

ACTIVIDAD CURRICULAR DE FORMACIÓN

Facultad o Instituto	:	Ciencias de la Ingeniería
Carrera	:	Ingeniería Civil Informática

I. IDENTIFICACIÓN DE LA ACTIVIDAD CURRICULAR

Nombre	:	Ingeniería de Software II					
Código	:	INF-514					
Semestre lectivo	:	IX Semestre					
Horas	:	Presencial:	54	Autónomas:	96	TOTAL:	150
Créditos SCT	:	5					
Duración	:	Trimestral		Semestral:	x	Anual:	
Modalidad	:	Presencial:	x	Semi-presencial:		A Distancia:	
Área de Formación	:	Disciplinar:		General:		Profesional:	x
						Práctica:	
Pre-requisito (Si los hubiese)	:	Ingeniería de Software I					

II. DESCRIPCIÓN Y CARACTERIZACIÓN DE LA ACTIVIDAD CURRICULAR

La actividad curricular de Ingeniería de Software II se desarrolla en el noveno semestre del Plan de estudios de la Carrera de Ingeniería Civil Informática. Tal actividad curricular, localizada en el ciclo avanzado de formación, tiene un carácter teórico-práctico y pertenece al área curricular de Formación Profesional.

El énfasis de ingeniería se centra en aspectos de diseño de software; construcción de software; diseño y aplicación de pruebas usando herramientas; mantenimiento de software.

Se espera que el estudiante comprenda y aplique los procesos de diseño de software en la construcción software que satisface los requisitos de los clientes.

La metodología de enseñanza y aprendizaje empleada será con clases expositivas-participativa, aprendizaje basado en análisis de caso, aprendizaje colaborativo, aprendizaje orientado a proyectos, y lectura guiada.

La evaluación será por medio de productos, tales como artefactos de proyectos; presentaciones orales con rúbricas; talleres con sus respectivas rúbricas; y pruebas escritas con sus pautas de corrección.

III. COMPETENCIAS DEL PERFIL DE EGRESO ASOCIADAS A LA ACTIVIDAD CURRICULAR.

III.1 COMPETENCIAS PROFESIONALES.

COMPETENCIA	SUBCOMPETENCIA
Gestionar proyectos de software, aplicando estándares y metodologías en el desarrollo, operación y mantenimiento de software de calidad que satisfacen necesidades organizacionales	Gestionar proyectos de software de calidad, que satisfacen necesidades organizacionales reales.

III.2 COMPETENCIAS GENÉRICAS.

COMPETENCIA	SUBCOMPETENCIA
Demostrar coherencia ética entre sus postulados valóricos y sus acciones, respetando los derechos humanos y participando activamente en las organizaciones comunitarias, haciendo primar la responsabilidad social desde una perspectiva cristiana.	Juzgar sus actuaciones basándose en fuentes primarias del cristianismo y referentes espirituales.
Desarrollar la identidad regional, generando instancias de integración recíproca entre la Universidad y la comunidad.	Contribuir al desarrollo de programas en la comunidad regional y local.
Comunicar ideas, tanto en la lengua materna como en el idioma inglés, haciendo uso de las tecnologías de la información para desenvolverse en diversos escenarios, dando soluciones a diversas problemáticas de la especialidad.	Comunicarse de forma oral en inglés de acuerdo a lenguaje científico haciendo uso de las tecnologías de la información en contextos propios de su profesión.

IV. RESULTADOS DE APRENDIZAJE - APRENDIZAJE ESPERADO.

RESULTADOS DE APRENDIZAJES
1.-Elaborar una propuesta de diseño de software considerando la metodología de gestión de proyecto y ambientes de desarrollo establecidos y de acuerdo a los requerimientos analizados a satisfacer de un contexto específico de la comunidad regional o local.
2.- Utilizar framework y case/entorno en la construcción de software a partir de las especificaciones de diseño, considerando las buenas prácticas de desarrollo de software de calidad y su formalización en documentos técnicos en inglés o español.
3.- Aplicar pruebas de software a partir de su planeación y/o automatización según las especificaciones de requerimientos acordados, considerando una fluida comunicación de los involucrados, una autoevaluación personal y del equipo, que permita definir su puesta en producción.
4.- Planear el mantenimiento de software de acuerdo al contexto, considerando buenas prácticas del proceso de mantenimiento de software y técnicas de documentación.

V. UNIDADES DE APRENDIZAJE Y EJES TEMÁTICOS

R. AP.	UNIDAD	EJE(S) TEMÁTICO(S)
1	Diseño de software	<ul style="list-style-type: none"> -Visión general de PMBOK -Métodos ágiles de la gestión de software -Transición del análisis al diseño -Conceptos y principios del diseño -El modelo del diseño -Diseño arquitectónico y estilos de arquitectura: <ul style="list-style-type: none"> -Arquitectura de sistemas y de aplicaciones -Diseño orientado a objetos -Diseño interfaz de usuario -Artefactos, proceso y roles del diseño de software -SOA
	Construcción de software	<ul style="list-style-type: none"> -UML en la fase de construcción de software -Framework, entorno y software en la construcción de software. -Patrones de diseño -Técnicas de codificación -Verificación y validación de software -Herramientas de integración continua -BPMS en la construcción de software.
3	Pruebas y depuración de software	<ul style="list-style-type: none"> -Fundamentos de las pruebas de software. -Tipos de técnicas de prueba de software. -Herramientas para ejecutar pruebas. -Depuración de programas. -Implantación de software.
4	Mantenimiento de software	<ul style="list-style-type: none"> -Fundamentos de mantenimientos de software. -Proceso de mantenimiento de software -Documentación de mantenimiento

VI. ESTRATEGIAS DE ENSEÑANZA-APRENDIZAJE

De acuerdo al Modelo Educativo de la Universidad Católica del Maule, la metodología de trabajo propuesta para el desarrollo de la actividad curricular se basa en un enfoque activo-participativo. Esto implica que el docente fomenta una participación activa del estudiante y provee orientaciones y lineamientos que permitan construir su propio aprendizaje. Para lograr este objetivo, las distintas clases consideran una serie de estrategias metodológicas seleccionadas por el docente, tales como:

- Aprendizaje orientado a proyectos.
- Aprendizaje Colaborativo.
- Aprendizaje basado en estudio de casos.
- Metodología expositiva.
- Lectura guiada.
- Uso de entorno, framework y software

VII. PROCEDIMIENTOS DE EVALUACION DE APRENDIZAJES.

RESULTADO DE APRENDIZAJES	INDICADORES	INSTRUMENTO Y/O TÉCNICA EVALUATIVA	PONDERACIÓN (%)
1	<ul style="list-style-type: none"> -Describe las fases de la administración de proyectos. -Explica una propuesta ágil de gestión de proyecto - Compara gestión de proyecto tradicional y ágil. - Explica y ejemplifica los estilos de arquitectura. -Aplica refinamiento de modelos de análisis para la fase de diseño. -Elabora propuesta de diseño a partir de especificaciones previas de un contexto regional o local. 	<ul style="list-style-type: none"> Mapa conceptual/Rúbrica Control escrito/pauta de corrección. Informe caso de estudio/rúbrica Avance Proyecto/ Rúbrica 	20%
2	<ul style="list-style-type: none"> -Integra framework /case en la construcción de software. -Explica la diferencia entre los diferentes patrones de diseño. 	<ul style="list-style-type: none"> Informe de Modelo / Rúbrica Taller/ pauta de corrección Avance Proyecto/ Rúbrica 	30%

	<ul style="list-style-type: none"> -Aplica patrones de diseño en la construcción de software -Explica la importancia de la codificación en el contexto del desarrollo y mantención de software. -Explica la importancia de la integración continua -Aplica una herramienta de integración -Elabora documentos técnicos del desarrollo de software en inglés o español. 	Interfaz de usuario/ Rúbrica	
3	<ul style="list-style-type: none"> -Localiza dentro de un proceso la etapa de prueba de software -Explica los conceptos asociados con las técnicas de prueba de software. -Diseña prueba de software. -Usa herramienta para hacer pruebas de software. Realiza depuración de software. Evalúa desempeño del equipo y se autoevalúa. Explica un proceso de implantación. Realiza una implantación de software. 	Taller/ Rúbrica Informe Caso de estudio/Rúbrica Presentación Proyecto/ Rúbrica	30%
4	<ul style="list-style-type: none"> -Explica los factores que contribuyen con el mantenimiento de software -Explica las fases de un proceso de mantenimiento 	Taller/ pauta de corrección Proyecto/ pauta de corrección	20%

	<p>-Contrasta los diferentes tipos de mantenimientos.</p> <p>-Desarrolla un análisis de impacto sobre una solicitud de mantenimiento.</p> <p>-Emplea plantillas para documentar el mantenimiento de software.</p> <p>-Define un plan de actividades para realizar un mantenimiento de software.</p>		
--	---	--	--

VIII. RECURSOS DE INFRAESTRUCTURA

Sala de clases, data, internet, laboratorio computación, Servidor (hardware) para proyectos y software, LMS-UCM

IX. RECURSOS BIBLIOGRÁFICOS

	Autor, Título, Editorial, Año de Edición	Biblioteca donde se encuentra	N° Libros Disponibles
BÁSICA OBLIGATORIA	<p>- Ian Sommerville. Software engineering. Addison-Wesley, 2011.</p> <p>- Pierre Bourque, Richard E. Fairley. Guide to the Software Engineering Body of Knowledge (SWEBOK(R)): Version 3.0. IEEE Computer Society (Author),</p> <p>- Scrum: The Art of Doing Twice the Work in Half the Time Sep 30, 2014 by Jeff Sutherland and JJ Sutherland</p> <p>- Jez Humble and David Farley. Continuous delivery.</p>	- - -	- - -
COMPLEMENTARIA	<p>-- Project Management Institute. 2013. A Guide to the Project Management Body of Knowledge (PMBOK Guide) Newtown Square, Pa: Project Management Institute. 2013.</p> <p>- Stephen A. White, Derek Miers, BPMN: guía de referencia y modelado: comprendiendo y utilizando BPMN:</p>	- -	- -

	<p>desarrolle representaciones gráficas de procesos de negocios, que sean rigurosos pero al mismo tiempo de fácil comprensión, Stephen A. White, Derek Miers; Future Strategies, 2010.</p> <p>- Pierre Bourque, Richard E. Fairley, "Guide to the Software Engineering Body of Knowledge (SWEBOK(R)):", Version 3.0. IEEE Computer Society (Author).</p>		
--	--	--	--

X. OTROS RECURSOS

Nombre Recurso	Tipo de Recurso
<i>Videos seleccionados de Internet de eventos, cursos, entrevistas de apoyo al curso</i>	<i>Videos</i>
<i>Rational Rose, Bizagi, Git, Framework fron-end y framework back-end,</i>	<i>software</i>
<i>Servidor Web para hospedar software de la asignatura y proyectos de los estudiantes</i>	<i>Hardware</i>