

PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE  
ESCUELA DE INGENIERÍA  
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

**IIC3143      DESARROLLO DE SOFTWARE**

<b>Créditos y horas:</b>	10 créditos / 10 horas (3 h. Clases; 1,5 Laboratorio; 5,5 Trabajo individual)
<b>Profesor:</b>	Rodrigo Sandoval
<b>Coordinador:</b>	Rosa Alarcón
<b>Bibliografía:</b>	<ul style="list-style-type: none"><li>- Beck, K.; Fowler, M. (2000) Planning extreme programming. Addison-Wesley</li><li>- CMMI product team. (2010) CMMI for development, version 1.3. CMU/SEI</li></ul>
<b>Descripción:</b>	Este curso se basa en el aprendizaje adquirido en el curso de ingeniería de software, donde se enseña el proceso de producción de un sistema de software completo. Este curso profundiza en la fase de desarrollo. Para ello enseña marcos de trabajo y métodos, prácticas de análisis, prácticas de desarrollo, prácticas de gestión, medición de resultados y calidad y como pasar de la teoría a la práctica.
<b>Requisitos:</b>	IIC2143 Ingeniería de software
<b>Co-requisitos:</b>	Ninguno
<b>Tipo de curso:</b>	Mínimo
<b>Objetivos de aprendizaje:</b>	<ol style="list-style-type: none"><li>1. Encontrar requisitos usando las técnicas de análisis de necesidades, análisis de objetivos, y análisis de casos de uso.</li><li>2. Organizar y priorizar requisitos.</li><li>3. Validar requisitos usando criterios de factibilidad, claridad, no ambigüedad, etc.</li><li>4. Representar requisitos funcionales y no funcionales para distintos tipos de sistemas usando técnicas formales e informales.</li><li>5. Especificar y medir atributos de calidad.</li><li>6. Negociar entre diversos interesados para acordar un conjunto de requisitos</li></ol>
<b>Criterios ABET relacionados al curso:</b>	<ul style="list-style-type: none"><li>b. Diseñar y realizar experimentos: analizar e interpretar datos.</li><li>e. Identificar, formular y resolver problemas de Ingeniería.</li><li>i. Reconocer la necesidad y capacidad de la educación continua.</li><li>j. Conocimiento de temas contemporáneos.</li><li>k. Técnicas, habilidades y herramientas modernas para la práctica de la Ingeniería.</li></ul>

**Contenidos:**

1. Marcos de trabajo y métodos: RUP y el enfoque de los procesos de desarrollo; MSF y el enfoque en los roles del proyecto; CMMI y el énfasis en la calidad; Scrum, XP y los métodos ágiles.
2. Prácticas de análisis: Técnicas para descubrir requisitos, y lenguajes y modelos para representarlos; análisis de necesidades, objetivos, y casos de uso; el contexto de la ingeniería de sistemas; especificación y medición de desempeño, confiabilidad, disponibilidad, inocuidad, y seguridad.
3. Prácticas de desarrollo: Programación en pares, desarrollo dirigido por pruebas, propiedad del código, control de versiones, refactoring, y desarrollo incremental.
4. Prácticas de gestión: Conformación del equipo, valor para el cliente versus funcionalidad desarrollada, gestión de riesgos, control de cambios, y derechos del cliente y de los desarrolladores; iteraciones y "daily builds"; balance de recursos, tiempo, funcionalidad, y calidad.
5. Medición de resultados y calidad: Documentación, instancias y estrategias de revisión, formalización de hitos, control de avance, y receptibilidad.
6. De la teoría a la práctica: Análisis de casos de estudio reales y aspectos que influyeron en su éxito o fracaso.