

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

IIC2154 PROYECTO DE ESPECIALIDAD

Créditos y horas:	10 créditos / 10 horas (3 h. Profesor guía; 3 h. Coaching; 4 h. Desarrollo de proyecto)
Profesor:	Andrés Neyem
Coordinador:	Andrés Neyem
Bibliografía:	Applying UML and patterns: An introduction to object-oriented analysis and design and the unified process, Larman, C., (2002). Prentice hall PTR upper Saddle River, NJ, USA.
Descripción:	Aplicar conocimientos previos, de manera integrada, de desarrollo de software, hardware, diseño de redes; experimentar problemas reales relativos al desarrollo de software en grupos. En estos proyectos, es necesario integrar conocimiento de software y hardware con el fin de elaborar la mejor solución posible para el cliente, entregar recomendaciones de arquitectura para garantizar el mejor rendimiento del software. Adicionalmente, los alumnos aprenderán a trabajar en equipos realizando diversos roles, siguiendo estándares de calidad y buenas prácticas en la gestión, diseño y desarrollo.
Requisitos:	IIC2143 Ingeniería de software y IIC2413 Bases de datos
Co-requisitos:	Ninguno
Tipo de curso:	Mínimo
Objetivos de aprendizaje:	<ol style="list-style-type: none">1. Llevar a cabo los pasos que subyacen al desarrollo de software: definición de requerimientos, diseño funcional, características e integración.2. Desarrollar habilidades técnicas y no técnicas en su desarrollo profesional.
Criterios ABET relacionados al curso:	<ol style="list-style-type: none">a. Conocimiento de matemáticas, ciencias e Ingeniería.b. Diseñar y realizar experimentos: analizar e interpretar datos.c. Diseñar sistemas, componentes o procesos.e. Identificar, formular y resolver problemas de Ingeniería.f. Responsabilidad ética y profesionali. Reconocer la necesidad y capacidad de la educación continua.j. Conocimiento de temas contemporáneos.k. Técnicas, habilidades y herramientas modernas para la práctica de la Ingeniería.

Contenidos:

1. Planificación y control de proyecto
 - 1.1. Planificación
 - 1.2. Recursos
 - 1.3. Gestión de proyecto
 - 1.4. Desarrollo, testing y producción
 - 1.5. Herramientas de soporte, diseño de marcos
2. Coordinación basada en roles
3. Modelos de desarrollo de software
 - 3.1. Agile process: Scrum, XP
4. Análisis de requerimientos
5. Testing
6. Calidad y desarrollo de software
7. Configuración
 - 7.1. Herramientas de soporte
 - 7.2. Control de versiones