



IIC2143 - Ingeniería de Software

– Programa de curso –

Profesor	- Josefa Paz España (jpespana@alumni.uc.cl) - María Trinidad Vargas (mtvargas1@uc.cl)
Requisitos	- IIC2233 o IIC1222
Sitio Web	- Canvas (http://cursos.canvas.uc.cl/)
Clases	- Lunes y Miércoles, módulo 3
Ayudantías	- Viernes, módulo 3

1. Presentación del curso

En este curso los estudiantes aprenderán conceptos, técnicas y metodologías que se utilizan para el desarrollo de software confiable y robusto. Los estudiantes serán capaces de entender las distintas etapas del proceso de desarrollo de software, incluyendo la especificación de requisitos, el diseño, el desarrollo, la gestión, técnicas de verificación y validación de software.

2. Objetivos de aprendizaje

Al finalizar el curso, el estudiante estará capacitado para:

- Aplicar el desarrollo de un sistema de una manera metódica, considerando requisitos, diseño modular, para su implementación que identifique y minimice los riesgos, codificando para su integración de manera colaborativa, usando métodos para identificar y prevenir fallas.
- Desarrollar requisitos claros, concisos y precisos para el desarrollo de un nuevo producto de software (sistema), basados en las necesidades de los usuarios y otros interesados.
- Aplicar principios y patrones al diseñar un sistema y al evaluar el diseño de un sistema pensando en su escalabilidad y mantenibilidad (abstracción, descomposición, ocultación de información, acoplamiento, cohesión, etc.)
- Crear diagramas de clases en UML que modelan el dominio de un problema y la arquitectura de software de un sistema.
- Crear diagramas de secuencia, de estados, y de actividades en UML que modelen los casos de uso y, más en general, el comportamiento de un sistema.

- Aplicar técnicas de testing simples a distintos niveles de un producto de software para verificar y validar la correcta funcionalidad del producto.

3. Contenido

A continuación se presenta un desglose detallado de los contenidos del curso:

1. Motivación

- La necesidad de ingeniería de software
- Software como servicio
- Desafíos y oportunidades

2. Proceso

- La necesidad de proceso modelo de cascada
- Procesos iterativos
- Procesos incrementales
- Métodos ágiles
- Scrum y Kanban

3. Requisitos

- Funcionales y no funcionales
- Relatos de usuarios

4. Gestión del Proyecto

- Actividades de gestión: estimaciones
- Planeación de producto, release y sprint
- Gestión de personas

5. Diseño

- Modelo de dominio
- Atributos de un buen diseño: acoplamiento y cohesión
- Diagramas UML de clases, secuencia y estados patrones de diseño

6. Arquitectura

- Conceptos fundamentales
- Atributos que impactan la arquitectura: patrones arquitectónicos
- Arquitecturas cliente servidor y multicapas
- Arquitectura orientada a servicios
- Microservicios

7. Aseguramiento de Calidad (QA)

- Definiciones de calidad
- Prevención de defectos
- Detección y eliminación de defectos (testing)

4. Metodología

El curso se desarrolla en clases expositivas, tutoriales, actividades, tareas y un proyecto. El curso contará con dos canales de información. Los anuncios y enlaces relevantes serán publicados a través de la plataforma canvas. Todo el material del curso, incluyendo los apuntes de clases, enunciados de tareas, pautas, y ayudantías, estarán disponibles en el sitio web del curso.

El curso contará con actividades evaluativas sobre el contenido de la cátedra (**NC**). Las actividades incluyen: (i) una tarea (**T**) que permitirá a los estudiantes a familiarizarse con las tecnologías que utilizaremos en el curso, (ii) dos pruebas prácticas (**PP**) de los contenidos del curso y un examen final (**E**) donde se evalúan todos los contenidos de la cátedra. Finalmente, a lo largo del semestre los estudiantes trabajarán en grupos para desarrollar una aplicación web (**NP**). Durante el desarrollo de su aplicación, los estudiantes tendrán que aplicar diferentes conceptos, prácticas, y principios de desarrollo de software ágil visto en clases.

5. Evaluaciones

El curso se evalúa mediante los siguientes tipos de actividades.

- **Catedra (50 %):** La nota de catedra (**NC**) considerará una tarea, las dos pruebas prácticas y el examen.
 - **Tarea (T):** Al inicio del semestre los estudiantes realizarán una tarea (**T**).
 - **Pruebas Prácticas (PP):** Se tomarán dos pruebas que midan el conocimiento de los temas vistos en su respectiva parte del semestre. La nota de pruebas prácticas **PP** será el promedio simple entre ambas.
 - **Exámen (E):** Al final del semestre los estudiantes rendirán un exámen de todos los contenidos vistos en el semestre cuya nota será **E**.

La nota de cátedra (**NC**) será asignada con la siguiente ecuación:

$$NC = T * 0,1 + PP * 0,6 + E * 0,3$$

Requisitos de Eximición. El estudiante debe contar con un promedio entre ambas pruebas prácticas igual o superior a 5.5, además de contar con nota igual o superior a 3.95 en cada una de las pruebas, es decir, **PP** \geq **5,50**, **PP1** \geq **3,95** y **PP2** \geq **3,95**. Si el estudiante cumple este requisito su nota de cátedra (**NC**) será asignada con la siguiente ecuación:

$$NC = T * 0,1 + PP * 0,9$$

- **Proyecto (50 %):** El resto del semestre los estudiantes trabajarán en grupos de 3 integrantes para desarrollar una aplicación web. El proyecto será evaluado periódicamente, con entregas parciales (**EP**) (al término de cada iteración), una entrega final (**EF**) y una presentación final (**PF**). La nota del proyecto (**NP**) será asignada con la siguiente ecuación: **NP** = **EP** * **0,6** + **EF** * **0,2** + **PF** * **0,2**.

Requisitos de Aprobación. El estudiante debe aprobar la catedra y el proyecto por separado, es decir, tener **NC** \geq **3,95** y **NP** \geq **3,95**. Si el estudiante cumple este requisito la nota final se calcula de la siguiente manera:

$$NF = 0,5 * NC + 0,5 * NP$$

En caso que el estudiante no cumpla el requisito anterior, la nota final se calcula de la siguiente forma:

$$NF = \text{minimo}(NC, NP)$$

Fechas Actividades Evaluativas. Las fecha de las diferentes actividades evaluativas son:

- **Tarea:** La entrega de la tarea será formato en línea el día 28 de Marzo.
- **Pruebas prácticas:** Se tomará la primera prueba el día 4 de Abril y la segunda el día 27 de Mayo.
- **Exámen:** Se tomará el exámen del curso, para aquellos estudiantes que no cumplan con el requisito de eximición el día 30 de Junio.
- **Proyecto:** La fecha de entrega de las diferentes actividades evaluativas del proyecto se anunciarán con anticipación a lo largo del semestre.

Trabajo en Grupo. Los integrantes del grupo serán evaluados de manera conjunta en las distintas etapas del proyecto (entrega parcial, entrega final y presentación definitiva), a menos que uno o varios miembros del grupo informen al ayudante coordinador sobre la falta de colaboración activa de un integrante en el desarrollo del proyecto. En tal caso, se llevará a cabo una evaluación entre pares con el objetivo de analizar el aporte individual de cada miembro del equipo en las diferentes entregas. La calificación asignada reflejará proporcionalmente la contribución de cada integrante.

Importante: Si se observa una falta de participación activa por parte de un integrante en una entrega, se le otorgará una calificación en base a una entrevista con su product owner (tarjeta amarilla). La falta de participación activa implica que el integrante no realizó los cambios necesarios para cumplir sus tareas asignadas para la entrega o no demuestra conocimiento y dominio sobre esos cambios (se revisaran la asignación de tareas en **github projects**). En caso de que esa misma persona demuestre nuevamente una falta de participación activa en una segunda entrega, se considerará que ha reprobado el proyecto con la calificación mínima (tarjeta roja).

Estas medidas buscan asegurar la equidad en la evaluación y fomentar la colaboración y el compromiso de todos los integrantes del grupo en el desarrollo exitoso del proyecto.

6. Normativas adicionales

- Los estudiantes deben llegar puntuales a las evaluaciones del curso. Para el caso de la prueba, quienes se presenten con un atraso mayor de 30 minutos no rendiran la actividad y tendrán la calificación mínima en la misma en caso de no contar con justificación validada por la DIPRE.

7. Bibliografía

- Fundamentals of Software Engineering (2nd Edition), Carlo Ghezzi, Mehdi Jazayeri, Dino Mandrioli, Prentice Hall; September 29, 2002.
- Software Engineering: (Update) (8th Edition), Ian Sommerville; Addison Wesley, June 4, 2006.
- Software Engineering: A Practitioner's Approach /(7 edition), Roger Pressman; McGraw-Hill Science/Engineering/Math, January 20, 2009.
- Clean Code: A Handbook of Agile Software Craftsmanship. Robert. C. Martin.
- User Stories Applied: For Agile Software Development. Mike Cohn.
- Software Engineering: A Modern Approach. Marco Tulio Valente.
- Apuntes de Ingenieria de Software. Juan Pablo Sandoval Alcocer y Jaime Navón.

8. Política de Integridad Académica

Los/as estudiantes de la Escuela de Ingeniería de la Pontificia Universidad Católica de Chile deben mantener un comportamiento acorde a la Declaración de Principios de la Universidad. En particular, se espera que mantengan altos estándares de honestidad académica. Cualquier acto deshonesto o fraude académico está prohibido; los/as estudiantes que incurran en este tipo de acciones se exponen a un Procedimiento Sumario. Es responsabilidad de cada estudiante conocer y respetar el documento sobre Integridad Académica publicado por la Dirección de Docencia de la Escuela de Ingeniería.

Específicamente, para los cursos del Departamento de Ciencia de la Computación, rige obligatoriamente la siguiente política de integridad académica. Todo trabajo presentado por un/a estudiante para los efectos de la evaluación de un curso debe ser hecho individualmente por el/la estudiante, sin apoyo en material de terceros. Por “trabajo” se entiende en general las interrogaciones escritas, las tareas de programación u otras, los trabajos de laboratorio, los proyectos, el examen, entre otros.

En particular, si un/a estudiante copia un trabajo, o si a un/a estudiante se le prueba que compró o intentó comprar un trabajo, obtendrá nota final 1.1 en el curso y se solicitará a la Dirección de Pregrado de la Escuela de Ingeniería que no le permita retirar el curso de la carga académica semestral.

Por “copia” se entiende incluir en el trabajo presentado como propio, partes hechas por otra persona. En caso que corresponda a “copia” a otros estudiantes, la sanción anterior se aplicará a todos los involucrados. En todos los casos, se informará a la Dirección de Pregrado de la Escuela de Ingeniería para que tome sanciones adicionales si lo estima conveniente.

También se entiende por copia extraer contenido sin modificarlo sustancialmente desde fuentes digitales como Wikipedia o mediante el uso de asistentes inteligentes como ChatGPT o Copilot. Se entiende que una modificación sustancial involucra el análisis crítico de la información extraída y en consecuencia todas las modificaciones y mejoras que de este análisis se desprendan. Cualquiera sea el caso, el uso de fuentes bibliográficas, digitales o asistentes debe declararse de forma explícita, y debe indicarse cómo el/la estudiante mejoró la información extraída para cumplir con los objetivos de la actividad evaluativa.

Obviamente, está permitido usar material disponible públicamente, por ejemplo, libros o contenidos tomados de Internet, siempre y cuando se incluya la referencia correspondiente.

Lo anterior se entiende como complemento al Reglamento del Estudiante de la Pontificia Universidad Católica de Chile (<https://registrosacademicos.uc.cl/reglamentos/estudiantiles/>). Por ello, es posible pedir a la Universidad la aplicación de sanciones adicionales especificadas en dicho reglamento.

Compromiso del Código de Honor. Este curso suscribe el Código de Honor establecido por la Universidad, el que es vinculante. Todo trabajo evaluado en este curso debe ser propio. En caso que exista colaboración permitida con otros/as estudiantes, el trabajo deberá referenciar y atribuir correctamente dicha contribución a quien corresponda. Como estudiante es un debe conocer el Código de Honor (<https://www.uc.cl/codigo-de-honor/>)