



UNIVERSIDAD NACIONAL DE ENTRE RÍOS

FACULTAD DE CIENCIAS DE LA ADMINISTRACIÓN

LICENCIATURA EN SISTEMAS

PLAN 2012

PROPUESTA DE CURSO PARA RÉGIMEN DE CRÉDITOS

CURSO:

“MÉTRICAS DE SOFTWARE”

Docente a Cargo: Mg. Martín Mauricio Pérez

AÑO 2014

CONTENIDO:

- 1- INTRODUCCIÓN AL TÓPICO DE LAS MÉTRICAS DE SOFTWARE.
- 2- OBJETIVOS DEL CURSO.
- 3- CONDICIONES DE CURSADO.
- 4- PROGRAMA SINTÉTICO.
- 5- PROGRAMA ANALÍTICO.
- 6- BIBLIOGRAFÍA BÁSICA Y DE REFERENCIA.
- 7- METODOLOGÍA PEDAGÓGICA.
- 8- RÉGIMEN DE CURSADO Y EVALUACIÓN.
- 9- PLANTEL DOCENTE.

1- Introducción al Tópico de las Métricas de Software

Las mediciones son el centro de muchos sistemas que gobiernan nuestras vidas. Sin mediciones económicas, físicas, médicas, o meteorológicas, sería inimaginable el funcionamiento de la mayoría de las tecnologías actuales.

Pero las mediciones no sólo son importantes en el dominio tecnológico o profesional, sino que todos nosotros las usamos en nuestra vida cotidiana. Sin darnos cuenta, todo el tiempo manejamos precios, alturas, tamaños, distancias, tiempo... Por lo tanto, las mediciones nos ayudan a comprender nuestro mundo, a interactuar con lo que nos rodea y a mejorar nuestras vidas.

Aunque no siempre se reconocen como algo esencial en la ingeniería de software, las métricas de software cumplen un rol muy importante. Se utilizan mediciones para evaluar situaciones, observar progresos, calcular efectividad, etc. Pero la brecha entre cómo se efectúa una medición y cómo *podría* efectuarse todavía es mayor de lo que debería ser. Una razón clave para esta brecha entre lo potencial y lo real ha sido la ausencia de un encuadre coordinado e integral para comprender y usar las mediciones.

Los buenos desarrolladores miden características del software para determinar si los requerimientos son consistentes y completos, si el diseño es de buena calidad, y si el código está listo para ser probado. Los buenos directores de proyectos miden atributos del proceso y del producto para poder afirmar con certeza cuándo se podrá entregar el software y si el presupuesto se verá excedido. Y los clientes informados miden aspectos del producto final para determinar si éste cumple con los requerimientos y si tiene la calidad adecuada.

2- Objetivos del Curso

- ◆ Examinar las bases de la medición y experimentación, proveyendo una noción básica de por qué medimos y cómo esa medición sustenta la investigación del uso y efectividad de herramientas y técnicas de la ingeniería de software.
- ◆ Explorar en detalle las mediciones de la ingeniería de software, con información sobre métricas específicas y sus usos.
- ◆ Alentar la aplicación de métricas a los procesos, productos y recursos, como una parte natural y útil de las actividades regulares de desarrollo y mantenimiento de software.
- ◆ Ofrecer una perspectiva gerencial de la medición de software, explicando prácticas actuales y anticipando el futuro de las mediciones y métricas.

3- Condiciones de Cursado

- ◆ Ser alumno regular de la carrera Licenciatura en Sistemas.
- ◆ Tener aprobadas todas las materias de 1° y 2° año.
- ◆ Tener regularizada (ambos parciales aprobados) la asignatura “Lenguajes de Programación”.
- ◆ Haber aprobado al menos el primer parcial de Probabilidad y Estadística.

4- Programa Sintético

Parte I: **Principios de la Medición y la Experimentación** (6 horas)

Parte II: **Mediciones en la Ingeniería de Software** (6 horas)

Parte III: **La Norma NM ISO/IEC 9126** (3 horas)

5- Programa Analítico

Parte I: Principios de la Medición y la Experimentación

Medición: qué es y por qué hacerla

Mediciones en la vida cotidiana y en la ingeniería de software. Alcance de las métricas de software.

Fundamentos de la medición

La teoría figurativa de la medición. Mediciones y modelos. Escalas de medición y tipos de escalas. Significatividad en las mediciones.

Un encuadre basado en objetivos para la medición de software

Clasificación de medidas de software. Determinación de qué medir. Aplicación del encuadre. Validación de mediciones de software. Validación en la práctica.

Investigación empírica

Cuatro principios de la investigación. Planificación de experimentos formales. Planificación de estudios de casos.

Recolección de datos

Qué son los buenos datos. Cómo definir y recolectar los datos. Cuándo recogerlos. Cómo almacenar y recoger los datos.

Análisis de los datos resultantes de mediciones del software

Introducción. Análisis de los resultados de experimentos. Ejemplos de técnicas simples de análisis. Métodos avanzados. Visión general de pruebas estadísticas.

Parte II: Mediciones en la Ingeniería de Software

Medición de atributos internos del producto: tamaño

Aspectos del tamaño del software. Longitud. Reuso. Funcionalidad. Complejidad.

Medición de atributos internos del producto: estructura

Tipos de medidas estructurales. Estructura del flujo de control. Atributos de la modularidad y del flujo de información. Métricas orientadas a objetos. Estructura de datos. Problemas con las medidas generales de complejidad.

Medición de atributos externos del producto

Modelado de la calidad del software. Medición de aspectos de calidad.

Confiabilidad del software: medición y predicción

Fundamentos de la teoría de la confiabilidad. El problema de la confiabilidad del software. Modelos paramétricos del crecimiento de la confiabilidad. Precisión predictiva. La importancia del ambiente funcional. Aspectos más amplios de la confiabilidad del software.

Medición de recursos: productividad, equipos, y herramientas

El significado de la productividad. ¿Productividad de qué? Medición de la productividad. Equipos, herramientas, y métodos.

Predicciones sobre el proceso

Buenas estimaciones. Estimación de costos: problemas y enfoques. Modelos de costo y esfuerzo. Problemas de los métodos de modelado existentes. Incidencia en la predicción de procesos.

Parte III: La Norma NM ISO/IEC 9126

Modelo de Calidad

Objeto y campo de aplicación. Marco de trabajo del modelo de calidad. Modelo de calidad para calidad interna y externa. Modelo de calidad para calidad en el uso.

Métricas Externas e Internas

Métricas para: Funcionalidad, Confiabilidad, Facilidad de uso, Eficiencia, Facilidad de mantenimiento, Portabilidad.

Métricas de Calidad de Uso

Métricas para: Efectividad, Productividad, Seguridad Física, Satisfacción.

6- Bibliografía Básica y de Referencia

- **“Software Metrics – A Rigorous and Practical Approach”, Second Edition.**

Norman Fenton y Shari Pfleeger.

International Thomson Computer Press – Londres, 1996.

- **Norma Mercosur NM ISO/IEC 9126. Primera Edición, 2008.**

Asociación Mercosur de Normalización.

- **“Software Metrics for Product Assessment”.**

R. Bache y G. Bazzana.

McGraw-Hill, 1993.

- **“Software Engineering Metrics and Models”.**

Conte, Dunsmore y Shen.

Benjamin-Cummings, 1986.

- **“Applied Software Measurement”.**

C. Jones.

McGraw-Hill, 1991.

- **“Derivation and Validation of Software Metrics”.**

M. Shepperd y D. Ince.

Clarendon Press, 1993.

7- Metodología Pedagógica

El curso está organizado para dictarse en 5 clases presenciales de 3 horas cada una, más el tiempo necesario extra áulico para la resolución por parte de los alumnos de los trabajos prácticos propuestos, y la elaboración del trabajo final previsto. Se estima que estas tareas no presenciales demandarán a los estudiantes un mínimo de 15 horas adicionales.

Las clases presenciales serán teórico-prácticas, y se contará con el apoyo de equipo de proyección de diapositivas (PC – cañón). Si bien los alumnos no deberán hacer un uso continuo de computadoras durante las clases, es deseable que las mismas se desarrollen en una sala de máquinas para que los alumnos puedan, de ser necesario, acceder a información de interés, revisar programas, u otras actividades que requieran del uso de una PC.

El Campus Virtual de la Universidad será también una herramienta utilizada como apoyo en el proceso de enseñanza-aprendizaje. El espacio virtual con que contará el curso servirá como ámbito de intercambio de material, de comunicación entre participantes, de evacuación de dudas, entre otras posibilidades que ofrece la plataforma.

Se recomienda que se establezca un cupo máximo de 20 alumnos inscriptos al curso. De esta manera se espera atender adecuadamente las tareas de dictado, revisión de producciones de los alumnos, respuestas a consultas, y seguimiento en general de los alumnos.

8- Régimen de Cursado y Evaluación

Para conservar su condición de regularidad, los alumnos deberán cumplir con un mínimo del 75% de asistencia a las clases presenciales.

Para la aprobación del curso se exigirá la resolución individual de una serie de trabajos prácticos propuestos, más la presentación, aprobación y defensa de un trabajo final integrador grupal.

En la evaluación se tendrán en cuenta la calidad de los trabajos presentados, la participación en clase y en los foros virtuales, y la solvencia en la defensa de los trabajos finales.

9- Docente a Cargo

Martín Mauricio Pérez. Magíster en Ciencias de la Computación. Profesor Adjunto Ordinario / Titular Interino, Cátedra “Algoritmos y Complejidad”, del 5° año de la Licenciatura en Sistemas. Investigador Categoría IV del Programa de Incentivos.

El dictado del curso se realizará dentro del marco de la Dedicación Exclusiva del docente, no insumiendo nuevos puntos docentes.