PROCESO DE GESTIÓN DE FORMACIÓN PROFESIONAL INTEGRAL

FORMATO GUÍA DE APRENDIZAJE

1. IDENTIFICACIÓN DE LA GUIA DE APRENDIZAJE

- Denominación del Programa de Formación: Técnico en Programación De Software
- Código del Programa de Formación: 233104 V2
- Nombre del Proyecto Formativo: Creación de soluciones informáticas a la medida, en algunos centros formativos de la región de Risaralda.
- Fase del Proyecto: Análisis
- Actividad de Proyecto Formativo: Identificar los requerimientos del software, a través de un documento que contenga el levantamiento de la información de lo que el usuario requiere.
- Competencia:
    - Establecer requisitos de la solución de software de acuerdo con estándares y procedimiento técnico
- Resultados de Aprendizaje:
    - Caracterizar la información a recolectar de acuerdo con las necesidades del cliente.
    - Interpretar el informe de requisitos de la solución conforme a la identificación de las necesidades del negocio
- Duración de la Guía de Aprendizaje (horas): 144

2. PRESENTACIÓN

El levantamiento de requerimientos es mucho más que una etapa en el desarrollo de software: es el puente que conecta las ideas del cliente con soluciones tecnológicas innovadoras. ¡Imagina la satisfacción de ser quien traduce las necesidades en realidad! Esta habilidad no solo es esencial para proyectos exitosos, sino también una puerta abierta al crecimiento profesional.

Aprender a levantar requerimientos fortalece competencias clave como la comunicación efectiva, el análisis crítico y la resolución de problemas. Es una experiencia que te reta a pensar como un detective, a explorar problemas complejos y a diseñar soluciones claras y efectivas.

Además, este proceso fomenta el trabajo autónomo, permitiéndote desarrollar independencia en la investigación, y a la vez, promueve el trabajo colaborativo, integrando diversas perspectivas en equipo. Dominar esta etapa no solo mejora tu entendimiento del ciclo de vida del software, sino que también te convierte en un profesional adaptable en un mundo dinámico.

¡Atrévete a tomar el desafío del levantamiento de requerimientos! Cada habilidad que desarrolles en este camino será una herramienta valiosa para destacar en el mundo del desarrollo de software y liderar proyectos de impacto. ¡Es tu momento de brillar y construir soluciones que cambien la vida de las personas!

3. FORMULACIÓN DE LAS ACTIVIDADES DE APRENDIZAJE

En esta guía encontrará diversas actividades de aprendizaje. Para completarlas con éxito, será importante leer, comprender y trabajar en equipo. ¡Comencemos!

3.1 Actividades de reflexión inicial:

Comentado [1]: les recomiendo estructurar actividades. es decir: REflexionar sobre la importancia de los sistemas de sortware al interior de la organización por medio de estudio de caso.

Analizar las necesidades del cliente respecto a las soluciones o los requerimientos solicitados a través de caso de estudio

Descripción de la actividad

Carlos es dueño de una barbería y/o salón de uñas en la cual trabajan 3 personas haciendo los servicios pertinentes y 1 administrador. Se quiere crear un software que permita mejorar el servicio y la imagen del negocio, tanto para conseguir nuevos clientes como para fidelizar los que ya se tienen.

Si deseamos identificar las necesidades y expectativas para establecer los requisitos del software:

¿A quiénes deberíamos entrevistar?

¿Qué pregunta(s) deberíamos hacer a cada uno según su rol?

Responda de manera individual.

Posteriormente, en equipos de 3 participantes redactarán un informe escrito final, donde condensa y resume las respuestas de cada uno, para luego ser socializadas al grupo.

Ambiente requerido: Sala de sistemas
Estrategias o técnicas didácticas activas: Estudio de caso
Materiales de formación: computador
Material de apoyo: documentos suministrados por el instructor e internet
Duración de la actividad: 5 horas.

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

Analizar situaciones reales de su comunidad que pueden ser abordadas a través de técnicas de elicitación

Descripción de la actividad:

1. Consulta el documento "Técnicas de Elicitación.pdf" que se encuentra en la plataforma.
2. De manera grupal, identificar un problema en tu colegio o comunidad y plantear una posible solución mediante una aplicación de software.
3. Utilizar el método que mejor consideren para recopilar la información relevante, asegurando adaptarla a las necesidades de la institución y/o comunidad.
4. Una vez que hayan reunido los datos, elaborar una presentación con un informe de requisitos en función de las necesidades.
5. Posteriormente, realizar una socialización en plenaria en la que presenten el problema identificado, el método empleado para recopilar la información y la solución propuesta.
   ¡Es la oportunidad de compartir ideas y proponer una solución innovadora para tu colegio o comunidad! ¡Anímate y demuestra tu talento!

Ambiente requerido: Sala de sistemas

Estrategias o técnicas didácticas activas: Aprendizaje basado en problemas

Materiales de formación: computador

Material de apoyo: Técnicas de Elicitación.pdf

Duración de la actividad: 5 horas.

3.3 Actividades de apropiación:

3.3.1 Reconocer diferentes técnicas de desarrollo de software a partir de requerimientos e historia de usuario y necesidades del sector productivo.

3.3.1.1. Reconocer la importancia de los requisitos de software a través de video

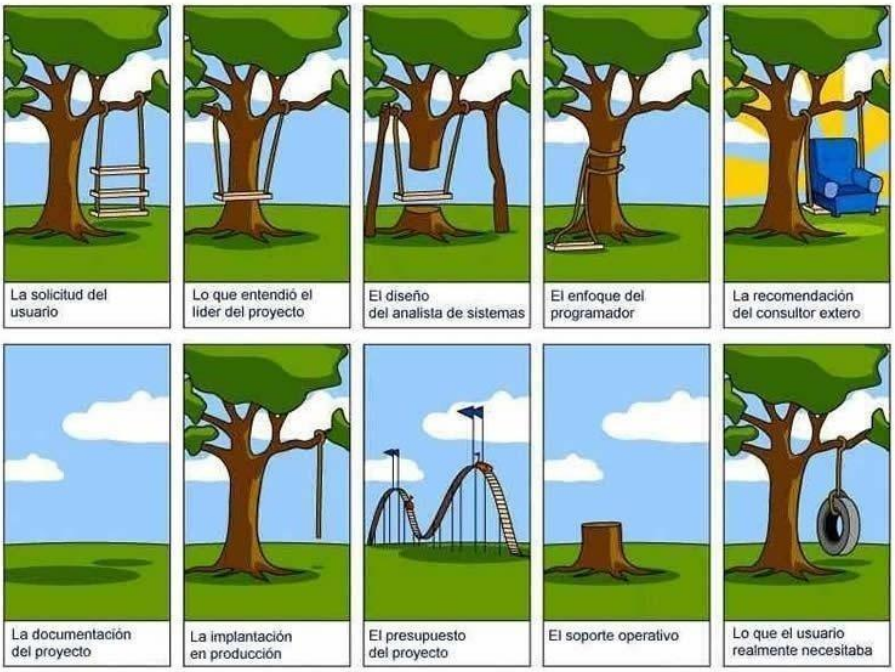
Descripción de la actividad

Aprendices, llegamos al punto donde haremos el levantamiento de los requerimientos técnicos.

Por favor dirígete al siguiente video: https://www.youtube.com/watch?v=rINH97CcWPc

La importancia de los requisitos del software son claves para el proceso de desarrollo de software, vamos a mirar un poco más de cerca el problema.

- Consulta en internet el significado de la palabra ELICITACIÓN y Stakeholders.
- Observa detenidamente la siguiente imagen.



En una hoja o documento de word contesta las siguientes incógnitas:

1. Cual es el problema que notas entre cada una de las personas interesadas y el resultado que tuvo el columpio.

Persona interesada	**Problema**

2. ¿Qué cosas harías, teniendo en cuenta el video visto, para que esta situación mejorará?
3. ¿Crees que el famoso dicho: “El cliente siempre tiene la razón” aplica en el desarrollo de software?, ¿si?,¿no?, ¿por qué?
4. Ahora ponte en los zapatos de alguna de las personas interesadas en la construcción del columpio, ¿Cómo te sentirías?

Ubicarse en el grupo que organizó el instructor

Comenta tu punto de vista con el grupo y escribe una conclusión final teniendo en cuenta la opinión de tus compañeros y la intervención realizada por el instructor alrededor del tema.

Ambiente requerido: Sala de sistemas

Estrategias o técnicas didácticas activas: Aprendizaje basado en problemas

Materiales de formación: computador

Material de apoyo: referenciar los docs documentos suministrados por el instructor e internet

Evidencias de aprendizaje: Participa de manera activa en el foro

Instrumentos de evaluación: Lista de chequeo

Duración de la actividad: 10 horas.

Comentado [2]: Tener en cuenta: Esta no es la información del formato, faltan evidencias e instrumentos de evaluación que tienen las actividades 3.3. y 3.4

3.3.1.2. Identificar los componentes en el ciclo de vida del software a partir de procesos de investigación

Descripción de la actividad

Llegó la hora de leer, dirígete al material de apoyo "Introducción elicitación.pdf", la carpeta "elicitación" y el documento llamado "Elicitación de requerimientos.pdf".

Investiga de manera individual otras técnicas de elicitación de requerimientos.

Luego de esto, realiza un cuadro comparativo de ventajas y desventajas de todas las técnicas incluyendo la entrevista, justifica los motivos por los cuales considera cuál es la técnica más apropiada, entrega la actividad resuelta en un documento, según indicaciones del instructor.

Ambiente requerido: Sala de sistemas

Estrategias o técnicas didácticas activas: Aprendizaje basado en problemas

Materiales de formación: computador

Material de apoyo: documentos suministrados por el instructor e internet

Evidencias de aprendizaje: Cuadro comparativo diligenciado

Instrumentos de evaluación: Lista de chequeo

Duración de la actividad: 10 horas.

Comentado [3]: la misma recomendación de 3.3

3.3.2 Aplicar técnicas de desarrollo de software en ambiente simulado o real que permita validar el funcionamiento de la propuesta

3.3.2.1 Diseñar técnica de elicitación a partir de las necesidades del cliente

Descripción de la Actividad:

1. Basado en problema encontrado, actividad 3.2, utilizar una de las técnicas de elicitación que consideren más apropiado para el proceso de recolección de la información.
2. Construir métodos para la recolección de la información como entrevistas, escenarios, prototipos, reuniones moderadas u observación, utilizar 1 o más técnicas de elicitación si lo consideras necesario.
3. Entregar la actividad resuelta en un documento, según indicaciones del instructor.

Ambiente requerido: Sala de sistemas

Estrategias o técnicas didácticas activas: Aprendizaje basado en problemas

Materiales de formación: computador

Material de apoyo: documentos suministrados por el instructor e internet

Evidencias de aprendizaje: Documento Normas APA

Instrumentos de evaluación: Lista de chequeo

Duración de la actividad: 14 horas.

3.3.2.2 Realizar documento de conceptualización de requerimientos teniendo en cuenta el material de referencia.

Descripción de la actividad:

1. El instructor le invita a conformar equipos de trabajo de tres (3) Aprendices.
2. Los aprendices recibirán documento de lectura y apropiación; para su uso durante la actividad.

3. Los equipos de trabajo tendrán 30 minutos para apropiar el documento de lectura.
4. Los aprendices se familiarizan con los conceptos y clasificación de las diferentes metodologías de desarrollo.
5. Posteriormente, cada aprendiz ingresará a la plataforma donde resolverá el cuestionamiento asignado.
6. Siguiendo con este razonamiento, en la parte inferior del documento, los participantes deberán responder a las siguientes preguntas reflexivas:

¿Qué modelo de desarrollo se adapta mejor a las necesidades específicas de mi problema-proyecto? (Considera modelos como Cascada, Ágil, Scrum, Kanban, etc.)
¿Cómo afecta la elección del modelo de desarrollo a la flexibilidad y adaptabilidad del problema-proyecto? (Reflexiona sobre la capacidad de respuesta a cambios en los requisitos.)
¿Qué riesgos están asociados con el modelo de desarrollo elegido? (Identifica posibles desafíos y cómo mitigarlos).
¿Cómo se gestionan las pruebas y la calidad del software en el modelo seleccionado? (Evalúa la importancia de las pruebas en cada etapa del desarrollo.)
¿Qué papel juega la comunicación en el modelo de desarrollo que estamos utilizando? (Analiza la efectividad de la comunicación entre equipos y partes interesadas.)
¿Cómo se recopilan y documentan los requisitos de manera efectiva? (Considera métodos como entrevistas, encuestas y prototipos.)

¿Qué técnicas se utilizan para priorizar los requisitos?
- Reflexiona sobre el uso de matrices de priorización o técnicas como MoSCoW.

¿Cómo se gestionan los cambios en los requisitos durante el ciclo de vida del proyecto?
- Evalúa el proceso de gestión de cambios y su impacto en el desarrollo.

¿Qué tan bien se comunican los requisitos a todos los miembros del equipo?
- Considera la claridad y comprensión de los requisitos por parte de todos los involucrados.

¿Cómo se valida que los requisitos se están cumpliendo durante el desarrollo?
- Reflexiona sobre las técnicas de validación y verificación utilizadas.

¿Cómo se equilibran las expectativas de los stakeholders con las limitaciones del proyecto?

- Considera la gestión de expectativas y la comunicación continua.

¿Qué lecciones se pueden aprender de proyectos anteriores sobre la gestión de requisitos y modelos de desarrollo?

- Reflexiona sobre la experiencia pasada y su aplicación en proyectos futuros.

¿Cómo se asegura la sostenibilidad y escalabilidad del software en el modelo elegido?

- Piensa en la planificación a largo plazo y la arquitectura del software.

Durante esta actividad, el instructor brindará apoyo a cada equipo de trabajo, guiándolos y resolviendo dudas para garantizar la correcta aplicación de lo aprendido.

Ambiente requerido: Sala de sistemas

Estrategias o técnicas didácticas activas: Aprendizaje basado en problemas y solución de preguntas.

Materiales de formación: computador

Material de apoyo: Referenciar documentos suministrados por el instructor e internet

Evidencias de aprendizaje: Documento Normas APA-Solución del cuestionario en un archivo

Instrumentos de evaluación: Cuestionario

Duración de la actividad: 4 horas.

3.4 Actividades de Transferencia el Conocimiento:

Presentar documento de levantamiento de información de requerimientos del software que contenga las necesidades del usuario.

Taller de Levantamiento de Información de Requerimientos

Desarrollar y presentar un documento que recoja de manera efectiva los requisitos del software, asegurando que se alineen con las necesidades y expectativas de los usuarios.

Descripción de la actividad:

Simulación de Entrevistas (1:30 hora)

- Formación de Grupos: Dividir a los participantes en grupos pequeños.
- Asignación de Roles: Un grupo actúa como entrevistadores y otro como usuarios.
- Simulación: Realizar una entrevista de requisitos sobre un software ficticio.
- Debriefing: Cada grupo comparte sus experiencias y aprendizajes.

Creación de Encuestas (1:30 hora)

- Diseño de Encuestas: Cada grupo debe crear una encuesta para recoger requisitos de usuarios sobre un software.
- Discusión: Compartir las encuestas creadas y recibir retroalimentación de otros grupos.

Análisis de Requisitos (30 minutos)

- Revisión de Ejemplos: Analizar ejemplos de documentos de requisitos.
- Identificación de Mejores Prácticas: Discutir qué hace que un documento de requisitos sea efectivo.

Cierre y Reflexión (30 minutos)

- Reflexión Grupal:¿Qué aprendieron? ¿Cómo aplicarán estos conocimientos en sus proyectos?
- Feedback del Taller: Recoger opiniones sobre la actividad.

Ambiente requerido: Sala de sistemas

Estrategias o técnicas didácticas activas: Aprendizaje basado en problemas

Materiales de formación: computador

Material de apoyo: documentos suministrados por el instructor e internet

Evidencias de aprendizaje: Documento Normas APA-

Duración de la actividad: 4 horas.

4. PLANTEAMIENTO DE EVIDENCIAS DE APRENDIZAJE PARA LA EVALUACIÓN EN EL PROCESO FORMATIVO.

Fase del proyecto formativo	Actividad del proyecto formativo	Actividad de Aprendizaje	Evidencias de Aprendizaje	Criterios de Evaluación	Técnicas e Instrumentos de Evaluación
Análisis	Identificar los requerimientos del software, a través de un documento que contenga el levantamiento de la información de lo que el usuario requiere.	Reconocer diferentes técnicas de desarrollo de software a partir de requerimientos e historia de usuario y necesidades del sector productivo.	Recolección de información necesaria para identificar los requerimientos utilizando técnicas o métodos de elicitación Descripción de la evidencia: Cada aprendiz debe construir métodos para la recolección de la información como entrevistas, cuestionarios, encuestas.	●Determina las técnicas de elicitación de acuerdo con las necesidades del cliente. ●Aplica los instrumentos de recolección de información según las técnicas seleccionadas	Cuestionario Evaluación de conocimiento Lista de chequeo Evaluación de Desempeño
		Aplicar técnicas de desarrollo de software en ambiente simulado o real que permita validar el funcionamiento de la propuesta	Ficha técnica historia de usuario Descripción de la evidencia: Construir la ficha técnica precisando las necesidades del usuario como número, nombre de usuario, nombre de la historia, objetivo del requerimiento, prioridad en negocio, iteración asignada, programador responsable ,validación y demás descripciones contenidas en el formato.	●Utiliza los estándares de especificación de requisitos de acuerdo con las características del proyecto. ●Documenta la especificación de requisitos según las necesidades del cliente.	Lista de chequeo Valoración de producto

5. GLOSARIO DE TÉRMINOS

- **Requerimientos Funcionales:** Describen las funciones específicas que un sistema debe realizar. Por ejemplo, "el sistema debe permitir a los usuarios registrarse utilizando su correo electrónico".

- **Requerimientos No Funcionales**: Especifican las características de calidad del sistema, como rendimiento, seguridad, escalabilidad y facilidad de uso. Por ejemplo, "el sistema debe cargar en menos de 2 segundos".

- **Stakeholders:** Son todas las personas o grupos interesados en el proyecto. Esto incluye usuarios finales, clientes, desarrolladores, gerentes y otros involucrados.

- **Elicitación:** Es el proceso de obtener, analizar y documentar los requerimientos de un sistema a través de técnicas como entrevistas, encuestas y prototipos.

- **Análisis de Requerimientos:** Consiste en examinar y detallar los requerimientos obtenidos durante la elicitación para asegurarse de que sean claros, completos y alcanzables.

- **Validación de Requerimientos:** Proceso de confirmar que los requerimientos reflejan correctamente las necesidades de los stakeholders y que son realizables técnicamente.

- **Caso de Uso:** Un documento o diagrama que describe cómo un usuario interactuará con el sistema para cumplir un objetivo específico. Ejemplo: "El usuario inicia sesión y accede a su perfil".

- **Historia de Usuario:** Una descripción breve y sencilla de la funcionalidad que un usuario necesita, escrita desde su perspectiva. Formato típico: "Como [tipo de usuario], quiero [acción o funcionalidad] para [lograr un objetivo]".

- **Entrevista:** Técnica de elicitación que consiste en realizar preguntas estructuradas o no estructuradas a los stakeholders para obtener información sobre sus necesidades y expectativas.

- **Encuesta o Cuestionario:** Una herramienta para recopilar información estructurada de varios usuarios o stakeholders, ideal para proyectos con un gran número de participantes.

- **Prototipo:** Representación visual o funcional preliminar de un sistema o aplicación, utilizada para obtener retroalimentación temprana de los usuarios.

- **Observación Directa:** Técnica que implica observar cómo los usuarios realizan sus tareas cotidianas para identificar oportunidades de mejora o necesidades específicas.

- **Requisitos Ambiguos:** Requerimientos que no están claramente definidos, lo que puede llevar a malentendidos o problemas durante el desarrollo del sistema.

- **Priorización de Requerimientos:** El proceso de determinar qué requisitos son más importantes o urgentes para implementarse en el proyecto.

- **Documento de Requerimientos:** Un informe detallado que recopila todos los requerimientos del sistema, incluyendo funcionales y no funcionales, para su aprobación y seguimiento.

- **Sesión de Brainstorming:** Reunión creativa para generar ideas y posibles soluciones en equipo, frecuentemente utilizada para explorar opciones iniciales de funcionalidades o características.

- **Alcance del Proyecto:** Define los límites del sistema, especificando qué funcionalidades se incluirán y qué aspectos están fuera del alcance.

- **Backlog de Producto:** Lista priorizada de tareas, funcionalidades y requerimientos pendientes en un proyecto ágil, que sirve como guía para el desarrollo.

- **Diagramas de Flujo:** Representaciones gráficas que muestran los pasos o procesos de un sistema, utilizados para entender y comunicar cómo funcionará.

- **Iteración:** Un ciclo repetitivo dentro del proceso de desarrollo en el que se revisan y refinan los requerimientos y las soluciones.

6. REFERENTES BILBIOGRÁFICOS

- Boehm, B., & Ross, R. (1989). *Theory-W Software Project Management: Principles and Examples*. IEEE Transactions on Software Engineering, 15(7), 902–916. https://doi.org/10.1109/TSE.1989.5070455

- Sommerville, I. (2015). *Software Engineering* (10th ed.). Pearson.

- Pressman, R. S., & Maxim, B. R. (2019). *Software Engineering: A Practitioner's Approach* (9th ed.). McGraw-Hill Education.
- IEEE. (2014). *IEEE Standard for Software Requirements Specifications (IEEE 29148-2014)*. IEEE.

- Wiegers, K., & Beatty, J. (2013). *Software Requirements* (3rd ed.). Microsoft Press.

- Karlsson, J. (1997). *Software Requirements Prioritizing*. Proceedings of the IEEE International Conference on Requirements Engineering, 110–116. https://doi.org/10.1109/ICRE.1997.602048

- Cohn, M. (2004). *User Stories Applied: For Agile Software Development*. Addison-Wesley.

- Cockburn, A. (2001). *Writing Effective Use Cases*. Addison-Wesley.

- Glinz, M. (2007). *On Non-Functional Requirements*. Proceedings of the 15th IEEE International Requirements Engineering Conference, 21–26. https://doi.org/10.1109/RE.2007.45

- Jacobson, I., Booch, G., & Rumbaugh, J. (1999). *The Unified Software Development Process*. Addison-Wesley.

7. CONTROL DEL DOCUMENTO

	Nombre	Cargo	Dependencia	Fecha
Autor (es)	**Germán Osorio** **Jhon Faber Cortés**	**Instructor**	**Conexión Vocacional**	**26/02/25**

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

	Nombre	Cargo	Dependencia	Fecha	Razón del Cambio
Autor (es)					