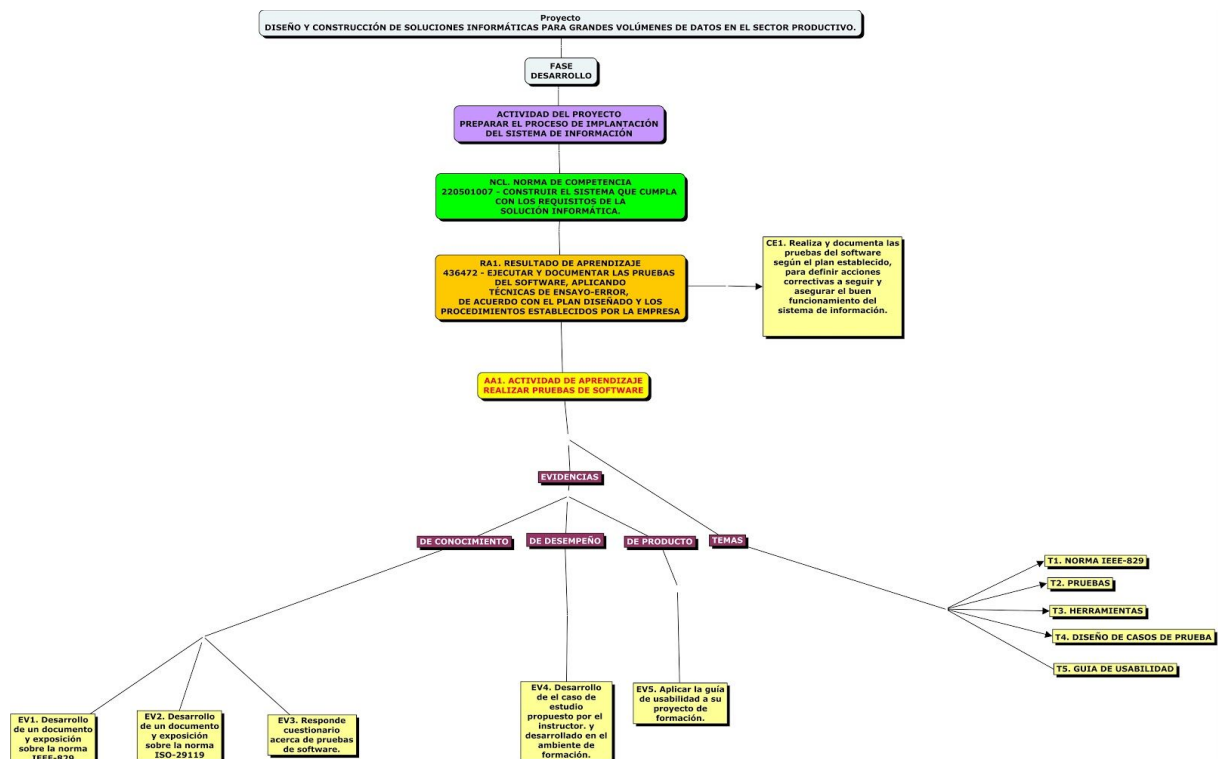


PROCESO DIRECCIÓN DE FORMACIÓN PROFESIONAL INTEGRAL

FORMATO GUÍA DE APRENDIZAJE

IDENTIFICACIÓN DE LA GUÍA DE APRENDIZAJE

- Denominación del Programa de Formación: Análisis y Desarrollo de Sistemas de Información
- Código del Programa de Formación: 228106
- Nombre del Proyecto (si es formación Titulada) DISEÑO Y CONSTRUCCIÓN DE SOLUCIONES INFORMÁTICAS PARA GRANDES VOLUMENES DE DATOS EN EL SECTOR PRODUCTIVO.
- Fase del Proyecto (si es formación Titulada) DESARROLLO
- Actividad de Proyecto (si es formación Titulada) PREPARAR EL PROCESO DE IMPLANTACIÓN DEL SISTEMA DE INFORMACIÓN
- Competencia:
 - 220501007 - CONSTRUIR EL SISTEMA QUE CUMPLA CON LOS REQUISITOS DE LA SOLUCIÓN INFORMÁTICA.
- Resultados de Aprendizaje Alcanzar:
 - 436472 - EJECUTAR Y DOCUMENTAR LAS PRUEBAS DEL SOFTWARE, APLICANDO TÉCNICAS DE ENSAYO-ERROR, DE ACUERDO CON EL PLAN DISEÑADO Y LOS PROCEDIMIENTOS ESTABLECIDOS POR LA EMPRESA
- Duración de la Guía: 44H



2. PRESENTACIÓN



Los principales aspectos para evaluar en un producto software son la funcionalidad (hace lo que debe), la Fiabilidad (resistente a fallos), y el Rendimiento (lleva a cabo su trabajo de manera efectiva). Las pruebas pueden hacerse a diferentes niveles dependiendo del objetivo de las mismas, parte de esta información se documenta a través del objeto de aprendizaje “Pruebas de software”, el cual le presenta información sobre diferentes tipos de pruebas y diferentes estrategias para su realización, complementado con un video que contextualiza al respecto de la importancia de las pruebas de software.

“La mayoría de los buenos programadores lo son no porque esperen que se les pague o por la adulación del público, sino porque les divierte programar”

Linus Torvalds.

3. FORMULACIÓN DE LAS ACTIVIDADES DE APRENDIZAJE

3.1 ACTIVIDADES DE REFLEXIÓN INICIAL

Durante la realización de procesos de pruebas es importante mantener las habilidades comunicativas, de análisis, ser metódicos y ordenados, así como mantener unas buenas relaciones interpersonales y en general vivenciar principios éticos en todas nuestras acciones.

Debido a que debemos afianzar estas características se plantea la realización de una autoevaluación donde se reflexionará sobre algunos principios, su interpretación y referentes, lo cual nos llevará a generar un plan de mejoramiento personal.

Esta es una invitación a desarrollar las acciones propuestas con actitud crítica, propositiva y con la seguridad que serán de gran aporte en su vida profesional.

En el desarrollo de software es importante el aseguramiento de la calidad en el producto que se ha desarrollado, teniendo en cuenta aspectos como: funcionalidad, fiabilidad y el rendimiento para esto el aprendiz debe reflexionar mediante un ensayo donde responda lo siguiente:

- ¿Por qué se debe realizar pruebas al proyecto de formación de acuerdo con los estándares internacionales?
- ¿Qué incidencias se presentan al no realizar pruebas a las aplicaciones desarrolladas en el proceso de implantación?
- Investigue sobre las certificaciones internacionales para las pruebas de software.
- ¿Por qué utilizar herramientas de pruebas que automaticen éstas y que resultados puedo obtener?
- Desarrolle una matriz de productos para pruebas de software, donde se describa brevemente su uso, licencia, plataforma.

Reúnanse en equipo con otros compañeros y compartan el conocimiento que tienen respecto a estos temas de reflexión, posteriormente se socializa con el resto del grupo, donde el instructor centralizará y dará soporte a las conclusiones presentadas.

3.2 ACTIVIDADES DE CONTEXTUALIZACIÓN E IDENTIFICACIÓN DE CONOCIMIENTOS NECESARIOS PARA EL APRENDIZAJE.



Antes de entrar a realizar las pruebas al software se hace necesario contextualizar acerca de los siguientes temas:

- Conceptos sobre pruebas de software y estándares internacionales IEEE-829, IEC 29119.
- Niveles
- Tipos
- Enfoques
- Plan de pruebas
- Diseño de casos de prueba
- Herramientas tecnológicas para pruebas
- Documentación de las pruebas (unitarias, carga, estrés, integración).

Para esto, el aprendiz debe realizar el taller de aprendizaje “INTRODUCCIÓN A LAS PRUEBAS DE SOFTWARE” y “SENA TESTING DE SOFTWARE.pdf”, dispuesto en la plataforma. LMS, Leer los siguientes temas:

- Orientaciones Generales.
- Actividades
- Pruebas
- Diseño de casos de prueba
- Depuración de errores
- Documentación de pruebas

Una vez explorado el objeto que presenta las diferentes pruebas, se invita al aprendiz a desarrollar un mapa conceptual, un glosario de términos que relacione los principales conceptos encontrados en el material de estudio.

3.3 ACTIVIDADES DE APROPIACIÓN

Para esto el aprendiz debe realizar la lectura sobre “OVAPruebas.pdf”, donde encontrará los recursos necesario para construir los artefactos necesarios para realizar un prueba de software.

De lo anterior, se puede decir que las pruebas de software son estrictamente necesarias, para determinar de manera dinámica la calidad del software; de esta manera, se garantiza que se ha construido el sistema correcto y de la forma correcta.

3.4 ACTIVIDADES DE TRANSFERENCIA DEL CONOCIMIENTO

De acuerdo a la contextualización realizada sobre pruebas de software, aplicamos sobre un caso de estudio propuesto por el instructor y aplicando el estándar IEEE-829, elaborando los siguientes documentos:



Documento 1. Plan de pruebas

Documento eje sobre el cual se desarrollan las pruebas

Describe:

Alcance

Enfoque

Recursos

Calendarización de actividades de prueba

Identifica los ítems y características a probar

Identifica las tareas de prueba a desarrollar, los responsables de cada tarea y los riesgos asociados

prueba.

Se determina QUÉ necesita ser probado

Se determina cómo sería una prueba exitosa

Se deriva de los requerimientos

Documento 2. Especificación del diseño de

Documento 3. Especificación de casos de prueba.

Valores exactos de entrada y otros que se requieran

Valores exactos de salida y cambios del sistema esperados

Pasos para ejecutar las pruebas

Documento 4. Procedimientos de prueba.

Describe cómo el tester ejecutará físicamente la prueba y los pasos necesarios

Documento 5. Reporte de transmisión de ítems de pruebas.

Describe los ítems para prueba

Dónde encontrarlos y da la aprobación para su liberación

Garantiza al tester de que los ítems están listos para ser probados

Documento 6. Log de pruebas.

Registra los detalles sobre qué casos de pruebas se han ejecutado, en qué orden y sus resultados (pass/fail)

Si hay inconformidades, se levanta o actualiza un reporte de incidentes

Documento 7. Reporte de incidentes de prueba.

Descripción de los detalles encontrados cuando la prueba no pasó

Documento 8. Reporte de pruebas.

Resume la información importante sobre las pruebas:

Evaluación de qué tan bien se realizaron las pruebas

Número de incidentes reportados

Evaluación sobre la calidad del sistema

Gracias a este documento se decide si la calidad del sistema es suficiente para continuar

4. ACTIVIDADES DE EVALUACIÓN

EVIDENCIA 1: Desarrollo de un documento y exposición sobre la norma IEEE-829

Conforma grupo de 4 aprendices y desarrolla un documento acerca de la la norma IEEE-829 y la socializa con los demás aprendices, se evalúa mediante una lista de chequeo dispuesta en la plataforma LMS.

EVIDENCIA 2: Desarrollo de un documento y exposición sobre la norma ISO-29119

Conforma grupo de 4 aprendices y desarrolla un documento acerca de la la norma ISO-29119 y la socializa con los demás aprendices, se evalúa mediante una lista de chequeo dispuesta en la plataforma LMS

EVIDENCIA 3: Responde cuestionario acerca de pruebas de software.

Responde preguntas acerca de pruebas de software.

EVIDENCIA 4: Desarrollo de el caso de estudio propuesto por el instructor. y desarrollado en el ambiente de formación.

Desarrolla el caso de estudio “ALQUIPC”, donde se le aplica las siguientes pruebas:

- Caja Negra
- Caja Blanca
- Regresión
- Unitarias.

Realizar los artefactos necesarios de acuerdo a los estándares de pruebas de software, importante desarrollar el taller “[TALLER PRUEBAS DE CAJA NEGRA, CAJA BLANCA Y REGRESION.pdf](#)”.

EVIDENCIA 5: Aplicar la guía de usabilidad a su proyecto de formación

De acuerdo a la [guía de usabilidad de MINTIC](#), construya una lista de chequeo donde aplique a su proyecto de formación y que sirva para otro proyecto de la ficha,

Evidencias de Aprendizaje	Criterios de Evaluación	Técnicas e Instrumentos de Evaluación
---------------------------	-------------------------	---------------------------------------

<div><div><div>Evidencias de Conocimiento :</div><div><ul style="list-style-type: none">EV1. Desarrollo de un documento y exposición sobre la norma IEEE-829EV2. Desarrollo de un documento y exposición sobre la norma ISO-29119EV3. Responde cuestionario acerca de pruebas de software.</div></div><div><div>Evidencias de Desempeño</div><div><ul style="list-style-type: none">EV4. Desarrollo de el caso de estudio propuesto por el instructor. y desarrollado en el ambiente de formación.</div></div><div><div>Evidencias de Producto:</div><div><div>EV5. Aplicar la guía de usabilidad a su proyecto de formación.</div></div></div></div> <td><div>Realiza y documenta las pruebas del software según el plan establecido, para definir acciones correctivas a seguir y asegurar el buen funcionamiento del sistema de información.</div></td> <td><table><tr><th>Técnicas</th><th>Instrumentos</th></tr><tr><td>Observación Directa</td><td>Lista de chequeo</td></tr><tr><td>Formulación de preguntas</td><td>Cuestionarios</td></tr><tr><td>Valoración de productos</td><td>Listas de verificación o de chequeo</td></tr><tr><td>Simulación de situaciones</td><td>Lista de chequeo</td></tr><tr><td>Solución de Problemas</td><td>Prueba de habilidad y ejercicio practico</td></tr><tr><td>Entrevista</td><td>Cuestionario</td></tr></table><div><div>I1EV1 Lista de verificación.</div><div>I2EV2 Lista de verificación.</div><div>I3EV3. Cuestionarios</div><div>I4EV4 Lista de verificación</div><div>I5EV5 Lista de verificación</div></div></td>	<div>Realiza y documenta las pruebas del software según el plan establecido, para definir acciones correctivas a seguir y asegurar el buen funcionamiento del sistema de información.</div>	<table><tr><th>Técnicas</th><th>Instrumentos</th></tr><tr><td>Observación Directa</td><td>Lista de chequeo</td></tr><tr><td>Formulación de preguntas</td><td>Cuestionarios</td></tr><tr><td>Valoración de productos</td><td>Listas de verificación o de chequeo</td></tr><tr><td>Simulación de situaciones</td><td>Lista de chequeo</td></tr><tr><td>Solución de Problemas</td><td>Prueba de habilidad y ejercicio practico</td></tr><tr><td>Entrevista</td><td>Cuestionario</td></tr></table> <div><div>I1EV1 Lista de verificación.</div><div>I2EV2 Lista de verificación.</div><div>I3EV3. Cuestionarios</div><div>I4EV4 Lista de verificación</div><div>I5EV5 Lista de verificación</div></div>	Técnicas	Instrumentos	Observación Directa	Lista de chequeo	Formulación de preguntas	Cuestionarios	Valoración de productos	Listas de verificación o de chequeo	Simulación de situaciones	Lista de chequeo	Solución de Problemas	Prueba de habilidad y ejercicio practico	Entrevista	Cuestionario
Técnicas	Instrumentos															
Observación Directa	Lista de chequeo															
Formulación de preguntas	Cuestionarios															
Valoración de productos	Listas de verificación o de chequeo															
Simulación de situaciones	Lista de chequeo															
Solución de Problemas	Prueba de habilidad y ejercicio practico															
Entrevista	Cuestionario															

5. GLOSARIO DE TÉRMINOS

- Prueba: Proceso mediante el cual se ejecuta de manera sistemática un conjunto de actividades (métodos y técnicas) para encontrar errores.
- Caso de prueba: Conjunto de condiciones, datos o variables que servirán para determinar si los requisitos del sistema se cumplen de manera parcial, completa, o no se cumplen.

- Error: Discrepancia entre el valor calculado y el valor teórico o esperado, con responsabilidad del desarrollador.
- Defecto software: Desviación en el valor esperado por una cierta característica. Defecto de calidad.
- Fallo: Consecuencia de un error o un defecto software.
- Overflow: Significa desbordamiento en el buffer, cuando la cantidad de datos supera la capacidad pre-asignada. Es un fallo de programación.
- Underflow: Significa subdesbordamiento del buffer, cuando se carga datos a una velocidad inferior a la de procesamiento, provocando bloqueos.

6. REFERENTES BIBLIOGRÁFICOS

- BOLAÑOS, D., SIERRA, A., & ALARCÓN, M. (2008). Pruebas de Software y JUnit. Madrid: Pearson Prentice Hall.
- CATALDI, Z. (2000). Metodología de diseño, desarrollo y evaluación de software educativo. Tesis de Magíster en Informática. Argentina: Facultad de Informática. Universidad Nacional de la Plata (UNLP).
- PRESSMAN, R. (2006). Ingeniería del Software: Un enfoque práctico. Sexta edición. McGrawHill.

<http://compromiso.sena.edu.co/documentos/vista/descarga.php?id=1306>

7. CONTROL DEL DOCUMENTO

	Nombre	Cargo	Dependencia	Fecha
Autor (es)	JOSE FERNANDO GALINDO SUREZ	INSTRUCTOR	CGMLTI	14/01/2020

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

	Nombre	Cargo	Dependencia	Fecha	Razón del Cambio
Autor (es)					