Ministerio de Cultura y Educación
Universidad Nacional de San Luis
Facultad de Ciencias Físico Matemáticas y Naturales
Departamento: Informatica
Area: Area IV: Pr. y Met. de Des. del Soft.

(Programa del año 2016)
(Programa en trámite de aprobación)
(Presentado el 29/09/2016 16:29:04)

I - Oferta Académica

Materia	Carrera	Plan	Año	Período
(OPTATIVA) ADMINISTRACIÓN DE PROYECTOS DE SOFTWARE II	LIC.CS.COMP.	32/12	2016	2° cuatrimestre

II - Equipo Docente

Docente	Función	Cargo	Dedicación
MONTEJANO, GERMAN ANTONIO	Prof. Responsable	P.Asoc Exc	40 Hs

III - Características del Curso

Credito Horario Semanal				
Teórico/Práctico	**Teóricas**	**Prácticas de Aula**	**Práct. de lab/ camp/ Resid/ PIP, etc.**	**Total**
2 Hs	Hs	1 Hs	2 Hs	5 Hs

Tipificación	Periodo
E - Teoria con prácticas de aula, laboratorio y campo	2º Cuatrimestre

Duración			
Desde	**Hasta**	**Cantidad de Semanas**	**Cantidad de Horas**
08/08/2016	18/11/2016	15	75

IV - Fundamentación

Actualmente es imprescindible que un profesional de Informática tenga capacidad de gerenciar proyectos de desarrollo de software con la calidad que impone el mundo moderno, inserto en un esquema global, donde no existen fronteras para la producción de software y para el consumo del mismo, donde se exige el aseguramiento de la calidad de proyectos de software como premisa básica, donde los estándares de aseguramiento de calidad son requeridos para certificar la producción de software.

Es necesario que la administración de proyectos de software esté basada en métricas de software adecuadas, y que, en el planeamiento de proyectos de software se realice un exhaustivo análisis del riesgo de los proyectos.

V - Objetivos / Resultados de Aprendizaje

Consolidar y ampliar los conceptos y desarrollar las habilidades adicionales relacionadas con aspectos específicos de la Administración de Proyectos de Software.

Introducir a las características, normalmente complejas, del desarrollo de software en la actualidad.

Discutir y aplicar las prácticas, prioridades y conformación de equipos para la producción de software con criterio industrial.

VI - Contenidos

Unidad 1: Generalidades acerca de un enfoque actual de la Administración de Proyectos de Software

La producción de software vista como un Proceso. Características de los Procesos. Calidad de los Procesos. Diseño y Control de Procesos. Tecnologías asociadas a la producción de software. Métodos para gestionar y motivas equipos de producción de software. Visión general acerca de la forma en la que la tecnología, la gente y la economía impactan en la Administración de Proyectos de Software. Visión general de métricas para estimar costos, calendarios de proyectos y calidad en el contexto de la producción de software.

Unidad 2: Replanteo de la Administración de Proyectos de Software
Administración de Proyecto de Software "tradicional". Evolución de los aspectos relacionados con los conceptos y las estimaciones cuantitativas en Proyectos de Software. Análisis comparativo entre las Administración "tradicional" y un enfoque moderno del Gerenciamiento de Proyectos de Software.

Unidad 3: Proceso de Software y Fases del Ciclo de Vida
Comparación del concepto de Proceso de Software con los Modelos de Ciclo de Vida Lineal Secuencial, en Cascada, Lineal Incremental, "RAD" (Desarrollo Rápido de Aplicaciones) y en Espiral. Elaboración de una WBS (Descomposición de la Estructura de Trabajo) de un Proyecto de Software orientada a las Actividades y orientada a los Productos. Estimación del esfuerzo asociado a cada una de las "hojas" de la WBS. Elaboración del cronograma del Proyecto (Gantt / PERT) a partir de la WBS. Elaborar el presupuesto del Proyecto. Definición de los perfiles de cada uno de los integrantes del equipo de desarrollo.

Unidad 4: Artefactos de un Proyecto
Artefactos de un Proyecto según el Proceso Unificado. Artefacto entendido como producto tangible del proceso de desarrollo de software. Ejemplos de artefactos de un Proyecto: Casos de Uso, Diagrama de Clases, etc. Otros artefactos: Plan del Proyecto, Casos de Negocios, Enfoque del tratamiento del Riesgo del Proyecto, Documentación del Proyecto.

Unidad 5: Arquitecturas de Software y Proyectos de Software
Arquitectura como organización fundamental del futuro sistema descripta según: Sus componentes, relación entre ellos y con el ambiente, principios que guían su diseño y evolución. Arquitectura Inicial. Arquitectura de Referencia. Documento de Descripción de arquitectura (SAD): Subsistemas, Componentes, Arquitectura Runtime. Arquitecturas y Modelos.

Unidad 6: Workflows de Procesos de Software
Definición y administración de los workflows de Procesos de Software. Uso del enfoque BPMN: Business Process Modeling Notation - Notación para el Modelado de Procesos de Negocio. Uso de herramientas basadas en BPMN. Simulación del workflow generado. Definición y utilización de los checkpoints del Proceso de Software.

Unidad 7: Disciplinas asociadas en un Proyecto de Software
Modelado de Negocio, Requerimientos, Análisis y Diseño, Implementación, Pruebas, Despliegue, Gestión de Configuraciones y Cambios, Gestión del Proyecto y Gestión del Entorno.

Unidad 8: Roles en un Proyecto de Software
Relación entre roles, actividades, tareas, pasos y artefactos en el contexto de un Proyecto de Software. Promotor del Proyecto; Gerente de Proyecto; Arquitecto; Programador; Tester; Cliente; "Coacher"; "Tracker", etc.

VII - Plan de Trabajos Prácticos

Práctico de Aula:
- Modelado (instancia) de un Proceso de Software (Proceso Unificado) de una PYME de Software
- Elaboración de un cuadro comparativo "Gestión de Proyectos de Software tradicional" vs. "Gestión de Proyectos de Software moderna"

Práctico de Laboratorio:
- Producción de Software como Proceso: Diseño y Modelado de Procesos, metodología BPMN, uso de herramientas del tipo Bizagi
- Elaboración de un Modelo de Negocios de una empresa de importante dimensiones (Arcor, Bridas, Siderar, etc.)

- Elaboración de la Arquitectura inicial de un ERP

Práctico de Campo:
- Relevamiento de la definición de Roles en Proyectos de Software en empresas del Parque Industrial del Software de San Luis (no menos de dos empresas a ser estudiadas)

VIII - Regimen de Aprobación

Para regularizar la asignatura el alumno debe:
- entregar en tiempo y forma y aprobar los trabajos prácticos de aula y de laboratorio,
- aprobar y realizar una presentación oral y pública en base al práctico de campo,
- aprobar un examen integrador o su correspondiente recuperación.

Para promocionar la asignatura el alumno debe cumplir con las condiciones de regularización, haber asistido al menos al 80% de las clases y aprobar los trabajos prácticos de aula, de laboratorio y de campo y el examen integrador con nivel superior o igual a siete puntos sobre un total de diez.

El seguimiento continuo de los alumnos que cursan se realiza mediante la observación e interacción sistemática durante las clases prácticas (aula, laboratorio y campo), la evaluación de los prácticos y la evaluación final integradora.

La evaluación final integradora está basada en un esquema de coloquio, tomando como línea base de construcción del mismo el resultado de los trabajos prácticos de aula, de laboratorio y de campo, este último expuesto y defendido con formato de presentación oral y pública.

Para los alumnos que trabajen, la cátedra se rige según las Ordenanzas C.S. N° 26/97, 15/00 y 13/03.

No se aceptan alumnos libres debido al trabajo práctico de campo que deben realizar.

IX - Bibliografía Básica

[1] Royce, Walker, “Software Project Management : A Unified Framework”, Addison Wesley, Second Edition
[2] Kruchten, P., “The Rational Unified Process: An Introduction, Addison Wesley, Third Edition
[3] Pressman, R., “Software Engineering: A Practitioner's Approach”, Editorial Mc Graw Hill, 7th Edition, 2009.
[4] Sommerville, I., “Software Engineering (Fifth Edition)”, Addison-Wesley, 1996
[5] Método BPMN, uso de herramientas y tutorial en línea: http://www.bizagi.com
[6] Ghezzi, Carlo y otros "Fundamentals of Software Engineering", Prentice Hall, 1991
[7] Manifiesto para el Desarrollo de Software Ágil, http://www.agilemanifesto.org - Martín Fowler, La Nueva Metodología, http://www.programacion.net
[8] Sitio Web del Instituto de Ingeniería de Software (CMU), http://www.cmu.edu,
[9] Ambler, S., “Agile Modeling: Effective practices for Extreme Programming and the Unified Process”, John Wiley & Sons, 2002.
[10] Scott Ambler, S., “Agile Modeling and the Unified Process”,
http://www.agilemodeling.com/essays/agileModelingRUP.htm, 2002
[11] Apuntes de la Cátedra.
[12] Notas de clase preparadas por el Área.

X - Bibliografia Complementaria

[1] Harindranath H., Wojtkowski W.G., Zupancic J., Rosenberg D., Wojtkowski W., Wrycza S., Sillince J.A.A. (Editors), “New Perspectives on Information Systems Development: Theory, Methods and Practice”, Springer, 2002

XI - Resumen de Objetivos

Consolidar y ampliar los conceptos y desarrollar las habilidades adicionales relacionadas con aspectos específicos de la Administración de Proyectos de Software. Introducir a las características, normalmente complejas, del desarrollo de software en la actualidad. Discutir y aplicar las prácticas, prioridades y conformación de equipos para la producción de software con

criterio industrial.

XII - Resumen del Programa

Generalidades acerca de un enfoque actual de la Administración de Proyectos de Software
Replanteo de la Administración de Proyectos de Software
Proceso de Software y Fases del Ciclo de Vida
Artefactos de un Proyecto
Arquitecturas de Software y Proyectos de Software
Workflows de Procesos de Software
Disciplinas asociadas en un Proyecto de Software
Roles en un Proyecto de Software

XIII - Imprevistos

XIV - Otros

ELEVACIÓN y APROBACIÓN DE ESTE PROGRAMA	
	Profesor Responsable
Firma:	
Aclaración:	
Fecha:	