

Nombre de la asignatura: **INGENIERÍA DE SOFTWARE ORIENTADA A PROCESOS.**

Línea de Trabajo: **Ingeniería de Software**

Tiempo de dedicación del estudiante a las actividades de:

DOC	TIS	TPS	Horas Totales	Créditos
48	20	100	168	6

**DOC:** Docencia; **TIS:** Trabajo independiente significativo; **TPS:** Trabajo profesional supervisado

### 1. Historial de la Asignatura.

Fecha de Elaboración o Revisión	Participantes	Observaciones (cambios y justificaciones)
ITCV 21 Noviembre 2011	MC. Pedro Luis Sánchez Orellana MC. Osvaldo Daniel Fernández Bonilla	

### 2. Pre-requisitos

Pre-requisitos asignatura Ingeniería de Software.

### 3. Objetivo de la Asignatura

Analizar e implementar procesos de ingeniería de software basados en modelos de calidad para la definición de actividades de administración, desarrollo, mantenimiento y operación de sistemas de software.

### 4. Aportación del Perfil del Graduado

Esta materia contribuye a incrementar el conocimiento del alumno desde el punto de vista administrativo del desarrollo de software. Para lo cual el alumno aprenderá aspectos que abarcan, desde la gestión de proyectos de software, hasta el soporte y mantenimiento de los mismos.

### 5. Contenido Temático

Unidad	Temas	Subtemas
1	El proceso: una visión general.	<ul style="list-style-type: none"><li>1.1 Definición de procesos en ingeniería de software.</li><li>1.2 Procesos de infraestructura.<ul style="list-style-type: none"><li>1.2.1 Grupos de proceso en ingeniería de software.</li><li>1.2.2 Modelos de recursos.</li></ul></li><li>1.3 Procesos de medición.<ul style="list-style-type: none"><li>1.3.1 Metodologías para el proceso de medición.</li><li>1.3.2 Paradigmas de medición</li></ul></li><li>1.4 Procesos de definición.<ul style="list-style-type: none"><li>1.4.1 Definición de los tipos de procesos.</li><li>1.4.2 Ciclos de vida en los modelos de procesos.</li><li>1.4.3 Ciclo de vida del software.</li></ul></li><li>1.5 Proceso de análisis cualitativo.</li><li>1.6 Procesos de implementación y actualización.<ul style="list-style-type: none"><li>5.61. Paradigmas para los procesos de implementación</li></ul></li></ul>

		y cambios. 1.6.2 Evaluación de los resultados de procesos. 1.6.3 Evaluación de los procesos de implementación y actualización.
2	El proceso de gestión de proyectos de software.	2.1 Gestión de proyectos. 2.1.1 Planificación. 2.1.2 Organización. 2.1.3 Staffing. 2.1.4 Dirección. 2.2. Universalidad de la gestión 2.2.1 Diseño de datos. 2.2.2 Diseño arquitectónico. 2.2.3 Diseño a nivel de componentes. 2.2.4 Diseño de la interfaz. 2.3 Habilidades de Gestión y la Jerarquía organizacional. 2.3.1 Habilidades Técnicas. 2.3.2 Habilidades Humanas. 2.3.3 Habilidades Conceptuales. 2.3.4 Modularidad. 2.3.5 Arquitectura de software. 2.3.6 Jerarquía de control (estructura del 2.4 Planificación de un proyecto de ingeniería de software 2.5 Organización de un proyecto de ingeniería de software 2.6 Dirección de un proyecto de ingeniería de software
3	El proceso de administración de la configuración del software	3.1 Proceso de manejo en redes de suministros (SCM). 3.1.1. Contexto organizacional 3.1.2 Restricciones y guía para SCM 3.1.3 Planeación de SCM 3.2 Identificación de la configuración de software. 3.2.1 Configuración de Software 3.2.2 Relaciones entre componentes 3.2.3 Relaciones entre versiones 3.2.4 Librería de software 3.3 Control de configuración del software 3.3.1 Solicitud, evaluación y aprobación de cambios en el software. 3.3.2 Implementación de cambios en el software 3.4 Auditoría de configuración de software 3.4.1 Auditoría sobre la configuración funcional del software 3.4.2 Auditoría sobre la configuración física del software 3.5 Manejo de liberación en el software 3.5.1 Construcción del software 3.5.2 Administración de liberación del software.
4	El proceso de desarrollo de software	4.1 Diseño y desarrollo de software 4.1.1 Herramientas de construcción 4.1.2 Evaluación integrada en el desarrollo 4.1.3 Técnicas para el desarrollo de software 4.2 Reducción de la complejidad 4.2.1 Métodos lingüísticos de construcción 4.2.2 Métodos de construcción formal 4.2.3 Métodos de construcción visual

		4.3 Anticipación de la diversidad. 4.3.1 Documentación embebida 4.3.2 Métodos orientados a objetos 4.3.3 Archivos de configuración 4.3.4 Re-utilización de repositorios. 4.3.5 Parametrización de macros 4.3.6 Especificación en la configuración visual. 4.4 Estructuras para la validación 4.4.1 Diseño modular 4.4.2 Programación estructurada 4.4.3 Máquinas de estado lógicas 4.4.4 Sistemas redundantes - auto diagnosticables. 4.5 Uso de estándares externos durante el desarrollo 4.5.1 Leguajes de programación estándar 4.5.2 Lenguajes de descripción estándar 4.5.3 Alfabetos de representación estándar
5	El proceso de mantenimiento de software y el proceso de operación y soporte de sistemas de software	5.1 Conceptos básicos 5.1.1. Definiciones y terminología 5.1.2 Costos de mantenimiento 5.1.3 Naturaleza del mantenimiento 5.1.4 Evolución del software 5.2 Proceso de mantenimiento 5.2.1 Modelos de procesos 5.2.2 Actividades de mantenimiento 5.3 Problemas clave en el mantenimiento 5.3.1 Problemas técnicos 5.3.2 Problemas administrativos 5.3.3 Estimación de costo 5.3.4 Medición del mantenimiento 5.4 Técnicas para el mantenimiento 5.4.1 Re-ingeniería 5.4.2 Ingeniería inversa 5.4.3 Análisis de impacto

## 6. Metodología de Desarrollo del Curso

Queda a elección del docente manejar un problema específico para cada una de las unidades de, o bien un proyecto que represente todo el proceso de gestión y administración del software.

## 7. Sugerencias de Evaluación

- Exámenes parciales escritos al final de cada unidad.
- Elaboración de un proyecto de gestión, administración de software.
- Ejercicios del proceso para el desarrollo de software
- Ensayos de las estrategias adecuadas para el mantenimiento y soporte de sistemas de software.

## 8. Bibliografía y Software de Apoyo

- Stephen Schach, Object-Oriented and Classical Software Engineering (8th Edition), ISBN: 0073376183 Pages: 688, Publisher: McGraw-Hill Science/Engineering/Math, 2010-07-19.

- Alain Abran, Pierre Bourque, Robert Dupuis, James W. Moore, and Leonard L. Tripp. Guide to the Software Engineering Body of Knowledge - SWEBOK. IEEE Press, Piscataway, NJ, USA, 2004 version edition, 2004.
- Scacchi, W. Process Models in Software Engineering, Encyclopedia of Software Engineering, 2nd Edition, John Wiley and Sons, Inc, New York, December 2001.
- Sarah Beecham, Nathan Baddoo, Tracy Hall, Hugh Robinson, Helen Sharp, Motivation in Software Engineering: A systematic literature review, Information and Software Technology, Volume 50, Issues 9-10, August 2008, Pages 860-878, ISSN 0950-5849.
- Michiel van Genuchten, Analysis and improvement of software engineering processes, Information & Management, Volume 25, Issue 1, J. 1993, Pages 43-49, ISSN 0378-7206, 10.1016/0378-7206(93)90024-N.
- Susan Ferreira, James Collofello, Dan Shunk, Gerald Mackulak, Understanding the effects of requirements volatility in software engineering by using analytical modeling and software process simulation, Journal of Systems and Software, Volume 82, Issue 10, October 2009, Pages 1568-1577, ISSN 0164-1212.
- A Chatzigeorgiou, G Antoniadis, Efficient management of inspections in software development projects, Information and Software Technology, Volume 45, Issue 10, 15 July 2003, Pages 671-680, ISSN 0950-5849, 10.1016/S0950-5849(03)00074-0.
- Ian R McChesney, Séamus Gallagher, Communication and co-ordination practices in software engineering projects, Information and Software Technology, Volume 46, Issue 7, June 2004, Pages 473-489, ISSN 0950-5849.

## 9. Actividades Propuestas

Se sugiere que las prácticas propuestas sean realizadas por equipos para estar en concordancia con la finalidad de fomentar la discusión de ideas que plantea el curso. En este sentido, se proponen las siguientes prácticas por unidad:

Unidad	Actividad
1	Realizar exposiciones y mesas redondas respecto al significado de un proceso dentro del ámbito de la ingeniería de software.
2	Discusiones guiadas respecto a las estrategias a seguir durante la gestión de software.
3	Crear un mapa conceptual de los procesos involucrados en la administración de configuración
4	Realizar un ensayo de las estrategias para el desarrollo de software y la reducción de su complejidad.
5	Desarrollar un ejemplo de sistema de software y las estrategias para su mantenimiento y soporte

## 10. Nombre y Firma del Catedrático Responsable:



Dr. Pedro Luis Sánchez Orellana