



PROCESO DE GESTIÓN DE FORMACIÓN PROFESIONAL INTEGRAL FORMATO GUÍA DE APRENDIZAJE

IDENTIFICACIÓN DE LA GUIA DE APRENDIZAJE

- Denominación del Programa de Formación: **Técnico en Programación de Software**
- Código del Programa de Formación: **233104 v.2**
- Nombre del Proyecto: **DESARROLLO Y PROGRAMACIÓN DE SOFTWARE A LA MEDIDA EN LAS INSTITUCIONES EDUCATIVAS EN EL DEPARTAMENTO DE CÓRDOBA**
- Fase del Proyecto: Fase I - Análisis
- Actividad de Proyecto: **DETERMINAR LAS ESPECIFICACIONES Y NECESIDADES PARA EL DESARROLLO DEL SOFTWARE**
- Competencia: **220501092 - ESTABLECER REQUISITOS DE LA SOLUCIÓN DE SOFTWARE DE ACUERDO CON ESTÁNDARES Y PROCEDIMIENTO TÉCNICO**
- Resultados de Aprendizaje Alcanzar: **RAP 2. INTERPRETAR EL INFORME DE REQUISITOS DE LA SOLUCIÓN CONFORME A LA IDENTIFICACIÓN DE LAS NECESIDADES DEL NEGOCIO.**
- Duración de la Guía: **96 horas**

2. PRESENTACIÓN

En el mundo del desarrollo del software, antes de construir dicho sistema hay que planearlo bien y esta planeación no las da los requerimientos, esos datos de gran relevancia que nos da el cliente que va a obtener dicho sistema; es decir así como un ingeniero civil necesita de planos y un diseño arquitectónico antes de construir la torre. Igualmente el programador necesita antes de construir software dichos planos de este, que vienen hacer las características, los datos a procesar, el tipo de software, entre otros; te invitamos a conocer de manera mas profunda todo lo que conlleva a la recolección de requisitos de software en esta guía.



3. FORMULACIÓN DE LAS ACTIVIDADES DE APRENDIZAJE

3.1 Actividad de Reflexión inicial.

En la actualidad el área de desarrollo de software ha tomado gran relevancia, y es por la facilidad de como los softwares, aplicaciones o plataformas, agilizan, controlan y le dan seguridad a los procesos y datos, vemos las grandes ventajas de la tecnología aplicadas a la sistematización de procesos u operaciones. Es por eso que cada vez mas empresas migran sus procesos a la realización de un software, ya sea web o de escritorio, ha sabiendas que este nuevo elementos va a permitir no solo agilizar sus procesos si no también un crecimiento económico o reducción de costos.

Por tal motivo antes de crear un software es necesario recolectar cierta información de lo que va hacer este; a lo que llamamos requerimientos, en el siguiente video conocerás un poco a cerca de este proceso ANTES DE pasar a construir el software

<https://www.youtube.com/watch?v=Hedi-Qobgbc>

Posteriormente responda los siguientes interrogantes

- ¿Qué se entiende por requerimientos de software y cual es su importancia
- ¿Cual es el propósito de los requerimientos

Nota: Este ejercicio tiene como finalidad orientar y motivar en el desarrollo de los temas de esta actividad de aprendizaje, por tal razón no es calificable. No obstante, es primordial realizarlo.

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

Identificar las generalidades sobre los conceptos:

- Estándares IEEE 830: Especificación de requisitos, conceptos y características
- Plantilla ERS, actividades y características.

En el ambiente de aprendizaje, el instructor iniciará una lluvia de ideas con base a los temas anteriormente mencionados, para que los aprendices identifiquen los conocimientos que poseen de estos temas.

Para esta actividad los aprendices tendrán un tiempo de 1 hora.



3.3 Actividades de apropiación del conocimiento (conceptualización y teorización)

3.3.1 Foro temático: Especificación de Requisitos de Software

Este foro es un espacio que está diseñado para que usted investigue sobre la aplicabilidad De ERS. Que es especificación de software , que es un software entre otros

Apoye del material de estudio: (Ingeniería del Software - Especificaciones de Requisitos de Software - Raquel Martínez España)

<https://www.youtube.com/watch?v=gagdT0gJyqA>

Especificación de Requisitos de software

- Especificación
 - Documento que define de forma completa, precisa y verificable, los requisitos, el diseño y el comportamiento u otras características de un sistema o componente de un sistema.
- Software
 - Instrucciones (programas), estructuras de datos para facilitar la manipulación la información y documentos que describen la operación y uso de los programas.
- ERS se puede definir como la documentación de requisitos esenciales del software y de sus interfaces externas.



Posteriormente participe con sus comentarios frente a los temas planteados a continuación:

1. Que es especificación de requisitos, que es software.
2. Importancia de los requerimientos, ventajas y otros conceptos relevantes vistos en el anterior video.

Posteriormente usted debe interpretar las publicaciones de sus compañeros frente a la temática sugerida y responderles a dos de los comentarios, indicando su posición crítica frente al concepto de su compañero.

Para realizar su participación en el foro usted debe ingresar a la plataforma o metodología indicada por el instructor.

3.3.2 Clases magistrales, conceptos y características análisis de requerimientos, IEEE 830

El instructor realizará la presentación de los conceptos relacionados con el resultado a alcanzar, estándar IEEE 830 conceptos y características, análisis de requisitos, requerimientos funcionales y no funcionales, técnicas de análisis y definición, características.

Material web para complementar los conceptos:

IEEE 830 conceptos y características:

<https://www.fdi.ucm.es/profesor/gmendez/docs/is0809/ieee830.pdf>

http://virtual.umng.edu.co/distancia/ecosistema/odin/odin_desktop.php?path=Li4vb3ZhcY9pbmdlbmllcmllhX2luZm9ybWF0aWNhL2luZ2VuaWVyaWFfZGVfcmVxdWVyaW1pZW50b3MvdW5pZGFkXzMv#slide_1

Importancia de los requerimientos:

<https://www.northware.mx/blog/requerimientos-en-el-desarrollo-de-software-y-aplicaciones/>

Requerimientos funcionales y no funcionales:

<https://www.youtube.com/watch?v=tF88eNhNSb4>
<https://www.youtube.com/watch?v=Lv7XbZtnQ6A>

De esta forma se realizarán talleres, investigaciones, actividades propuestas por el instructor en donde alcancen y apropien los conceptos y material estudiado. en estas actividades los aprendices las desarrollaran en equipos de trabajo y presentaran las soluciones gráficas y descritas de los talleres en plenaria.



Evidencia: en esta actividad los aprendices individualmente deben realizar una prueba de conocimiento con los temas mencionados en esta actividad, la cual será propuesta por el instructor.

3.3.3 Desarrollo Principios de la Ingeniería de Requerimientos

El instructor realizará la presentación y sustentación sobre los principios de la ingeniería de requerimientos con la ayuda del siguiente video:
<https://www.youtube.com/watch?v=fLcWpOY3h7k>

los estudiantes complementaran con lo anteriormente aprendido y resaltarán aun más la importancia de todo este proceso antes de pasar a la construcción del software.

REQUERIMIENTOS - ESPECIFICACIÓN - DISEÑO - PROGRAMA

Los 4 principios :

- El negocio de tras del sistema, es el criterio para incluir, rechazar y priorizar requerimientos
- Los requerimientos están en el dominio de la aplicación (sector industrial en el cual se va a construir el software)
- Los requerimientos no funcionales, también son requerimientos
- Ir a la reunión con el cliente con un prototipo

El instructor les suministrará la siguiente **actividad** :

Los aprendices en equipos de trabajo (grupo de 4 aprendices) realizarán un ensayo sobre las siguientes temáticas :

- Que son los requerimientos funcionales y no funcionales, importancia y ejemplos de estos
- Principios de la ingeniería de requerimientos.

resaltando lo más relevantes de los conceptos y las temáticas vistas, el ensayo debe tener la presentación de los aprendices y estar en normatividad APA, enviar al correo del instructor o plataforma territorio SENA.



3.3.4 Desarrollo de temáticas: Plantilla ERS, Técnicas de análisis

El instructor realizará la presentación y sustentación, como redactar un documentó de requisitos de software, apoyándose del siguiente sitio web y video para complementar el objetivo del aprendizaje. Los aprendices deberán entrar a estos recursos para ser estudiados.

Cómo redactar un documento de requisitos de software:

<https://asana.com/es/resources/software-requirement-document-template>

Ver el siguiente video: (Como crear un documento de Especificación de Requerimientos de Software)

<https://www.youtube.com/watch?v=AotyBHVkp8I>

Posteriormente los aprendices participaran con sus aportes frente a los temas planteados a través de una mesa redonda y en grupos de 4 aprendices, tomando un vocero de cada grupo el cual hará su aporte de acuerdo a las siguientes preguntas :

¿Qué es un documento de especificación de requisitos de software?

¿Por qué conviene usar una especificación de requisitos de software?

¿Qué incluir en una especificación de requisitos de software?

¿Que es una Plantilla para documentos de requisitos de software?

Tiempo de la actividad : 4 horas

Plantillas para documento de especificación de requisitos de software - ERS(Link descarga)

<https://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/456>

<https://assets.asana.biz/m/f5d651dad259045/original/software-requirement-document-template-es.pdf>

TÉCNICAS DE LEVANTAMIENTO DE REQUERIMIENTOS DE SOFTWARE

Una etapa fundamental en proyectos de ingeniería de software, es la identificación y documentación de los requerimientos del futuro sistema al comienzo del proyecto, pues en numerosas ocasiones se ha demostrado que es cuando pueden prevenirse errores que puedan significar el fracaso del proyecto. (PMOinformatica.com, 2016)

El instructor hará la presentación sobre técnicas de requerimientos de software, a través de clase magistral y apoyándose en el material de este sitio web:



<http://www.pmoinformatica.com/2016/08/tecnicas-levantamiento-requerimientos.html>

los aprendices en grupos de trabajo investigaran, que otras técnicas o métodos de recolección existen, aparte de los expuestos por el instructor SENA.

Posteriormente en grupo de 4 apéndices, realizaran una mapa mental acerca de las técnicas de análisis y obtención de requerimientos, este lo deben presentar de forma digital y enviar al correo del instructor o subirlo en plataforma territorio.

3.3.5 Identificar Lenguaje Estándar de Modelado (UML)



Identificar que es UML, tipos características , diseños

- Lenguaje estándar de modelado de sistemas orientados a objetos
- Estándar actual para análisis y diseño (UML)
- Diagramas UML



Los aprendices en compañía del instructor realizarán la lectura de la Introducción y observarán críticamente el vídeo propuesto. La importancia de UML en el desarrollo de sistemas <https://www.youtube.com/watch?v=tOayFoHRn-q>

Posterior a esta actividad; en compañía del instructor tutor los aprendices por grupos de trabajo expresaran sus ideas y conclusiones a cerca del video visto.

Duración: 4 horas

El docente formador realizará charlas magistrales donde expondrá con claridad a los alumnos los diversos temas relacionados con:

- DISEÑO DE DIAGRAMAS DE CASOS DE USO
- DISEÑO DE DIAGRAMAS DE OBJETOS
- DISEÑO DE DIAGRAMAS SECUENCIA
- DISEÑO DE DIAGRAMAS DE CLASES
- DISEÑO DE DIAGRAMAS DE PAQUETES
- DISEÑO DE DIAGRAMAS DE DESPLIEGUE
- DISEÑO DE DIAGRAMAS DE COLABORACIÓN
- MANEJO DE HERRAMIENTAS CASE (START UML, MINDOMO, CANVA, GOCONQR, ENTRE OTRAS)



Los aprendices conformados en grupos de 4 desarrollaran un mapa mental de los siguientes diagramas:

- **DISEÑO DE DIAGRAMAS DE CLASES**
- **DISEÑO DE DIAGRAMAS DE CASOS DE USO**
- **DISEÑO DE DIAGRAMAS DE OBJETOS**

Posterior a esto el moderador del grupo socializara con sus compañeros, el trabajo realizado Con las debidas correcciones del instructor tutor.

Duración: 16 horas

3.4 Actividad de Transferencia del Conocimiento



Construir el informe de requerimientos de software para evidenciar las actividades desarrolladas en el proceso formativo.

El instructor entregará a través de la plataforma virtual, un caso de estudio sobre un informe técnico de diseño, para que los aprendices desarrollen los conocimientos adquiridos en un informe sobre los requerimientos que se tienen del software, desde el punto de vista técnico, argumentando los conceptos y temáticas que se deben abordar para su solución, incluyendo los diagramas de UML, requerimientos funcionales y no funcionales y los conceptos de fundamentación en programación de software.

Evidencia de desempeño de producto: Informe con los requerimientos necesarios para iniciar el proceso de construcción del software

Duración: 10 horas



4. ACTIVIDADES DE EVALUACIÓN

Tome como referencia la técnica e instrumentos de evaluación citados en la guía de Desarrollo Curricular

Evidencias de Aprendizaje	Criterios de Evaluación	Técnicas e Instrumentos de Evaluación
Evidencias de Conocimiento: Evidencias de Desempeño Evidencias de Producto: Construir el informe de requerimientos de software con base a estándares y principios Diseña Diagramas UML (caso de usos, de clases y de objetos) para representar el modelado del sistema, utilizando herramientas CASE.	UTILIZA LOS ESTÁNDARES DE ESPECIFICACIÓN DE REQUISITOS DE ACUERDO CON LAS CARACTERÍSTICAS DEL PROYECTO. DOCUMENTA LA ESPECIFICACIÓN DE REQUISITOS SEGÚN LAS NECESIDADES DEL CLIENTE	Técnicas Instrumentos de Evaluación: Prueba de conocimiento Instrumentos de Evaluación: Lista de verificación



5. GLOSARIO DE TÉRMINOS

HERRAMIENTAS CASE: *Computer Aided Software Engineering*, Ingeniería de Software Asistida por Computadora. Son diversas aplicaciones informáticas o programas informáticos destinadas a aumentar la productividad en el desarrollo de software reduciendo el costo de las mismas en términos de tiempo y de dinero.

DIAGRAMAS DE CASOS DE USO: se usan para modelar las interacciones del usuario/sistema. Ellos definen el comportamiento, requisitos y restricciones en la forma de scripts o escenarios.

DIAGRAMAS DE OBJETOS: muestran como las instancias de elementos estructurales se relacionan y usan en tiempo de ejecución.

DIAGRAMAS DE MÁQUINA DE ESTADOS: son esenciales para el entendimiento de la condición momento a momento o “estado de ejecución” de un modelo cuando éste se ejecuta.

DIAGRAMAS SECUENCIA: están estrechamente relacionados a los diagramas de comunicaciones y muestran la secuencia de mensajes pasadas entre los objetos usando una línea de tiempo vertical.

DIAGRAMAS DE CLASES: definen los bloques de construcción básica de un modelo: los materiales generales, clases y tipos que se usan para construir un modelo completo.

DIAGRAMAS DE PAQUETES: se usan para dividir el modelo en contenedores lógicos o “paquetes” y describen las interacciones entre ellos a un nivel más alto.

DIAGRAMAS DE DESPLIEGUE: muestran la disposición de artefactos significativos dentro de una configuración real.

DIAGRAMAS DE COLABORACIÓN: muestran la red y la secuencia de mensajes de comunicaciones entre objetos en tiempo de ejecución durante una instancia de colaboración.

ARGOUML: es una aplicación de diagramado de UML escrita en Java y publicada bajo la Licencia EPL.



6. REFERENTES BIBLIOGRÁFICOS

- Modelado en Sistemas UML. Popkin Software and Systems. Centro de Recursos de UML en www.popkin.com
- Entendiendo UML: La guía del desarrollador, con una aplicación java basada en web, por Paul Harmon y Mark Watson; Morgan Kauffman Publishers, Inc., 1998 (www.mkp.com/books_catalog/1-55860-465-0.asp).
- ¿Qué es el lenguaje unificado de modelado (UML)?: <https://www.lucidchart.com/pages/es/que-es-el-lenguaje-unificado-de-modelado-uml>
- Manual de UML. Paul Kimmel. MacGraw Hill Profesional.
- Tutorial UML 2.3: http://www.sparxsystems.com.ar/resources/uml2_tutorial.html

7. CONTROL DEL DOCUMENTO

	Nombre	Cargo	Dependencia	Fecha
Autor (es)	Andrés Ramos Morales	Instructor	CCIT	17-04-2023

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

	Nombre	Cargo	Dependencia	Fecha	Razón del Cambio
Autor (es)					