


**TÉCNICO SUPERIOR UNIVERSITARIO EN
TECNOLOGÍAS DE LA INFORMACIÓN ÁREA
DESARROLLO DE SOFTWARE MULTIPLATAFORMA
EN COMPETENCIAS PROFESIONALES**

**ASIGNATURA DE ESTÁNDARES Y MÉTRICAS PARA EL DESARROLLO DE
SOFTWARE**

1. Competencias	Implementar soluciones multiplataforma, en la nube y software embebido, en entornos seguros mediante la adquisición y administración de datos e ingeniería de software para contribuir a la automatización de los procesos en las organizaciones.
2. Cuatrimestre	Cuarto
3. Horas Teóricas	23
4. Horas Prácticas	22
5. Horas Totales	45
6. Horas Totales por Semana Cuatrimestre	3
7. Objetivo de aprendizaje	El alumno evaluará las metodologías y modelos existentes en la industria mediante el uso de estándares y métricas para asegurar la calidad de proyectos de desarrollo de software.

Unidades de Aprendizaje	Horas		
	Teóricas	Prácticas	Totales
I. Introducción a la calidad en el desarrollo de software	3	3	6
II. Técnicas de estimación	6	12	18
III. Metodologías y modelo de madurez en el desarrollo de software	14	7	21
Totales	23	22	45


ELABORÓ:	Comité técnico académico de diseño curricular del subsistema de CGUTyP de la familia de carreras de TSU en Tecnologías de la Información	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2018	

ESTÁNDARES Y MÉTRICAS DE DESARROLLO DE SOFTWARE

UNIDADES DE APRENDIZAJE

1. Unidad de aprendizaje	I. Introducción a la calidad en el desarrollo de software
2. Horas Teóricas	3
3. Horas Prácticas	3
4. Horas Totales	6
5. Objetivo de la Unidad de Aprendizaje	El alumno identificará los conceptos generales asociados a la calidad para aplicar las métricas en el desarrollo de software.


Temas	Saber	Saber hacer	Ser
Generalidades y conceptos de la calidad en el desarrollo de software.	Identificar conceptos de calidad, normas, estándares y procesos aplicables al desarrollo de software.		Organizado. Analítico. Sistemático. Autodidacta.
Conceptos y métricas de calidad en el desarrollo de software.	Identificar el concepto de métrica. Identificar los tipos de métricas asociados a los factores y características que determinan la calidad del software.	Seleccionar las métricas de acuerdo al contexto del desarrollo de software.	Organizado. Analítico. Sistemático. Autodidacta.

ELABORÓ:	Comité técnico académico de diseño curricular del subsistema de CGUTyP de la familia de carreras de TSU en Tecnologías de la Información	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2018	

ESTÁNDARES Y MÉTRICAS DE DESARROLLO DE SOFTWARE

PROCESO DE EVALUACIÓN

Resultado de aprendizaje	Secuencia de aprendizaje	Instrumentos y tipos de reactivos
<p>Elaborará un documento que contenga:</p> <ul style="list-style-type: none"> • Mapa conceptual de los institutos, estándares y normas que regulan la calidad en el desarrollo de software. • Tabla de factores y características que determinan la calidad en el desarrollo de software. • Cuadro sinóptico de aplicación de métricas. 	<ol style="list-style-type: none"> 1. Identificar los conceptos generales asociados a la calidad. 2. Analizar los factores y características que determinan la calidad. 3. Comprender el concepto de métrica. 4. Identificar los tipos de métricas considerando factores y características a medir. 	<ul style="list-style-type: none"> - Estudio de casos. - Listas de cotejo.

ELABORÓ:	Comité técnico académico de diseño curricular del subsistema de CGUTyP de la familia de carreras de TSU en Tecnologías de la Información	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2018	


ESTÁNDARES Y MÉTRICAS DE DESARROLLO DE SOFTWARE

PROCESO ENSEÑANZA APRENDIZAJE

Métodos y técnicas de enseñanza	Medios y materiales didácticos
<ul style="list-style-type: none"> - Aprendizaje basado en proyectos. - Estudio de casos. - Discusión en grupo. 	Pizarrón. Plumones. Computadora. Internet. Equipo multimedia. Plataformas virtuales.

ESPACIO FORMATIVO

Aula	Laboratorio / Taller	Empresa
X	X	


ELABORÓ:	Comité técnico académico de diseño curricular del subsistema de CGUTyP de la familia de carreras de TSU en Tecnologías de la Información	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2018	

ESTÁNDARES Y MÉTRICAS DE DESARROLLO DE SOFTWARE

UNIDADES DE APRENDIZAJE

1. Unidad de aprendizaje	II. Técnicas de estimación
2. Horas Teóricas	6
3. Horas Prácticas	12
4. Horas Totales	18
5. Objetivo de la Unidad de Aprendizaje	El alumno empleará las técnicas de estimación para determinar el tamaño del software y el esfuerzo requerido en el desarrollo.


Temas	Saber	Saber hacer	Ser
Puntos de función.	Identificar el procedimiento de la estimación de puntos de función.	Calcular el tamaño del software de acuerdo a la cuenta ajustada de puntos de función.	Organizado. Analítico. Sistemático. Autodidacta.
Puntos de casos de uso	Identificar el procedimiento de la estimación del esfuerzo utilizando la técnica de casos de uso.	Calcular el esfuerzo requerido en el desarrollo de software de acuerdo a los casos de uso del software.	Organizado. Analítico. Sistemático. Autodidacta.

ELABORÓ:	Comité técnico académico de diseño curricular del subsistema de CGUTyP de la familia de carreras de TSU en Tecnologías de la Información	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2018	

ESTÁNDARES Y MÉTRICAS DE DESARROLLO DE SOFTWARE

PROCESO DE EVALUACIÓN

Resultado de aprendizaje	Secuencia de aprendizaje	Instrumentos y tipos de reactivos
<p>Elaborará un documento con base en un caso de estudio que contenga:</p> <ul style="list-style-type: none"> • Estimación de la complejidad por puntos de función. • Estimación del esfuerzo por casos de uso. 	<ol style="list-style-type: none"> 1. Comprender el procedimiento para hacer el cálculo y estimar la complejidad por puntos de función. 2. Comprender el procedimiento para calcular el esfuerzo requerido para el desarrollo de software con base en casos de uso. 	<ul style="list-style-type: none"> - Estudio de casos. - Lista de cotejo.

ELABORÓ:	Comité técnico académico de diseño curricular del subsistema de CGUTyP de la familia de carreras de TSU en Tecnologías de la Información	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2018	


ESTÁNDARES Y MÉTRICAS DE DESARROLLO DE SOFTWARE

PROCESO ENSEÑANZA APRENDIZAJE

Métodos y técnicas de enseñanza	Medios y materiales didácticos
<ul style="list-style-type: none"> - Aprendizaje basado en proyectos. - Estudio de casos. - Discusión en grupo. 	Pizarrón. Plumones. Computadora. Internet. Equipo multimedia. Plataformas virtuales.

ESPACIO FORMATIVO

Aula	Laboratorio / Taller	Empresa
X	X	


ELABORÓ:	Comité técnico académico de diseño curricular del subsistema de CGUTyP de la familia de carreras de TSU en Tecnologías de la Información	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2018	

ESTÁNDARES Y MÉTRICAS DE DESARROLLO DE SOFTWARE


UNIDADES DE APRENDIZAJE

1. Unidad de aprendizaje	III. Metodologías y modelo de madurez en el desarrollo de software.
2. Horas Teóricas	14
3. Horas Prácticas	7
4. Horas Totales	21
5. Objetivo de la Unidad de Aprendizaje	El alumno identificará las metodologías de evaluación de desempeño y los modelos de aseguramiento de la calidad para el proceso de desarrollo de software

Temas	Saber	Saber hacer	Ser
Proceso Personal de Desarrollo de software (PSP).	Identificar los elementos y procedimientos del PSP.	Determinar el nivel personal de desempeño de acuerdo a la medición de tiempos y defectos.	Organizado. Analítico. Sistemático. Autodidacta. Trabajo en equipo. Comunicación oral y escrita.
Proceso de Desarrollo de Software en Equipo (TSP).	Identificar los elementos y procedimientos del TSP.	Determinar el nivel de desempeño del equipo de acuerdo a la medición de tiempos y defectos.	Organizado. Analítico. Sistemático. Autodidacta. Trabajo en equipo. Comunicación oral y escrita.
Integración de Modelos de Madurez de Capacidades (CMMI).	Identificar los niveles de madurez de la norma CMMI	Determinar el alcance de los componentes de las áreas claves del proceso en el nivel 2 de CMMI.	Organizado. Analítico. Sistemático. Autodidacta. Trabajo en equipo. Comunicación oral y escrita.

ELABORÓ:	Comité técnico académico de diseño curricular del subsistema de CGUTyP de la familia de carreras de TSU en Tecnologías de la Información	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2018	


Modelo de Proceso de Software (MoProSoft).	Identificar los niveles de madurez del estándar MoProSoft de acuerdo a la norma mexicana de calidad en desarrollo de software NMX-I-059/02.	Determinar la estructura y procesos de MoProSoft.	Organizado. Analítico. Sistemático. Autodidacta. Trabajo en equipo. Comunicación oral y escrita.
--	---	---	---

ELABORÓ:	Comité técnico académico de diseño curricular del subsistema de CGUTyP de la familia de carreras de TSU en Tecnologías de la Información	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2018	

ESTÁNDARES Y MÉTRICAS DE DESARROLLO DE SOFTWARE

PROCESO DE EVALUACIÓN

Resultado de aprendizaje	Secuencia de aprendizaje	Instrumentos y tipos de reactivos
<p>Elaborará un documento a partir de un caso de estudio que contenga:</p> <ul style="list-style-type: none"> ● Plantillas de PSP. ● Plantillas TSP. ● Tabla comparativa de los dos modelos de calidad (CMMI y MoProSoft). 	<ol style="list-style-type: none"> 1. Identificar los elementos y formatos para PSP y TSP. 2. Comprender el proceso de evaluación de PSP y TSP. 3. Comprender el alcance de la norma a nivel 2 de CMMI. 4. Comprender el alcance del estándar MoProSoft. 	<ul style="list-style-type: none"> - Estudio de casos. - Lista de cotejo.

ELABORÓ:	Comité técnico académico de diseño curricular del subsistema de CGUTyP de la familia de carreras de TSU en Tecnologías de la Información	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2018	


ESTÁNDARES Y MÉTRICAS DE DESARROLLO DE SOFTWARE

PROCESO ENSEÑANZA APRENDIZAJE

Métodos y técnicas de enseñanza	Medios y materiales didácticos
<ul style="list-style-type: none"> - Aprendizaje basado en proyectos. - Estudio de casos. - Discusión en grupo. 	Pizarrón. Plumones. Computadora. Internet. Equipo multimedia. Plataformas virtuales.

ESPACIO FORMATIVO


Aula	Laboratorio / Taller	Empresa
X	X	

ELABORÓ:	Comité técnico académico de diseño curricular del subsistema de CGUTyP de la familia de carreras de TSU en Tecnologías de la Información	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2018	


ESTÁNDARES Y MÉTRICAS DE DESARROLLO DE SOFTWARE

CAPACIDADES DERIVADAS DE LAS COMPETENCIAS PROFESIONALES A LAS QUE CONTRIBUYE LA ASIGNATURA

Capacidad	Criterios de Desempeño
Identificar la propuesta de solución a través de técnicas y herramientas de modelado, para determinar los requerimientos técnicos del sistema de información	Entrega un documento formal que incluya: <ul style="list-style-type: none"> ● Modelado de procesos: Casos de uso y diagrama de actividades ● Recursos: Humanos, Materiales, Financieros y Tiempos ● Riesgos ● Partes involucradas ● Propuesta de solución ● Costo
Establecer requerimientos funcionales y no funcionales de la solución mediante técnicas y metodologías de análisis de requerimientos para atender la necesidad planteada.	Entrega un documento formal de requerimientos que incluye: <ul style="list-style-type: none"> ● Requerimientos funcionales: Clave, descripción, reglas de negocio, criterios de aceptación, prioridad, usuarios y responsables ● Requerimientos no funcionales: tipo, fiabilidad, respuesta en el tiempo, capacidad de almacenamiento, restricciones de dispositivos de entrada / salida y la representación de datos que se utiliza en la interfaz del sistema. ● Requerimientos técnicos del sistema: tipo, función, característica, sistema operativo
Diseñar arquitectura del software mediante el modelado de los procesos y componentes para satisfacer los requerimientos técnicos y operacionales de la solución.	Entrega Un documento que incluya los diagramas UML de acuerdo a la propuesta de solución: <ul style="list-style-type: none"> ● Caso de uso ● Clases ● Secuencia ● Actividades ● Componentes ● Colaboración ● Estados ● Distribución
Implementar soluciones de software a través de la instalación y puesta en marcha para la liberación y cierre del proyecto.	Entrega la solución del software y lo documenta en: <ol style="list-style-type: none"> Plan de instalación que incluya: <ul style="list-style-type: none"> - Requerimientos de hardware y software - Requerimientos de infraestructura Plan de puesta en marcha y operación <ul style="list-style-type: none"> - Capacitación a usuarios - Pilotaje Acta de cierre de proyecto: <ul style="list-style-type: none"> - Empresa - Nombre del proyecto

ELABORÓ:	Comité técnico académico de diseño curricular del subsistema de CGUTyP de la familia de carreras de TSU en Tecnologías de la Información	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2018	


	<ul style="list-style-type: none"> - Cliente - Líder del proyecto - Módulos - Fecha de entrega - Firma de aceptación
--	---

ELABORÓ:	Comité técnico académico de diseño curricular del subsistema de CGUTyP de la familia de carreras de TSU en Tecnologías de la Información	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2018	

ESTÁNDARES Y MÉTRICAS DE DESARROLLO DE SOFTWARE

FUENTES BIBLIOGRÁFICAS

Autor	Año	Título del Documento	Ciudad	País	Editorial
Guillermo Pantaleo	2016 987383205X	<i>Calidad en el Desarrollo de Software</i>		México	Alfaomega
IEEE Computing Society, Pierre Bourque	2014 0769551661	<i>SWEBOK V3.0 Guide to The Software Engineering Body of Knowledge</i>		Estados Unidos	IEEE Computing Society
Ian Sommerville	2016 9780133943030	<i>Software Engineering</i>	Londres	Inglaterra	Pearson
Roger S. Pressman / Bruce Maxim	2015 9780078022128	<i>Software Engineering a Practitioner's Approach</i>	Nueva York	Estados Unidos	Mc Graw Hill
Nina S. Godbole	2016 9781842657027	<i>Software Quality Assurance: Principles and Practices</i>	Londres	Inglaterra	Alpha Science International
Mario Piattini	2016 8499645305	<i>Calidad de Sistemas de Información</i>		España	RA MA Editorial
Víctor Gómez Adán	2017 9781522070153	<i>Fundamentos de la Calidad del Software</i>			Publicación Independiente

ELABORÓ:	Comité técnico académico de diseño curricular del subsistema de CGUTyP de la familia de carreras de TSU en Tecnologías de la Información	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2018	