



Programa de Estudios de la Unidad de Aprendizaje: LABORATORIO DE DESARROLLO DE SOFTWARE II																										
Clave: 4FP-FM1067			Créditos: 4.5			Programa Académico: TÉCNICO EN PROGRAMACIÓN																				
						Nivel:						1°	2°	3°	4°	5°	6°									
Ramas de Conocimiento						Unidades Académicas donde se Imparte:																				
Ingeniería y Ciencias Físico Matemáticas	<input checked="" type="checkbox"/>	Ciencias Sociales Administrativas		Ciencias Médico Biológicas		TODAS LAS U.A.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	CET1
Área de Formación Curricular						Tiempos Asignados:																				
Institucional		Científica, Humanística y Tecnológica Básica		Profesional	<input checked="" type="checkbox"/>	Global: <u>72</u> Hrs/18 semanas/Semestre Aula: <u>0</u> Hrs/Semana Total: _____ Hrs/Semestre Taller: <u>0</u> Hrs/Semana Total: <u>0</u> Hrs/Semestre Laboratorio: <u>4</u> Hrs/Semana Total: <u>72</u> Hrs/Semestre Otros ambientes de aprendizaje: <u>0</u> Hrs/Semana Total: <u>0</u> Hrs/Semestre																				
Tipo de Espacio																										
Aula		Taller		Laboratorio	<input checked="" type="checkbox"/>	Otros ambientes de Aprendizaje																				
Modalidad																										
Escolarizada	<input checked="" type="checkbox"/>	No Escolarizada		Mixta																						
Vigencia a Partir:		ENERO 2024																								
Proceso de Diseño y Autorización:					Día	Mes	Año	Organización																		
								Por Unidad de Aprendizaje:	<input checked="" type="checkbox"/>	Por Área:		Por Módulo:														
					Firma y Sello de Autorización:																					
Elaborado por:	REP. ACAD. NMS	Fecha de Elaboración:	20	02	2023																					
Revisado por:	DEMS	Fecha de Revisión:	10	10	2023																					
Aprobado por:	CTCE-NMS	Fecha de Aprobación:	24	10	2023																					
Autorizado por:	CPA-CGC	Fecha de Autorización:	23	11	2023																					
						M. EN E.N.A MARÍA ISABEL ROJAS RUÍZ Directora de Educación Media Superior																				

Programa Académico: Técnico en Programación

Unidad de Aprendizaje: Laboratorio de Desarrollo de Software II

FUNDAMENTACIÓN

Enseguida se presentan los argumentos que exponen las bases epistémicas y didácticas, así como la relevancia del Programa de Estudios de la unidad de aprendizaje **Laboratorio de Desarrollo de Software II**, para la formación del estudiante.

La unidad de aprendizaje **Laboratorio de Desarrollo de Software II**, pertenece al área de formación profesional del Bachillerato Tecnológico Bivalente del Nivel Medio Superior del Instituto Politécnico Nacional, se ubica en el cuarto nivel del Plan de Estudios del Programa Académico Técnico en Programación y se imparte en la modalidad escolarizada, de manera obligatoria, en la rama del conocimiento de Ingeniería y Ciencias Físico-Matemáticas.

Esta unidad de aprendizaje coadyuva a comprender el **Laboratorio de Desarrollo de Software II**, como una dimensión científica, técnica, tecnológica, social, responsable, metodológica y sustentable, que incentiva la adquisición, desarrollo y aplicación de las propiedades del paradigma de programación, la arquitectura para la resolución a problemas y diversas habilidades cognitivas. Introduce al estudiante al campo conceptual, procedimental y actitudinal, para el desarrollo de proyectos de software, mediante la identificación, planeación y aplicación de metodologías de desarrollo de proyectos de software, para dar solución a problemáticas del campo laboral, económico y social. La adquisición de estas destrezas y habilidades relacionadas con el paradigma de programación favorecerán en el estudiante el desarrollo de una visión crítica, cuya puesta en práctica, de manera colaborativa y en forma autónoma, en el futuro le coadyuvará a responder en forma eficiente y eficaz a los retos que se le presenten al incorporarse a estudios superiores o de igual forma al campo laboral.

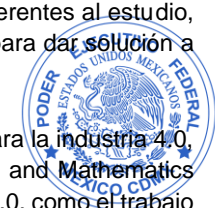
Esta unidad de aprendizaje retoma algunos aprendizajes previos del tercer nivel, es decir el ciclo de desarrollo del software (planificación, análisis, diseño, pruebas e Implementación), la metodología de desarrollo del software, el aprendizaje basado en proyectos y se trabaja de forma colaborativa en equipos de hasta 4 estudiantes, también hay una vinculación con las unidades de aprendizaje de base de datos, programación con nuevas tecnologías y programación intermedia que son del mismo nivel.

No se consideran las prácticas, para esta unidad de aprendizaje, ya que se utiliza como metodología de aprendizaje, el aprendizaje basados en proyectos, de manera que los estudiantes proponen el proyecto que van a desarrollar de forma colaborativa, es decir equipos de 4 estudiantes, de tal forma resultan 11 proyectos diferentes, por lo que las actividades desde la semana 1, son en base a un proyecto de software propio y que son diferentes entre sí.

La unidad de aprendizaje **Laboratorio de Desarrollo de Software II**, contribuye al desarrollo del talento requerido por la Industria 4.0 y para la transformación del país, Esto debido a que adquirirá, desarrollará y aplicará conceptos, técnicas y métodos que favorecen el análisis, desarrollo y resolución de problemas computacionales, mediante la escritura de programas en un lenguaje de programación, así como el desarrollo de diseñar y escribir programas para dar solución a problemas cotidianos mediante el paradigma de programación, arquitectura y trabajo en equipo extrapolando en una solución informática en un lenguaje de programación.

Laboratorio de Desarrollo de Software II, es una unidad de aprendizaje enfocada al desarrollo de habilidades técnicas, cognitivas y socioemocionales inherentes al estudio, análisis, aplicación y resolución de problemas, mediante la identificación, planeación y aplicación de metodologías de desarrollo de proyectos de software, para dar solución a problemáticas del campo laboral, económico y social.

La unidad de aprendizaje **Laboratorio de Desarrollo de Software II**, estará fundamentada en el Modelo Educativo Institucional vigente y en la Educación para la industria 4.0, por esto, se emplearán metodologías didácticas activas como, el Aprendizaje Basado en proyectos, Gamificación, Science, Technology, Engineering Arts and Mathematics (STEAM), Aprendizaje Basado en la Colaboración y el Diálogo, entre otras; esto con el propósito de que el estudiante desarrolle competencias de un entorno 4.0, como el trabajo





Programa Académico: Técnico en Programación

Unidad de Aprendizaje: Laboratorio de Desarrollo de Software II

colaborativo, trabajo en equipo, reto al cambio, autodirección, resolución de problemas cercanos a la realidad, autogestión del aprendizaje y resiliencia. Además, se emplearán herramientas tecnológicas que fomentarán la colaboración e interacción presenciales y virtuales, en forma asíncrona, que corresponden a la Educación 4.0. También se emplearán aplicaciones informáticas para el análisis, diseño, codificación y pruebas de problemas computacionales comunes, entre otros.

El rol del docente titular y de los 3 docentes auxiliares serán de mediadores entre el estudiante y los contenidos didácticos a abordar, puesto que se centrará en la creación, organización, supervisión y mediación de los espacios de trabajo, incluidas las aulas virtuales, los ciberespacios, sin dejar de atender las necesidades técnicas, de conocimientos, apoyo logístico y metodológico en los procesos de aprendizaje individual y grupal, con el objetivo de generar ambientes que favorezcan la educación técnica, inclusiva, flexible, sustentable y con perspectiva de género.

El estudiante desarrollará un trabajo autónomo en diferentes ambientes de aprendizaje, organizará el trabajo de manera independiente y articulará saberes de diversos campos del conocimiento, que le posibilitarán construir y expresar su propio conocimiento en beneficio de la sociedad; también adquirirá habilidades tanto tecnológicas como personales que promoverán la comunicación asertiva, la creatividad, la negociación, la gestión del tiempo, la motivación, el liderazgo y la responsabilidad social vinculada a la protección del medio ambiente, la erradicación de toda manifestación de violencia de género, la inclusión y la accesibilidad.

La evaluación se efectuará en el marco de la evaluación auténtica, por esto, comprenderá tres momentos: diagnóstica, formativa y sumativa. La evaluación diagnóstica se llevará a cabo mediante una charla informal sobre conocimientos previos con evaluación y retroalimentación durante el mismo momento, con la finalidad de que el docente efectúe los ajustes didácticos pertinentes y de ser necesario, nivele y ajuste los conocimientos previos adquiridos en otras unidades de aprendizaje para que establezca conexiones significativas con las unidades didácticas de la unidad de aprendizaje. Un segundo momento de la evaluación hace referencia a la evaluación formativa, que se desarrollará a lo largo del proceso de enseñanza-aprendizaje mediante las secuencias didácticas y actividades de aprendizaje formativas que estimulen el aprendizaje activo y significativo del estudiante. Este momento se enriquecerá con diversos tipos de evaluación, como la autoevaluación y la coevaluación, puesto que coadyuvarán a dar seguimiento al desarrollo de los saberes y habilidades en contexto. Cabe señalar que estas clases de evaluación serán reforzadas a través de la retroalimentación efectiva y constante.

En el tercer momento de la evaluación, con fines de acreditación, se diseñarán escenarios con problemas comunes que permitan recuperar el nivel de logro y conducir al estudiante a la meta cognición en la unidad de aprendizaje **Laboratorio de Desarrollo de Software II**, mediante evidencias de conocimiento, comprensión y diseño de proyectos, escritura de programas en algún lenguaje de programación, compilación de los programas escritos, entre otras evidencias de aprendizaje, cuyos criterios, aspectos e indicadores serán conocidos por los estudiantes en forma previa. Las evidencias de evaluación formativa e integradora mostrarán el saber hacer de manera reflexiva de los estudiantes, utilizando el conocimiento que van adquiriendo durante el proceso didáctico para luego transferir y aplicar este aprendizaje en contextos escolares, sociales y laborales.

Con base a la flexibilidad curricular y en el reconocimiento de aprendizajes múltiples, también podrá aplicarse una evaluación general para verificar que los conocimientos adquiridos por el estudiante demuestren que domina los saberes, objetivos y alcances del **Laboratorio de Desarrollo de Software II**, al finalizar el periodo ordinario. De esa forma, el programa de estudios de esta unidad de aprendizaje establece estándares para el desarrollo de conocimientos, habilidades socioemocionales, actitudes y valores.





Programa Académico: Técnico en Programación

Unidad de Aprendizaje: Laboratorio de Desarrollo de Software II

DESCRIPCIÓN DE LA UNIDAD DE APRENDIZAJE

Unidad de Aprendizaje: Laboratorio de Desarrollo de Software II		
<p>Desarrolla proyectos de software, a través de las propiedades del paradigma de programación, arquitectura y aplicación de metodologías, para solucionar diversas necesidades en general, de forma colaborativa y creativa.</p>		
Unidad 1: Análisis y Planeación		
Unidad de competencia	Aprendizajes esperados	Contenidos de aprendizaje
<p>Define el análisis y planeación del desarrollo de proyectos de software, usando las propiedades del paradigma de programación y la arquitectura para dar soluciones a problemáticas variadas.</p>	<p>Lleva a cabo la planeación y el análisis de una problemática, empleando alguna de las metodologías para el desarrollo de proyectos de software, y así proponer soluciones a problemas.</p>	<p>Conceptual: Fundamentos del proyecto, ciclo de vida de un proyecto. Los principios de desarrollo de proyectos de software. Las metodologías de desarrollo de proyectos de software El plan estratégico. El diagrama de Gantt y ruta crítica. El proyecto de software.</p> <p>Procedimental: Diagnostica la problemática de un proyecto de software describiendo sus componentes. Desarrolla el cronograma de actividades. Planea el proyecto a través del diagrama de Gantt y ruta crítica. Emplea alguna de las metodologías de desarrollo de proyectos de software, para dar solución a la problemática. Identifica las propiedades del paradigma de programación y la arquitectura</p> <p>Actitudinal: Trabaja en equipo con sus compañeros y docentes permitiendo el desarrollo de las actividades de una manera armoniosa y colaborativa. Tiene un pensamiento creativo y analítico para la resolución de diversos problemas. Utiliza la Innovación en la planeación del proyecto.</p>
	<p>Hace uso de la planeación de un proyecto, realizando el análisis y ejecución de cada una de las etapas de la metodología utilizada, para solucionar un problema de la vida real.</p>	<p>Conceptual: El proceso del negocio y el estudio de mercado la viabilidad y el alcance de un proyecto. Etapas de la metodología de solución de problemas. Planeación del proyecto de software. El análisis de requerimientos funcionales, no funcionales y de sistema.</p> <p>Procedimental: Describe el proceso del negocio y el estudio de mercado.</p>





Programa Académico: Técnico en Programación

Unidad de Aprendizaje: Laboratorio de Desarrollo de Software II

		<p>Propone soluciones al problema por medio del análisis del proyecto de software.</p> <p>Elige una propuesta de proyecto de software, para dar solución a una problemática.</p> <p>Actitudinal: Trabaja en equipo con sus compañeros y docentes permitiendo el desarrollo de las actividades de una manera armoniosa y colaborativa. Muestra un autodesarrollo y disposición hacia el aprendizaje. Tiene proactividad e iniciativa para la solución de diversas problemáticas comunes, así como tiene la capacidad para anticiparse a problemas o necesidades futuras.</p>
Unidad 2: Diseño y codificación		
Unidad de competencia	Aprendizajes esperados	Contenidos de aprendizaje
<p>Elabora el proyecto de software a través del lenguaje de modelado considerando la arquitectura y la codificación de programas en un lenguaje de programación, de forma comprometida y autónoma.</p>	<p>Utiliza los elementos del diseño de software, dando solución a la problemática planteada de forma analítica adaptándolo a los cambios tecnológicos.</p>	<p>Conceptual: Lenguaje de modelado para el diseño del software. Los Algoritmos para solución de problemas. El modelo cliente-servidor. Las páginas web y su seguridad. El diseño y la codificación del proyecto de software mediante un lenguaje de programación.</p> <p>Procedimental: Diseña la aplicación de software a través de un lenguaje de modelado. Elabora la solución de problemas reales usando algún paradigma de programación. Efectúa la codificación en un lenguaje de programación de alto nivel.</p> <p>Actitudinal: Tiene un pensamiento creativo y analítico para la resolución de diversos problemas. Aprende a ser resiliente al comprometerse y seguir adelante con lo encomendado. Aprende de forma autónoma. Se adapta rápidamente a los cambios tecnológicos de la 4T, entendiendo de manera responsable el uso del software.</p>





Programa Académico: Técnico en Programación

Unidad de Aprendizaje: Laboratorio de Desarrollo de Software II

	<p>Elabora en un lenguaje de programación de alto nivel la codificación del proyecto de software, planteado de forma colaborativa y empleando una base de datos.</p>	<p>Conceptual: Mapa de navegación. El Look & Feel. El lenguaje de programación Programación orientada a objetos o algún otro paradigma. Entornos de desarrollo integrados (IDE). La programación con nuevas tecnologías. Modelado de la base de datos (modelo entidad-relación, diccionario de datos y modelo relacional) y/o Modelado de la base de datos no relacional.</p> <p>Procedimental: Realiza el mapa de navegación y el Look & Feel . Utiliza un lenguaje de programación para la codificación del proyecto de software. Hace uso de un entorno de desarrollo integrado. Realiza el modelado y la codificación de la base de datos relacional o no relacional.</p> <p>Actitudinal: Aprende de forma autónoma. Aprecia el pensamiento creativo y de resolución de problemas. Trabaja en equipo con sus compañeros y docentes permitiendo el desarrollo de las actividades de una manera armoniosa y colaborativa.</p>
--	--	---

Unidad 3 Pruebas e implementación

Unidad de competencia	Aprendizajes esperados	Contenidos de aprendizaje
<p>Implementa las pruebas al proyecto de software desarrollado, a través de la planeación de pruebas, ocupando técnicas, para garantizar la calidad del producto, de manera colaborativa</p>	<p>Garantiza la calidad del proyecto de software desarrollado por medio de pruebas.</p>	<p>Conceptual: Tipos de pruebas de software Plan de pruebas Calidad del software Presentación del proyecto de software</p> <p>Procedimental: Efectúa algunos tipos de pruebas de software. Elabora el plan de pruebas, para asegurar la funcionalidad.</p>





Programa Académico: Técnico en Programación

Unidad de Aprendizaje: Laboratorio de Desarrollo de Software II

		<p>Realiza la presentación del proyecto de software que dé solución a la problemática planteada.</p> <p>Actitudinal: Muestra Resiliencia, e Iniciativa. Capacidad de toma de decisiones de manera responsable. Se muestra competitivo y proactivo en su participación. Aprecia el pensamiento creativo y resolución de problemas.</p>
	<p>Implementa el proyecto de software, cumpliendo con los requerimientos determinados, y de acuerdo a la gestión del proyecto.</p>	<p>Conceptual: La implementación del proyecto informático. El plan de mejora continua. La administración del proyecto de software.</p> <p>Procedimental: Realiza la implementación del proyecto de software, de acuerdo a lo requerido. Elabora el plan de mejora continua. Hace la gestión del proyecto de software.</p> <p>Actitudinal: Se muestra competitivo. Muestra pensamiento creativo y de resolución de problemas. Se interesa en actualizar permanentemente habilidades. Muestra resiliencia Muestra responsabilidad social</p>





Programa Académico: Técnico en Programación

Unidad de Aprendizaje: Laboratorio de Desarrollo de Software II

➔ **MATRIZ DE VINCULACIÓN** ⬅

COMPETENCIAS PARA EL SIGLO XXI HABILIDADES BLANDAS Y SOCIOEMOCIONALES	Unidad de Competencia 1			Unidad de Competencia 2			Unidad de Competencia 3		
	AE 1	AE 2		AE 1	AE 2		AE 1	AE 2	
Trabajo en equipo	X	X			X				
Pensamiento creativo	X	X		X	X		X	X	
Innovación	X	X		X					
Autodesarrollo		X		X					
Resiliencia				X			X	X	
Iniciativa							X	X	
Autonomía				X	X				
Proactividad		X					X		
Toma de decisiones responsable							X		
Disposición hacia el aprendizaje							X		
Resolución de problemas	X			X	X		X	X	



Programa Académico: Técnico en Programación

Unidad de Aprendizaje: Laboratorio de Desarrollo de Software II

PERFIL DOCENTE

La unidad de aprendizaje **Laboratorio de Desarrollo de Software II**, es de las primeras bases que se sientan para poder comenzar a programar, debido al tamaño de los grupos y que se necesita de una supervisión más personalizada se requiere de 4 docentes, ya que existen 4 horas en laboratorios en esta unidad de aprendizaje, donde aplicará lo teórico a la práctica y cada docente podrá atender varios alumnos, cubriendo la totalidad del grupo, de lo contrario el estudiante se atrasa, no resuelve sus dudas, originando que su aprendizaje no sea el adecuado, con la repercusión en las otras unidades de aprendizaje del mismo nivel y las siguientes unidades de aprendizaje del quinto nivel, siendo pertinente incluir 3 docentes auxiliares, para apoyar al docente titular, con el mismo número de horas teóricas y prácticas.

El Docente titular 1 y los 3 Docentes auxiliares que impartan la Unidad de Aprendizaje **Laboratorio de Desarrollo de Software II**, contarán con las habilidades en el manejo de los saberes disciplinares y/o profesionales, así como su disposición, autoridad y tolerancia en el manejo de grupos de aprendizaje. Por lo tanto, debe poseer las habilidades que favorezcan el desarrollo del talento 4.0.

Habilidades docentes en el desarrollo del Talento

En el campo de su especialización:

- Habilidades y conocimientos profesionales que se requiere para la impartición de la Unidad de Aprendizaje.
- Adquirir habilidades digitales, desarrollarlas y actualizarlas.

En el campo pedagógico:

- Fomentar procesos de enseñanza que le permitan interpretar y resolver las necesidades de aprendizaje de los alumnos, tomando en cuenta sus capacidades, habilidades, vocación e intereses.
- Desarrollar procesos de enseñanza aprendizaje, utilizando métodos basados en administración de proyectos reales, aprovechando espacios educativos distintos a las aulas, para mejorar la calidad y pertinencia de la enseñanza.

En el campo de la investigación:

- Fortalecer el trabajo académico a partir del aprovechamiento de los resultados y productos de los proyectos de investigación.

Perfil Profesional

- Titulado en Ing. en Sistemas Computacionales, Inteligencia Artificial, Ciencia de Datos, y/o Licenciado en Informática, Comunicaciones y Electrónica, Telemática, Computación o Maestría en Ciencias de la Computación, Sistemas Computacionales, Maestría en Gestión de la innovación en Tecnologías de la información y comunicación, maestría o especialización en docencia o afín, con experiencia de dos años en el área docente y experiencia comprobable de tres años en la iniciativa pública o privada aplicando los conocimientos de la unidad de aprendizaje

Se requiere de un docente titular y tres docentes auxiliares, con el mismo número de horas teóricas y prácticas frente a grupo, La dinámica de trabajo en el laboratorio se hace con la formación de varios equipos de trabajo, los cuales desarrollan proyectos de software con herramientas de programación adecuadas a la problemática a solucionar. Por esta situación se debe contar con un Docente titular y otros 3 Docentes, con competencias en diferentes herramientas de programación y en administración de proyectos de tecnologías de la información, para poder asesorar, guiar, verificar a los distintos equipos de trabajo, facilitar la generación de juicios, valorar una parte individual y una parte grupal, la evaluación se realiza a los estudiantes en función de los conocimientos aplicados y el proceso es tan importante como la solución al problema el cual es supervisado por los todos los docentes.



Programa Académico: Técnico en Programación

Unidad de Aprendizaje: Laboratorio de Desarrollo de Software II

ESTRUCTURA DIDÁCTICA

Unidad Didáctica 1:	Unidad 1: Análisis y Planeación	Nivel:	Cuarto
Propósito General:	Desarrolla proyectos de software, a través de las propiedades del paradigma de programación, arquitectura y aplicación de metodologías, para solucionar diversas necesidades en general, de forma colaborativa y creativa.		
Unidad de Competencia No 1:	Define el análisis y planeación del desarrollo de proyectos de software, usando las propiedades del paradigma de programación y la arquitectura para dar soluciones a problemáticas variadas.		
Aprendizaje Esperado No 1:	Lleva a cabo la planeación y el análisis de una problemática, empleando alguna de las metodologías para el desarrollo de proyectos de software, y así proponer soluciones a problemas.	Tiempo estimado para obtener el Aprendizaje Esperado:	8 horas
Contenidos de Aprendizaje			
Conceptuales	Procedimentales	Actitudinales	
Fundamentos del proyecto, ciclo de vida de un proyecto. Los principios de desarrollo de proyectos de software. Las metodologías de desarrollo de proyectos de software. El plan estratégico. El diagrama de Gantt y ruta crítica. El proyecto de software.	Diagnostica la problemática de un proyecto de software describiendo sus componentes. Desarrolla el cronograma de actividades. Planea el proyecto a través del diagrama de Gantt y ruta crítica. Emplea alguna de las metodologías de desarrollo de proyectos de software, para dar solución a la problemática.	Trabaja en equipo con sus compañeros y acompañado de los docentes, permitiendo el desarrollo de las actividades de una manera armoniosa y colaborativa. Tiene un pensamiento creativo y analítico para la resolución de diversos problemas. Utiliza la Innovación en la planeación del proyecto.	
Estrategia Didáctica y Ambiente de Aprendizaje			
Aprendizaje Basado en Proyectos, y el trabajo colaborativo, en el desarrollo de un proyecto de software, bajo los paradigmas de programación, con la conformación de equipos de 4 estudiantes, en el laboratorio de computación, así como el uso de la computadora y aplicaciones en Internet para algunas consultas e investigaciones, también se contempla la publicación de apuntes y ejemplos, haciendo uso de una plataforma virtual, como Classroom o Teams.			
Apertura: El docente, hace una exploración de conocimientos de manera general, con una lluvia de ideas y conceptos sobre la programación, los proyectos, las tecnologías, el software, etc. Para generar con ello reflexión, a partir de las respuestas y conocimientos previos del estudiante sobre estos temas. Además, se realiza el análisis de un proyecto.			
Desarrollo: El docente hace una presentación electrónica sobre los fundamentos y conceptos básicos de la programación y su relación con los proyectos de software, explica sus características y las partes constitutivas de un proyecto, explica las metodologías de desarrollo de un proyecto de software y ejemplifica diversas problemáticas a ser resueltas mediante el desarrollo de proyectos de software. Los estudiantes buscan, discriminan y sintetizan información para construir su conceptualización de los elementos de programación y de proyectos, el docente retroalimenta las aportaciones de los estudiantes.			
Cierre: Los estudiantes, realizan la planeación estratégica y dos propuestas de proyecto de software, por equipo o empresa, escogen una metodología de desarrollo de proyectos, elaboran el cronograma de actividades y planean el proyecto a través del diagrama de Gantt y ruta crítica, el docente está abierto a las preguntas o dudas que el estudiante llegará a tener para facilitar su aprendizaje.			
Ambiente de Aprendizaje:			





Programa Académico: Técnico en Programación

Unidad de Aprendizaje: Laboratorio de Desarrollo de Software II

De forma presencial en el laboratorio de computación, además el docente creará un aula virtual en alguna plataforma tecnológica como Teams, Classroom, etc. Para poder tener comunicación con los alumnos de manera asíncrona y con ello complementar la comunicación, resolver dudas y crear un ambiente híbrido (presencial y virtual).

Herramientas Tecnológicas y Recursos Didácticos	Evidencia de Aprendizaje para la Evaluación Formativa	Instrumento y Criterios de Evaluación
<p>Herramientas Tecnológicas: Equipo de Cómputo (PC) Programas (aplicaciones) orientados a la unidad de competencia Conexión a internet. Retroproyector.</p> <p>Plataformas Software orientado a la unidad didáctica desarrollada, ejemplo el Classroom o Teams.</p> <p>Recursos Didácticos: Presentaciones electrónicas Videos Repositorios de contenidos y material de consulta Apuntes orientados a los temas expuestos Infografías enfocadas en la resolución y procesos metodológicos, algorítmicos y lógicos. Pizarrón y Marcadores. Papelería escolar</p>	<p>Portafolio de evidencias del proceso de solución de las propuestas de proyectos Formación de equipos de hasta 4 estudiantes. Entrega en un archivo electrónico, el plan estratégico de la empresa o equipo. Entrega en un archivo electrónico, el diagrama de Gantt y ruta crítica del proyecto de software</p>	<p>Instrumento de evaluación:</p> <ul style="list-style-type: none"> • Lista de cotejo. <p>Criterios de evaluación de forma:</p> <ul style="list-style-type: none"> • El portafolio de evidencias cuenta con una portada con los datos de identificación del estudiante. • Los archivos se encuentran ordenados y con la nomenclatura indicada. • Los textos están creados con fuente Arial 12, interlineado intermedio. • La evidencia está conformada por una introducción, desarrollo y conclusión • La redacción mantiene un tono académico, profesional, crítico y responsable <p>Criterios de evaluación de fondo:</p> <ul style="list-style-type: none"> • Aplica el pensamiento lógico para la solución a los problemas y casos del proyecto de software. • Identifica el proyecto de software, de acuerdo al paradigma de programación. • Realiza el plan estratégico, el diagrama de Gantt y la ruta crítica, con los puntos requeridos





Programa Académico: Técnico en Programación

Unidad de Aprendizaje: Laboratorio de Desarrollo de Software II

Unidad Didáctica 1:	Unidad 1: Análisis y Planeación	Nivel:	Cuarto
Propósito General:	Desarrolla proyectos de software, a través de las propiedades del paradigma de programación, arquitectura y aplicación de metodologías, para solucionar diversas necesidades en general, de forma colaborativa y creativa.		
Unidad de Competencia No 1:	Define el análisis y planeación del desarrollo de proyectos de software, usando las propiedades del paradigma de programación y la arquitectura para dar soluciones a problemáticas variadas.		
Aprendizaje Esperado No 2:	Hacer la planeación de un proyecto, realizando el análisis y ejecución de cada una de las etapas de la metodología, utilizada para solucionar un problema de la vida real.	Tiempo estimado para obtener el Aprendizaje Esperado:	12 horas
Contenidos de Aprendizaje			
Conceptuales	Procedimentales	Actitudinales	
El proceso del negocio y el estudio de mercado la viabilidad y el alcance de un proyecto Etapas de la metodología de solución de problemas Planeación del proyecto de software El análisis de requerimientos funcionales, no funcionales y de sistema	Describe el proceso del negocio y el estudio de mercado. Propone soluciones al problema por medio del análisis del proyecto de software. Elige una propuesta de proyecto de software, para dar solución a una problemática	Trabaja en equipo con sus compañeros y acompañado de los docentes, permitiendo el desarrollo de las actividades de una manera armoniosa y colaborativa. Muestra un autodesarrollo y disposición hacia el aprendizaje. Tiene proactividad e iniciativa para la solución de diversas problemáticas comunes, así como tiene la capacidad para anticiparse a problemas o necesidades futuras	
Estrategia Didáctica y Ambiente de Aprendizaje			
El Aprendizaje Basado en Proyectos y el trabajo colaborativo, en el desarrollo de un proyecto de software, para guiar el aprendizaje del estudiante, se podrá hacer uso de la computadora y aplicaciones en Internet, así como la publicación de guías y manuales haciendo uso de la plataforma Classroom, además de formación de equipos de trabajo de hasta 4 estudiantes			
Apertura: los estudiantes en acompañamiento de los docentes identifican las etapas de la metodología de un proyecto, las variables, datos de entrada y de salida, comprenden la definición de análisis y la relación que tiene con la planeación de un proyecto de software en la solución de problemas. Además, el docente les explica el análisis y diseño de un proyecto,			
Desarrollo: el docente titular mostrará ejemplos de la metodología de solución de un proyecto, empleando el paradigma de programación para que posteriormente el estudiante por medio de ésta resuelva problemas aplicados a situaciones cotidianas, considerando datos de entrada, variables, procesamiento, interacciones y datos de salida, las actividades podrán ser individual o colaborativas, ya sea presencial o por medio de una plataforma de aprendizaje.			
Cierre: al terminar la solución de los casos o problemas, se debe promover la reflexión y el intercambio de ideas a través de foros presenciales o en línea y las evidencias deberán quedar respaldadas de forma individual en un portafolio de evidencias.			
Ambiente de Aprendizaje:			
Presencial en el laboratorio de computación, pero además el docente creará un aula virtual en alguna plataforma tecnológica como Teams, Classroom, etc. Para poder tener comunicación con los alumnos de manera asíncrona y con ello complementar la comunicación, resolver dudas y crear un ambiente híbrido (presencial y virtual).			
Herramientas Tecnológicas y Recursos Didácticos	Evidencia de Aprendizaje para la Evaluación Formativa	Instrumento y Criterios de Evaluación	
Herramientas Tecnológicas: Equipo de Cómputo (PC)	Portafolio de evidencias que muestre el proceso de solución de los problemas, el proceso del negocio, y estudio de mercado	Instrumento de evaluación: <ul style="list-style-type: none">Lista de cotejo	





Programa Académico: Técnico en Programación

Unidad de Aprendizaje: Laboratorio de Desarrollo de Software II

<p>Programas (aplicaciones) orientados a la unidad de competencia Conexión a internet. Retroproyector.</p> <p>Plataformas Software orientado a la unidad didáctica desarrollada, ejemplo el Classroom</p> <p>Recursos Didácticos: Presentaciones electrónicas Videos Repositorios de contenidos y material de consulta Repositorio de casos para la solución Pizarrón y Marcadores. Papelería escolar</p>	<p>Archivo electrónico con propuesta de proyecto de software, para dar solución a una problemática</p>	<p>Criterios de evaluación de forma:</p> <ul style="list-style-type: none"> • El portafolio de evidencias cuenta con una portada con los datos de identificación del estudiante. • Los archivos se encuentran ordenados y con la nomenclatura indicada. • Los textos están creados con fuente Arial 12, interlineado intermedio. • La evidencia está conformada por una introducción, desarrollo y conclusión. • La redacción mantiene un tono académico y profesional, crítica y responsable. <p>Criterios de evaluación de fondo:</p> <ul style="list-style-type: none"> • Aplica el paradigma de programación y arquitectura para la solución a los problemas y casos del proyecto de software • Identifica el proceso del negocio y el estudio de mercado. • Presenta la propuesta de proyecto de manera viable y organizada.
---	--	---





Programa Académico: Técnico en Programación

Unidad de Aprendizaje: Laboratorio de Desarrollo de Software II

Unidad Didáctica 2:	Unidad 2: Diseño y codificación	Nivel:	Cuarto
Propósito General:	Desarrolla proyectos de software, a través de las propiedades del paradigma de programación, arquitectura y aplicación de metodologías, para solucionar diversas necesidades en general, de forma colaborativa y creativa.		
Unidad de Competencia No 2:	Elabora el proyecto de software a través del lenguaje de modelado considerando la arquitectura y la codificación de programas en un lenguaje de programación, de forma comprometida y autónoma.		
Aprendizaje Esperado No 1:	Utiliza los elementos del diseño de software, dando solución a la problemática planteada de forma analítica adaptándolo a los cambios tecnológicos.	Tiempo estimado para obtener el Aprendizaje Esperado:	16 horas
Contenidos de Aprendizaje			
Conceptuales	Procedimentales	Actitudinales	
Lenguaje de modelado para el diseño del software. Los Algoritmos para solución de problemas. El modelo cliente-servidor. Las páginas web y su seguridad. El diseño y la codificación del proyecto de software mediante un lenguaje de programación.	Diseña la aplicación de software a través de un lenguaje de modelado. Elabora la solución de problemas reales usando algún paradigma de programación. Efectúa la codificación en un lenguaje de programación de alto nivel.	Tiene un pensamiento creativo y analítico para la resolución de diversos problemas. Aprende a ser resiliente al comprometerse y seguir adelante con lo encomendado. Aprende de forma autónoma. Se adapta rápidamente a los cambios tecnológicos de la 4T, entendiendo de manera responsable el uso del software.	
Estrategia Didáctica y Ambiente de Aprendizaje			
<p>Aprendizaje Basado en Proyectos, y el trabajo colaborativo, en el desarrollo de un proyecto de software, bajo los paradigmas de programación, con la conformación de equipos de 4 estudiantes, en el laboratorio de computación, así como el uso de la computadora y aplicaciones en Internet para algunas consultas e investigaciones, también se contempla la publicación de apuntes y ejemplos, haciendo uso de una plataforma virtual, como Classroom o Teams.</p> <p>Apertura: El docente realiza una presentación ante los estudiantes, sobre el lenguaje de modelado para el diseño de software, los algoritmos, el modelo cliente-servidor, las páginas web y su seguridad, explicando la relación que tienen los anteriores conceptos, con el desarrollo del proyecto de software.</p> <p>Desarrollo: el docente titular mostrará algunos ejemplos con una presentación, sobre el lenguaje de modelado para el diseño de software, la arquitectura el modelo cliente-servidor, las páginas web y su seguridad, para que posteriormente el estudiante por medio de esta lo aplique al proyecto que está desarrollando con su equipo de trabajo. o por medio de una plataforma de aprendizaje.</p> <p>Cierre: El docente hace una reflexión e intercambio de ideas con los estudiantes, a través de foros presenciales o en línea de la actividad desarrollada y pide que las evidencias deberán quedar respaldadas de forma grupal o individual en un portafolio de evidencias.</p> <p>Ambiente de Aprendizaje:</p> <p>Presencial en el laboratorio de computación, pero además el docente creará un aula virtual en alguna plataforma tecnológica como Teams, Classroom, etc. Para poder tener comunicación con los alumnos de manera asincrónica y con ello complementar la comunicación, resolver dudas y crear un ambiente híbrido (presencial y virtual).</p>			





Programa Académico: Técnico en Programación

Unidad de Aprendizaje: Laboratorio de Desarrollo de Software II

Herramientas Tecnológicas y Recursos Didácticos	Evidencia de Aprendizaje para la Evaluación Formativa	Instrumento y Criterios de Evaluación
<p>Herramientas Tecnológicas: Equipo de Cómputo (PC) Programas (aplicaciones) orientados a la unidad de competencia Conexión a internet. Retroproyector.</p> <p>Plataformas Software orientado a la unidad didáctica desarrollada, ejemplo el Classroom</p> <p>Recursos Didácticos: Videos Presentaciones electrónicas Repositorios de contenidos y material de consulta Repositorio de casos para la solución Infografías enfocadas en la resolución y procesos metodológicos, algorítmicos y lógicos. Pizarrón y Marcadores. Papelería escolar</p>	<p>Portafolio de evidencias que muestre el diseño de la aplicación de software, con un lenguaje de modelado. La codificación del proyecto de software en un lenguaje de programación del alto nivel.</p>	<p>Instrumento de evaluación:</p> <ul style="list-style-type: none"> • Lista de cotejo, rubrica <p>Criterios de evaluación de forma:</p> <ul style="list-style-type: none"> • El portafolio de evidencias cuenta con una portada con los datos de identificación del estudiante. • Los archivos se encuentran ordenados y con la nomenclatura indicada. • Los textos están creados con fuente Arial 12, interlineado intermedio. • La evidencia está conformada por una introducción, desarrollo y conclusión. • La redacción de la evidencia mantiene un tono académico y profesional, crítica y responsable <p>Criterios de evaluación de fondo:</p> <ul style="list-style-type: none"> • El diseño de la aplicación de software, fue realizada en un lenguaje de modelado • Presenta los algoritmos de solución del proyecto • Aplica el paradigma de programación, la arquitectura y modelo de desarrollo para la solución al caso. • La codificación del proyecto da solución al proyecto de software.





Programa Académico: Técnico en Programación

Unidad de Aprendizaje: Laboratorio de Desarrollo de Software II

Unidad Didáctica 2:	Diseño y codificación	Nivel:	Cuarto
Propósito General:	Desarrolla proyectos de software, a través de las propiedades del paradigma de programación, arquitectura y aplicación de metodologías, para solucionar diversas necesidades en general, de forma colaborativa y creativa		
Unidad de Competencia No 2:	Elabora el proyecto de software a través del lenguaje de modelado considerando la arquitectura y la codificación de programas en un lenguaje de programación, de forma comprometida y autónoma		
Aprendizaje Esperado No 2:	Elabora en un lenguaje de programación de alto nivel la codificación del proyecto de software, planteado de forma colaborativa y empleando una base de datos	Tiempo estimado para obtener el Aprendizaje Esperado:	24 horas

Contenidos de Aprendizaje

Conceptuales	Procedimentales	Actitudinales
Mapa de navegación. El Look & Feel. El lenguaje de programación Programación orientada a objetos o algún otro paradigma. Entornos de desarrollo integrados (IDE) La programación con nuevas tecnologías Modelado de la base de datos (modelo entidad-relación, diccionario de datos y modelo relacional) o modelado de la base de datos no relacional.	Realiza el mapa de navegación y el Look & Feel. Utiliza un lenguaje de programación para la codificación del proyecto de software. Hace uso de un entorno de desarrollo de integrado. Realiza el modelado y la codificación de la base de datos relacional o no relacional.	Aprende de forma autónoma. Aprecia el pensamiento creativo y de resolución de problemas. Trabaja en equipo con sus compañeros y docentes permitiendo el desarrollo de las actividades de una manera armoniosa y colaborativa.

Estrategia Didáctica y Ambiente de Aprendizaje

El Aprendizaje Basado en Proyectos, además de integrar recursos digitales para guiar el aprendizaje del estudiante, hacer uso de la computadora y aplicaciones en Internet, así como la publicación de apuntes y manuales haciendo uso de una plataforma virtual, se propone la formación de equipos de trabajo de hasta 4 estudiantes

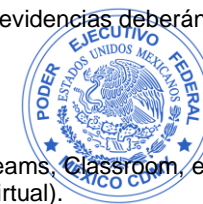
Apertura: El docente realiza una presentación ante los estudiantes, del mapa de navegación, el Look & Feel, el lenguaje de programación, la programación orientada a objetos o algún otro paradigma, los entornos de desarrollo integrados (IDE), La programación con nuevas tecnologías, el modelado de la base de datos (modelo entidad-relación, diccionario de datos y modelo relacional) y/o el modelado de la base de datos no relacional, explicando la relación que tienen los anteriores conceptos, con el desarrollo del proyecto de software.

Desarrollo: el docente titular mostrará algunos ejemplos con una presentación, del mapa de navegación, el Look & Feel, el lenguaje de programación, la programación orientada a objetos o el paradigma usado, los entornos de desarrollo integrados (IDE), La programación con nuevas tecnologías, el modelado de la base de datos (modelo entidad-relación, diccionario de datos y modelo relacional), para que posteriormente el estudiante por medio de esta lo aplique al proyecto que está desarrollando con su equipo de trabajo o por medio de una plataforma de aprendizaje.

Cierre: El docente hace una reflexión e intercambio de ideas con los estudiantes, a través de foros presenciales o en línea de la actividad desarrollada y pide que las evidencias deberán quedar respaldadas de forma grupal o individual en un portafolio de evidencias.

Ambiente de Aprendizaje:

Es un ambiente de aprendizaje presencial en el laboratorio de computación, pero además el docente creará un aula virtual en alguna plataforma tecnológica como Teams, Classroom, etc. Para poder tener comunicación con los alumnos de manera asincrónica y con ello complementar la comunicación, resolver dudas y crear un ambiente híbrido (presencial y virtual).





Programa Académico: Técnico en Programación

Unidad de Aprendizaje: Laboratorio de Desarrollo de Software II

Herramientas Tecnológicas y Recursos Didácticos	Evidencia de Aprendizaje para la Evaluación Formativa	Instrumento y Criterios de Evaluación
<p>Herramientas Tecnológicas: Equipo de Cómputo (PC) Programas (aplicaciones) orientados a la unidad de competencia Conexión a internet. Retroproyector.</p> <p>Plataformas Software orientado a la unidad didáctica desarrollada, ejemplo el Classroom</p> <p>Recursos Didácticos: Videos Presentaciones electrónicas. Repositorios de contenidos y material de consulta Repositorio de casos para la solución Pizarrón y Marcadores. Papelería escolar</p>	<p>Portafolio de evidencias que muestre el mapa de navegación y el Look & Feel,</p> <p>Repositorio digital de actividades de tareas individuales y en equipo de la codificación del proyecto de software.</p> <p>Archivo electrónico del modelado del proyecto y la codificación de la base de datos relacional o no relacional</p>	<p>Instrumento de evaluación:</p> <ul style="list-style-type: none"> • Lista de cotejo, <p>Criterios de evaluación de forma:</p> <ul style="list-style-type: none"> • El portafolio de evidencias cuenta con una portada con los datos de identificación del estudiante. • Los archivos se encuentran ordenados y con la nomenclatura indicada. • Los textos están creados con fuente Arial 12, interlineado intermedio. • La evidencia está conformada por una introducción, desarrollo y conclusión. • La redacción a la reflexión mantiene un tono académico y profesional, crítica y responsable. <p>Criterios de evaluación de fondo:</p> <ul style="list-style-type: none"> • El Look & Feel, contiene todas las pantallas del mapa de navegación. • El mapa de navegación, está basado en los requerimientos del proyecto • Aplica el paradigma de programación, arquitectura y modelo de desarrollo para la solución a los problemas y casos.





Programa Académico: Técnico en Programación

Unidad de Aprendizaje: Laboratorio de Desarrollo de Software II

Unidad Didáctica 3:	Pruebas e implementación	Nivel:	Cuarto
Propósito General:	Desarrolla proyectos de software, a través de las propiedades del paradigma de programación, arquitectura y aplicación de metodologías, para solucionar diversas necesidades en general, de forma colaborativa y creativa		
Unidad de Competencia No 3:	Aplica las pruebas al proyecto de software desarrollado considerando la planeación de pruebas, ocupando técnicas, para garantizar la calidad del producto		
Aprendizaje Esperado No 1:	Garantizar la calidad del proyecto de software desarrollado por medio de pruebas.	Tiempo estimado para obtener el Aprendizaje Esperado:	8 horas

Contenidos de Aprendizaje

Conceptuales	Procedimentales	Actitudinales
Tipos de pruebas de software. Plan de pruebas. Calidad del software. Presentación del proyecto de software.	Efectúa algunos tipos de pruebas de software. Elabora el plan de pruebas, para asegurar la funcionalidad. Realiza la presentación del proyecto de software que dé solución a la problemática planteada.	Muestra Resiliencia e Iniciativa. Capacidad de toma de decisiones de manera responsable. Se muestra competitivo y proactivo en su participación. Aprecia el pensamiento creativo y resolución de problemas.

Estrategia Didáctica y Ambiente de Aprendizaje

Aprendizaje Basado en Proyectos, y el trabajo colaborativo, en el desarrollo de un proyecto de software, bajo los paradigmas de programación, con la conformación de equipos de 4 estudiantes, en el laboratorio de computación, así como el uso de la computadora y aplicaciones en Internet para algunas consultas e investigaciones, también se contempla la publicación de apuntes y ejemplos, haciendo uso de una plataforma virtual, como Classroom o Teams.

Apertura: El docente realiza una presentación ante los estudiantes, de los tipos de pruebas de software, el plan de pruebas, la calidad del software, la presentación del proyecto de software, explicando la relación que tienen los anteriores conceptos, con el desarrollo del proyecto de software.

Desarrollo: el docente titular mostrará algunos ejemplos con una presentación, de los tipos de pruebas de software, el plan de pruebas, la calidad del software, y la presentación del proyecto de software, para que posteriormente el estudiante por medio de esta, lo aplique al proyecto que está desarrollando con su equipo de trabajo, o por medio de una plataforma de aprendizaje.

Cierre: El docente hace una reflexión e intercambio de ideas con los estudiantes, de forma presencial o en línea de la actividad desarrollada y pide que las evidencias deberán quedar respaldadas de forma grupal o individual en el portafolio de evidencias.

Ambiente de Aprendizaje:

Es un ambiente de aprendizaje presencial en el laboratorio de computación, pero además el docente creará un aula virtual en alguna plataforma tecnológica como Teams, Classroom, etc. Para poder tener comunicación con los alumnos de manera asíncrona y con ello complementar la comunicación, resolver dudas y crear un ambiente híbrido (presencial y virtual).





Programa Académico: Técnico en Programación

Unidad de Aprendizaje: Laboratorio de Desarrollo de Software II

Herramientas Tecnológicas y Recursos Didácticos	Evidencia de Aprendizaje para la Evaluación Formativa	Instrumento y Criterios de Evaluación
<p>Herramientas Tecnológicas: Equipo de Cómputo (PC) Programas (aplicaciones) orientados a la unidad de competencia Conexión a internet. Retroproyector.</p> <p>Plataformas Software orientado a la unidad didáctica desarrollada, ejemplo el Classroom o Teams.</p> <p>Recursos Didácticos: Videos Presentaciones electrónicas Repositorios de contenidos y material de consulta Repositorio de casos para la solución</p>	<p>Portafolio de evidencias que muestre: El tipo de pruebas de software, El plan de pruebas, para asegurar la funcionalidad. La presentación del proyecto de software con la solución a la problemática planteada</p>	<p>Instrumento de evaluación:</p> <ul style="list-style-type: none"> • Lista de cotejo <p>Criterios de evaluación de forma:</p> <ul style="list-style-type: none"> • El portafolio de evidencias cuenta con una portada con los datos de identificación del estudiante. • Los archivos se encuentran ordenados y con la nomenclatura indicada. • Los textos están creados con fuente Arial 12, interlineado intermedio. • La evidencia está conformada por una introducción, desarrollo y conclusión. • La redacción a la reflexión mantiene un tono académico y profesional, crítica y responsable. <p>Criterios de evaluación de fondo:</p> <ul style="list-style-type: none"> • Aplica el paradigma de programación, arquitectura y modelo de desarrollo para la solución a los problemas y casos. • Realiza las pruebas al proyecto de software, de acuerdo plan de pruebas. • Hace la presentación del proyecto de software, ante los estudiantes y docentes, y prueba la funcionalidad del sistema de software realizado.





Programa Académico: Técnico en Programación

Unidad de Aprendizaje: Laboratorio de Desarrollo de Software II

Unidad Didáctica 3:	Pruebas e implementación	Nivel:	Cuarto
Propósito General:	Desarrolla proyectos de software, a través de las propiedades del paradigma de programación, arquitectura y aplicación de metodologías, para solucionar diversas necesidades en general, de forma colaborativa y creativa		
Unidad de Competencia No 3:	Aplica las pruebas al proyecto de software desarrollado considerando la planeación de pruebas, ocupando técnicas, para garantizar la calidad del producto.		
Aprendizaje Esperado No 2:	Realiza la implementación del proyecto de software, cumpliendo con los requerimientos determinados, y de acuerdo a la gestión del proyecto.	Tiempo estimado para obtener el Aprendizaje Esperado:	4 horas

Contenidos de Aprendizaje

Conceptuales	Procedimentales	Actitudinales
La implementación del proyecto informático El plan de mejora continua. La administración del proyecto de software.	Realiza la implementación del proyecto de software, de acuerdo a lo requerido. Elabora el plan de mejora continua. Hace la gestión del proyecto de software.	Se muestra competitivo. Aprecia el pensamiento creativo y resolución de problemas. Se interesa en actualizar permanentemente habilidades. Valora la positividad, el orden. Aprecia la responsabilidad y es comprometido.

Estrategia Didáctica y Ambiente de Aprendizaje

Aprendizaje Basado en Proyectos, y el trabajo colaborativo, en el desarrollo de un proyecto de software, bajo los paradigmas de programación, con la conformación de equipos de 4 estudiantes, en el laboratorio de computación, así como el uso de la computadora y aplicaciones en Internet para algunas consultas e investigaciones, también se contempla la publicación de apuntes y ejemplos, haciendo uso de una plataforma virtual, como Classroom o Teams.

Apertura: El docente realiza una presentación de la implementación del proyecto informático, el plan de mejora continua, la administración del proyecto de software, explicando la relación que tienen los anteriores conceptos, con el desarrollo del proyecto de software.

Desarrollo: el docente titular mostrará algunos ejemplos con una presentación, de la implementación del proyecto informático, el plan de mejora continua, la administración del proyecto de software, para que posteriormente el estudiante por medio de esta, lo aplique al proyecto que está desarrollando con su equipo de trabajo o por medio de una plataforma de aprendizaje.

Cierre: El docente hace una reflexión e intercambio de ideas con los estudiantes, a través de foros presenciales o en línea de la actividad desarrollada y pide que las evidencias deberán quedar respaldadas de forma grupal o individual en un portafolio de evidencias.

Ambiente de Aprendizaje:

Es un ambiente de aprendizaje presencial en el laboratorio de computación, pero además el docente creará un aula virtual en alguna plataforma tecnológica como Teams, Classroom, etc. Para poder tener comunicación con los alumnos de manera asíncrona y con ello complementar la comunicación, resolver dudas y crear un ambiente híbrido (presencial y virtual).





Programa Académico: Técnico en Programación

Unidad de Aprendizaje: Laboratorio de Desarrollo de Software II

Herramientas Tecnológicas y Recursos Didácticos	Evidencia de Aprendizaje para la Evaluación Formativa	Instrumento y Criterios de Evaluación
<p>Herramientas Tecnológicas: Equipo de Cómputo (PC) Programas (aplicaciones) orientados a la unidad de competencia Conexión a internet. Retroproyector.</p> <p>Plataformas Software orientado a la unidad didáctica desarrollada, ejemplo el Classroom</p> <p>Recursos Didácticos: Videos Presentaciones electrónicas. Repositorios de contenidos y material de consulta Repositorio de casos para la solución Infografías enfocadas en la resolución y procesos metodológicos, algorítmicos y lógicos. Pizarrón y Marcadores. Papelería escolar</p>	<p>Portafolio de evidencias que muestre: la implementación del proyecto de software, de acuerdo a lo requerido, el plan de mejora continua, y la gestión del proyecto de software</p>	<p>Instrumento de evaluación:</p> <ul style="list-style-type: none"> • Lista de cotejo, rubrica <p>Criterios de evaluación de forma:</p> <ul style="list-style-type: none"> • El portafolio de evidencias cuenta con una portada con los datos de identificación del estudiante. • Los archivos se encuentran ordenados y con la nomenclatura indicada. • Los textos están creados con fuente Arial 12, interlineado intermedio. • La evidencia está conformada por una introducción, desarrollo y conclusión. • La redacción a la reflexión mantiene un tono académico y profesional, crítica y responsable. <p>Criterios de evaluación de fondo:</p> <ul style="list-style-type: none"> • Aplica la implementación del proyecto de software, en función de lo requerido. • Aplica el paradigma de programación elegido, la arquitectura y modelo de desarrollo para la solución a los problemas y casos. • Presenta el plan de mejora continuo del proyecto de software.





Programa Académico: Técnico en Programación

Unidad de Aprendizaje: Laboratorio de Desarrollo de Software II

PLAN DE EVALUACIÓN SUMATIVA

N°	Unidad de Competencia	Evidencia Integradora	Criterios e Instrumento de Evaluación	Porcentaje de Acreditación
1	Define el análisis y planeación del desarrollo de proyectos de software, usando las propiedades del paradigma de programación y la arquitectura para dar soluciones a problemáticas variadas.	El portafolio de evidencias, con la planeación del proyecto de software	Instrumento de evaluación: <ul style="list-style-type: none"> • Lista de cotejo Criterios de evaluación de forma: <ul style="list-style-type: none"> • El portafolio de evidencias cuenta con una portada con los datos de identificación del estudiante. • Los archivos se encuentran ordenados y con la nomenclatura indicada. • Los textos están creados con fuente Arial 12, interlineado intermedio. Criterios de evaluación de fondo: <ul style="list-style-type: none"> • La evidencia está conformada por una introducción, desarrollo y conclusión. • La redacción a la reflexión mantiene un tono académico y profesional, crítica y responsable. • Aplica el paradigma de programación, arquitectura y modelo de desarrollo para la solución a los problemas y casos. 	30%
2	Elabora el proyecto de software a través del lenguaje de modelado considerando la arquitectura y la codificación de programas en un lenguaje de programación, de forma comprometida y autónoma.	El portafolio de evidencias, con el diseño y la codificación del proyecto de software, en algún lenguaje de programación	Instrumento de evaluación: <ul style="list-style-type: none"> • Lista de cotejo Criterios de evaluación de forma: <ul style="list-style-type: none"> • El portafolio de evidencias cuenta con una portada con los datos de identificación del estudiante. • Los archivos se encuentran ordenados y con la nomenclatura indicada. 	50%





Programa Académico: Técnico en Programación

Unidad de Aprendizaje: Laboratorio de Desarrollo de Software II

			<ul style="list-style-type: none"> Los textos están creados con fuente Arial 12, interlineado intermedio. <p>Criterios de evaluación de fondo:</p> <ul style="list-style-type: none"> La evidencia está conformada por una introducción, desarrollo y conclusión. La redacción a la reflexión mantiene un tono académico y profesional, crítica y responsable. Aplica el paradigma de programación, arquitectura y modelo de desarrollo para la solución a los problemas y casos. 	
3	<p>Aplica las pruebas al proyecto de software desarrollado considerando la planeación de pruebas, ocupando técnicas, para garantizar la calidad del producto.</p>	<p>El portafolio de evidencias, con el plan de pruebas, la implementación y el plan de mejora del proyecto de software desarrollado</p>	<p>Instrumento de evaluación:</p> <ul style="list-style-type: none"> Lista de cotejo <p>Criterios de evaluación de forma:</p> <ul style="list-style-type: none"> El portafolio de evidencias cuenta con una portada con los datos de identificación del estudiante. Los archivos se encuentran ordenados y con la nomenclatura indicada. Los textos están creados con fuente Arial 12, interlineado intermedio. <p>Criterios de evaluación de fondo:</p> <ul style="list-style-type: none"> La evidencia está conformada por una introducción, desarrollo y conclusión. La redacción a la reflexión mantiene un tono académico y profesional, crítica y responsable. Aplica el paradigma de programación, arquitectura y modelo de desarrollo para la solución a los problemas y casos. 	20%





Programa Académico: Técnico en Programación

Unidad de Aprendizaje: Laboratorio de Desarrollo de Software II

EVIDENCIA INTEGRADORA

Propósito de la Unidad de Aprendizaje	Evidencia Integradora	Criterios e Instrumento de Evaluación	Porcentaje de Acreditación
<p>Desarrolla proyectos de software, a través de las propiedades del paradigma de programación, arquitectura y aplicación de metodologías, para solucionar problemáticas en general, de forma colaborativa y creativa.</p>	<p>El portafolio de evidencias, y el proyecto de software</p>	<p>Instrumento de evaluación:</p> <ul style="list-style-type: none"> • Lista de cotejo <p>Criterios de evaluación de forma:</p> <ul style="list-style-type: none"> • El portafolio de evidencias cuenta con una portada con los datos de identificación del estudiante. • Los archivos se encuentran ordenados y con la nomenclatura indicada. • Los textos están creados con fuente Arial 12, interlineado intermedio. <p>Criterios de evaluación de fondo:</p> <ul style="list-style-type: none"> • La evidencia está conformada por una introducción, desarrollo y conclusión. • La redacción a la reflexión mantiene un tono académico y profesional, crítica y responsable. • Aplica el paradigma de programación, arquitectura y modelo de desarrollo de proyectos de software, para la solución a los problemas y casos. 	<p>100%</p>





Programa Académico: Técnico en Programación

Unidad de Aprendizaje: Laboratorio de Desarrollo de Software II

PROGRAMA SINTÉTICO

PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

Desarrolla proyectos de software, a través de las propiedades del paradigma de programación, arquitectura y aplicación de metodologías, para solucionar problemáticas en general, de forma colaborativa y creativa.

N°	UNIDAD DE COMPETENCIA	APRENDIZAJES ESPERADOS	CONTENIDOS DE APRENDIZAJE/SABERES
1	Define el análisis y planeación del desarrollo de proyectos de software, usando las propiedades del paradigma de programación y la arquitectura para dar soluciones a problemáticas variadas.	<p>1.1 Lleva a cabo la planeación y el análisis de una problemática, empleando alguna de las metodologías para el desarrollo de proyectos de software, y así proponer soluciones a problemas.</p> <p>1.2 Hacer la planeación de un proyecto, realizando el análisis y ejecución de cada una de las etapas de</p>	<p>Conceptual:</p> <p>Fundamentos del proyecto, ciclo de vida de un proyecto. Los principios de desarrollo de proyectos de software Las metodologías de desarrollo de proyectos de software El plan estratégico El diagrama de Gantt y ruta crítica El proyecto de software</p> <p>Procedimental:</p> <p>Diagnostica la problemática de un proyecto de software describiendo sus componentes. Desarrolla el cronograma de actividades. Planea el proyecto a través del diagrama de Gantt y ruta crítica. Emplea alguna de las metodologías de desarrollo de proyectos de software, para dar solución a la problemática.</p> <p>Actitudinal:</p> <p>Trabaja en equipo con sus compañeros y docentes permitiendo el desarrollo de las actividades de una manera armoniosa y colaborativa. Tiene un pensamiento creativo y analítico para la resolución de diversos problemas Utiliza la Innovación en la planeación del proyecto</p> <p>Conceptual:</p> <p>El proceso del negocio y el estudio de mercado la viabilidad y el alcance de un proyecto Etapas de la metodología de solución de problemas Planeación del proyecto de software</p>





Programa Académico: Técnico en Programación

Unidad de Aprendizaje: Laboratorio de Desarrollo de Software II


		<p>la metodología utilizada, para solucionar un problema de la vida real.</p>	<p>El análisis de requerimientos funcionales, no funcionales y de sistema</p> <p>Procedimental:</p> <p>Describe el proceso del negocio y el estudio de mercado. Propone soluciones al problema por medio del análisis del proyecto de software. Elige una propuesta de proyecto de software, para dar solución a una problemática.</p> <p>Actitudinal:</p> <p>Trabaja en equipo con sus compañeros y docentes permitiendo el desarrollo de las actividades de una manera armoniosa y colaborativa. Muestra un autodesarrollo y disposición hacia el aprendizaje. Tiene proactividad e iniciativa para la solución de diversas problemáticas comunes, así como tiene la capacidad para anticiparse a problemas o necesidades futuras</p>
2	<p>Elabora el proyecto de software a través del lenguaje de modelado considerando la arquitectura y la codificación de programas en un lenguaje de programación, de forma comprometida y autónoma.</p>	<p>2.1 Utiliza los elementos del diseño de software, dando solución a la problemática planteada de forma analítica adaptándolo a los cambios tecnológicos.</p>	<p>Conceptual:</p> <p>Lenguaje de modelado para el diseño del software Elabora la solución de problemas reales usando algún paradigma de programación. El modelo cliente-servidor Las páginas web y su seguridad El diseño y la codificación del proyecto de software mediante un lenguaje de programación</p> <p>Procedimental:</p> <p>Diseña la aplicación de software a través de un lenguaje de modelado Elabora la solución de problemas reales usando algún paradigma de programación. Efectúa la codificación en un lenguaje de programación de alto nivel.</p> <p>Actitudinal:</p> <p>Tiene un pensamiento creativo y analítico para la resolución de diversos problemas.</p>





Programa Académico: Técnico en Programación

Unidad de Aprendizaje: Laboratorio de Desarrollo de Software II

		<p>2.2 Elabora en un lenguaje de programación de alto nivel la codificación del proyecto de software, planteado de forma colaborativa y empleando una base de datos</p>	<p>Aprende a ser resiliente al comprometerse y seguir adelante con lo encomendado Aprende de forma autónoma. Se adapta rápidamente a los cambios tecnológicos de la 4T, entendiendo de manera responsable el uso del software</p> <p>Conceptual: Mapa de navegación. El Look & Feel El lenguaje de programación Programación orientada a objetos o algún otro paradigma. Entornos de desarrollo integrados (IDE) La programación con nuevas tecnologías Modelado de la base de datos (modelo entidad-relación, diccionario de datos y modelo relacional) y/o Modelado de la base de datos no relacional.</p> <p>Procedimental: Realiza el mapa de navegación y el Look & Feel Utiliza un lenguaje de programación para la codificación del proyecto de software. Hace uso de un entorno de desarrollo integrado. Realiza el modelado y la codificación de la base de datos relacional o no relacional.</p> <p>Actitudinal: Aprende de forma autónoma Aprecia el pensamiento creativo y de resolución de problemas. Trabaja en equipo con sus compañeros y docentes permitiendo el desarrollo de las actividades de una manera armoniosa y colaborativa.</p>
3	<p>Aplica las pruebas al proyecto de software desarrollado considerando la planeación de pruebas, ocupando técnicas, para garantizar la calidad del producto.</p>	<p>3.1 Garantizar la calidad del proyecto de software desarrollado por medio de pruebas.</p>	<p>Conceptual: Tipos de pruebas de software Plan de pruebas Calidad del software Presentación del proyecto de software</p> 



Programa Académico: Técnico en Programación

Unidad de Aprendizaje: Laboratorio de Desarrollo de Software II

		<p>3.2 Realiza la implementación del proyecto de software, cumpliendo con los requerimientos determinados, y de acuerdo a la gestión del proyecto</p>	<p>Procedimental:</p> <p>Efectúa algunos tipos de pruebas de software Elabora el plan de pruebas, para asegurar la funcionalidad. Realiza la presentación del proyecto de software que dé solución a la problemática planteada</p> <p>Actitudinal</p> <p>Muestra Resiliencia, Iniciativa. Capacidad de toma de decisiones de manera responsable. Se muestra competitivo y proactivo en su participación. Aprecia el pensamiento creativo y resolución de problemas.</p> <p>Conceptual:</p> <p>La implementación del proyecto informático El plan de mejora continua. La administración del proyecto de software</p> <p>Procedimental:</p> <p>Realiza la implementación del proyecto de software, de acuerdo a lo requerido. Elabora el plan de mejora continua. Hace la gestión del proyecto de software</p> <p>Actitudinal:</p> <p>Se muestra competitivo. Aprecia el pensamiento creativo y de resolución de problemas. Se interesa en actualizar permanentemente habilidades. Valora la positividad, el orden. Aprecia la responsabilidad y es comprometido</p>
--	--	---	---





Programa Académico: Técnico en Programación

Unidad de Aprendizaje: Laboratorio de Desarrollo de Software II

BIBLIOGRAFÍA BÁSICA Y COMPLEMENTARIA

Número y Nombre de la Unidad Didáctica	FORMATO APA	CLASIFICACIÓN	
		Básico	Consulta
Unidad didáctica 1: : Análisis y Planeación	Pressman, Roger (2010). Ingeniería del Software. Un enfoque Práctico. Mc Graw Hill	X	
Unidad didáctica 1: : Análisis y Planeación	Aguilar, Joyanes. Luis (2022). Fundamentos De Programación (4.a ed.). Mc. GRAW HILL EDUCATION.	X	
Unidad didáctica 1: : Análisis y Planeación	Peña, Leonel (2020). Como Desarrollar un Proyecto de Software. Edición Kindle	X	
Unidad 2: Diseño y codificación	Sznajdleder, Pablo (2012). Algoritmos a Fondo con Implementaciones en C y Java. Alfaomega	X	
Unidad 2: Diseño y codificación	Somerville, Ian (2010). Ingeniería del Software. Editorial Pearson	X	
Unidad 3 Pruebas e implementación	Pressman, Roger (2005). Ingeniería del Software. Un enfoque Práctico. Mc Graw Hill	X	
Unidad 3 Pruebas e implementación	Somerville, Ian (2010). Ingeniería del Software. Editorial Pearson		X

