución: Mínimo 720 p.
- Para hacer el envío de la evidencia remítase al área de la actividad correspondiente y acceda al espacio**: Video sustentación-GA2-240201528-AA3-EV01**

3.3.4 Actividad de aprendizaje GA2-240201528-AA4: crear un algoritmo para sistematizar el cálculo de perímetros, áreas y volumen de figuras planas y sólidos regulares.

Esta actividad consiste en proponer un algoritmo que permita calcular el área y perímetro de figuras planas y el volumen de sólidos regulares valiéndose de herramientas computacionales.

Duración: 12 horas

Materiales de formación: para el desarrollo adecuado de las evidencias el aprendiz debe apropiar los contenidos del componente formativo "**Avanzado**".

Evidencias:

A continuación, se describen las acciones y las correspondientes evidencias que conforman la actividad de aprendizaje:

- **Evidencia GA2-240201528-AA4-EV01: algoritmo para el cálculo de áreas y volúmenes.**

Se recomienda lo siguiente:

a. Consultar las figuras geométricas y las fórmulas que definen tanto el área como el volumen según sea el caso. Para ello se recomienda el apoyo en recursos multimedia y otras alternativas bibliográficas a las que usted tenga acceso.
b. Puede guiarse por el siguiente material multimedia, el cual le ayudará a orientarse frente al diseño del algoritmo. (Revise la videografía que se encuentra en https://www.youtube.com/watch?v=XJNdP-kxgUE).
c. Después de tener la información completa y organizada diseñe un algoritmo.
d. Piense en la siguiente pregunta ¿Si tuviera un sólido irregular qué método utilizaría para calcular el volumen?
e. Realice una presentación donde explique la solución al problema dado.

Para la entrega de esta evidencia tenga en cuenta que la presentación debe tener:

- Introducción.
- Problema planteado.
- La información recolectada de fórmulas y figuras.
- El algoritmo diseñado (incluyendo el proceso).
- Conclusiones.

Lineamientos generales para la entrega de la evidencia:

- **Producto a entregar:** algoritmo para el cálculo de áreas y volúmenes.
- **Formato:** PDF.
- Para hacer el envío de la evidencia remítase al área de la actividad correspondiente y acceda al espacio**: Algoritmo para el cálculo de áreas y volúmenes. GA2-240201528-AA4-EV01.**

3.4 Actividades de aprendizaje de la competencia interactuar en lengua inglesa de forma oral y escrita dentro de contextos sociales y laborales según los criterios establecidos por el Marco Común Europeo de Referencia para las Lenguas-240202501.

Durante el desarrollo de las actividades se debe comprender vocabulario y expresiones en inglés en contextos personales acorde con nivel principiante A1.2 de acuerdo con el MCERL (La construcción de los diferentes programas de la formación en inglés emplea el MCERL como instrumento de referencia para proveer los conocimientos requeridos para alcanzar el nivel)

3.4.1 Actividad de aprendizaje GA2-240202501-AAA1: reportar opiniones sobre situaciones cotidianas y laborales, pasadas y futuras en contextos sociales.

Dentro del desarrollo de la competencia comunicativa en inglés, es fundamental ser capaz de expresar opiniones y puntos de vista acerca de las situaciones a las que el aprendiz se enfrenta diariamente, tanto en su contexto social como su contexto laboral.

El aprendiz está llamado a ser observador y crítico, por esta razón es muy importante que conozca la estructura y el vocabulario apropiado para dar su opinión acerca de diferentes temáticas relacionadas directamente con su quehacer profesional.

Es importante el seguir las indicaciones del instructor en cuanto a la mejor manera de apropiar el material del curso y la correlación que tiene este con el programa de formación.

Duración: 48 horas.

Material de formación: para el desarrollo adecuado de las evidencias el aprendiz debe apropiar los contenidos del componente formativo **“LEVEL 2 - MCER A1.2”.**

Evidencias:

A continuación, se describen las acciones y las correspondientes evidencias que conforman la actividad de aprendizaje:

- **Evidencia GA2-240202501-AA1-EV01: cuestionario.**

Presenta un cuestionario (evaluación en línea) para evaluar comprensión lectora y gramatical del nivel (LEVEL 2 - MCER A1.2), el cual consta de quince preguntas (15) y un tiempo aproximado de 45 minutos.

Lineamientos para la entrega de la evidencia:

- **Producto a entregar**: cuestionario con preguntas de acuerdo a los temas relacionados del segundo nivel "LEVEL 2 - MCER A1.2".
- Para responder el cuestionario (evaluación en línea), remítase al área de la actividad correspondiente y acceda al espacio: C**uestionario. GA2-240202501-AA1-EV01.**

- **Evidencia GA2-240202501-AA1-EV02: video entrevista virtual.**

Con el apoyo de uno de sus compañeros y una vez apropiado el material de estudio del segundo nivel realicé un video, en el cual realicé una presentación de una entrevista de trabajo, postulándose para un empleo relacionado con su perfil profesional. Luego de ser entrevistado intercambie el rol con su compañero para ser el entrevistador.

Durante la entrevista deberá responder las siguientes preguntas:

1. *Tell us about yourself*
2. *What motivates you?*
3. *What are your hobbies?*
4. *What are your future plans if you get this job?*
5. *What are your future goals or what are your career goals?*
6. *What do you see as a major success in your life?*
7. *Why did you leave your last job?*
8. *Where do you see yourself in 5 years?*

Para realizar la grabación de la entrevista virtual puede programar un encuentro sincrónico a través de Skype, Zoom, Meet o Teams con su compañero de trabajo y realizar la grabación de la entrevista para luego guardarla.

Lineamientos generales para la entrega de la evidencia:

- **Producto a entregar:** video de la entrevista virtual de trabajo. Una vez finalizada la grabación de la entrevista debe cargar el vídeo a YouTube o Vimeo, con su cuenta de correo personal o institucional; compruebe que no tenga restricciones de visualización para que pueda compartir el enlace de visualización.
- **Formato:** PDF con los datos básicos del entrevistado, entrevistador y la URL del vídeo. Tenga en cuenta que se debe garantizar la visualización del video.
- **Extensión:** de 3 a 5 minutos.
- Para hacer el envío de la evidencia debe dirigirse al área de la actividad correspondiente y acceder al espacio: V**ideo entrevista virtual. GA2-240202501-AA1-EV02.**

- **Evidencia GA2-240202501-AA1-EV03: crónica.**

Seleccione un personaje icónico y con trayectoria importante en el mundo de acuerdo a su perfil profesional (Ej. En moda un personaje icónico es Agatha Ruiz) y busque en internet información acerca de él (ella), a partir del personaje seleccionado realice una crónica escrita con los hechos más importantes y las situaciones más relevantes en la trayectoria profesional del personaje.

Lineamientos generales para la entrega de la evidencia:

- **Producto a entregar:** crónica escrita.
- **Extensión:** el documento escrito será máximo de 1 página y una extensión entre 200 y 500 palabras, con tipo de letra Arial, tamaño 12 e interlineado 1,5. Es ideal tener cuidado con las estructuras gramaticales utilizadas deben ser coherente a las temáticas estudiadas en el material de estudio.
- **Formato:** documento escrito en Word y se deberá exportar a PDF.
- Para hacer el envío de la evidencia remítase al área de la actividad correspondiente y acceda al espacio: C**rónica. GA2-240202501-AA1-EV03.**

3.4.2 Actividad de aprendizaje GA2-240202501-AA2: expresar opiniones sobre situaciones cotidianas y laborales actuales, pasadas y futuras en contextos sociales.

Dentro de los procesos de aprendizaje de lengua, el exteriorizar lo aprendido es uno de los más valiosos aportes. Dentro de este contexto se abordarán temas como el vecindario y las actividades que se están realizando.

Es importante el seguir las indicaciones del instructor en cuanto a la mejor manera de apropiar el material del curso y la correlación que tiene este con el programa de formación.

Duración: 48 horas.

Material de formación: para el desarrollo adecuado de las evidencias el aprendiz debe apropiar los contenidos del componente formativo **“LEVEL 3 - MCER A2.1”.**

Evidencias:

A continuación, se describen las acciones y las correspondientes evidencias que conforman la actividad de aprendizaje:

- **Evidencia GA2-240202501-AA2-EV01: cuestionario.**

Presenta un cuestionario de acuerdo con el contenido formativo del tercer nivel “LEVEL 3 - MCE A2.1”. El cuestionario consta de quince preguntas (15) y un tiempo aproximado de 30 minutos.

Lineamientos generales para la entrega de la evidencia:

- **Producto a entregar:** cuestionario con preguntas de acuerdo a la información brindada en el tercer nivel.
- Para responder el cuestionario (evaluación en línea), remítase al área de la actividad correspondiente y acceda al espacio: **Cuestionario. GA2-240202501-AA2-EV01.**

- **Evidencia GA2-240202501-AA2-EV02: presentación.**

De acuerdo con la temática estudiada en el tercer nivel realiza una presentación en inglés y en formato vídeo de un lugar turístico de su región por medio de un corto video activando su cámara web con el fin de identificar de forma oral en el cual incluye la descripción y razón de popularidad del municipio o ciudad.

Para la elaboración del vídeo con la cámara web, tenga en cuenta el desarrollo de un guion o estructura a través de una herramienta como PowerPoint, Emaze, Prezi en la que se dispongan algunas diapositivas con imágenes y textos para dar cuenta de la presentación del lugar.

Diapositivas: durante el desarrollo de la presentación oral es ideal que presente entre 3 y 5 diapositivas, donde se incluyan los siguientes elementos:

- Diapositiva de portada (datos básicos del aprendiz, nombre del curso, instructor y nombre de la actividad).
- Dónde se encuentra.
- ¿Cómo es? (Use adjetivos para describirlo como colores, comparativos…)
- ¿Qué elementos hay en ese lugar? (Ejemplo: montes, ríos, casas…)
- Describa algún elemento importante de ese lugar de forma detallada.
- ¿Qué sensaciones transmite ese lugar? (alegría, paz, felicidad, estrés, agobio, tristeza, miedo misterio, terror…)
- Opinión personal. ¿Por qué has elegido ese lugar?

Estructura del vídeo con cámara web: cuando grabe el video, tenga en cuenta que las fotos, imágenes y textos que use en las diapositivas deben ser un apoyo visual para denotar buena pronunciación y su aprendizaje inicial sobre las temáticas del tercer nivel.

Para la realización de la emisión del vídeo deberá encender su cámara web, mostrar la pantalla con las diapositivas creadas. La recomendación es utilizar alguna herramienta digital que permite grabar el vídeo y pantalla como; Screencast-o-Matic, Loom, Camtasia, recordscreen.io, scrnrcrd.com e incluso existen aplicaciones como X Recorder para que pueda realizarlo desde su teléfono móvil. Lo importante es mostrar las diapositivas, su cámara web e ir realizando su presentación de forma oral.

Una vez finalizado el vídeo debe cargarlo a YouTube o Vimeo, con su cuenta de correo personal o institucional; compruebe que no tenga restricciones de visualización para que pueda compartir el enlace de visualización.

Lineamientos generales para la entrega de la evidencia:

- **Producto a entregar:** documento con los datos del aprendiz y enlace del video.
- **Formato:** PDF con la URL del vídeo.
- **Extensión:** de 2 a 5 minutos.
- Para hacer el envío de la evidencia remítase al área de la actividad correspondiente y acceda al espacio: **Presentación. GA2-240202501-AA2-EV02.**

- **Evidencia GA2-240202501-AA2-EV03: documento escrito.**

Elabora un documento escrito a modo de correo electrónico donde se realiza la postulación a un puesto de trabajo incluyendo datos relevantes de la hoja de vida y haciendo uso efectivo de la Netiqueta.

La estructura general del documento escrito debe ser la siguiente:

Título (asunto): debe enmarcar la idea general del mensaje y describir brevemente la temática de este. No limitarse a una sola palabra.
Saludo: redacte un saludo que exprese un grado de cordialidad el cual debe ser coherente y mantenerse a lo largo del mensaje.
Cuerpo del mensaje: incluya la palabra "solicitud" y el cargo al cual se postula. El objetivo principal de escribir el correo electrónico para la solicitud de empleo es que la entidad que puede contratarlo conozca algunas de las posesiones imperativas que deben formar parte de su saber profesional:

- El propósito de la solicitud de empleo.
- Qué trabajo está solicitando.
- Habilidades para desempeñar el trabajo al cual se postula.
- Qué tiene que ofrecer a la empresa.
- Datos básicos para que el destinatario pueda ponerse en contacto.

Despedida: cierre el mensaje en el que se reafirme el grado de formalidad utilizado en el saludo y en el cuerpo de la carta.
Dato remitente: ubique los datos conformados como nombres, lugar de origen, profesión.

- La **extensión** del mensaje escrito en un documento será de 1 a 3 páginas, con tipo de letra Times New Roman, tamaño 12 e interlineado 1,5.

Lineamientos generales para la entrega de la evidencia:

- **Extensión:** el mensaje de correo escrito a modo correo electrónico será de máximo de 1 página a 3 páginas y una extensión de entre 200 y 400 palabras, con tipo de letra Arial, tamaño 12 e interlineado 1,5. Es ideal tener cuidado con redacción y las estructuras gramaticales utilizadas deben ser coherentes con las temáticas estudiadas en el material de estudio.
- **Formato:** documento escrito en Word y se deberá exportar a PDF.
- Para hacer el envío de la evidencia remítase al área de la actividad correspondiente y acceda al espacio: **Documento escrito. GA2-240202501-AA2-EV03.**

4. ACTIVIDADES DE EVALUACIÓN

Evidencias de Aprendizaje	Criterios de Evaluación	Técnicas e Instrumentos de Evaluación
Evidencia de conocimiento: Foro temático: Lenguaje Unificado de Modelado (UML). **GA2-220501093-AA1-EV01.**	Interpreta el informe de requisitos para modelar las funciones del software.	**Lista de chequeo:** IE-GA2-220501093-AA1-EV01.

Comentado [U18]: Lo que se evalúa en los instrumentos técnicos, en ninguna parte se pide en las evidencias. Por ello esto debe decir en las evidencias, porque sino se evalúa algo que no se pide. Acomodar todas las evidencias técnicas con lo que se solicita en cada IE, más los comentarios que dejo en cada una.

Comentado [19]: *Respuesta a Usuario (17/11/2021, 22:07): "..."*
Se agrega lo que se evalua a las evidencias...

Comentado [U20]: Este criterio no se incluyó en ningún IE.

Comentado [21]: *Respuesta a Usuario (17/11/2021, 22:13): "..."*
Se incluye el criterio en los instrumentos que hacia falta.

Evidencia de desempeño: Elaboración de diagramas y plantillas para casos de uso del proyecto. **GA2-220501093-AA1-EV02.**	Elabora diagramas de casos de uso de acuerdo con los estándares actuales mediante Lenguaje Unificado de Modelado (UML). Genera plantillas extendidas de casos de uso expresando la intención de las acciones a desarrollar. Realiza diagramas de actividades exponiendo detalles de los casos de uso. Documenta las actividades de análisis a través de un informe.	**Lista de chequeo:** IE-GA2-220501093-AA1-EV02.
Evidencia de desempeño: Elaboración de historias de usuario del proyecto. **GA2-220501093-AA1-EV03.**		**Lista de chequeo:** IE-GA2-220501093-AA1-EV03.
Evidencias de producto: Diagramas y documentación de actividades del proyecto. **GA2-220501093-AA1-EV04.**		**Lista de chequeo:** IE-GA2-220501093-AA1-EV04.
Evidencia de desempeño: Elaboración de los diagramas del modelo de dominio del proyecto. **GA2-220501093-AA2-EV01.**	Representa el negocio en términos de clases abstractas generando un modelo de dominio consistente.	**Lista de chequeo**: IE-GA2-220501093-AA2-EV01.
Evidencia de conocimiento: Mapa conceptual sobre validación de documentos. **GA2-220501093-AA3-EV01.**	Elabora listas de chequeo para validación de la documentación de análisis. Evalúa el informe de análisis teniendo en cuenta la calidad de los artefactos generados y la respuesta al cumplimiento de requisitos. Realiza mejoras a la documentación de análisis de acuerdo con los resultados de la evaluación.	**Lista de chequeo:** IE-GA2-220501093-AA3-EV01.
Evidencia de desempeño: Elaboración del informe de análisis con listas de chequeo para la validación de artefactos. **GA2-220501093-AA3-EV02.**		**Lista de chequeo:** IE-GA2-220501093-AA3-EV02.

	Realiza prototipo inicial del software de acuerdo con los casos de uso identificados.	
Evidencia de conocimiento: Foro temático: Licenciamiento de software. **GA2-220501094-AA1-EV01.**	Dimensiona las licencias de software de acuerdo con las características de la solución y las necesidades de la empresa. Elabora referentes técnicos de acuerdo con normatividad legal vigente.	**Lista de chequeo:** IE-GA2-220501094-AA1-EV01.
Evidencia de desempeño: Diseño de las fichas técnicas para la recolección de la información. **GA2-220501094-AA1-EV02.**	Diseña formatos de fichas técnicas de acuerdo con estándares	**Lista de chequeo:** IE-GA2-220501094-AA1-EV02.
Evidencia de producto: Especificación de los referentes técnicos del hardware - software y estimación de las condiciones económicas**.** **GA2-220501094-AA1-EV03.**	Dimensiona las licencias de software de acuerdo con las características de la solución y las necesidades de la empresa Estima condiciones técnicas y económicas de acuerdo con las características de la solución de software y las necesidades de la empresa.	**Lista de chequeo:** IE-GA2-220501094-AA1-EV03.
Evidencia de producto: Ficha técnica de los productos requeridos, los términos de referencia para el contrato y lista de chequeo para evaluación de proveedores.	Estima costos de software y hardware de acuerdo con las características del negocio. Documenta la propuesta técnica según las especificaciones de la solución de software	**Lista de chequeo:** IE-GA2-220501094-AA2-EV01.

Comentado [U22]: No está en ningún Instrumento este criterio. Además, acá en la guía y en la planeación mencionan que los criterios son los mismos para las tres evidencias, pero en cada IE los mencionan de manera diferenciada.

Comentado [MFMC23R22]: Este criterio ya se agregó al EV01
Desde la parte técnica cada instrumento presenta un criterio de evaluación diferente,
Ej NO podemos estimar condiciones económicas en el foro temático de lic de software ni tampoco en diseño de fichas técnicas para la recolección de información, de igual manera para las otras.

GA2-220501094-AA2-EV01		
Evidencia de desempeño: Diseño de tablas comparativas sobre presupuestos de hardware y software. **GA2-220501094-AA3-EV01.**	Estima costos de software y hardware de acuerdo con las características del negocio.	**Lista de chequeo:** IE-GA2-220501094-AA3-EV01.
Evidencia de producto: Propuesta técnica y económica para la implementación del proyecto. **GA2-220501094-AA3-EV02.**	Documenta la propuesta técnica según las especificaciones de la solución de software.	**Lista de chequeo:** IE-GA2-220501094-AA3-EV02.
Evidencia de conocimiento: Taller de especificación del modelo conceptual y estructurar la propuesta técnica del proyecto de software. **GA2-220501094-AA4-EV01.**	Estima costos de software y hardware de acuerdo con las características del negocio. Documenta la propuesta técnica según las especificaciones de la solución de software.	**Lista de chequeo:** IE-GA2-220501094-AA4-EV01.
Evidencia de producto: Propuestas técnica y económica con ajustes de acuerdo a la negociación tecnológica. **GA2-220501094-AA4-EV02.**	Estima costos de software y hardware de acuerdo con las características del negocio. Documenta la propuesta técnica según las especificaciones de la solución de software.	**Lista de chequeo:** IE-GA2-220501094-AA4-EV02.
Evidencia de conocimiento: Cuestionario procedimientos aritméticos.	Aplica procedimientos aritméticos y algebraicos según el problema planteado.	**Cuestionario:** IE-GA2-240201528-AA1-EV01.

Comentado [U24]: No corresponden con los criterios que se mencionan en el Instrumento.

Comentado [MFMC25R24]: Si corresponde, por favor revisar

Comentado [U26]: A nivel general los instrumentos técnicos carecen de profundización, por lo general siempre se dejan entre 2 y 3 ítems, que son muy similares. Se deben ampliar y ser más específicos a lo que se solicite en cada evidencia, cuando estas se ajusten a lo requerido en cada una.

Comentado [MFMC27R26]: ¡¡Ajustado, mil gracias1!!

GA2-240201528-AA1-EV01.	Plantea ecuaciones o sistemas de ecuaciones de acuerdo con la relación entre las variables. Presenta la relación entre dos cantidades o variables según los fundamentos matemáticos.	
Evidencia de desempeño: Planteamiento de ecuación. **GA2-240201528-AA2-EV01.**	Resuelve ecuaciones o sistemas de ecuaciones de acuerdo con principios matemáticos. Presenta solución a problemas mediante figuras geométricas. Define el problema a resolver de acuerdo con las necesidades de su entorno.	**Lista de chequeo:** IE-GA2-240201528-AA2-EV01.
Evidencia de producto: Video sustentación. **GA2-240201528-AA3-EV01.**	Representa conjunto de datos de acuerdo con la variable estadística. Realiza conversiones según las equivalencias entre sistemas de medida. Define el problema a resolver de acuerdo con las necesidades de su entorno.	**Lista de chequeo:** IE-GA2-240201528-AA3-EV01.
Evidencia de desempeño: Algoritmo para el cálculo de áreas y volúmenes. **GA2-240201528-AA4-EV01.**	Selecciona las herramientas computacionales para la verificación de los resultados de acuerdo con los requerimientos matemáticos. Elabora una propuesta de solución alternativa a partir de los	**Lista de chequeo:** IE-GA2-240201528-AA4-EV01.

	procedimientos matemáticos inicialmente planteados. Calcula perímetros, áreas y volúmenes de acuerdo con los elementos de la figura geométrica.	
Evidencia de conocimiento: Cuestionario. **GA2-240202501-AA1-EV01.**	Comprende información concreta relativa a temas cotidianos y laborales en textos orales y escritos.	**Cuestionario:** IE-GA2-240202501-AA1-EV01.
Evidencia de desempeño: Video entrevista virtual. **GA2-240202501-AA1-EV02.**		**Lista de chequeo:** IE-GA2-240202501-AA1-EV02.
Evidencia de producto: Crónica. **GA2-240202501-AA1-EV03.**		**Lista de chequeo:** IE-GA2-240202501-AA1-EV03.
Evidencia de conocimiento: Cuestionario. **GA2-240202501-AA2-EV01.**	Describe de manera sencilla y clara asuntos, acciones, experiencias, sentimientos, planes relacionados con temas de su interés y cotidianidad, siguiendo una secuencia lineal de elementos.	**Cuestionario:** IE-GA2-240202501-AA2-EV01.
Evidencia de desempeño: Presentación**.** **GA2-240202501-AA2-EV02.**		**Lista de chequeo:** IE-GA2-240202501-AA2-EV02.
Evidencia de producto: documento escrito. **GA2-240202501-AA2-EV03.**		**Lista de chequeo:** IE-GA2-240202501-AA2-EV03.

5. GLOSARIO DE TÉRMINOS

Algoritmo: un algoritmo es una secuencia lógica y finita de pasos que permite solucionar un problema o cumplir con un objetivo.

Los algoritmos deben ser precisos e indicar el orden lógico de realización de cada uno de los pasos, debe ser definido y esto quiere decir que si se ejecuta un algoritmo varias veces se debe obtener siempre el mismo resultado, también debe ser finito o sea debe iniciar con una acción y terminar con un resultado o solución de un problema

Aritmética: la aritmética se encarga de realizar con números y simbología en conjunto con las operaciones antes mencionadas, el desarrollo de propiedades y habilidades las cuales pueden ser usadas en la vida cotidiana y materias de estudio que impliquen a la matemática como base fundamental de aprendizaje (Adrián, 2020).

Metacrilato: este termoplástico rígido se caracteriza por su gran transparencia, ligereza y resistencia a la intemperie. Visualmente es parecido al vidrio, aunque es un material mucho más ligero y aguanta unas diez veces más los golpes.

A su vez, este **plástico rígido** se trabaja con facilidad. Es decir, puede perforarse, lijarse, partirse o cortarse utilizando herramientas mecánicas. Asimismo, es un plástico fácil de moldear con la aplicación de calor.

Proporción: es la relación de igualdad entre dos razones matemáticas, o la comparación entre dos razones matemáticas.

Razón: es una razón binaria entre magnitudes, se expresa como a es a b, o a: b, numéricamente una razón se puede expresar como una fracción o un decimal

6. REFERENTES BIBLIOGRÁFICOS

Guerrero, J., Zúñiga, K., Certuche, C. & Pardo, C. (2019). *What is there about DevOps? XIV Jornadas Iberoamericanas de ingeniería de Software e Ingeniería del Conocimiento* (JIISIC'2019).

Lwakatare, L. E., Kuvaja, P. & Oivo, M. (2015). *Dimensions of DevOps. Lecture Notes in Business Information Processing*. 212–217. https://doi.org/10.1007/978-3-319-18612-2_19

Pressman, R. (2010). *Ingeniería del software, un enfoque práctico* (Séptima edición). Bogotá: McGraw-Hill.

7. CONTROL DEL DOCUMENTO

	Nombre	Cargo	Dependencia	Fecha

Comentado [U28]: No está el Glosario técnico ni tampoco de inglés.

Comentado [U29]: No están las referencias de matemáticas, ni tampoco los de inglés.

Comentado [DERM30]: La razón por la que no fueron incluidos los autores es que las normas APA establece que solo deben incluirse en este apartado los autores citados en el documento. El nombre mas adecuado para este apartado sería bibliografía que no es lo mismo que referencias bibliográficas, sin embargo, se incluyen siguiendo su solicitud.

Comentado [U31]: No están los autores de matemáticas, ni tampoco los de inglés.

Autor (es)	Mario Fernando Meneses Calvache	Experto Temático	Regional Cauca - Centro de Teleinformática y Producción Industrial	Noviembre 2021
	Carlos Hernán Muñoz Carvajal	Experto Temático	Regional Cauca - Centro de Teleinformática y Producción Industrial	Noviembre 2021
	Deivis Eduard Ramírez Martínez	Diseñador Instruccional	Regional Distrito Capital - Centro para la Industria de la Comunicación Gráfica	Noviembre 2021
	Andrés Felipe Velandia Espitia	Revisor Metodológico y Pedagógico	Regional Distrito Capital - Centro de Diseño y Metrología	Noviembre 2021
	Rafael Neftalí Lizcano Reyes	Asesor Pedagógico	Regional Santander - Centro Industrial del Diseño y la Manufactura	Noviembre 2021

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

	Nombre	**Cargo**	**Dependencia**	**Fecha**	**Razón del Cambio**
Autor (es)					

FORMATO PARA EL DESARROLLO DE COMPONENTE FORMATIVO

PROGRAMA DE FORMACIÓN	**ANÁLISIS Y DESARROLLO DE SOFTWARE**

COMPETENCIA	220501094- Elaboración de la propuesta técnica del software	**RESULTADOS DE APRENDIZAJE**	**220501094-02** Elaborar propuesta técnica del software de acuerdo con las especificaciones técnicas definidas. **220501094-03** Validar las condiciones de la propuesta técnica del software de acuerdo con los intereses de las partes.

NÚMERO DEL COMPONENTE FORMATIVO	**11**
NOMBRE DEL COMPONENTE FORMATIVO	**Propuesta técnica**
BREVE DESCRIPCIÓN	La Propuesta técnica es el documento que debe contar con las respectivas normas, cuyo fin es proporcionar la información detallada sobre una propuesta de solución de software para resolver un problema de negocio.
PALABRAS CLAVE	**Artefactos, Trazabilidad, requisitos de software, informe de análisis, instrumentos de verificación, Stakeholders.**

ÁREA OCUPACIONAL	**2 - CIENCIAS NATURALES, APLICADAS Y RELACIONADAS**
IDIOMA	**Español**

A. TABLA DE CONTENIDOS:

Introducción
1.0 presupuesto
1.1 Conceptos
1.2 teoría
1.3 Costos

2.0 Propuesta Técnica
2.1 estructura
2.2 Términos de Referencia
2.3 Elaboración
2.4 Presentación

3.0 Validación propuesta técnica
3.1 Contratación
3.2 Fundamentos
3.3 Tipos

3.4 Normatividad
3.5 Acuerdos

B. DESARROLLO DE CONTENIDOS:

Introducción

Realizar una correcta propuesta de softwares es fundamental al momento de ofrecer nuestros servicios, ya que de este documento depende las probabilidades de éxito del proyecto, aquí debemos convencer al cliente que la solución que se ofrece es la más adecuada para las necesidades de su empresa.

1.0 presupuesto

En los primeros acercamientos con el posible cliente, hay que indagar sobre la capacidad económica de la empresa y con qué presupuesto disponen para el proyecto, con el fin de amoldarnos a él o presentar una propuesta adecuada a sus recursos, para realizar esta consulta podemos ingresar a la página https://www.rues.org.co/ y adquirir un certificado de la empresa con la cual se trabajará, esto evita tratar con intermediarios y tener un concepto del tipo y tamaño de la empresa.

Figura 1 Presupuesto



Fuente(https://www.freepik.es/fotos-vectores-gratis/facturacion)

1.1 Conceptos

Contrato: es un documento legal en el que dos o mas partes acuerdan o pactan cumplir una serie de condiciones teniendo como contraprestación un fin económico.
Adición de contrato: como su nombre lo indica es una adición que se puede hacer en tiempo y valor, sí es en valor SOLO se puede adicionar hasta el 50% del valor inicial del contrato y nunca pude exceder este tope.
Activos: Son el conjunto de todos los bienes y derechos con valor monetario que son propiedades de una empresa, institución o individuo
Pasivo: se refiere a aquel conjunto de obligaciones pendientes, tales como deudas no satisfechas
CDP: Certificado de disponibilidad presupuestal, en este documento se garantiza o respalda la existencia de recursos para atender un gasto especifico.
Monetización: Es el proceso por medio del cual una determinada accione es convertida en dinero, por lo general este término se lo utiliza en redes sociales y plataformas con streaming
Gasto: consumo de un bien o servicio a cambio de una contraprestación

Licitación: es un procedimiento formal por cual un organismo público demanda obras, servicios, bienes, entre otros, al sector privado.
Freelance: Trabajador independiente

1.2 Teoría

En cuanto a presupuesto es un plan enfocado a cumplir una meta propuesta “El software o entregables”, esta se representa en valores y términos financieros que se deben cumplir en un cierto tiempo y bajo unas condiciones pactadas, para este fin debemos tener en cuenta los siguientes aspectos:

- **Establecer una relación cordial con el cliente:** esto permite generar una conexión más cercana con la cual podemos darnos a conocer y descifrar lo que busca el cliente.
- **Identificar a los interesados o stakeholders:** la propuesta debe realizarse teniendo en cuenta a la persona que se entrevista ya que cada funcionario tiene expectativas y necesidades diferentes de acuerdo a su cargo y rol dentro de la empresa.
- **Identificar necesidades:** para esto debemos solicitar información sobre como interactúa la compañía, su modelo de negocios, procesos y procedimientos, comprendiendo las necesidades del cliente y sugiriendo nuevas soluciones a problemas que aún no detectan.
- **Conocer el presupuesto:** Debes solicitar conocer el presupuesto que el cliente tiene destinado para ejecutar este proyecto, en algunos casos debes amoldarte a él y cubrir las necesidades de mayor prioridad y en otros casos puedes entrar en el campo de la negociación ofreciendo más servicios o módulos extras, generado así más recursos.
-

Figura 2 Contrato



Fuente (https://www.freepik.es/search?dates=any&format=search&page=1&query=contrato&sort=popular)

1.3 Costos

Es necesario, entonces, recordar que el costo, también llamado coste, es el valor económico que se cobra por realizar un producto, prestar un servicio o vender un bien, aquí debemos tener en cuenta la adquisición de insumos y los gastos que se derivan de la producción como el pago de la mano de obra, administrativos etc.

En este caso en particular la estimación del costo de desarrollo de software es un factor suma mente importante en el análisis del proyecto informático y propuesta técnica, hay que tener en cuenta el esfuerzo, costo y tiempo de las actividades que se realizará en el proyecto.

Aspectos a tener en cuenta al momento de realizar la planeación de costos:

El tamaño del software que se realizará
El tiempo estimado de producción
¿Se realizará el trabajo por módulos?
¿Se necesita contratar personal de apoyo para la producción y/o soporte?
¿Como se dará soporte técnico? “Por cuanto Tiempo, presencial o remoto”
¿Se necesita adquirir equipos tecnológicos?

2.0 Propuesta Técnica

La propuesta técnica es el documento que se entrega al posible cliente, en la cual se detalla la estrategia a desarrollar para darle solución a un problema existente, el contenido o formato de una propuesta puede variar según la empresa que lo solicita, si es privada, gubernamental, fundación o ONG, en estos casos se debe adaptar la propuesta a los formatos requeridos de las empresas. Una propuesta debe ser clara, concreta, explicita y fácil de entender.

2.1 Estructura

Cuando se tiene el insumo principal que es toda la información requerida y analizada se empieza a estructurar la propuesta que debe tener el siguiente esquema.

- Saludos y Presentación
- Alcance de la propuesta
- Opciones de las propuestas de trabajo (Dos o tres)
- Cotización por cada propuesta presentada
- Condiciones (Forma de pago)
- Conclusión y despedida

2.2 Términos de Referencia TdR

Es el documento que realiza una empresa en el que plasma las oportunidades de mejora, necesidades o problemas y se las presenta a una empresa o al experto informático para darles solución. Los TdR sirven para que las empresas tengan en claro los servicios que requieren contratar y la contraparte sobre en que condiciones ofrecerán dichos servicios y poder elaborar una propuesta técnica adecuada.

Los **TdR** incluyen el servicio que necesita la empresa de forma clara y concisa, los objetivos que se espera lograr con la contratación, en algunos casos se describe las especificaciones técnicas de la solución informática, los requisitos del contratista, el tiempo de ejecución del contrato y las condiciones de pago; la estructura de este documento tiene una Justificación, alcances del servicio, perfil de la empresa a contratar y condiciones contractuales.

Cuando una empresa es nueva y no maneja los **TdR**, lo que el experto informático debe hacer es acordar una reunión con las partes interesadas o también llamadas stakeholders, y hacerles preguntas sobre el funcionamiento de la empresa, los procesos que manejan, servicios que prestan y en qué sector económico están, también se debe dialogar por el presupuesto proyectado para la solución de software y la matriz DOFA de la empresa para tener claras sus necesidades y ofrécele los servicios adecuados en una propuesta técnica.

Figura 3 Stakeholders



Fuete(https://www.freepik.es/vector-gratis/empleados-empresa-compartiendo-pensamientos-e-ideas_8270975.htm#page=1&query=proposal&position=2)

2.3 Elaboración: Es necesario, entonces, recordar que debemos tener una papelería adecuada que represente a nuestro negocio, el formato de documento debe contar con un diseño sobrio pero atractivo, el documento debe tener un encabezado en el cual debe ir el nombre de la empresa y el NIT o tú nombre si eres un freelance, este debe ir acompañado del logo. En el pie de pagina debe ir la dirección de la empresa y datos de contacto como números telefónicos correos, WhatsApp, Facebook, Instagram etc. Se debe tener en cuenta como insumos principales los **TdR** o la entrevista al momento de elaborar la propuesta.

En el cuerpo del documento utilizamos la siguiente estructura, hay que recordar que se debe utilizar las hojas requeridas no hay que extenderse demasiado a no ser que se necesario

Fecha según formato utilizado Ciudad DD/MM/AÑO

Estimado señor
"Nombre del cliente o empresa"

2.3.1 Presentación y saludos: Este párrafo es muy corto hay que ser formales concisos y cordiales. Ejemplo

“Gracias por reunirse conmigo el anterior día, fue agradable compartir tiempo para profundizar y tener una conversación objetiva y concreta sobre las necesidades tecnológicas de su reconocida empresa, en las páginas siguientes encontrará (1,2 o 3) propuestas que pueden satisfacer las necesidades de su empresa, cada opción representa la solución a una necesidad con un costo y servicio diferente”

2.3.2 Alcance de la Propuesta: Aquí se define cuáles son los objetivos reales que la implementación de software tendrá en la empresa y los límites de la propuesta, así se puedes asegurarte que las metas y objetivos se logren sin contratiempos.

2.3.3 Valoración de la situación

Desde el punto de vista del cliente según la entrevista que anteriormente realizaron o las TrD, describe la situación deseada.

Desde tu punto de vista como crees que contribuirás con la implementación de software a la transformación y mejorar de la productividad de la empresa, puede ser un párrafo o también se puede listar con viñetas

2.3.4 Opciones de Propuesta Se pude presentar de una a tres propuestas dentro de este documento, brindándole al cliente más opciones para elegir, teniendo en cuenta el tamaño del proyecto lo podemos hacer por faces o de manera sencilla.

Para esto utilizamos la siguiente estructura:

- **Opción 1:** Titulo breve y descriptivo
  Un párrafo que describa lo que se hará

Tabla 1 Ejemplo Propuesta

Fase	Funciones o Actividades	Entregables
FASE 1	Construcción del plan de Diagnóstico	Diagnóstico del estado Inicial de la organización
	Cronograma de trabajo	Cronograma de cumplimiento con fechas establecidas
FASE 2	Implementación de la Solución de Software	Solución de Software implementada y ajustada
	Pruebas de Seguridad del Software	
	Ajustes de la Solución de Software	
FASE 3	Capacitación al personal	Cronograma de Capacitación al personal
	Elaboración de Manual de Usuario	Entrega del Manual de Usuario
	Soporte Técnico	El Tiempo que crea conveniente de acuerdo con la propuesta y presupuesto indicado

Autor

Esto mismo se debe hacer con cada opción de propuesta presentada una, dos o tres según el caso.

2.3.5 Cotización: Se debe presentar una cotización por cada propuesta Ejemplo:
Cotización de la propuesta 1 $ 5.000.000
Cotización de la propuesta 2 $ 7.500.000

Cotización de la propuesta 3 $ 17.000.000

Hay que tener en cuenta que la cotización es un precio fijo y no estimaciones por lo cual se ha enviado más de una propuesta para que el cliente tenga más alternativas que se adecuen a su presupuesto.

2.3.6 Términos y condiciones
Aquí se detalla cómo deben ser los pagos, adelanto, fechas, cantidades o porcentaje de cada cuota, esto conforme a las fases o entregas, lo ideal es tener una cuenta bancaria activa ya que el medio de pago más utilizado es la transferencia o consignación bancaria.

2.4 Presentación
La propuesta debe ser entregada o socializada de acuerdo con los términos acordados con el cliente o según los requerimientos de la licitación

Si el cliente solicita que la propuesta sea socializada, hay que contar con recursos o medios audios visuales como "Video beam, apuntadores y presentaciones digitales etc. "para realizar un optima presentación.

3.0 Validación propuesta técnica

3.1 Contratación
La contratación se realiza después que el cliente haya aceptado una de las propuestas técnicas presentadas, o te hayan adjudicado una licitación; para legalizar este trámite es necesario cerrar el acuerdo por medio de un contrato escrito, en el cual estén estipulado todos los términos y condiciones que se acordaron en la propuesta técnica o en el pliego de condiciones.

3.2 Fundamentos
El fin de los principios o fundamentos en contratación, es garantizarles a los ciudadanos un proceso publico transparente y con igualdad, al que pueden acceder todas las personas o empresas que cumplan con los requisitos establecidos en un pliego de condiciones publicado por las entidades estatales, que pueden ser SENA, DIAN, Fuerzas militares, ICBF, DANE, Universidades, alcaldías, gobernaciones etc.

3.3 Tipos En Colombia existen dos tipos de contratación la privada y la pública o estatal.
3.3.1 Contratación Privada: es el acuerdo al que llegan dos partes que pueden ser personas naturales o jurídicas para que se haga un producto, una labor o se deje de hacer algo, en este documento se pactan términos, acuerdos, multas y la forma de remuneración económica.

3.3.2 Contratación Estatal: De acuerdo con el artículo 3 de la Ley 80 de 1993, la contratación estatal tiene como fin, "la continua y eficiente prestación de los servicios públicos y la efectividad de los derechos e intereses de los administrados que colaboran con ellas en la consecución de dichos fines", Es decir que el estado necesita de particulares para para el cumplimiento de sus fines, esto se realiza por medio del concurso de proveedores o más conocida como licitación.

3.3.3 La Licitación: Es un método por el cual una entidad del estado otorga un servicio u obra a una empresa o persona que ofrece las mejores condiciones.

¿Qué requisitos debe cumplir una empresa o persona para participar de una licitación estatal?

- Contar con RUT Registro Único tributario "lo expide la DIAN"
- Tener la empresa inscrita en registro mercantil "Gestione en Cámara de Comercio"
- Tener registrada la empresa o trabajadores a una ARL "Administradora de riesgos laborales"
- Llevar la contabilidad de la empresa según la guía técnica 184 "Apoyo de un Contador Público"
- Los trabajadores de la empresa deben estar afiliados al sistema de seguridad social "Salud, Pensión y caja de compensación"

Existen dos tipos de licitación que son:

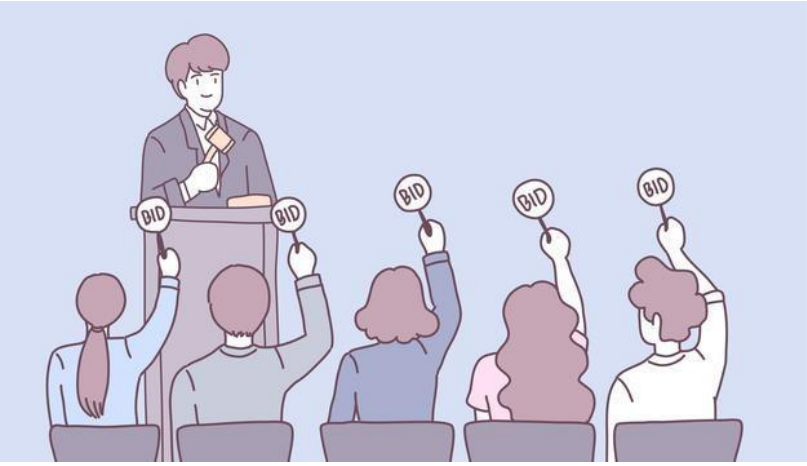
3.3.4 Licitación Privada: Solo pueden participar las empresas a las cuales se hayan enviado invitaciones.

3.3.5 licitación Pública: Es abierta a todos los participantes cualquier persona puede participar en el proceso.

El proceso de licitación pública tiene los siguientes pasos:

- La entidad estatal que tiene una necesidad de obras bienes o servicios elabora un plan de compras el cual debe ser aprobado.
- La entidad estatal elabora un pliego de condiciones "Borrador" que se publica en la página SECOP https://colombiacompra.gov.co/ aquí esta publicado 10 días hábiles, aunque también se publica por la página web de la entidad, los interesados deben estar pendiente de los diferentes procesos publicados.
- Las personas interesadas "también llamadas **Proponentes u Oferentes"** en el término de los 10 días hábiles pueden solicitar si es necesario aclaraciones del pliego de condiciones.
- Publicación del pliego de condiciones definitivo: con la resolución de apertura del proceso en el cual incluye el cronograma de presentación de propuestas.
- Audiencia de Asignación revisión y distribución de riesgos: es realizada por la entidad para que los futuros proponentes aclaren sus dudas y realicen observaciones sobre el proceso.
- Entrega de Propuestas: conforme al cronograma los proponentes entregan sus propuestas en el lugar, hora y en los términos señalados.
- Etapa de Evaluación: la entidad inicia la verificación de los requisitos mínimos y se otorgan los puntajes de las ofertas presentas.
- La Entidad presta el informe de la Evaluación: por un término de 5 días hábiles los oferentes pueden pronunciarse y presentar observaciones.
- La entidad Analiza las observaciones presentadas y cita para definir el proceso de audiencia publica de acuerdo al cronograma.
- Audiencia de Adjudicación: la entidad pública con antelación en el SECOP publicará el borrador del acto administrativo de adjudicación, si algún oferente cumple con todos los requisitos del pliego y con las condiciones mas favorables en cuanto a costos y calidad la entidad adjudica el proceso a un oferente, de lo contrario se declara el proceso desierto

Figura 3 Licitación



Fuente(https://www.freepik.es/vector-gratis/licitaciones-o-subastas-que-involucran-compra-articulo_11879371.htm#page=1&query=BIDDING&position=6)

3.4 Normatividad

La normatividad que rige la contratación en Colombia es:

- Ley 80 de 1993 - Modalidades de selección, publicidad, selección objetiva,
- Ley 1150 de 2007 - Contratación con recursos Públicos
- Ley 1712 de 2014 - Ley de Trasparencia
- Ley 1474 - Anticorrupción
- Ley 1150 de 2007 – Eficiencia y Transparencia
- Ley 4170 – Agencia de contratación Publica

3.5 Acuerdos

El acuerdo que rige al sistema de contratación en Colombia se llama acuerdo marco, Según la página de Colombia compra eficiente dice que:

El Acuerdo Marco es un instrumento de agregación de demanda. El Acuerdo Marco es un contrato entre Colombia Compra Eficiente, como representante de los compradores públicos y uno o más proveedores para adquirir bienes, obras o servicios que pueden ser estandarizados. El Acuerdo Marco contiene la identificación del bien o servicio, el precio máximo de adquisición, el plazo máximo de entrega, las garantías mínimas y las condiciones en las cuales los compradores pueden vincularse al Acuerdo.

ACUERDOS MARCO.(S.F). que es un acuerdo macro. Colombia compra eficiente. Recuperado de: https://colombiacompra.gov.co/ciudadanos/preguntas-frecuentes/acuerdos-marco [25/09/2021].

C. ACTIVIDADES DIDÁCTICAS (OPCIONALES SI SON SUGERIDAS)

DESCRIPCIÓN DE ACTIVIDAD DIDÁCTICA	
Nombre de la Actividad	Arrastrar y enlazar
Objetivo de la actividad	**Afianzar algunos de los conceptos más importantes asociados al proceso de la propuesta técnica**
Tipo de actividad sugerida	**Enlazar los conceptos que correspondan al significado**
Archivo de la actividad (Anexo donde se describe la actividad propuesta)	**Anexo documento en Word llamado Actividad didáctica 1**

D. MATERIAL COMPLEMENTARIO:

Tema	Referencia APA del Material	Tipo de material (Video, capítulo de	Enlace del Recurso o Archivo del documento o material

		libro, artículo, otro)	
Propuesta Técnica	Starfish Host. (2020, 19 mayo). *Cómo hacer una propuesta de Software - ¡Presenta una propuesta de software exitosa!* youtube. https://www.youtube.com/watch?v=6zI4ZcRS8yg	Video Youtube	https://www.youtube.com/watch?v=6zI4ZcRS8yg
Contratación Licitaciones	Ministerio de Comercio, Industria y Turismo. (2013, 7 junio). Contratación Publica. http://www.aplicaciones-mcit.gov.co/. http://www.aplicaciones-mcit.gov.co/cincopasos/c2.html	Articulo	http://www.aplicaciones-mcit.gov.co/cincopasos/c1.html

E. GLOSARIO:

TÉRMINO	SIGNIFICADO
Cotización	La **cotización** es el hecho de cotizar y significa establecer un precio, estimarlo o pagar una cuota. Determina el valor real de un bien, de un servicio
Freelance	Un FreeLancer es una persona que ofrece servicios profesionales o productos a diferentes clientes de manera independiente, es decir, por su propia cuenta.
Licitación	Es una fase del procedimiento de contratación del sector público para elegir a las empresas privadas. La licitación es la fase de publicidad, donde la administración pública se encarga de publicitar el contrato que va a ofertar
Stakeholders	Partes Interesadas, desempeñan un papel clave en la elicitación, especificación y validación de los requerimientos de software.

TdR	Termino de Referencia: documento que elabora una empresa que necesita contratar a otras empresas para que le proporcionen determinados servicios para mejorar sus sistemas de mercadeo, de producción, de organización, sus finanzas o bien su administración en general.

F. REFERENCIAS BIBLIOGRÁFICAS:

- Anton.agency. (2020, 17 julio). ¿Cómo presentar una propuesta de proyecto de software?

  https://anton.agency/. https://anton.agency/como-presentar-una-propuesta-proyecto-software/

- Colombia compra eficiente. (2020, 20 noviembre). ACUERDOS MARCO.

  https://colombiacompra.gov.co/ciudadanos/preguntas-frecuentes/acuerdos-marco

- deconceptos.com. (2021, 12 diciembre). Conceptos. deconceptos. https://deconceptos.com/?s=contrato
- Gestiopolis. (2019, 7 noviembre). Teoría de los presupuestos. https://www.gestiopolis.com/.

  https://www.gestiopolis.com/teoria-de-los-presupuestos

- Ministerio de Comercio, Industria y Turismo. (2013, 7 junio). Contratación Publica.

  http://www.aplicaciones-mcit.gov.co/. http://www.aplicaciones-mcit.gov.co/cincopasos/c2.html

- NormasApa. (2020, 2 mayo). Normas APA. Normas Apa. https://normasapa.in/

G. CONTROL DEL DOCUMENTO

	Nombre	**Cargo**	**Dependencia** *(Para el SENA indicar Regional y Centro de Formación)*	**Fecha**
Autor (es)	Mario Fernando Meneses Calvache	Experto Temático	Regional Cauca, Centro de teleinformática y producción industrial	Septiembre 2021

	Jonathan Guerrero Astaiza	Instructor	Regional Cauca, Centro de teleinformática y producción industrial	Septiembre 2021
	Zulema León	Instructor	Regional Cauca, Centro de teleinformática y producción industrial	Septiembre 2021
	Carlos Hernan Muñoz	Experto Temático	Regional Cauca, Centro de teleinformática y producción industrial	Septiembre 2021
	Rafael Neftalí Lizcano Reyes	Asesor pedagógico	Regional Santander. Centro Industrial del Diseño y la Manufactura.	Septiembre 2021

H. CONTROL DE CAMBIOS

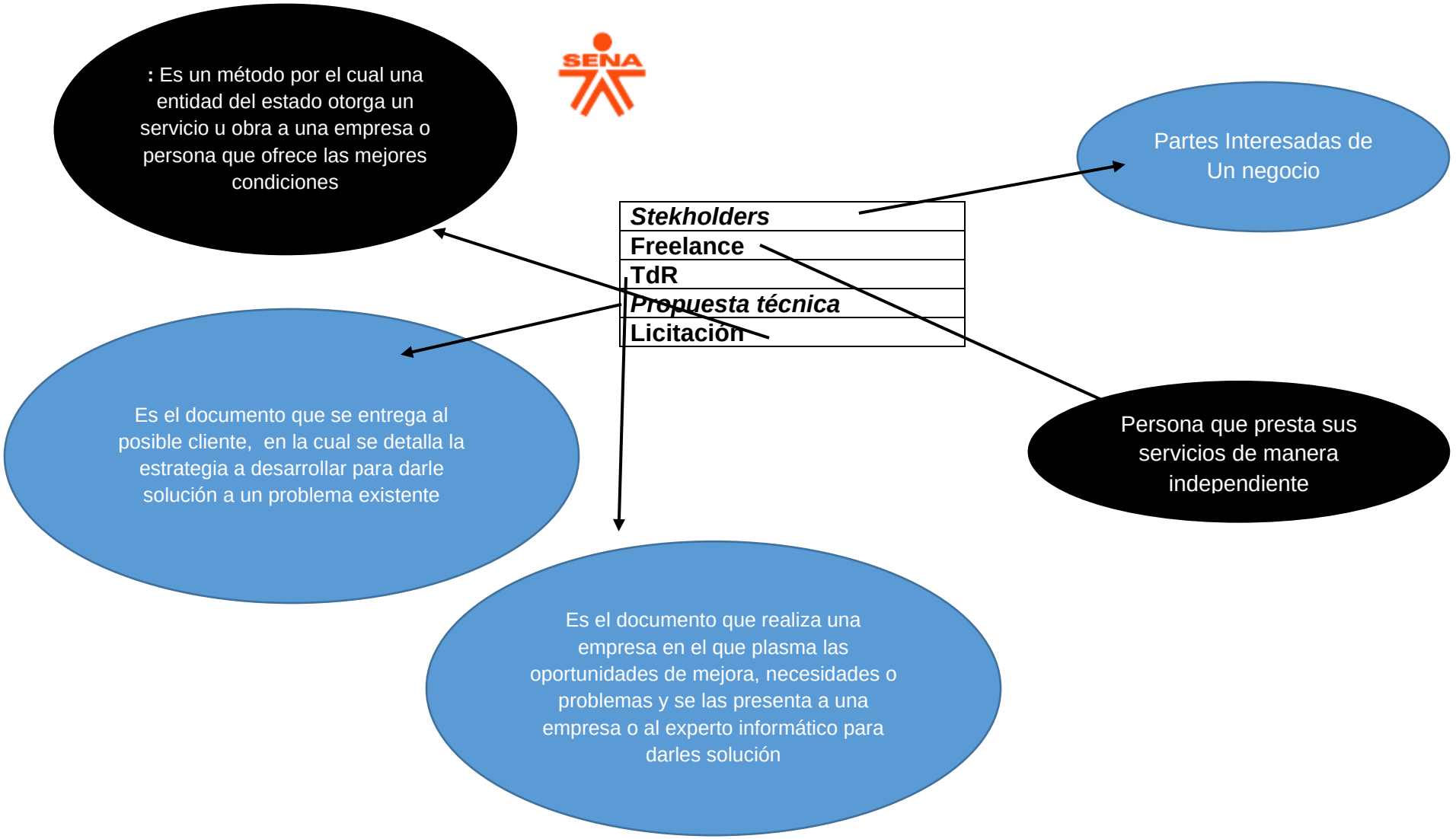
(Diligenciar únicamente si realiza ajustes a la Unidad Temática)

	Nombre	Cargo	Dependencia	Fecha	Razón del Cambio
Autor (es)					

DESCRIPCIÓN DE ACTIVIDAD DIDÁCTICA	
Nombre de la Actividad	Arrastrar y enlazar
Objetivo de la actividad	Afianzar algunos de los conceptos más importantes asociados al proceso de Propuesta técnica
Tipo de actividad sugerida	Enlazar los conceptos que correspondan al significado
Archivo de la actividad (Anexo donde se describe la actividad propuesta)	Anexo documento en Word llamado Actividad didáctica 1

Desarrollo de la actividad:

La actividad planteada consiste en enlazar la imagen con el significado correspondiente.


: Es un método por el cual una entidad del estado otorga un servicio u obra a una empresa o persona que ofrece las mejores condiciones
Partes Interesadas de Un negocio
Stekholders
Freelance
TdR
Propuesta técnica
Licitación
Es el documento que se entrega al posible cliente, en la cual se detalla la estrategia a desarrollar para darle solución a un problema existente
Persona que presta sus servicios de manera independiente
Es el documento que realiza una empresa en el que plasma las oportunidades de mejora, necesidades o problemas y se las presenta a una empresa o al experto informático para darles solución

FORMATO PARA EL DESARROLLO DE COMPONENTE FORMATIVO

PROGRAMA DE FORMACIÓN	ANÁLISIS Y DESARROLLO DE SOFTWARE

COMPETENCIA	220501096 desarrollar la solución de software de acuerdo con el diseño y metodologías de desarrollo.	RESULTADOS DE APRENDIZAJE	220501096-05 Realizar pruebas al software para verificar su funcionalidad

NÚMERO DEL COMPONENTE FORMATIVO	38
NOMBRE DEL COMPONENTE FORMATIVO	Las pruebas de software
BREVE DESCRIPCIÓN	Son el grupo de actividades enfocadas en encontrar los posibles fallos en el funcionamiento un software
PALABRAS CLAVE	calidad, diseño de prueba, herramientas de prueba, niveles de prueba, software, testing

ÁREA OCUPACIONAL	2 - CIENCIAS NATURALES, APLICADAS Y RELACIONADAS
IDIOMA	Español

A. TABLA DE CONTENIDOS:

1.0 Generalidades de las pruebas de Software
1.1 Tipos de pruebas y sus características
1.2 Los casos de prueba
1.3 Desarrollo guiado por pruebas

B. DESARROLLO DE CONTENIDOS:

Introducción

El intercambio y manejo de información es esencial en todos los aspectos ya que involucra los niveles operativos, económicos, culturales e intelectuales de la humanidad; como parte fundamental de este son los sistemas de información que se desarrollan para optimizar el intercambio y almacenamiento de estos. (Sanz, 2005) afirma que las pruebas de software son parte fundamental en el desarrollo de los sistemas de información ya que son el conjunto de procesos encaminados a encontrar posibles fallos de funcionamiento, usabilidad o configuración de un programa o aplicación

Las pruebas de software también conocidas como “Software Testing” son la actividad de control de calidad más realizada en los proyectos de desarrollo, o mantenimiento de software, en ellas se encuentran errores como los (bugs), que a simple vista son insignificantes, pero pueden ser trágicos en los procesos que el programa desarrolla.

Ilustración 1



Fuente: https://www.pngwing.com/es/free-png-dfysl

1.0 Generalidades de las pruebas de Software

Las pruebas software son parte fundamental del proceso de calidad y adopción de buenas prácticas, estas deben realizarse en todo el proceso de desarrollo de software, de acuerdo a Mera (2016) este proceso debe incluir “Revisión de los requerimientos, realización de análisis documentales, identificación de defectos, pruebas funcionales y no funcionales, pruebas dinámicas y estáticas, pruebas de integración, informes de confianza en el nivel de calidad, información para la toma de decisiones, planes de mejora continua”

El mismo autor afirma que dentro de las actividades de un proceso de prueba se debe incluir el control, comprobación de los resultados, elaboración de informes del proceso, aplicación objeto de las pruebas y bitácoras de experiencias.

Es de suma importancia realizar una bitácora y una matriz con el fin de ejecutar los casos de prueba y así concluir si el caso funciona como se esperaba para agregarlo como una conformidad, en caso contrario pasaría a ser una no conformidad, la matriz ayudan a establecer métricas que ayudan a definir tiempos, esfuerzos, asignaciones, y riesgos logrando optimizar recursos y talento humano, asegurando una óptima calidad del producto final

Según la organización ISO (2020) Estas pruebas se regulan por la norma ISO/IEC 25000 Sistemas y Requisitos de Calidad y Evaluación de Software (SQuaRE) . Pruebas de software la cual es aplicable a todos aquellos que deseen crear especificaciones de prueba, crear marcos correspondientes o crear automatización de pruebas

1.1 Tipos de pruebas

En los últimos años se ha propuesto algunos principios con el fin de establecer pautas universales para que los programadores de software las adapten a sus procesos de pruebas

Mera (2016) lista los siete principios:

Las pruebas muestran la presencia de defectos: Las pruebas son herramientas que facilitan detectar defectos; no obstante, no garantizan que no haya defectos ocultos y no es una evidencia de que el software esté optimo

Pruebas tempranas: Encontrar los errores en etapas tempranas, permite un importante ahorro de recursos

No es posible realizar pruebas exhaustivas: hay que realizar un análisis riesgos, para identificar prioridades y tener una óptima toma de decisiones, centralizando esfuerzos y utilizando el talento humano y recursos de manera correcta

Las pruebas dependen del contexto: estas dependen del contexto en el cual se ejecutan; por lo cual se debe dar más énfasis en las que sean para un sistema crítico como para un software financiero, para el cual se requiere realizar más pruebas, en comparación con otras aplicaciones con nivel críticos bajos.

Agrupación de defectos: Las pruebas deben agruparse, la mayoría de fallos operativos se enfocan en un número reducido de módulos.

La paradoja de los pesticidas: no hay que repetir la misma prueba varias veces, por lo cual es importante que los casos de prueba se revisen constantemente y escribir nuevos con el fin de encontrar más defectos.

Ausencia de error: de nada sirve corregir en el sistema si este no cumple con los requerimientos o necesidades del usuario.

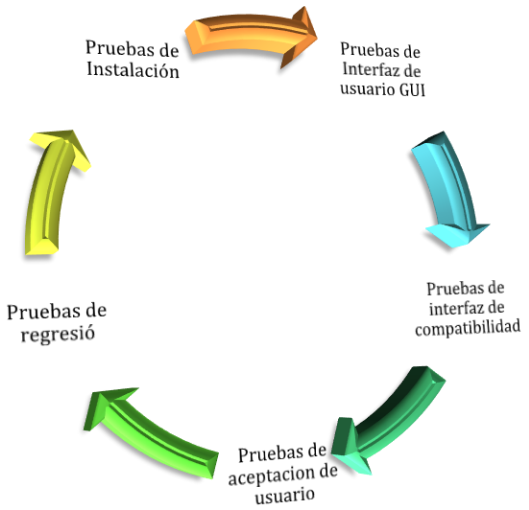
Ilustración 2



Fuente: https://www.pngwing.com/es/free-png-ysnoq

1.1.1 **Pruebas funcionales** IBM (2021) plantea que estas pruebas se basan en la ejecución y revisión de las funciones y en su interoperabilidad con sistemas específicos, se llevan a cabo en todos los niveles de prueba, buscan evaluar cada una de las opciones con las que cuenta el paquete informático, principalmente en el comportamiento externo del producto o aplicativo, también se las identifica como de tipo caja negra.

Ilustración 3



Autor

1.1.2 Pruebas unitarias de acuerdo a Mera (2016) también se las conoce como unit testing son pequeños tests en los cuales se revisa que el comportamiento de un objeto y su lógica funcione adecuadamente.

por lo general se realiza durante la fase de desarrollo de la aplicaciones o software, esta es ejecutada por los programadores, aunque también pueden realizarlas los aseguradores de calidad . las razones por las cuales desarrollamos este tipo de pruebas son:

- Son rápidas de realizar por lo que se pueden realizar en gran cantidad
- Sirven como documentación de apoyo para el proyecto
- Demuestran que la lógica del código funciona en todos los casos
- Permite a los programadores a entender bien el código base, lo que permite realizar cambios oportunos.
- Se obtendrá un código de calidad

Ilustración 4



Fuente: https://www.pngwing.com/es/free-png-nytaj

En algunos casos las pruebas pueden hacerse de manera manual pero lo mejor es usar herramientas que permitan ejecutar el servicio de la mejor manera posible, hay muchas herramientas en el mercado y estas varían en función del lenguaje de programación que se esté utilizando aquí mencionaremos algunas de las más conocidas.

- xUnit: Es una herramienta de pruebas unitarias para el framework .NET.
- Junit: Es una herramienta de pruebas de aplicaciones Java.
- NUnit: Es una herramienta de pruebas de plataformas .NET.
- PHPUnit: Es una herramienta de pruebas de programación PHP.

Para realizar las pruebas unitarias debemos tener en cuenta los siguientes criterios

- Se debe probar un sólo código a la vez.
- Hay que realizar pruebas frecuentemente mientras programas
- Se debe corrige los bugs identificados en las pruebas antes de continuar
- Cualquier cambio que ser realice también debe pasar el test

R.E.D Tema Pruebas Unitarias y Funcionales 1.1.1 y 1.1.2
Acordeón Tipo A

1.1.3 **Pruebas de integración** de acuerdo a Mera (2016) estas son los encargados de verificar el óptimo acople entre los componentes y de probar las interfaces entre los módulos, como el sistema de archivos en integración con el hardware y el componente de validación de usuario con el sistema operativo, hay que tener en cuenta los objetos típicos de prueba que son:

- Bases de datos de sub sistemas
- Datos de configuración
- Configuración del sistema
- Infraestructura
- interfaces

Ilustración 5



Fuente: https://www.pngwing.com/es/free-png-mrsfl

El mismo autor argumenta que en las pruebas de integración se revisan las interfaces entre componentes o subsistemas para cerciorarse que son llamados cuando se necesite y que los datos o mensajes que se transmiten son los adecuados, existen los siguientes tipos:

Integración incremental: se prueba el nuevo componente en conjunto con las que ya se probaron y se incrementa progresivamente con los nuevos componentes que ser probaran.

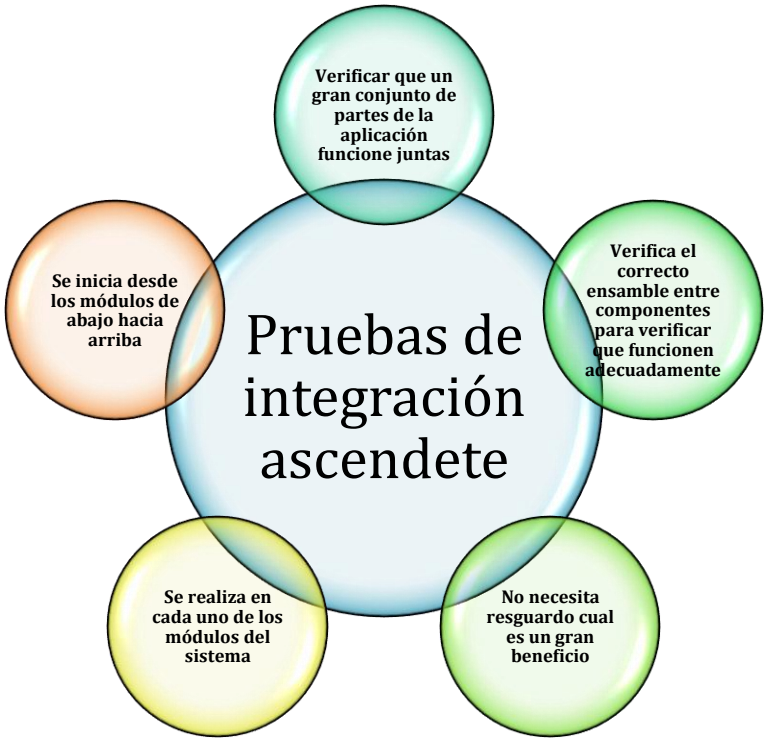
Descendente: también conocidas como **(***top-down***)** se realizan siguiendo el flujo de control ejemplo desde el GUI o menú principal y los componentes o sistemas se sustituyen por stubs.

Ascendente: también conocidas como **(***bottom-up***)** las pruebas se ejecutan desde la parte inferior del flujo hacia arriba y los componentes o sistemas se sustituyen por controladores.

Integración no incremental: también conocida como Big-bang se prueba cada componente por separado y al final se integran todos de una vez realizando las pruebas pertinentes.

Combinadas: se desarrolla combinando el enfoque ascendente y descendente, pero hay que realizar una planificación para que los componentes individuales se encuentren en el medio.

Ilustración 6



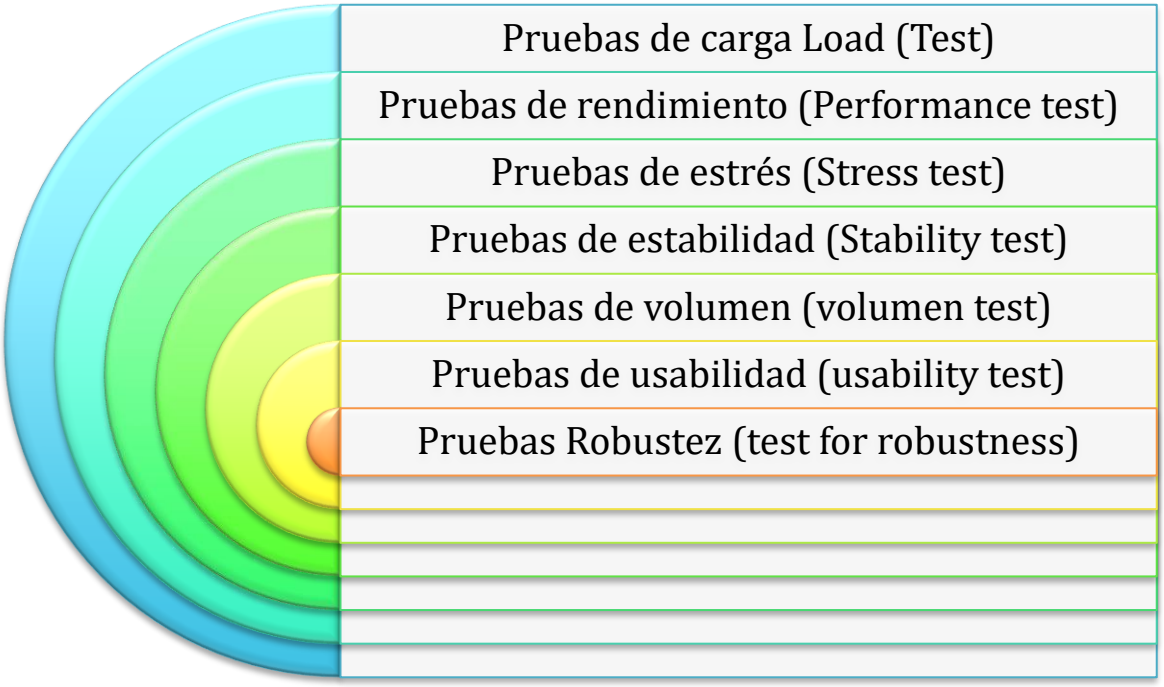
Autor

1.1.4 Pruebas no funcionales: Mera (2016) manifiesta que en este tipo de pruebas se comprueban los requisitos basados en la operación de un software, no en su funcionalidad, estas pruebas, pueden ayudarnos a establecer la carga que resiste el producto, si su rendimiento es el adecuado o si es estable a nivel de contacto con el servidor, en otras palabras, no dice si tienen un bajo desempeño o rendimiento en la producción. aquí encontramos:

- *Pruebas de carga:* Simulan la demanda esperada que puede tener una aplicación y mide el resultado
- *Prueba de rendimiento:* se calcula la respuesta del programa con diferentes medidas de peticiones del usuario.
- *Pruebas de volumen:* se prueba el funcionamiento del programa con ciertos volúmenes de datos
- *Pruebas de estrés:* cuantos usuarios o peticiones puede soportar el programa, con esta prueba determinamos si el sistema es estable.
- *Pruebas de configuración:* aquí validamos que efectos en el desempeño tienen los cambios de configuración que se pueden aplicar como en diferente hardware, software, versiones y navegadores

- *Prueba de resistencia:* aquí se prueba cómo se comporta una aplicación luego de un uso prolongado
- *Pruebas de cumplimiento:* se evalúa si la aplicación cumple con los objetivos para el cual fue diseñado.
- *Pruebas de usabilidad:* se centran en validar que tan fácil se utiliza el programa en cuanto a facilidad de aprendizaje, eficiencia, memorización, errores y satisfacción.

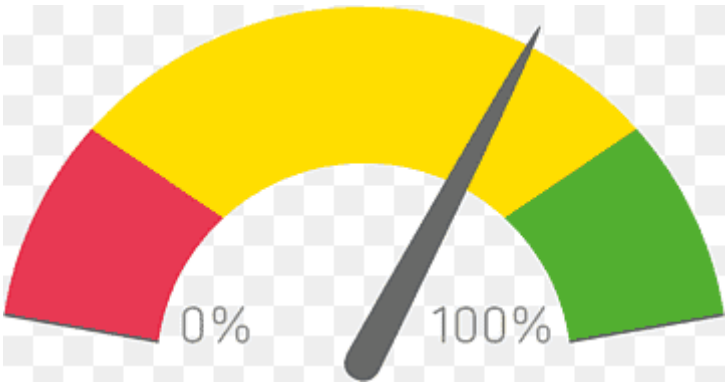
Ilustración 7



Autor

1.1.5 Pruebas de rendimiento Según IBM (2021) estas pruebas sirven para evaluar lo rápido que el programa realiza una tarea en condiciones específicas de trabajo, también evalúa la escalabilidad, fiabilidad y uso de los recursos

Ilustración 8



Fuente: https://www.pngwing.com/es/free-png-mcxfg

El mismo autor expresa que los objetivos de estas pruebas son:

- **Identificar y localizar problemas de rendimiento:** es permitir encontrar un fallo en el rendimiento, y también ayudar a localizar en que parte está ese problema de rendimiento para poder solucionarlo.
- **Verificar el cumplimiento de los SLA (Acuerdos de Nivel de Servicio)**. Se utiliza para medir los tiempos de respuesta del software en condiciones y tiempos específicos.
- **Localizar cuellos de botella**. Ayudar a detallar en que lugar se generan estos cuellos de botellas si se debe problemas de hardware como CPU, memoria, o es el ancho de banda

R.E.D Tema Pruebas integración No Funcionales y de rendimiento
Pestaña Tipo B

1.1.6 Ventajas y desventajas de los tipos de pruebas. Este punto se abordó en cada uno de los puntos anteriores.

Ventajas de estos tipos de prueba

Pruebas funcionales

- Optimiza las funcionalidades y la calidad del producto.
- Asegura que la seguridad y la arquitectura sean las adecuadas
- Asegura que el comportamiento del software y de todas sus funciones sea el esperado.
- Se asegura que el software o aplicación está libre de errores.

Pruebas unitarias

- Simplifica la integración y permite una buena documentación.
- Funciona con el proceso de desarrollo ágil.

- Detección temprana de errores en sus funcionalidades lo que minimiza tiempo y costos
- Optimiza la efectividad del código.

Ilustración 9



Fuente: https://www.pngwing.com/es/free-png-zlayv

Pruebas de integración

- Detalla los errores y conflictos en cada módulo y en sus conexiones
- Mejorar la calidad del código.
- Garantiza que las funciones del software serán confiables y estables entres sus módulos
- Garantiza que todos los módulos del software estén correctamente integrados y funcionen en conjunto como se necesita.

Pruebas de Rendimiento

- Mejoras las características del software o aplicación perfeccionando el sistema
- Identifica errores que no se detectan en las pruebas funcionales.
- Localiza los cuellos de botella para solucionarlos y optimizar su rendimiento.
- Valora la escalabilidad y velocidad de la aplicación o website

Desventajas de estos tipos de pruebas

Por lo general todas las pruebas de software traen beneficios al desarrollador, dependiendo del tipo de la prueba los beneficios pueden ser más altos y enfocados en un fin o propósito especifico, dando como resultado una mejor calidad de producto, mejorando su funcionalidad, reduciendo costos, tiempos, recursos humanos y mejorando la experiencia del usuario.

Se podría considerar como una desventaja, si se comparan los tipos de prueba ya que puede variar en:

Tiempo

Costo

Esfuerzo

Recursos humanos

Exactitud de las pruebas

Sin embargo, la única desventaja de las pruebas de software son no hacerlas ya que el omitir este paso desencadena en una nefasta serie de errores como un producto no funcional y colocarían en tela de juicio la profesionalidad de programador y su equipo de trabajo

1.2 Los de casos de pruebas

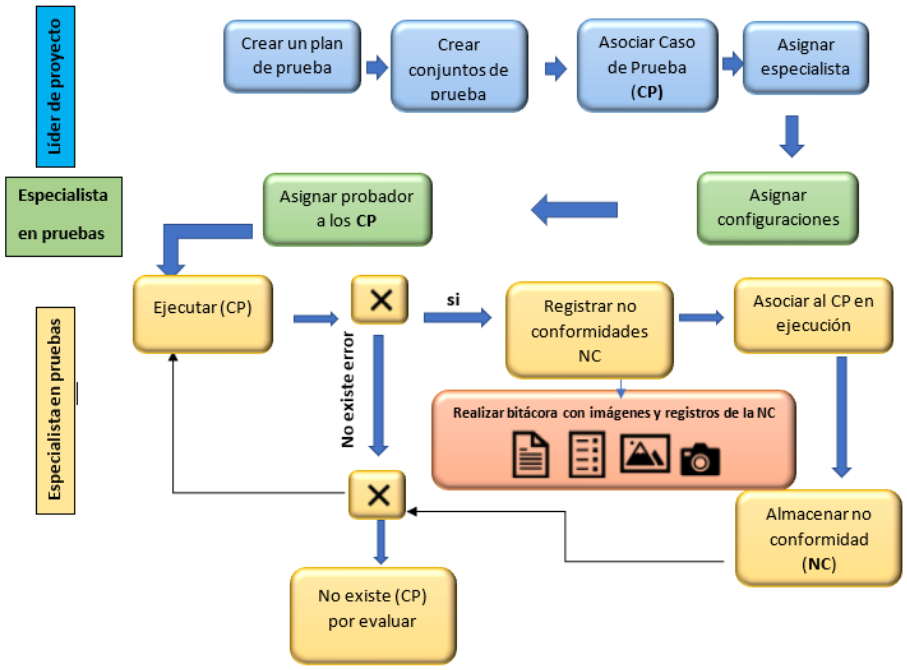
Ilustración 10



Fuente: https://www.pngwing.com/es/free-png-nvsxn

Son los documentos de los productos que se generan al momento de realizar una prueba de software, en el cual se plasma las precondiciones entradas y resultados esperados, incluyendo la implementación correcta, la identificación de errores, el chequeo de la calidad y otras informaciones consideradas importantes, en general no hay una plantilla especifica solo la bitácora, listas de chequeo, e imágenes en la cual se registran las no conformidades que se encuentran en las pruebas.

Ilustración 11



Autor

En el diseño de las pruebas de software se deben identificar y describir casos de prueba, por lo tanto para documentar un caso de pruebas se debe comprender que este contiene un conjunto de variables o condiciones en la cual un tester determina si un sistema funciona según lo esperado, así mismo cabe resaltar que la documentación de un caso de prueba se convierte en una actividad sin dificultad, si tenemos la información necesaria para su proceso de elaboración, teniendo en cuenta que al momento de probar un software es de mucho aporte, puesto que se transforma en una herramienta esencial en el proceso de registro, seguimiento y control.

A continuación, se nombran los elementos principales que debe contener un caso de prueba.

Tabla 1

Caso de prueba	
Identificador	Puede ser alfanumérico o numérico.
Nombre	Nombre del caso de prueba de manera concisa.

descripción	Objetivo del caso de prueba, también describe que probara, en ciertas ocasiones se incluye el ambiente de pruebas.
Número de orden **Ejecución**	Orden en la cual se ejecuta el caso de prueba, en la situación de que se tengan múltiples casos de prueba.
Requerimiento asociado	Si se plantea un caso de prueba se debe saber a qué requerimiento va asociado para mantener la trazabilidad
Precondición	Estado en la cual se debe encontrar el sistema antes de comenzar la prueba
Postcondición	El estado en que debe encontrarse el sistema luego de ejecutar la prueba
Resultado esperado	Objetivo que debe ser alcanzado posterior a ejecutarse la prueba.

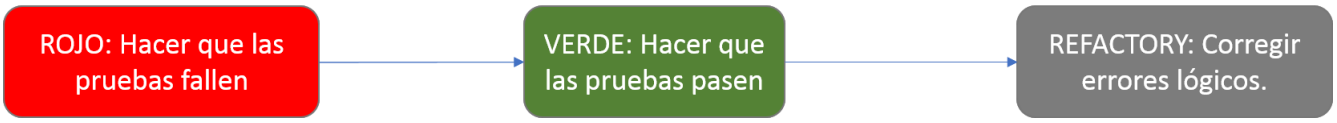
Fuente: Autor

1.3. Desarrollo guiado por pruebas.

Para Herranz (2021)" El desarrollo de software dirigido por pruebas es una técnica de programación en la cual las pruebas que normalmente son las unitarias se escriben primero y luego se escribe el código que pase la pruebas, con esta práctica se consigue entre otras cosas: un código más robusto, más seguro, más mantenible y una mayor rapidez en el desarrollo."

Cuando el TDD se combina con el desarrollo Ágil según Herranz (2021) "tienen como consecuencia un espacio más amplio el cual cubre el desarrollo de software hasta el diseño de software el cual propone el proceso denominado ROJO-VERDE-REFACTORY".

Proceso de TDD - Desarrollo Dirigido por pruebas



ROJO: Hacer que las pruebas fallen
VERDE: Hacer que las pruebas pasen
REFACTORY: Corregir errores lógicos.

Según Sánchez (2017) “Para usar TDD el desarrollador de software tiene que cambiar su método de trabajo. Ahora, en vez de crear los casos de uso y después implementarlos, va a intentar convertir un caso de uso en un ejemplo. Si se realiza esto con todos los casos de uso, llegará un momento en el que los ejemplos describirán por sí solos la próxima tarea a implementar.

El proceso de diseño de software, combinando TDD con metodologías ágiles, sería el siguiente:

- El cliente escribe su historia de usuario.
- Se definen en conjunto con el cliente los criterios de aceptación de la historia de usuario, descomponiéndola todo lo que se pueda para simplificar lo más posible.
- Se selecciona el punto de aceptación más sencillo y se transforma en la prueba unitaria.
- Se verifica que esta prueba falla.
- Se escriben las líneas de código para hacer que la prueba sea correcta.
- Se llevan a cabo todas las pruebas de carácter automático.
- Se vuelve a escribir el código para que quede limpio.
- Nuevamente se deben ejecutar las pruebas de carácter automático para verificar que todo sigue en orden funcionando.
- Se debe regresar al paso número 3 en cuanto se detecten criterios de aceptación que hagan falta y se debe repetir el ciclo una y otra vez hasta lograr la aplicación completa.

Es de comentar que este es un proceso de desarrollo relativamente nuevo, el cual está comenzando a ganar terreno y ser conocido en el ámbito del desarrollo de software en las organizaciones a nivel mundial, la cual se basa en evidencias anecdóticas y una serie de evaluaciones empíricas. Hablando del desarrollo tradicional las pruebas se usan para validar y verificar la calidad del software y su creación se realiza después de que existe la funcionalidad (Catalán San Martín ,2017).

En TDD, las pruebas adicionalmente al tema validación y verificación, se usan con fines de especificación, donde primero se desarrolla la prueba y posteriormente la funcionalidad. Esto permite que la calidad del software aumente y la productividad del desarrollador mejore, dado que se lleva un mejor esfuerzo en depurar y se avanza rápidamente con el progreso del producto software.

Beneficios del desarrollo guiado por pruebas

TDD permite una verificación de las funcionalidades desarrolladas, en ciclos más pequeños, lo que permite la detección de errores y así mismo su solución temprana.

La implementación de esta forma de trabajo permitirá elevar la fiabilidad del producto software, no solo desde el punto de vista de errores funcionales, sino que también en la consecución avanzar con las necesidades requeridas por el cliente.

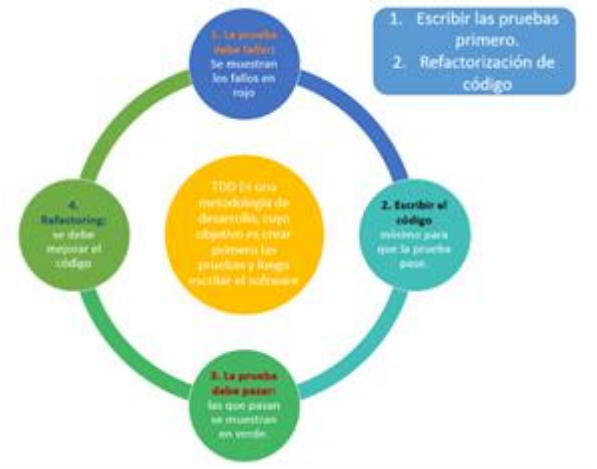
Otro de los beneficios es la optimización y mejora en el rendimiento del código, dado que se hace mucho énfasis en escribir solamente el código necesario para la resolución de la necesidad.

El proceso de realizar pruebas por cada nueva funcionalidad o requisito, facilita el proceso de desarrollo de forma modular, lo que permite la escalabilidad y el futuro mantenimiento del producto software.

Ciclo del desarrollo guiado por pruebas

Ahora que ya sabemos los beneficios que implica aplicar TDD, veamos el ciclo completo del proceso, revisando cada uno de sus pasos. En la siguiente imagen se puede observar el flujo de manera gráfica:

Ilustración Ciclo TDD 12

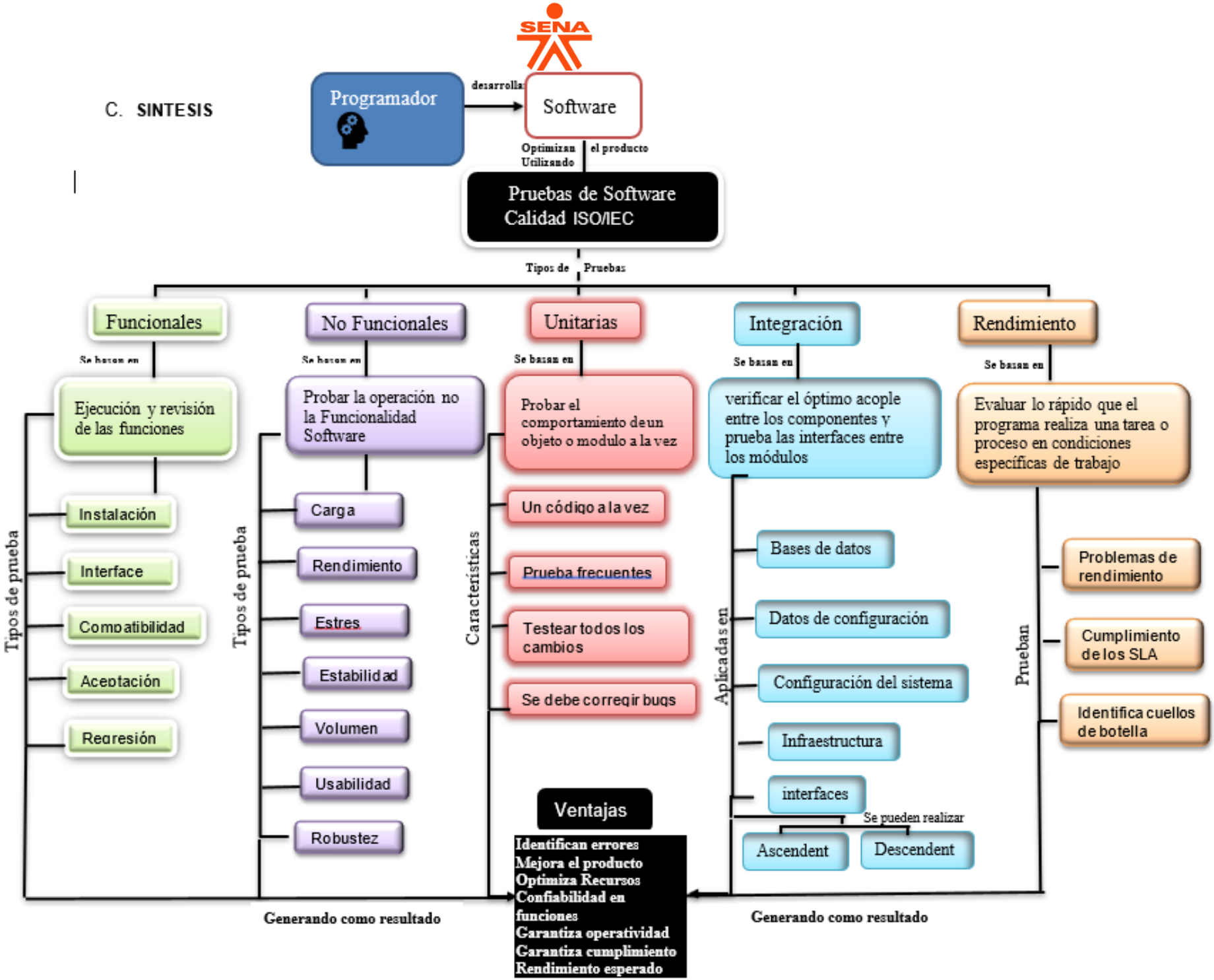


Fuente: https://ed.team/cursos/tdd

R.E.D Tema: Desarrollo guiado por pruebas
Pestaña Tipo B


SENA

C. **SINTESIS**


Programador
desarrolla
Software
Optimizan Utilizando
el producto
Pruebas de Software Calidad ISO/IEC
Tipos de Pruebas
Funcionales
No Funcionales
Unitarias
Integración
Rendimiento
Se basan en
Ejecución y revisión de las funciones
Probar la operación no la Funcionalidad Software
Probar el comportamiento de un objeto o modulo a la vez
verificar el óptimo acople entre los componentes y prueba las interfaces entre los módulos
Evaluar lo rápido que el programa realiza una tarea o proceso en condiciones específicas de trabajo
Tipos de prueba
Instalación
Interface
Compatibilidad
Aceptación
Regresión
Tipos de prueba
Carga
Rendimiento
Estres
Estabilidad
Volumen
Usabilidad
Robustez
Características
Un código a la vez
Prueba frecuentes
Testear todos los cambios
Se debe corregir bugs
Aplicadas en
Bases de datos
Datos de configuración
Configuración del sistema
Infraestructura
interfaces
Se pueden realizar
Ascendent
Descendent
Prueban
Problemas de rendimiento
Cumplimiento de los SLA
Identifica cuellos de botella
Ventajas
Identifican errores
Mejora el producto
Optimiza Recursos
Confiabilidad en funciones
Garantiza operatividad
Garantiza cumplimiento
Rendimiento esperado
Generando como resultado
Generando como resultado

D. ACTIVIDADES DIDÁCTICAS (OPCIONALES SI SON SUGERIDAS)

DESCRIPCIÓN DE ACTIVIDAD DIDÁCTICA	
Nombre de la Actividad	Conceptos básicos de Pruebas de Software
Objetivo de la actividad	Afianzar algunos Conceptos básicos de Pruebas de Software
Tipo de actividad sugerida	Arrastrar y soltar el término con la definición que corresponde
Archivo de la actividad (Anexo donde se describe la actividad propuesta)	Anexo documento en Word llamado Actividad didáctica 1

DESCRIPCIÓN DE ACTIVIDAD DIDÁCTICA	
Nombre de la Actividad	Conceptos básicos de Pruebas de Software
Objetivo de la actividad	Afianzar algunos Conceptos básicos de Pruebas de Software
Tipo de actividad sugerida	Selección múltiple
Archivo de la actividad (Anexo donde se describe la actividad propuesta)	Anexo documento en Word llamado Actividad didáctica 2

E. MATERIAL COMPLEMENTARIO:

Tema	Referencia APA del Material	Tipo de material (Video, capítulo de libro, artículo, otro)	Enlace del Recurso o Archivo del documento o material
Pruebas Funcionales	IBM (2021).Pruebas Funcionales.https://www.ibm.com/docs/es/rtw/9.1.0?topic=SSBLQQ_9.1.0/com.ibm.rational.test.ft.doc/topics/Getting_Started_With_Ivory.html	Articulo	https://www.ibm.com/docs/es/rtw/9.1.0?topic=SSBLQQ_9.1.0/com.ibm.rational.test.ft.doc/topics/Getting_Started_With_Ivory.html
Pruebas de Rendimiento	IBM (2021).Pruebas de rendimiento.https://www.ibm.com/docs/es/rtw/9.0.0?topic=phases-performance-testing	Articulo	https://www.loadview-testing.com/es/blog/tipos-de-pruebas-de-software-diferencias-y-ejemplos/

GLOSARIO:

TÉRMINO	SIGNIFICADO
bottom-up	pruebas Ascendente**.**
QA (Quality Assurance)	Especialistas en pruebas de software verifican que un software no contenga fallos**.**
REFACTORY	Acción de limpiar o reconstruir el código de una aplicación.
Sprint	Ciclo de desarrollo en metodología de desarrollo scrum.
TDD	Desarrollo dirigido por pruebas.
top-down	pruebas descendentes.

REFERENCIAS BIBLIOGRÁFICAS:

Herranz, J.I. (2021). TDD como metodología de diseño de software. https://www.paradigmadigital.com/dev/tdd-como-metodologia-de-diseno-de-software/

IBM (2021).Pruebas de rendimiento. https://www.ibm.com/docs/es/rtw/9.0.0?topic=phases-performance-testing

IBM (2021).Pruebas Funcionales.
https://www.ibm.com/docs/es/rtw/9.1.0?topic=SSBLQQ_9.1.0/com.ibm.rational.test.ft.doc/topics/Getting_Started_With_Ivory.html

Mera Paz, J. A. (2016). Análisis del proceso de pruebas de calidad de software. *Ingeniería solidaria*, *12*(20).

Organización ISO. (2020). Ingeniería de sistemas y software - Requisitos de calidad y evaluación de sistemas y software (SQuaRE) - Guía de SQuaRE.
https://www.iso.org/standard/64764.html

Sánchez, A.F (2017). Agile Testing. Estado del arte. Su aplicación en empresas TIC de Extremadura. Universidad De Extremadura

Sanz, L. F. (2005). Un sondeo sobre la práctica actual de pruebas de software en España. REICIS. Revista Española de Innovación, Calidad e Ingeniería del Software,1(2), 43-54.

C. CONTROL DEL DOCUMENTO

	Nombre	Cargo	Dependencia *(Para el SENA indicar Regional y Centro de Formación)*	Fecha
Autor (es)	Mario Fernando Meneses Calvache	Experto Temático	Regional Cauca, Centro de teleinformática y producción industrial	Octubre 2021
	Carlos Hernán Muñoz	Experto Temático	Regional Cauca, Centro de teleinformática y producción industrial	Octubre 2021

D. CONTROL DE CAMBIOS
(Diligenciar únicamente si realiza ajustes a la Unidad Temática)

	Nombre	Cargo	Dependencia	Fecha	Razón del Cambio
Autor (es)					

FORMATO PARA EL DESARROLLO DE COMPONENTE FORMATIVO

PROGRAMA DE FORMACIÓN	**Análisis y Desarrollo de Software**

COMPETENCIA	**220501093 - Análisis de la especificación de requisitos del software**	**RESULTADOS DE APRENDIZAJE**	Planear actividades de análisis de acuerdo con la metodología seleccionada

NÚMERO DEL COMPONENTE FORMATIVO	**6**
NOMBRE DEL COMPONENTE FORMATIVO	**Metodologías de desarrollo de software**
BREVE DESCRIPCIÓN	**Las metodologías de desarrollo de software proponen un conjunto de procesos y actividades que deben ser desarrolladas por el equipo de desarrollo de software con el fin de 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.**
PALABRAS CLAVE	**Scrum, Xp, Rup, Agíl, Marcos de trabajo**

ÁREA OCUPACIONAL	**2 - CIENCIAS NATURALES, APLICADAS Y RELACIONADAS**
IDIOMA	**Español**

A. TABLA DE CONTENIDOS:

Introducción

1. **Metodologías de desarrollo de Software**
2. **Marcos de trabajo tradicionales**

2.1. Cascada
2.2. RUP

3. **Marcos de trabajo Ágiles**

3.1. XP
3.2. RAD
3.3. SCRUM

4. **Planeación de proyectos de software**

B. DESARROLLO DE CONTENIDOS:

Introducción

La industria del software ha tenido constantes avances desde su creación adoptando prácticas y tecnologías cada vez más complejas, buscando siempre satisfacer las necesidades de las personas, que cada vez son más exigentes y en tiempos cada vez más reducidos.

La ingeniería de software que surgió a partir del año 1968 busca principalmente plantear mecanismos que permitan a la industria del software adaptarse a los cambios mejorando la calidad de los productos y servicios desarrollados (Sommerville, 2005). Si bien la calidad de un producto o servicio dependerá de muchos factores uno de ellos está relacionado con la forma en cómo este producto o servicio fue construido. Es en este espacio donde las metodologías de desarrollo de software cobran importancia.

1. Metodologías de desarrollo de Software

Es importante aclarar que existe un gran número de definiciones sobre lo que es una metodología, para evitar cualquier confusión en este material utilizamos la definición dada en (Maida & Pacienzia, 2015), donde se indica 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.
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 los cuales al final determinarán las tareas generales y específicas que se deberían realizar para alcanzar un objetivo.

Existen diferentes tipos de metodologías de desarrollo de software las cuales 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 nos enfrentamos a la hora de desarrollar un producto y servicio. Es decir, cada una de estas metodologías tienen 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.

2. Marcos de trabajo tradicionales

Para el desarrollo de un buen producto de software se debe iniciar por un buen 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 & Pacienzia, 2015).

Lo anterior supone que este tipo de enfoques son óptimos en proyectos en los cuales los requisitos están plenamente identificados y delimitados, donde no se producirá ningún cambio en lo establecido mientras el proyecto es finalizado.

A continuación se describen algunas metodologías que se enmarcan en los marcos tradicionales de desarrollo de software.

2.1. Cascada

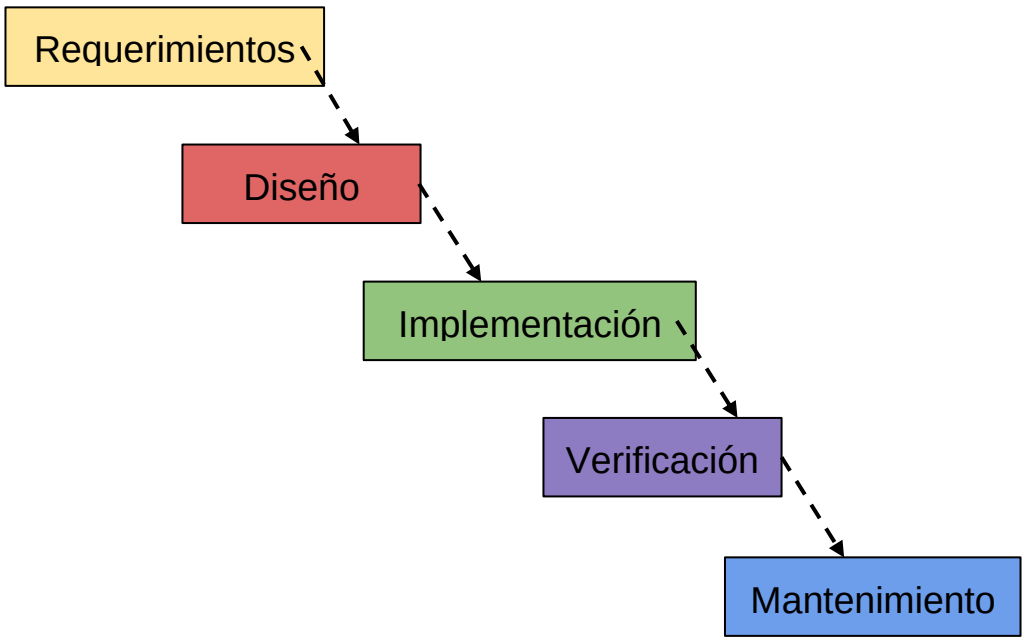
Este es uno de los modelos genéricos más ampliamente conocido en la ingeniería de software y se deriva de procesos de ingeniería de sistemas más generales (Royce, 1970). Este modelo plantea un proceso lineal donde las actividades de desarrollo de un producto o servicios de software se agrupan en un conjunto de

fases sucesivas donde estas son desarrolladas una única vez y los resultados de cada fase son la entrada requerida para cada fase subsiguiente, ninguna fase puede iniciar si la fase anterior no ha sido finalizada generalmente mediante un formalismo que puede ser un documento.

Según (Sommerville, 2005) el modelo en cascada se compone de 5 etapas principales, las cuales se asocian con actividades fundamentales en el proceso de desarrollo de software, las cuales son:

- Análisis: en esta fase se realiza una especificación muy detallada del sistema indicando los alcances, servicios a construir y restricciones, Esta fase se encarga principalmente en la especificación de los requisitos.
- Diseño: en esta fase se establece la arquitectura final del sistema, también se describen todas las abstracciones fundamentales del software y relaciones generalmente usando un lenguaje de modelado.
- Implementación: esta fase involucra la construcción del software a partir de los diseños del sistema y se hace una verificación inicial e individual del funcionamiento de cada uno de los módulos que se vayan construyendo.
- Verificación: esta fase se encarga de hacer el proceso formal de pruebas verificando fundamentalmente que el conjunto de módulos construidos funcione correctamente pero además cumpla con los requerimientos establecidos en la primera fase.
- Funcionamiento y mantenimiento: esta es la fase en la que entra el proyecto una vez el producto o servicio de software es entregado para ser usado por el cliente, por otra parte el mantenimiento hace referencia a la corrección de errores que no fueron detectados en las fases anteriores, implementación de mejoras del sistema y la incorporación de nuevos requerimientos.

Figura 1. Etapas del modelo Cascada.



Fuente: Autor.

En la Tabla 1 se presenta un cuadro comparativo donde se resumen las principales ventajas y desventajas de este modelo.

Tabla 1. Ventajas y desventajas del modelo cascada.

Ventaja	Desventaja
La definición clara de fases permite el desarrollo de una estructura sencilla que es ideal para proyectos sencillos y cortos	No se acopla muy bien a proyectos complejos con múltiples equipos trabajando en paralelo ya que las fases normalmente se traslapan y es difícil diferenciarlas.
Siguiendo este modelo se genera una muy buena documentación del proceso y es posible definir hitos claros.	Es difícil introducir cambios en el transcurso del proyecto.
Es más fácil realizar la estimación de costos al inicio del proyecto	Los usuarios finales y clientes normalmente son integrados al final del proceso lo que impide tener realimentación y ajustes en etapas tempranas.
Es fácil elaborar cronogramas de trabajo en base al desarrollo de las actividades de cada fase.	Existen fallos que solo son detectados cuando el sistema entra en funcionamiento, lo que puede ser desastroso para un proyecto que sigue este modelo.

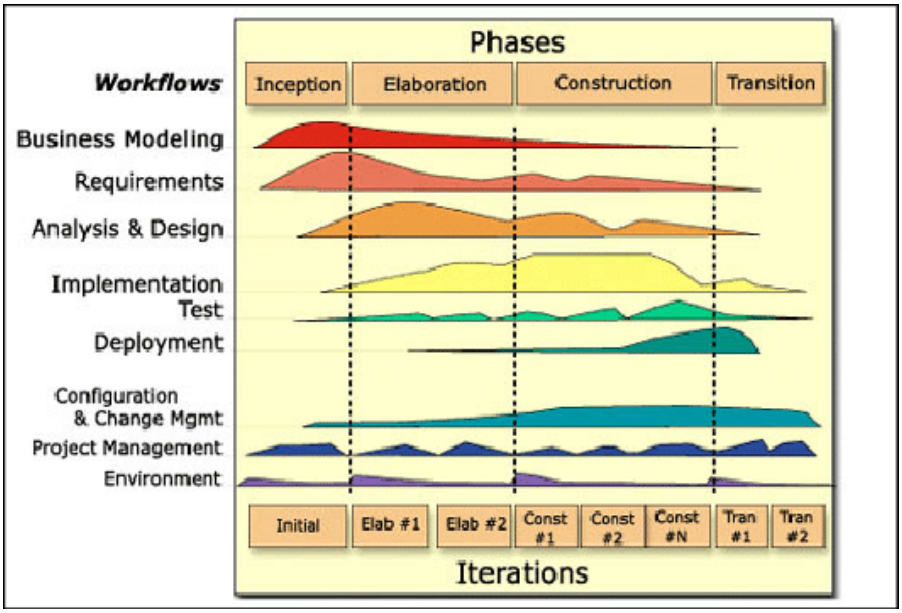
Fuente: Autor.

2.2. RUP

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) (Kruchten, 2003).

RUP divide el proceso de desarrollo en cuatro grandes fases, dentro de las cuales se realizan algunas iteraciones donde se desglosan en mayor o menor intensidad un conjunto de disciplinas según la fase que se está abordando. En la Figura 2 se observan las fases y disciplinas propuestas por RUP.

Figura 2. Fases y disciplinas de RUP.



Fuente: https://www.researchgate.net/profile/Mustafa-Man/publication/228820202/figure/fig6/AS:668418772578316@1536374862199/Phases-in-Rational-Unified-Process.png

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

La segunda fase corresponde a la elaboración, donde se centra la mayor parte del esfuerzo en la definición general de la arquitectura del sistema y el refinamiento de requisitos y modelado del negocio. RUP generalmente se apoya en el lenguaje de modelado UML para el modelado del negocio y descripción de la arquitectura.

La tercera fase corresponde a la construcción, donde se centran en las actividades relacionadas con la construcción del producto. Normalmente esta fase está constituida por varias iteraciones donde en cada una de ellas se desarrollan un subconjunto de requerimientos que normalmente están especificados como Casos de Uso. Cada una de estas pequeñas iteraciones se realiza utilizando el modelo cascada y el conjunto de todas las iteraciones producen al final una versión del producto.

La última fase se le denomina transición y en esta se desarrollan principalmente las disciplinas de pruebas y despliegue, es decir las actividades encaminadas a garantizar que el producto esté listo para entrega a sus usuarios. Es importante aclarar que dentro de cada una de las fases se desarrollan varias de las disciplinas, pero el esfuerzo y tiempo que se invierte en ellas tendrá mayor o menor enfoque dependiendo de la fase en la que se encuentre el proyecto.

Otro aspecto importante de esta metodología es que se apoya en un conjunto de artefactos que son desarrollados con el fin de especificar el proceso de análisis y diseño que soporta la construcción del software, entre los artefactos más comunes encontramos: documento de visión, documento de especificación de requisitos, diagramas de casos de uso, modelos conceptuales (clases y entidad relación), diagramas que representan la vista de implementación como diagramas de secuencia, estados y colaboración, entre otros.

Las disciplinas por otra parte representan un conjunto de actividades relacionadas con un área específica del proyecto y están inspiradas en el modelo en cascada. Rup establece las siguientes disciplinas: modelado de negocios, requerimientos, análisis y diseño, implementación, pruebas, despliegue, configuración y administración del cambio, administración de proyectos y ambiente (Kruchten, 2003).

La disciplina de modelado de negocios tiene como objetivo entender el problema a solucionar, medir el impacto organizacional y entender desde donde es posible obtener los requerimientos específicos para dar soporte a las necesidades de la organización. La disciplina de requerimientos se encarga de definir los límites del sistema, detallar los requerimientos y proveer una base para la estimación de costos y tiempos. La disciplina de análisis y diseño transforma todos los requerimientos en artefactos para el diseño general del sistema y su arquitectura. En la disciplina de implementación se transforman los diseños en código fuente y se realizan pruebas de unidad, generando por cada iteración versiones nuevas del sistema. La disciplina de pruebas realiza tareas enfocadas a garantizar la calidad del producto construido por medio de la validación del sistema en contraste con los requerimientos establecidos. La disciplina de despliegue realiza todas las tareas requeridas para llevar el sistema desarrollado a ambientes de producción para ser usado por los clientes finales y adicionalmente toda actividad relacionada con la entrega como por ejemplo procesos de capacitación.

RUP describe adicionalmente algunas disciplinas que son transversales a todas las fases, estas son: administración y configuración de cambio donde se utilizan métodos y herramientas para administrar todos los elementos que requieren registro específico de versiones para poder ser reproducidos en otros ambientes como por ejemplo los requisitos, el código fuente, infraestructura tecnológica, entre otros, La disciplina de administración de proyectos se encarga de las tareas de planeación y monitoreo general del proyecto lo cual es clave para proveer un marco de trabajo que permita hacer gestión del riesgo y finalmente la disciplina de ambiente que se encarga de desarrollar todas las tareas requeridas para proveer las herramientas y procesos que dan soporte al equipo de desarrollo.

3. Marcos de trabajo Ágiles

Los marcos de trabajo ágiles o metodologías ágiles para el desarrollo de software nacen como otra opción para abordar proyectos donde no es posible tener un detalle completo de los requerimientos y sus estimaciones en la primera fase del proyecto o donde es necesario hacer procesos de adaptabilidad a lo largo del proceso de desarrollo de software (Maida & Pacienzia, 2015).

Las metodologías ágiles proveen un conjunto de pautas y principios que buscan facilitar y priorizar la entrega de producto sobre procesos de documentación exhaustiva, haciendo procesos más simples donde interactúa el cliente final desde las primeras etapas del proyecto. El inicio de las metodologías ágiles nació en el año 2001 a partir del manifiesto ágil de software donde se establecen cuatro valores fundamentales (Manifiesto Ágil, 2001):

- Individuos e interacciones sobre procesos y herramientas
- Software funcionando sobre documentación extensiva
- Colaboración con el cliente sobre negociación contractual
- Respuesta ante el cambio sobre seguir un plan

Adicionalmente a los valores ágiles anteriormente listados, el manifiesto ágil establece 12 principios ágiles para materializar los valores definidos (Manifiesto Ágil, 2001):

- Nuestra principal prioridad es satisfacer al cliente a través de la entrega temprana y continua de software de valor.
- Son bienvenidos los requisitos cambiantes, incluso si llegan tarde al desarrollo. Los procesos ágiles se doblegan al cambio como ventaja competitiva para el cliente.
- Entregar con frecuencia software que funcione, en periodos de un par de semanas hasta un par de meses, con preferencia en los períodos breves.
- 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.
- El software que funciona es la principal medida del progreso.
- 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.
- La simplicidad como arte de maximizar la cantidad de trabajo que se hace, es esencial.
- Las mejores arquitecturas, requisitos y diseños emergen de equipos que se autoorganizan.
- En intervalos regulares, el equipo reflexiona sobre la forma de ser más efectivo y ajusta su conducta en consecuencia.

3.1. XP

XP es la abreviación comúnmente utilizada para referirse a Extreme Programming el cual es un marco de desarrollo de software ágil que busca producir software de alta calidad en contextos con requisitos altamente cambiantes, riesgos que involucran tiempos fijo con tecnologías nuevas y equipos de trabajo pequeños ubicados en un mismo sitio.

XP define cinco valores (Beck & Andres, 2004):

- Comunicación, el desarrollo de software requiere de trabajo en equipo por lo cual es importante la transferencia de conocimientos y utilizar medios de comunicación apropiados, se propone la discusión cara a cara con herramientas que permitan dibujar o escribir como por ejemplo un tablero.
- Simplicidad, esto se refiere a hacer solo las cosas que sean absolutamente necesarias evitando el desperdicio, elaborar las cosas de forma que sea fácil entender por otros y abordar solo requisitos conocidos.
- Retroalimentación, esto permite identificar áreas de mejora y revisión constante de las prácticas implementadas en el proceso de forma que se puedan establecer procesos de mejora permanentes.
- Coraje, se refiere al actuar sobre situaciones que pueden ser retadoras para el equipo como por ejemplo enfrentar problemas organizacionales, intentar implementar funcionalidades de formas diferente cuando lo convencional no funciona, aceptar comentarios, etc.
- Respeto, los miembros del equipo deben respetarse entre sí para comunicarse y trabajar en equipo. Cada persona contribuye en favor de lograr el objetivo del equipo.

Adicional a los valores XP se caracteriza por la definición de un conjunto de 12 prácticas de desarrollo de software que aunque pueden ser adoptadas de forma aislada tiene mayor relevancia con son desarrolladas en conjunto (Jeffries, 2011):

- El juego de la planificación: consiste en una reunión de planeación que realiza el equipo de desarrollo en conjunto con el cliente para discutir y probar las características a ser desarrolladas.
- Pequeños lanzamientos: esta práctica sugiere hacer iteraciones cortas con funcionalidades pequeñas que puedan ser probadas con el cliente de forma que se obtiene realimentación temprana y frecuente.
- Metáfora: Los nombres usados para definir cualquier tipo de identificador en el sistema debe ser coherente. El diseño y la estructura general debe ser comprensible para cualquier persona.
- Diseño simple: hacer código que funcione y que sea a la vez la solución más simple posible, evitando duplicación de código, menor número de métodos y clases.
- Pruebas: sugiere el uso de técnicas donde se enfatice en el proceso de pruebas constantes antes de iniciar la construcción del código como por ejemplo TTD (Desarrollo basado en pruebas)
- Refactorización: eliminación de funciones innecesarias, desacoplar elementos , eliminar cualquier tipo de redundancia de forma que se mantenga un código lo más limpio posible que sea fácil de entender y modificar.
- Programación en pareja: esta práctica propone que un código sea desarrollado por el trabajo de dos personas, donde una de ellas codifica y la otra está haciendo revisión, sugerencias y corrección de errores en el mismo momento de escritura.
- Propiedad colectiva: todo el equipo es dueño del sistema, esto promueve la cooperación y la libertad de cada miembro para aportar nuevas ideas.
- Integración continua: implica el uso de sistema de control de versiones de forma que el código pueda ser compartido o reutilizado pero también cada persona pueda trabajar de forma independiente y en paralelo sobre el mismo proyecto.

- Semana de 40 horas: es importante que los desarrolladores tengan espacios de trabajo apropiados donde puedan descansar y mantener un equilibrio entre la vida laboral y personal favoreciendo la eficiencia y puedan sostener la calidad durante todo el tiempo. Propone que no se deben exceder las 45 horas de trabajo semanales.
- Cliente in situ: El cliente final debe participar durante todo el proceso de construcción de software y estar presente para resolver cualquier pregunta que se pueda generar.
- Estándar de codificación: se deben utilizar los mismos formatos y estilos en la redacción de código

3.2. RAD

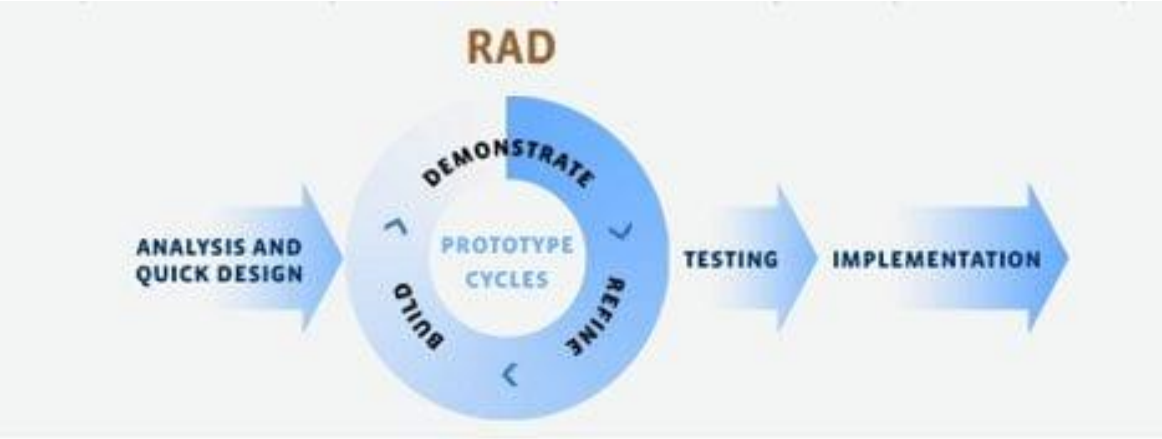
RAD (Desarrollo rápido de aplicaciones) es otra metodología de desarrollo de software ágil que se centra en el desarrollo rápido de aplicaciones mediante la realización de iteraciones frecuentes y realimentación constante y fue inventado por James Martin en 1991 (Martin, 1991).

Algunas características fundamentales de RAD son:

- Mayor flexibilidad y adaptabilidad a cualquier ajuste que deba realizarse durante el proceso de desarrollo.
- Iteraciones rápidas que reducen el tiempo de desarrollo y mantienen un ritmo de entrega acelerado.
- Se fomenta la reutilización de código.
- Mejor gestión del riesgo ya que las partes interesadas discuten y abordan cualquier vulnerabilidad mientras se construyen los productos

En la Figura 2 se describen las fases propuestas en RAD.

Figura 3. Fases definidas en RAD.



Fuente: https://blog.capterra.com/wp-content/uploads/2019/12/K_rOSJJEJ-Zl0G3VZGYWiGw3tFj1TuRQDvqPTpmVB3FrK4lnOONPG06gqWdwjBOagIbRUyCyD5R_WhvzZ4cHdglqaVyGyV4pWcn-PwIJn8JniGlQaeAQBJqLxqH_t4i7uCnBW9rp.png

La primera fase definida corresponde a la definición y finalización de los requisitos del proyecto. En esta fase las partes interesadas se reúnen para definir los requisitos del proyecto definiendo entre otros casos los objetivos, expectativas, plazos y presupuesto del proyecto.

La segunda fase aborda la construcción de prototipos los cuales son construidos, validados y mejorados a partir de la validación con los usuarios y una vez son aprobados pasan a la tercera fase donde estos prototipos son transformados en modelos funcionales.

En la cuarta fase se enfoca en la realización de pruebas exhaustivas para garantizar que todos los elementos construidos funcionan bien individualmente y también de forma colectiva y finalmente en la última fase se realizan todas las actividades de lanzamiento del producto lo que involucra el cargue inicial de datos y entrenamiento a los usuarios.

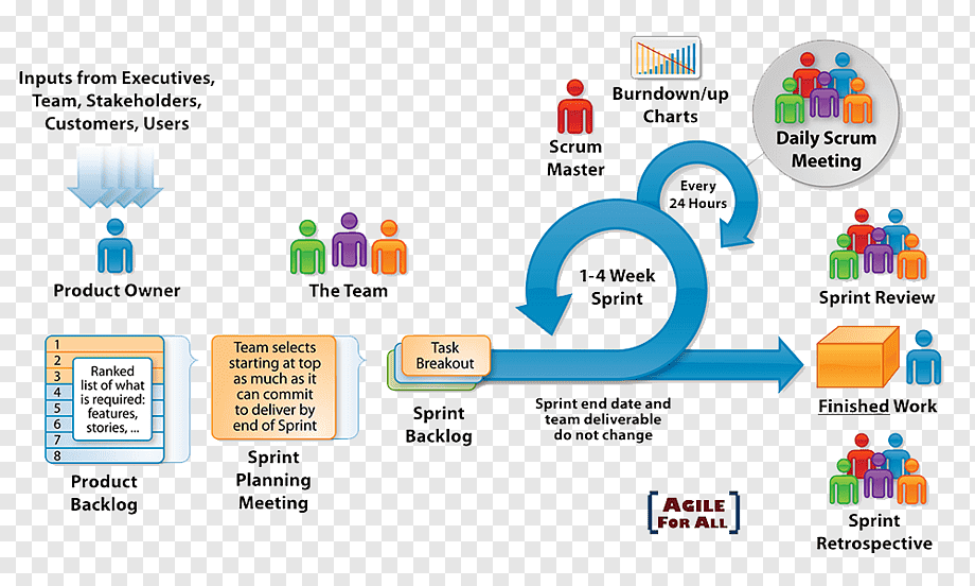
3.3. SCRUM

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

- Transparencia: hace referencia a que cualquier proceso de Scrum puede ser conocido por cualquiera. Esto es posible por medio de eventos como las reuniones de revisión y reuniones diarias, artefactos como la pila de Producto, cronogramas de lanzamiento y documentos de visión del proyecto, e instrumentos de seguimiento como lo son el BurnDown Chart o el tablero de Scrum (Scrum Board).
- Inspección: Permite que cualquiera pueda estar enterado de las actividades realizadas por otros y en general conocer el estado actual de los procesos.
- Adaptación: por medio de la transparencia y la inspección es posible fijar actividades de mejoras que permitan mejorar todo tipo de proceso en pro de lograr más altos estándares de calidad.

Adicionalmente este marco de trabajo ágil está estructurado por un conjunto de roles, eventos y artefactos que pueden ser observados en la Figura 2.

Figura 4. Marco de trabajo SCRUM.

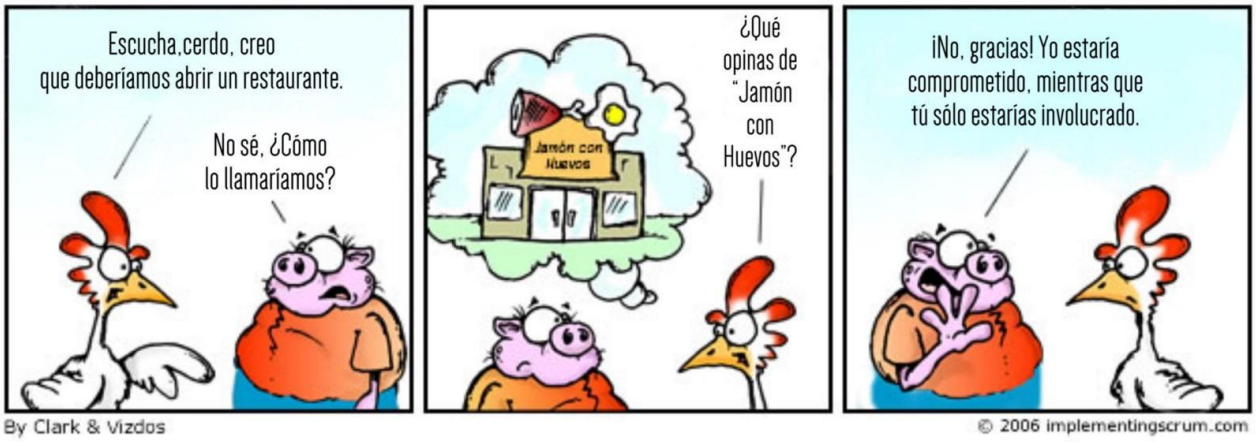


Fuente: https://www.pngwing.com/es/free-png-xqgjv

Dentro de los roles los roles hay una división en dos categorías fundamentales: los roles centrales que hacen referencia a los requeridos obligatoriamente para la creación de un producto, están altamente comprometidos y de los cuales depende el éxito o no de un proyecto, también están los roles no centrales que hace referencia a todos el personal interesado en el proyecto, pueden interactuar con el equipo pero no son los responsables del éxito del mismo, dentro de esta categoría entran los stakeholders, directivos, gerentes, marketing,

asesores, etc. Para representar mejor estas dos categorías se hace referencia a la historia del cerdo y la gallina que se observa en la Figura 5, los roles centrales son el cerdo en la historia y las gallinas los roles no centrales.

Figura 5. Fabula roles de Scrum.



*Fuente: https://cdn-images-1.medium.com/max/1600/1*znawlwBQPM_xAcopRXjQHw@2x.jpeg*

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

- Dueño del producto (Product Owner): persona con amplio conocimiento en el negocio del cliente, sus necesidades y las tendencias del mercado para el área específica. Este rol está encargado de maximizar el valor de negocio entregado al cliente y es el único responsable del control del Product Backlog (requerimientos) y su priorización. Este rol también representa al cliente en algunos procesos de demostración de avances y determina cuándo aprobar o no una entrega.

- Scrum Master: Es un rol que se encarga de facilitar los procesos al interior del equipo de trabajo removiendo cualquier impedimento y apoyando procesos de empoderamiento personal, debe velar porque los elementos propios del marco de trabajo Scrum se apliquen de manera correcta.

- Equipo de desarrollo (Developer team): son los responsables de la transformación de los requerimientos en código ejecutable a ser usado por el cliente, pero también son responsables de la planificación de las iteraciones, establecimiento de características para tener en cuenta en la verificación de un requerimiento terminado y presentación de avances a los clientes. Generalmente es un equipo autoorganizado y autogestionado.

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

- Sprint: es el corazón de Scrum y se refiere a una iteración que está acotada generalmente por un lapso entre 2 y 4 semanas donde se realiza un ciclo completo de actividades de análisis, diseño, construcción y pruebas para desarrollar una versión del producto potencialmente entregable al cliente.
- Planeación del Sprint: reunión realizada justo antes del inicio de un Sprint donde se definen el subconjunto de requerimientos (Sprint Backlog) a ser desarrollados en el siguiente Sprint y cómo será

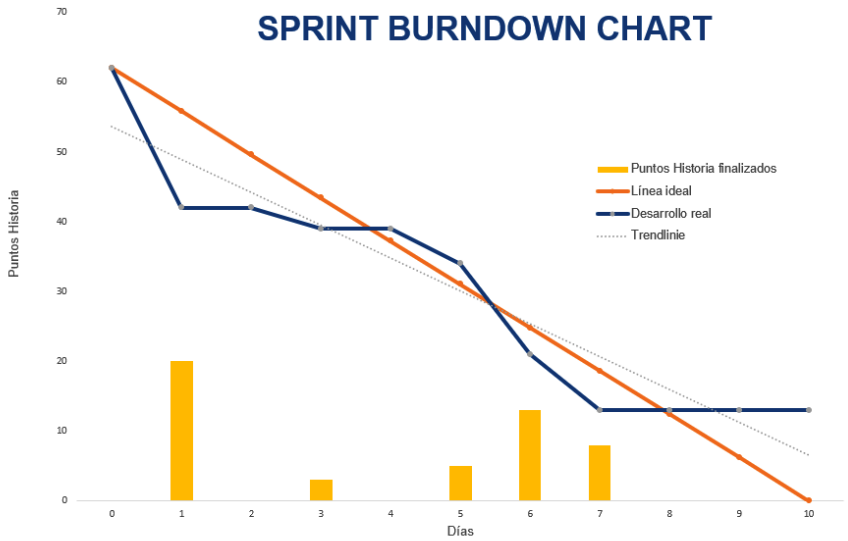
el proceso requerido para hacer la entrega del siguiente Sprint, lo cual incluye detallar los requerimientos en tareas concretas, estimación de tiempos/esfuerzo y distribución inicial de responsabilidades. Dependiendo de la duración del Sprint este tiempo de planificación puede variar pero la métrica establecida para Sprint de 4 semanas corresponde a una planeación de Sprint de 8 horas.

- Reunión diaria (Daily Meeting): Reunión realizada generalmente al inicio de cada día donde el equipo de trabajo informa en que ha venido trabajando, qué cosas realizará en el día y qué problemas se le han presentado. Es una reunión corta que se realiza de pie y que debe tener una duración alrededor de los 15 minutos. Esta reunión se alinea con los pilares de transparencia e inspección.
- Revisión del Sprint: reunión realizada al finalizar el sprint donde el equipo de desarrollo muestra los resultados del Sprint. Para Sprint de 4 semanas se usan reuniones de revisiones 4 horas.
- Reunión de retrospectiva: Última realizada luego de la revisión del Sprint y tiene como objetivo la autoevaluación personal y del grupo sobre el desempeño del Sprint que acaba de finalizar. En esta reunión se identifican y documentan los aprendizajes por medio de diferentes técnicas en las que generalmente se busca dar respuesta a las siguientes preguntas: ¿qué funcionó bien y se debe seguir haciendo?, ¿que no funcionó bien y se debe dejar de realizar? y ¿qué debemos empezar a mejorar? Para Sprint de 4 semanas se utilizan 3 horas para esta reunión.

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

- Pila de producto (Product Backlog): lista priorizada de requerimientos generalmente descritos en formato de historias de usuarios que representa todas las características del sistema a construir.
- Pila del Sprint (Sprint Backlog): lista de requerimientos seleccionados desde el Product backlog por el equipo de trabajo para ser desarrollados durante un Sprint particular. Este es uno de los artefactos generados en la reunión de planeación del Sprint.
- Burndown Chart: es un gráfico visual de dos ejes que muestra a los equipos la cantidad de trabajo pendiente por completar (eje Y) y el tiempo disponible para hacerlo (eje X). Este gráfico generalmente se realiza por cada Sprint ubicando la cantidad trabajo a realizar del Sprint Backlog (usualmente medido puntos de historia o horas de trabajo) en un tiempo 0 y por cada dia finalizado se resta la cantidad de puntos de historia u horas de cada tarea completada, también es posible usar este mismo gráfico para representar el avance general del proyecto ubicando en el eje Y la cantidad total de horas o esfuerzo del product Backlog y en el eje X la cantidad de Sprint proyectados. Cada uno de estos puntos se unen por medio de una línea y es posible determinar visualmente si el flujo de trabajo está en una situación óptima o no respecto al tiempo restante para completar el Sprint (ver Figura 6).

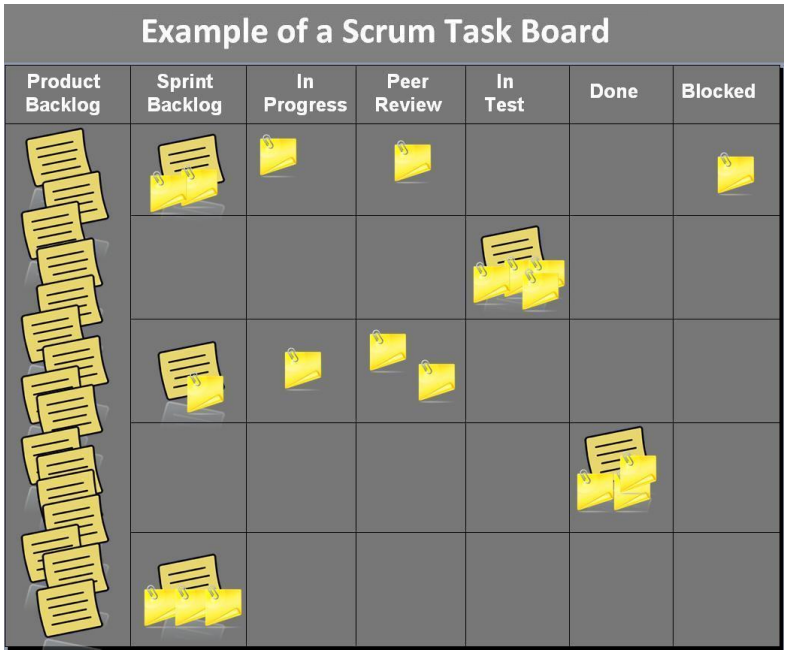
Figura 6. Burndown Chart.



Fuente: https://intl-blog.imgix.net/wp-content/uploads/2019/12/Burndown-chart.png?auto=format%2Cenhance

- Tablero de Scrum (Scrumboard): es un elemento visual donde se integra la mayor parte de los elementos del marco de trabajo Scrum, en él se indica la carga de trabajo, el estado actual de cada una de las actividades y sus respectivos responsables. Este es un elemento que se sincroniza de manera permanente y facilita la implementación de los pilares de transparencia, inspección y adaptabilidad (Ver Figura 7). Si bien se aconseja el uso de un tablero existen diferentes tipos de herramientas digitales que permiten la implementación de un tablero de Scrum.

Figura 7. Tablero de Scrum.



Fuente: https://upload.wikimedia.org/wikipedia/commons/f/fd/Scrum_task_board_example.jpg

Entre los principales beneficios del marco de trabajo Scrum encontramos:

- Es posible gestionar las expectativas del cliente de manera regular ya que este puede y debe participar en las reuniones de revisión por lo que está enterado todo el tiempo del estado actual del proyecto.
- El cliente puede obtener resultados importantes y utilizables desde las primeras iteraciones ya que la lista de producto está priorizada para ofrecer mayor valor en el menor tiempo posible y porque cada finalización de Sprint debe tener como resultado una versión totalmente funcional.
- El proyecto puede iniciar con requerimientos de muy alto nivel y es fácil administrar los cambios.
- La participación constante del cliente permite mitigar riesgos del proyecto desde sus primeras etapas.
- Los procesos de retrospectiva permiten establecer actividades permanentes de mejora continua en función de las experiencias del equipo.

4. Planeación de proyectos de software

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.

Según (SCRUMstudy, 2013) las actividades de planificación definidas dentro del marco de trabajo de Scrum son las siguientes:

- Creación de historias de usuario: En este proceso que es responsabilidad de Product Owner se crean las historias de usuarios con sus respectivos criterios de aceptación de forma que queden representados los requerimientos del sistema y sean comprendidos por los clientes y stakeholders.
- Estimación de historias de usuario: Proceso realizado por el equipo de desarrollo donde se valora el esfuerzo necesario para la realización de cada historia de usuario. Generalmente hay un proceso previo en el cual el Product Owner explica y presenta cada una de las historias de usuario y aclara cualquier duda que surja, así cuando se realiza el proceso de estimación todos son conscientes de las actividades que implican cada historia de usuario.
- Comprometer historias de usuario: En este proceso el equipo de desarrollo identifica las historias de usuario a ser desarrolladas en el Sprint, esto se hace mediante la reunión de planificación del Sprint.
- Identificar tareas: en este proceso cada una de las historias de usuario se desagregan más detalladamente por medio de tareas.
- Estimar tareas: En este proceso el equipo de desarrollo realiza una estimación de esfuerzo para la relación de la tarea de forma que es más sencillo asignar responsabilidades puntuales de acuerdo con las diferentes cargas del equipo.
- Crear el Sprint Backlog: el equipo de trabajo agrupa cada una de las tareas de las historias de usuario comprometidas en un artefacto llamado Sprint Backlog o pila del sprint.

Existen diferentes técnicas para la estimación de historias de usuario, entre las más destacadas se encuentran:

- Wideband Delphi: esta es una técnica de estimación basada en consenso para determinar la cantidad de trabajo necesario y el tiempo requerido para el mismo. En esta técnica cada individuo realiza estimaciones individuales de las historias y luego son socializadas al grupo para analizar los factores que influyen en cada estimación realizada, posterior a esto se realiza una nueva ronda de estimación individual siguiendo el mismo procedimiento hasta los valores de estimaciones se acerquen lo suficiente o se logre un consenso.

- Planning póker: es una técnica derivada de Wideband Delphi pero para la realización de las estimaciones se apoya en un conjunto de cartas especiales que vienen marcadas con números semejantes a la serie Fibonacci. Para el proceso de estimación se lee una historia de usuario y cada miembro del equipo selecciona una de las cartas y cuando se indica todos al mismo tiempo muestran sus cartas. En el caso de haber diversidad muy representativa en los valores se realiza una ronda donde los puntajes más altos y más bajos justifican sus decisiones, posterior a esto se hace una segunda ronda de estimación. Luego la estimación de cada historia se puede determinar por medio de consenso, democracia o promedio según se haya definido en el grupo de trabajo.
- Puño de cinco: en esta técnica se usan los dedos de la mano para expresar la estimación de cada miembro del equipo, al igual que planning póker se parte de una explicación inicial de la historia y se procese a realizar la votación, en caso de presentarse valores atípicos se procede a una sesión corta donde se justifican los valores asignados para posteriormente realizar una nueva votación.

C. ACTIVIDADES DIDÁCTICAS (OPCIONALES SI SON SUGERIDAS)

DESCRIPCIÓN DE ACTIVIDAD DIDÁCTICA	
Nombre de la Actividad	Emparejamiento entre término y definición
Objetivo de la actividad	Afianzar algunos de los conceptos más importantes asociados a las metodologías de desarrollo de software.
Tipo de actividad sugerida	Arrastrar y soltar el término con la definición que corresponde
Archivo de la actividad (Anexo donde se describe la actividad propuesta)	Anexo documento en Word llamado Actividad didáctica 1

D. MATERIAL COMPLEMENTARIO:

Tema	Referencia APA del Material	Tipo de material (Video, capítulo de libro, artículo, otro)	Enlace del Recurso o Archivo del documento o material
Marcos de trabajo tradicionales	Henao, C. (2018, June 9). *#1. Que son las metodologias tradicionales en el desarrollo de software* [Video]. YouTube. https://www.youtube.co	Video de Youtube	https://youtu.be/i8CPD1dW88k

	m/watch?v=i8CPD1dW88k&feature=youtu.be		
Marcos de trabajo Ágiles	Henao, C. (2018b, June 20). *#2. Que son las metodologias ágiles en el desarrollo de Software* [Video]. YouTube. https://www.youtube.com/watch?v=fHKsufzM7qQ&feature=youtu.be	Video de Youtube	https://youtu.be/fHKsufzM7qQ
SCRUM	Henao, C. (2018c, June 27). *#3. SCRUM en 6 minutos | Metodologías Ágiles* [Video]. YouTube. https://www.youtube.com/watch?v=HhC75IonpOU&feature=youtu.be	Video de Youtube	https://youtu.be/HhC75IonpOU

E. GLOSARIO:

TÉRMINO	SIGNIFICADO
Iteración	Hace referencia a un ciclo limitado por tiempo donde se ejecutan actividades de análisis, diseño, construcción y pruebas.
Product Owner	Rol central de Scrum encargado de la gestión de la pila de producto y representante del cliente dentro del grupo de trabajo.
Scrum Master	Rol central de Scrum encargado de facilitar el trabajo del equipo de desarrollo y de mantener la aplicación del marco de trabajo SCRUM.
Stakeholders	Persona, organización o empresa interesada en el desarrollo del proyecto.

F. REFERENCIAS BIBLIOGRÁFICAS:

- Beck, K., & Andres, C. (2004b). *Extreme Programming Explained: Embrace Change, 2nd Edition (The XP Series)* (2nd ed.). Addison-Wesley.

- Jeffries, R. (2011). *What is Extreme Programming?* Ronjeffries.Com. https://ronjeffries.com/xprog/what-is-extreme-programming/
- Kruchten, P. (2003). The Rational Unified Process: An Introduction (3rd Edition) (3rd ed.). Addison-Wesley Professional.
- Maida, EG, Pacienzia, J. (2015). Metodologías de desarrollo de software [en línea]. Tesis de Licenciatura en Sistemas y Computación. Facultad de Química e Ingeniería "Fray Rogelio Bacon". Universidad Católica Argentina, 2015. Disponible en: https://bit.ly/3hJMwXP [Fecha de consulta: 17-sep-2021]
- *Manifiesto por el Desarrollo Ágil de Software*. (2001). Agilemanifesto.Org. https://agilemanifesto.org/iso/es/manifesto.html
- Martin, J. (1991). *Rapid Application Development*. Macmillan Coll Div.
- Royce, W.W. (1970) Managing the Development of Large Software Systems. Proceedings of IEEE WESCON, 26, 328-388.
- SCRUMstudy. (2013). *A Guide to the Scrum Body of Knowledge (SBOK Guide)* (2013th ed.). VMEdu Inc.
- Sommerville, I., Galipienso, M. I. A., & Martinez, A. B. (2005). *Ingenieria del Software*. Pearson Educacion.

G. CONTROL DEL DOCUMENTO

	Nombre	Cargo	Dependencia	Fecha
Autor (es)	**Jonathan Guerrero Astaiza**	**Experto temático**	**Regional Cauca – centro de teleinformática y producción industrial**	**Septiembre de 2021**

H. CONTROL DE CAMBIOS

(Diligenciar únicamente si realiza ajustes a la Unidad Temática)

	Nombre	Cargo	Dependencia	Fecha	Razón del Cambio
Autor (es)					