



Universidad
Tecnológica
de Pereira



UNIVERSIDAD TECNOLÓGICA DE PEREIRA
FACULTAD DE INGENIERÍAS ELÉCTRICA, ELECTRÓNICA FÍSICA Y CIENCIAS DE LA
COMPUTACIÓN
PROGRAMA INGENIERIA DE SISTEMAS Y COMPUTACION

ASIGNATURA: Ingeniería del Software II

CÓDIGO: IS813

PRERREQUISITO: IS714 Ingeniería de software I

INTENSIDAD: 6 horas semanales

1. JUSTIFICACIÓN

Los fundamentos generales de la ingeniería de software son la base sobre la cual se construye la disciplina de la construcción y desarrollo de software, sin embargo, para dotar al estudiante de las competencias adecuadas que lo respalden en el mercado laboral se deben complementar con temas de actualidad que le amplíen el panorama del desarrollo de software.

2. OBJETIVO

2.1 Objetivo General

Presentar técnicas y herramientas propias de la ingeniería de software que le garanticen al estudiante mantener el nivel competitivo adecuado al mercado del desarrollo de software, de acuerdo con las tendencias internacionales.

2.2 Objetivos Específicos

Presentar conceptos relacionados con técnicas, herramientas y tendencias actualizadas relacionadas con la ingeniería de software de tal forma que al final del curso el estudiante esté en capacidad de:

- Conocer las mejores prácticas en el proceso de construcción de software.
- Evaluar y seleccionar los enfoques metodológicos para abordar el proceso de desarrollo de software.
- Utilizar herramientas de modelado de software.
- Evaluar y establecer metodologías de desarrollo de software orientado por aspectos.
- Reconocer la conveniencia de las diferentes arquitecturas de software en el diseño de soluciones.
- Identificar y exponer criterios de calidad en el proceso de desarrollo de software.

3. CONTENIDO

1. REFERENCIA HISTÓRICA A LAS METODOLOGÍAS DE DESARROLLO DE SOFTWARE
 - 1.1 Flujos de datos (Yourdon, Constantine y de Marco)
 - 1.2 Datos (Jackson y 4gl)
 - 1.3 Objetos: Rational Unified Process, Manifiesto ágil, XP, Scrum,
 - 1.4 Formales (Z)
2. GESTIÓN DE LA INGENIERÍA DE SOFTWARE (Capítulo 8 del SWEBOK)
 - 2.1 Iniciación y alcance

- 2.2 Planificación de un proyecto de software
- 2.3 Promulgación del proyecto de software
- 2.4 Revisión y evaluación
- 2.5 Cierre
- 2.6 Medidas de la ingeniería de software
- 3. GESTIÓN DE LA CONFIGURACIÓN DE SOFTWARE (Capítulo 7 del SWEBOK)
 - 3.1 Administración del proceso de gestión de la configuración
 - 3.2 Identificación de la configuración del software
 - 3.3 Control de la configuración
 - 3.4 Contabilizar el status de la configuración
 - 3.5 Auditar la configuración
 - 3.6 Administrar y distribución del software
- 4. PERSONALIZACIÓN DEL SOFTWARE
 - 4.1 Adaptatividad y adaptabilidad
 - 4.2 Modelado de usuario.
 - 4.3 Sistemas hipermedia adaptativos.
- 5. PROCESO DE INGENIERÍA DE SOFTWARE (Capítulo 9 del SWEBOK)
 - 5.1 Implementación y cambio del proceso
 - 5.2 Definición del proceso (Ciclo de vida)
 - 5.3 Evaluación del proceso
 - 5.4 Medida del proceso y del producto
- 6. ESTÁNDARES Y CALIDAD DE SOFTWARE.
 - 6.1 El proceso y la calidad del producto: calidad del software, el proceso, garantía de calidad del software.
 - 6.2 Mejoramiento de procesos: normas ISO, modelo CMMI, modelo ideal, ITIL, COBIT.
 - 6.3 Mejoramiento a nivel personal: PSP.
 - 6.4 Aseguramiento de la calidad.
 - 6.5 Métricas del software.
 - 6.6 Calidad en los sitios web.
- 7. IMPLANTACION DE SOFTWARE
 - 7.1 Instalación: hardware, software, comunicaciones.
 - 7.2 Pruebas.
 - 7.3 Llenado de datos
 - 7.4 Entrenamiento de los usuarios
 - 7.5 Salida a producción.

4. METODOLOGÍA

Los temas de exponen en forma magistral y se complementan con ejemplos presentados por el profesor y ejercicios propuestos por el profesor y desarrollados por los estudiantes.

5. BIBLIOGRAFÍA

BOURQUE, Pierre. DUPUIS, Robert. **SWEBOK Versión 2004: Guide to Software Engineering Body of Knowledge**. IEEE Computer Society, EEUU, 2004. ISBN 0-7695-2330-7.

BRUEGGE, Bernd y DUTOIT, Allen. *Ingeniería de Software Orientada a Objetos*. Prentice Hall – Pearson, 2002

CMMI Product Team. *Capability Maturity Model® Integration (CMMISM)*, Version 1.1 CMMISM for Software Engineering (CMMI-SW, V1.1) Staged Representation CMU/SEI-2002-TR-029. Carnegie Mellon University, 2002

FOWLER, Martin. *Refactoring: Improving the Design of Existing Code*, Segunda Edición, Addison-Wesley, 2003, ISBN: 0-201-485672, 464 pag.

HERNÁNDEZ BALLESTEROS, Juan Francisco y MINGUET MELIÁN, Jesus Ma. *La Calidad del Software y su medida*. Centro de Estudios Ramón Areces, 2003

HUMPHREY, Watts. *A Discipline for Software Engineering*. Addison-Wesley, 1995

HUMPHREY, Watts. *Introducción al Proceso Software Personal*. Addison-Wesley, 2001

HUMPHREY, Watts. *Introduction to the Team Software Process*. Addison-Wesley, 2000

ICONTEC. *Herramientas para implementar un sistema de Gestión de Calidad*. 2da Edición. 2006

ICONTEC. *Norma técnica NTC 5420-1 : Ingeniería de software. Calidad del producto de software, parte 1: Modelo de calidad*. Bogotá. 2006

ICONTEC. *Norma Técnica NTC- ISO 9000-3. Ingeniería de Software. Directrices para la aplicación de la NTC-ISO 9001:2000 a software de computador*. Bogotá, 2006

NAVA – JIMENEZ. *ISO 9000:2000 Estrategias para implantar la norma de calidad para la mejora continua*. Limusa 2005

PIATINNI, Mario y GARCIA, Rubio. *Calidad en el desarrollo y mantenimiento del Software*. Alfaomega, 2003

PRESSMAN, Roger. *Ingeniería del Software: Un enfoque práctico*, Sexta Edición. McGraw Hill, 2006.

YOURDON, Edward. **Análisis Estructurado Moderno**. 1ª Edición, México, Prentice Hall Hispanoamericana, 1993. ISBN: 968-880-303-0.

6. PROGRAMACIÓN

Tema	Semanas
Metodologías de desarrollo de software	3
Gestión de la ingeniería de software	2
Proceso de ingeniería de software	2
Estándares y calidad de software.	4
Implantación de software	2
Refactoring y reingeniería	1
Exámenes	1
Holgura para imprevisto	1
Total semanas	16