

PROGRAMA DE ESTUDIOS: MÉTRICAS DE SOFTWARE

PROCOLO

Fechas	Mes/año
Elaboración	05-2010
Aprobación	
Aplicación	09-2010

Clave			Semestre	9°		
Nivel	Licenciatura	X	Maestría		Doctorado	
Ciclo	Integración		Básico		Superior	X
Colegio	H. y C.S.		C. y T.	X	C. y H.	

Plan de estudios del que forma parte: Ingeniería de Software

Propósito(s) general(es):
El estudiante comprenderá y aplicará técnicas, para hallar y realizar métricas que pueda utilizar en las estimaciones para la administración de un proyecto.

Carácter		Modalidad				Horas de estudio semestral (16 semanas)					
Indispensable	X	Seminario		Taller		Con Docente	Teóricas	36	Autónomas	Teóricas	20
		Curso	X	Curso-taller			Prácticas	36		Prácticas	36
Optativa *		Laboratorio	X	Clínica		Carga horaria semanal: 4.5 + 3.5 = 8			Carga horaria semestral: 128		

Asignaturas Previas:	Asignaturas Posteriores:
Aseguramiento de la Calidad del software, Arquitectura de software	Diseño de experimentos en Ingeniería de Software

Requerimientos para cursar la asignatura	Conocimientos y habilidades: Haber comprendido la necesidad de asegurar la calidad del software de manera precisa basado en datos y conocer el proceso de software.
---	--

Perfil deseable del profesor:	Ingeniero en computación o alguna carrera afín, con maestría o doctorado de preferencia, en Ciencias de la Computación, o bien, amplios conocimientos y experiencia laboral en el área. Además, amplia experiencia docente y de investigación.
--------------------------------------	--

Academia responsable del programa: Informática	Diseñador (es): Candidato a M.C.C. Marcos López Chimil
--	--

* Aquellas en las que se ofrece la posibilidad de cursar una de las asignaturas, para cubrir un requisito INDISPENSABLE será considerada INDISPENSABLE.

INTRODUCCIÓN

La madurez de una organización, de acuerdo con el modelo Capability Maturity Model Integration, alcanza un alto grado cuando la organización tiene mediciones acerca de su proceso de software lo que le permite tener un proceso estable y predecible, objetivos altamente necesarios para lograr la calidad en los productos de software. La colección de estas métricas debe de hacerse en una organización determinada y utilizar adecuadamente la experiencia externa para lograr estimaciones y predicciones cada vez más precisas en la elaboración de los productos de software.

En este curso el estudiante aprenderá técnicas para elaborar y realizar métricas y estimaciones en las diferentes etapas del proceso de software de tal modo que aprenda a utilizarlas en la administración de un proyecto.

PROPÓSITOS GENERALES

El estudiante comprenderá y aplicará técnicas, para hallar y realizar métricas que pueda utilizar en las estimaciones para la administración de un proyecto.

CONTENIDOS

TEMAS Y SUBTEMAS	PROPÓSITOS ESPECIFICOS
1 Métricas	El estudiante comprenderá y aplicará las técnicas para realizar mediciones en el proceso de software.
1.1 Fundamentos de teoría de la medición 1.2 Medición del tamaño y de la complejidad 1.3 Medición de defectos, confiabilidad y disponibilidad	
2 Estimación	
2.1 Fundamentos de teoría de la estimación 2.2 Estimación de esfuerzo y tiempo 2.3 Predicción	El estudiante aprenderá a utilizar técnicas de estimaciones y con ello aprenderá a realizar predicciones
3 Las métricas en la organización	El estudiante comprenderá y aplicará métodos que utilizando métricas y haciendo estimaciones se reflejen en la madurez de la organización que elabora software.
3.1 Seguimiento de las mediciones 3.2 Mediciones financieras 3.3 Benchmarking 3.4 Métricas y CMMI	

METODOLOGÍA PARA EL CURSO

Este curso es teórico práctico. La materia se impartirá por medio de clases teóricas, prácticas y asesorías. Se estudiarán casos reales, de preferencia vinculados a procesos cercanos al estudiante, en los que pueda poner en práctica los conocimientos adquiridos en el curso durante el transcurso del semestre. El objetivo de este curso, es que los estudiantes aprendan a utilizar técnicas para proponer y elaborar métricas que permitan hacer estimaciones asegurando un proceso de software estable y predecible, para ello, es importante que el estudiante conozca casos reales que le sean significativos, a través de los cuales comprenda de modo evidente la necesidad de aplicar dichas métricas y estimaciones, además, el resultado conseguido a través de un proyecto en este curso será utilizado en los cursos que le siguen.

Puesto que la aplicación en la práctica de los conocimientos adquiridos en el aula es indispensable, se tendrán sesiones en laboratorios de cómputo una vez por semana, en ellas, el estudiante pondrá en práctica los conceptos tratados en el curso realizando prácticas que tengan como fin lograr elaborar software de calidad.

EVALUACIÓN DIAGNÓSTICA

Se aplicará un examen escrito sobre los temas contenidos en el curso de Aseguramiento de la calidad del Software. Esto con el fin de detectar los conocimientos con los que cuentan los estudiantes y sugerir acciones para que adquieran los que no tengan, por medio de un repaso general dentro de la clase o en asesorías individuales.

EVALUACIÓN FORMATIVA

Con el propósito de dar seguimiento al proceso de enseñanza aprendizaje, se propone aplicar tres evaluaciones formativas, correspondientes a los temas del curso, donde el estudiante mostrará sus conocimientos propuestos en dichos temas.

Los instrumentos para las evaluaciones formativas incluirán aspectos teóricos y prácticos, de tal modo que muestren sus avances con relación al uso de métricas y estimaciones, así como, la aplicación de los principios de la ingeniería de software aplicados a esta etapa. Además de las evaluaciones formativas se dejarán tareas a los estudiantes al terminar cada tema. Se hará una valoración junto con los estudiantes al final de cada tema para hacer sólida la comprensión de los mismos. Se pondrá especial atención en el cumplimiento de las prácticas en el laboratorio, así como en la realización de las tareas, la participación y asistencia a clases, y las correspondientes asesorías, fortaleciendo la formación del estudiante.

EVALUACIÓN DE CERTIFICACIÓN

El examen de certificación evaluará los conocimientos del estudiante sobre todos los temas tratados durante el curso. Este instrumento consta de dos partes: una evaluación teórica mediante examen escrito y la segunda mediante el desarrollo de un proyecto. En esta, al igual que en otras materias de final de la ingeniería de software, lo más importante es que el estudiante lleve a la práctica el material tratado en clase, de aquí que la certificación escrita tendrá al 40% de la calificación final, mientras que la parte práctica se evaluará por medio de un proyecto y representará el 60% de la calificación final. En este proyecto debe mostrar que realmente ha alcanzado los propósitos del curso, y deberá servir como apoyo para utilizarlos en las materias que le siguen.

BIBLIOGRAFÍA

La bibliografía que se menciona a continuación es la básica para el estudiante y el profesor:

Bibliografía	Temas para los que se recomienda
1. Laird Linda, Software Measurement and Estimation: A Practical Approach, Wiley-IEEE Computer Society Pr, 2006.	1, 2, 3
2. Kann, Stephen H., Metrics and Models in Software Quality Engineering, Addison-Wesley Professional, ed. 2da, 2002	1, 2
3. Grad, Robert B. Practical Software Metrics for Project Management and Process Improvement, Prentice Hall, 1992.	1, 2
4. Jones, Casper, Software Assessments, Benchmarks, and Best Practices, Addison-Wesley Professional, 2000	3
5. Mc Connell, Steve, Software Estimation: Demystifying the Black Art, Microsoft Press; ed 1ra, 2006.	3
6. Brooks, Frederick P. The Mythical Man-Month: Essays on Software Engineering, Anniversary Edition, Addison-Wesley Professional; 2 edition, 1995.	1

OTROS RECURSOS

Laboratorio de cómputo con acceso a Internet de alta velocidad, computadoras con un mínimo de 2 Gb en memoria principal, que tenga instalado una base de datos relacional y un Entorno de Desarrollo Integrado (IDE) (Eclipse, NetBeans y/o Visual Studio, cada una en su última versión), una herramienta case para sistemas orientados a objetos (ArgoUML), herramientas de depuración y pruebas (JUnit, Jtiger) y un paquete de oficina (OpenOffice). Además, pintarrón y proyector de video (cañón).