




Formato para registro de Unidades de aprendizaje 2021

INSTRUCTIVO para el correcto llenado del formato SIP-30

- El formato SIP-30 es un formato digital el cual puede ser completado con un procesador de texto y guardarse como archivo PDF para su envío.
- Adicionalmente será necesario anexar la solicitud firmada por el director de la Unidad Académica respectiva y el acuerdo de Colegio donde se avaló su registro; tenga listos los archivos al momento de ingresar su solicitud en el formulario en línea.
- El enlace de atención única para esta y otras gestiones es: <https://forms.office.com/r/c8DLS6VBv1> (copie y pegue en un navegador web si el enlace no funciona)
- Tome en cuenta los criterios establecidos en el Reglamento de Estudios de Posgrado ([REP 2017](#)) para el llenado de este formato, a continuación se presentan algunas definiciones útiles:
 - *Número de semanas por semestre del programa:* Es el número de semanas lectivas efectivas al semestre, indicadas en el acuerdo de creación del programa académico o en alguna actualización posterior del programa. En caso de haber tenido una actualización en este sentido, la misma deberá haber sido presentada y avalada en reunión del Colegio de Profesores de la Unidad Académica, además de haber sido aprobada por la SIP. El rango de semanas lectivas al semestre es mínimo 15 y máximo 18.
 - *Tipo de horas:* Las unidades de aprendizaje, en cuanto a las horas asignadas, están clasificadas como: Teóricas, Prácticas y Teórico-prácticas. Estas denominaciones son excluyentes, es decir, las unidades de aprendizaje solo pueden ser de un solo tipo, no pueden tener horas combinadas.
 - *Número de horas – semana:* Es el número de horas asignadas para ser impartida la Unidad de Aprendizaje a la semana.
 - *Total de horas al semestre:* Es el número de horas totales a impartir de la Unidad de Aprendizaje al semestre. Se calcula multiplicando Número de semanas por número de horas-semana.
 - *Créditos* (Reglamento de Estudios de Posgrado 2017): FÓRMULA DE CÁLCULO: $16 \text{ hrs.} = 1 \text{ crédito}$ (horas totales / 16), no deben asignarse fracciones, los créditos deben redondearse a número entero.
- Para el registro de unidades de aprendizaje de modalidad no escolarizada o mixta incluya adicionalmente los campos marcados con el color azul
- En todos los campos existen comentarios en forma de  globo que sirven de ayuda para el requisitado correspondiente, en caso de duda solicite apoyo del asesor didáctico de la UTEyCV de su Unidad Académica.



Formato para registro de Unidades de aprendizaje 2021

I.- Datos de identificación de la unidad de aprendizaje

Unidad académica:	Centro de Investigación en Computación (CIC)										
Programa académico:	Maestría en Ciencias de la Computación (MCC)										
	Doctorado				Orientación profesional						
X	Maestría				X	Orientado a la investigación					
	Especialidad				Con la industria						
					Especialidad médica						
	Sesión de colegio donde se propuso:				Fecha de propuesta:						
Nombre de unidad de aprendizaje:	Ingeniería de Software										
	Clave de la unidad de aprendizaje:						Créditos:		5		<i>REP 2017</i>
	Semanas del semestre		18	Horas a la semana:		4	Horas totales:		72		
Tipo de unidad de aprendizaje:	Obligatoria:		Optativa:		X	Observaciones: Actualización de la asignatura					
	Semestre:										
	Teórica (%):	100	Práctica (%):			Teórico-prácticas (%):					
Área del conocimiento:	Ingeniería y Ciencias Fisicomatemáticas		X	Ciencias Sociales y Administrativas			Ciencias Médico Biológicas		Interdisciplinario		
Modalidad no escolarizada:	No escolarizada		Nombre de la Plataforma:								
	Mixta		Presencial (%):			En plataforma (%):					
Horas establecidas en el programa de estudios:	Presenciales (si procede) (horas x semana)						En plataforma (horas x semana):				



Formato para registro de Unidades de aprendizaje 2021

I. Aprendizajes que el estudiante deberá demostrar al finalizar

Conocimientos	Habilidades y destrezas	Actitudes y valores
<ul style="list-style-type: none">• Dominar herramientas y técnicas desarrolladas en el marco de la Ingeniería de Software.• Evaluar y planificar la elaboración de los productos de software.• Análisis, diseño y estrategias del desarrollo de sistemas.• Evaluación del comportamiento de los sistemas de software.• Manejar métodos de pruebas e implantaciones viables.• Tomar decisiones mediante la aplicación de metodologías para contribuir en la eliminación de la crisis del software.• Adaptar los diferentes paradigmas de desarrollo en la industria.	<ul style="list-style-type: none">• Para la implementación de productos de software fiable y eficiente• Desarrollar habilidades para el correcto mantenimiento del producto de software• Para el cumplimiento de normas de calidad, teorías, políticas, metodologías y prácticas de la Ingeniería del Software• Para la estimación de servicios y productos de software que satisfagan los requerimientos de los clientes y usuarios• Para una comunicación adecuada con el cliente y usuarios• Para una correcta especificación de requerimientos, estimación de tiempos y costos dentro de los alcances del producto de software• Para proporcionar soluciones a los problemas de elaboración de software de las empresas con base en estrategias, estándares y tecnologías disponibles• En el proceso de resolución de problemas, análisis y diseño, desarrollo, implementación, verificación del producto de software• Para generar documentación que lleven a un proceso de construcción de	<ul style="list-style-type: none">• Adquirir <i>Soft skills</i> que son requeridas en el ingeniero de software:<ul style="list-style-type: none">• Capacidad de análisis y síntesis• Resolución de problemas• Organización y planificación• Comunicación oral y escrita• Uso de las tecnologías de la información y las comunicaciones• Autonomía• Compromiso• Responsabilidad• Ganas de aprender• Habilidades para trabajo en equipo (comunicación)• Motivación• Orientación a resultados



Formato para registro de Unidades de aprendizaje 2021

	<p>soluciones de software con base en un conocimiento adecuado de teorías, modelos y técnicas actuales</p> <ul style="list-style-type: none">• Para lograr una completa gestión de riesgos potenciales asociados al desarrollo de un proyecto de software• Para que las soluciones de software propuestas tengan la base de la Ingeniería de Software en el dominio de la aplicación; empleando metodologías o métodos que consideren atributos éticos, sociales, legales y económicos• Para lograr una adecuada estimación de costos y tiempos - seguimiento y control	
--	---	--

Resolución que aborda la propuesta con su enfoque disciplinar

Dentro de las Ciencias de la Computación y la informática, la Ingeniería de Software se ha convertido en una de las áreas clave para poder ofrecer productos de software y tecnología a las empresas y a la sociedad, que sea considerada eficiente y de calidad.

Los estudiantes del programa de maestría desarrollan su tesis de grado pueden proponer como parte de su trabajo de investigación el desarrollo de un producto de software o bien, un modelo o un método que puede ser útil para ser aplicado en las diferentes especialidades que tiene las ciencias de la computación y la Ingeniería de Software, les proporciona las bases para delimitar alcances, definir adecuadamente el problema a resolver y proponer una solución, creando una documentación adecuada que le permita dar seguimiento y control al producto (método, modelo, metodología, softwar) como resultado de su investigación.

Los objetivos de la ingeniería de software son diversos y se espera que, entre ellos, el estudiante al terminar el curso pueda: crear programas informáticos que satisfagan las necesidades de la sociedad y empresas; guiar y coordinar el desarrollo de un producto de software; intervenir en el ciclo de vida de un producto; estimar los costos y el plazo de ejecución de un proyecto; actuar como líder de un equipo de desarrollo de software; diseñar, desarrollar y administrar bases de datos; durante la creación de la aplicación, liderar y dirigir a los programadores; incluir procesos de calidad en las aplicaciones, como la medición de métricas y medidas y la evaluación de la calidad del software.



Formato para registro de Unidades de aprendizaje 2021

Áreas multi, inter y transdisciplinarias	Líneas de Generación y Aplicación de Conocimiento	Sector social
<ul style="list-style-type: none"> • Ciencias de la Computación (fundamentos de programación, análisis numérico) • Ingeniería de la Computación (algoritmos, arquitectura y organización, interacción humano-computadora, circuitos y sistemas) • Ergonomía del software (procesamiento léxico, adaptación humano-computadora) • Gestión (Marketing, ventas, política de negocio y estrategia) • Gestión de calidad (métodos, planeación, seguimiento y control de procesos de calidad) • Ingeniería de sistemas (levantamiento de requerimientos, diseño de sistemas correctamente acoplados, integración de equipos especializados) • Matemáticas (álgebra lineal, probabilidad y estadística, matemáticas discretas, análisis numérico) • Gestión de proyectos (administración de recursos, costos, riesgos, integración de proyectos) 	<ul style="list-style-type: none"> • Una persona (estudiante) con conocimiento de Ingeniería de Software, es capaz de desempeñarse como: <ul style="list-style-type: none"> - Director de TI. - Desarrollador y analista de aplicaciones de cualquier tipo (Web, móvil, etc.). - Desarrollador de software. - Ingeniero de software y hardware. - Director de proyectos de I+D+I (Investigación, desarrollo e innovación). - Administrador de Sistemas Informáticos - Diseñador de sitios web. - Jefe de proyecto. - Jefe de usabilidad y experiencia de usuarios. - Experto en ciberseguridad. - Especialista en integración y pruebas. - Consultor relacionado con tecnologías de la información (TI). - Gestor de Programación ágil. - Gestor/Responsable de calidad. 	<ul style="list-style-type: none"> • Sectores productivos o privados (centros de investigación, compañías de entretenimiento, etc.). • Sector público o social. • Sectores comunitarios. • Sector personal (Proyectos personales). • Sector investigación (Proyectos de Investigación). • Academia (Proyectos académicos). • Sector consultoría y asesoría. • Sector servicios. • Sector gubernamental. • Organismos descentralizados. • Sector financiero. • Sector de comunicaciones y transportes.
<p>Los conocimientos adquiridos en este curso le permiten al participante desempeñarse profesionalmente como líderes de proyecto, directores de tecnologías de la información, diseñadores de software, probadores de software, ingenieros (desarrollo, seguimiento y control de su propios proyectos como tesis, tesis de grado o investigadores en servicio social o dentro del programa BEIFI), participantes de proyectos (de la institución o vinculados o participantes como estudiantes BEIFI), así como todo profesionista con responsabilidad de la ejecución de proyectos (de investigación, desarrollo e innovación) de cualquier especialidad, en alguna de sus etapas, ya sean éstos de corto, mediano o largo plazo, también</p>		



Formato para registro de Unidades de aprendizaje 2021

los participantes tienen la posibilidad de desempeñarse adecuadamente como participantes de proyectos vinculados con instituciones públicas o privadas.

III Metodología de enseñanza – aprendizaje

Descripción

Evidencias como proceso de aprendizaje

Evidencias integradoras (resultados que contribuyen al curriculum)	Ponderación

IV. Descripción de la participación esperada en el estudiante

Receptiva	Resolutiva	Autónoma	Estratégica
-----------	------------	----------	-------------



Formato para registro de Unidades de aprendizaje 2021

--	--	--	--

Contenido temático

1 Conceptos fundamentales de la Ingeniería de Software.	4 HORAS
1.1 Especificación de requisitos del software	
1.2 Métodos y Modelos de Desarrollo de Software	
1.3 Diseño y Arquitectura de software	
1.4 Desarrollo integrado de productos y procesos	
1.5 Equipos de productos integrados	
1.6 Estructura de desglose del trabajo	
1.7 Estructura de desglose del Software	
1.8 Árboles de especificación y documentación	
1.9 Plan Maestro Integrado y Calendario	
1.10 Revisiones y auditorías	
1.11 Administración de la Configuración y Control del Cambio	
1.12 Análisis de compensaciones	
1.13 Manejo de riesgos	
1.14 Modelado y simulación	
2 Requisitos del Software	10 HORAS
2.1 Fundamentos de los requisitos del software	
2.2 Proceso de requisitos	
2.3 Levantamiento de requisitos	
2.4 Análisis de requisitos	
2.5 Especificación de requisitos	



Formato para registro de Unidades de aprendizaje 2021

2.6 Validación de requisitos	
2.7 Consideraciones prácticas de los requisitos del software	
2.8 Herramientas para tratar los requisitos del software	
3 Diseño del software	10 HORAS
3.1 Fundamentos del diseño del software	
3.2 Elementos clave en el diseño de software	
3.3 Arquitectura y estructura el software	
3.4 Diseño de Interfaces de usuario	
3.5 Análisis y evaluación de la calidad del diseño del software	
3.6 Notaciones para el diseño del software	
3.7 Métodos y estrategias para el diseño del software	
3.8 Herramientas para tratar el diseño del software	
4 Construcción del software	10 HORAS
4.1 Fundamentos de la construcción del software	
4.2 Gestión de la construcción	
4.3 Consideraciones prácticas	
4.4 Tecnologías para la construcción	
4.5 Herramientas para tratar la construcción del software	
5 Pruebas del software	6 HORAS
5.1 Fundamentos de las pruebas del software	
5.2 Niveles de pruebas	
5.3 Técnicas de pruebas	
5.4 Medidas relacionadas con las pruebas	
5.5 Proceso de pruebas	
5.6 Herramientas para tratar las pruebas del software	
6 Mantenimiento del software	3 HORAS
6.1 Fundamentos del mantenimiento del software	
6.2 Elementos clave del mantenimiento del software	
6.3 Proceso de mantenimiento	



Formato para registro de Unidades de aprendizaje 2021

6.4 Técnicas para el mantenimiento	
6.5 Herramientas para tratar el mantenimiento del software	
7 Gestión de la Configuración del Software	5 HORAS
7.1 Proceso para la gestión de la configuración del software	
7.2 Identificación de la configuración del software	
7.3 Control de la configuración del software	
7.4 Contabilización del estado de la configuración del software	
7.5 Auditoría de la configuración del software	
7.6 Gestión y entrega de software	
7.7 Herramientas para la gestión de la configuración del software	
8 Gestión de la ingeniería del software	5 HORAS
8.1 Inicio y definición del alcance	
8.2 Planeación de un proyecto de software	
8.3 Promulgación de proyectos de software	
8.4 Revisión y evaluación	
8.5 Cierre	
8.6 Métricas en ingeniería de software	
8.7 Herramientas para la gestión de la ingeniería de software	
9. Proceso de Ingeniería de Software	5 HORAS
9.1 Definición del proceso del software	
9.2 Ciclos de vida del software	
9.3 Evaluación y mejora de los procesos de software	
9.4 Métricas del software	
9.5 Herramientas para el proceso de Ingeniería de Software	
10. Métodos y modelos de la Ingeniería de Software	2 HORAS
10.1 Modelado	
10.2 Tipos de modelo	
10.3 Modelos del análisis	
10.4 Métodos de la Ingeniería de Software	



Formato para registro de Unidades de aprendizaje 2021

11. Calidad del software	2 HORAS
11.1 Fundamentos de la calidad del software	
11.2 Procesos de la gestión de la calidad del software	
11.3 Consideraciones prácticas de la calidad del software	
11.4 Herramientas para la gestión de la calidad del software	
12. Práctica profesional de la ingeniería del software	2 HORAS
12.1 Profesionalismo	
12.2 Psicología y Dinámica de grupos	
12.3 Habilidades de comunicación	
13. Economía de la Ingeniería de Software	2 HORAS
13.1 Fundamentos de la economía del software	
13.2 Economía del ciclo de vida	
13.3 Riesgo e incertidumbre	
13.4 Métodos de análisis económico	
13.5 Consideraciones Prácticas	
14. Fundamentos de la informática	2 HORAS
14.1 Técnicas de resolución de problemas	
14.2 Abstracción	
14.3 Fundamentos de programación	
14.5 Fundamentos de los lenguajes de programación	
14.6 Herramientas y técnicas de depuración	
14.7 Estructura y representación de datos	
14.8 Algoritmos y complejidad	
14.9 Concepto básico de un sistema	
14.10 Organización de las computadoras	
14.11 Conceptos básicos sobre el compilador	
14.12 Conceptos básicos sobre el sistema operativo	
14.13 Bases de datos y gestión de datos	
14.14 Aspectos básicos de la comunicación en red	
14.15 Computación paralela y distribuida	



Formato para registro de Unidades de aprendizaje 2021

14.16 Factores humanos básicos del usuario	
14.17 Factores humanos básicos del desarrollador	
14.18 Desarrollo y mantenimiento de software seguro	
15. Fundamentos matemáticos que debe conocer un Ingeniero de Software	2 HORAS
15.1 Conjuntos, relaciones y funciones	
15.2 Lógica básica	
15.3 Técnicas de prueba	
15.4 Fundamentos del conteo	
15.5 Grafos y arboles	
15.5 Probabilidad discreta	
15.6 Máquinas de estado finito	
15.7 Gramáticas	
15.8 Precisión numérica, exactitud y errores	
15.9 Teoría de los números	
15.10 Estructuras algebraicas	
16. Fundamentos de Ingeniería requeridos en la Ingeniería de Software	2 HORAS
16.1 Métodos empíricos y técnicas experimentales	
16.2 Análisis estadístico	
16.3 Medición	
16.4 Diseño en ingeniería	
16.5 Modelización, simulación y creación de prototipos	
16.6 Estándares	
16.7 Análisis de la causa raíz	
	TOTAL: 72 HORAS

V. Secuencia programática

No.	Tema	Objetivo de aprendizaje / competencia específica	Tiempo/Horas/Semanas
-----	------	--	----------------------



Formato para registro de Unidades de aprendizaje 2021

Actividad(es):	No. Nombre de la actividad: Descripción de la actividad:	Tipo de interacción(es):	
		Referencias (s):	
Evidencia(s):			

Tipo de interacción: ID–Instrucción directa, TC–Trabajo colaborativo, AC–Análisis en campo, RP–Reflexión personal, PE–Presentación expositiva

Indicar solo el número de las *Referencias* indizadas en la sección VII de este documento.

Nota: Replique esta sección las veces que sea necesario para cubrir toda la secuencia programática

VI. Habilitadores tecnológicos

Disposiciones		Especificaciones / descripción de efectos
	Conectividad	
	Habilidades digitales	
	Interoperabilidad	
	Datos abiertos	
	<i>Big Data</i>	
	<i>Machine Learning</i>	
	Simulación	
	Realidad aumentada	
	Otro...	

VII. Referencias



Formato para registro de Unidades de aprendizaje 2021

Conferencias magistrales

1. Modern Software Engineering. Dave Farley & Steve Smith GOTO Conferences 2022 https://www.youtube.com/watch?v=80LbSyTCKas
2. Keynote - The Last Twenty Years of Software Development Richard Campbell - NDC Conferences - London 2022 https://www.youtube.com/watch?v=YmPRDnCTtQ
3. The Next Decade of Software Development - Richard Campbell – NDC conferences - Oslo 2021 https://www.youtube.com/watch?v=a_LM3FJozr0

Notas complementarias

Documentales / electrónicas

<ol style="list-style-type: none">1. Bures, Tomas, Duchien, Laurence, Inverardi, Paola (Eds.). (2019). Software Architecture. 13th European Conference, ECSA 2019, Paris, France, September 9–13, 2019, Proceedings.2. Björn Regnell, D. D. (2012). "Requirements Engineering: Foundation for Software Quality". Springer. ISBN: 3642287131.3. Capers J. (2014). "The Technical and Social History of Software Engineering", Addison-Wesley& Pearson Education, Inc.4. Chang, S.K. (Shi Kuo) (2012). "Multimedia Software Engineering". Springer Science+Business Media New York. Originally published by Kluwer Academic Publishers 2000.5. Fairley, R. E. (1990). Ingeniería del software; MEXICO; McGraw-Hill.6. IEEE Standards Collection (1990). "Software Engineering: IEEE Computer Soc" Publisher: Institute of Electrical Electronics Engineering. ISBN: 1559370645.7. LNCS 6013 (2010). "Fundamental Approaches to Software Engineering". Publisher: Springer, 396 Pages. ISBN: 3642120288. This book constitutes the refereed proceedings of the 13th International Conference on Fundamental Approaches to Software Engineering, FASE 2010.8. Naveda J.F., Seidman S.B. (2006). "IEEE Computer Society Real-World Software Engineering Problems: A Self-Study Guide for Today's Software Professional". IEEE Computer Society. Wiley-Interscience A John Wiley & Sons, INC., Publication.9. Sage, A. and Rouse W. (2009). Handbook of Systems Engineering and Management, second ed. Hoboken, NJ, USA: John Wiley and Sons, Inc.10. Sicilia Urbán M.A., Rodríguez García D. (2012). "Ingeniería del software: un enfoque desde la guía SWEBOK". Alfaomega, 547 páginas.11. Sommerville I. (2011). Software Engineering. 9th Edition. Addison-Wesley. Pearson Education, Inc.12. SWEBOK Guide V3.0 (2014). IEEE Computer Society. ISBN-10: 0-7695-5166-1. May be downloaded free of charge for personal and academic use via www.swebok.org.13. W. Cazzola, R.J., Stroud T. (2000). "Reflection and Software Engineering (Lecture Notes in Computer Science)". Springer; Primera edition. English. August 25, 2000. ISBN: 3540677615. 227 pgs.



Formato para registro de Unidades de aprendizaje 2021

VIII. Créditos y responsabilidades

Responsabilidad	Nombre completo	Clave de nombramiento /No. de empleado
Coordinador (Autor)	Dra. Sandra Dinora Orantes Jiménez	14671-EH-19/6

VERIFICACIÓN GENERAL DE LA PLANEACIÓN DIDÁCTICA	REVISIÓN DE LA PLANEACIÓN DIDÁCTICA (VIABILIDAD)
Por la División de Operación y Promoción al Posgrado de la SIP	Por la Subdirección de Diseño y Desarrollo de la DEV
Nombre _____	Nombre _____
FIRMA _____	FIRMA _____

VERIFICACIÓN PARA SU PUESTA EN OPERACIÓN	REVISIÓN TÉCNICO-PEDAGÓGICA PARA LA MODALIDAD
Por la Dirección de Posgrado	Por la Dirección para la Educación Virtual
Nombre _____	Nombre _____
FIRMA _____	FIRMA _____
SELLO DE VALIDACIÓN	



Instituto Politécnico Nacional

Secretaría Académica
Dirección de Educación Virtual

Secretaría de Investigación y Posgrado
Dirección de Posgrado

SIP-30

Formato para registro de Unidades de aprendizaje 2021

--	--