

PROGRAMA DE CURSO

| Código | Nombre | | | |
|--|---|------------------|----------------------------------|---------------------------|
| CC6402 | Taller Avanzado de Desarrollo Ágil y Lean | | | |
| Nombre en Inglés | | | | |
| Workshop of Advanced Agile and Lean Development | | | | |
| SCT | Unidades Docentes | Horas de Cátedra | Horas Docencia Auxiliar | Horas de Trabajo Personal |
| 6 | 10 | 3,5 | 0 | 6,5 |
| Requisitos | | | Carácter del Curso | |
| CC6401 ó CC61A ó Autorización | | | Taller de Investigación Aplicada | |
| Resultados de Aprendizaje | | | | |
| <p>Los alumnos adquirirán un conocimiento avanzado y experiencia de primera mano de las metodologías, principios y prácticas derivados de las metodologías ágiles. En particular se abarcan tópicos que están en la frontera del conocimiento en el área, tales como:</p> <ul style="list-style-type: none"> - Desarrollo de productos de innovación sin que exista un problema de negocio bien definido, como sucede en los emprendimientos o en las investigaciones académicas. - Usabilidad de sistemas usando agilidad - Automatización de Pruebas de Aceptación usando Desarrollo Guiado por Comportamiento - Aplicación de Agilidad en Mantenimiento de Sistemas de Software usando Kanban - Cómo escalar la agilidad a organizaciones de gran tamaño - Formación de líderes en desarrollo ágil - Y en general cualquier tema que surja como novedad dentro de la industria tecnológica y en donde se esté usando la agilidad | | | | |

| Metodología Docente | Evaluación General |
|--|--|
| <p>El profesor realizará una pequeña inducción teórica a los tópicos avanzados que se investigarán en el curso, para que luego los alumnos profundicen en el tema mediante investigación y presentaciones al resto del curso. Luego, realizará en conjunto un proyecto de desarrollo de software en las sesiones del curso en donde se buscará aplicar los nuevos enfoques investigados, y se evaluará la efectividad de dichos enfoques a partir de lo experimentado.</p> | <ul style="list-style-type: none"> • Producto (20%) Como curso de ingeniería de software, está la meta de lograr un software satisfactorio en los tiempos y plazos del proyecto. Se basa en la calidad del producto de software desarrollado, Y la opinión del propio cliente. • Proceso (40%) Correcta aplicación de las prácticas Cumplimiento de tareas: puntualidad y cumplimiento requisitos estipulados por el profesor Co-evaluación: Evaluación recibida por el alumno de sus pares, de acuerdo a su dedicación, responsabilidad y capacidad de trabajo en equipo. • Comprensión e Investigación (40%) Dominio de los conceptos ágiles logrado Profundización aportada por el trabajo propio del alumno Presentaciones |

Unidades Temáticas

| Número | Nombre de la Unidad | Duración en Semanas |
|--|--|-------------------------------|
| | Introducción Teórica | 5 |
| Contenidos | Resultados de Aprendizajes de la Unidad | Referencias a la Bibliografía |
| <ul style="list-style-type: none"> - Introducción a los límites de las metodologías ágiles y los nuevos enfoques que han surgido a partir de ellas. - Elección, investigación y presentación de tópicos avanzados por parte de los alumnos | <ul style="list-style-type: none"> - Vislumbrar los bordes en el campo del conocimiento de la Agilidad y cuáles son los enfoques que han surgido para expandir dichas fronteras | Ver bibliografía |

| Número | Nombre de la Unidad | Duración en Semanas |
|--|--|-------------------------------|
| | Desarrollo de Proyecto | 10 |
| Contenidos | Resultados de Aprendizajes de la Unidad | Referencias a la Bibliografía |
| <ul style="list-style-type: none"> - Definición de Proyecto a realizar - Implementación de Proyecto usando desarrollo ágil e incorporando los nuevos enfoques estudiados por los alumnos | <ul style="list-style-type: none"> - Conocimiento de primera mano acerca de las implicaciones de aplicar en la práctica los enfoques estudiados | Ver bibliografía |

| Bibliografía |
|--|
| <ul style="list-style-type: none"> - Growing Object-Oriented Software, Guided by Tests <ul style="list-style-type: none"> • Steve Freeman, Nat Pryce - The Principles of Product Development Flow: Second Generation <ul style="list-style-type: none"> • Donald G. Reinertsen - Leading Lean Software Development <ul style="list-style-type: none"> • Mary & Tom Poppendieck - The Four Steps to Epiphany <ul style="list-style-type: none"> • Stephen Blank |



fcfm

FACULTAD DE CIENCIAS
FÍSICAS Y MATEMÁTICAS
UNIVERSIDAD DE CHILE

- Kanban
 - David J. Anderson
- Bridging the Communication Gap: Specification by Example and Agile Acceptance Testing
 - Gojko Adzic

| | |
|-----------------|----------------------|
| Vigencia desde: | Julio de 2010 |
| Elaborado por: | Agustín Villena Moya |