

2023-2 / IIC2173 — Programa del curso - Arquitectura de Sistemas de software

Tabla de contenidos

- [Equipo](#)
 - [Profesores](#)
 - [Ayudantes](#)
- [Objetivos](#)
- [Contenidos](#)
- [Evaluación](#)
 - [Interrogaciones y examen](#)
 - [Actividades](#)
 - [Proyecto](#)
 - [Criterio de aprobación](#)
- [Foro](#)
- [Bibliografía](#)
- [Política de integridad académica](#)

Equipo

Profesores

Nombre	Sección	GitHub	Email
Hans Findel	1	@hansfindel	hjfindel@uc.cl
Nicolás Acosta	2	@nacosta-cl	nlacosta@ing.puc.cl

Ayudantes

Nombre	Especialidad	GitHub	Email
Gerardo Crot (<i>coord</i>)	Coordinador	@gicrot	gicrot@uc.cl
Domingo Edwards	Experto generalista	@domingo2000	domingo.edwards@uc.cl
Edison Pizarro	Experto generalista	@edipizarro	edipizarro@uc.cl
Antonia Reyes	Corrección, ayudantías	@antonia-reyes	antonia.reyes@uc.cl
Jose Tomás Valdivia	Corrección	@jtvaldivia	jtvaldivia@uc.cl
Vicente Paiva	Guía proyecto	@vpaivag	vpaivag@uc.cl

Objetivos

- Tomar los requisitos para un sistema y desarrollar arquitecturas de software y diseños de alto nivel.
- Diseñar software distribuido.
- Diseñar software usando componentes COTS (commercial off-the-shelf)
- Aplicar frameworks y arquitecturas al diseñar una amplia variedad de software.
- Diseñar e implementar software usando tecnologías middleware.
- Manejar conceptos esenciales de cloud computing y devops
- Trabajar bajo incertidumbre y con libertades respecto a la implementación para lograr objetivos

Contenidos

- Modelado y diseño de software flexible al nivel arquitectónico
- Fundamentos de arquitectura dirigida por modelos
- Estilos y patrones arquitectónicos
- Armazones (frameworks) para middleware y para aplicaciones
- Diseño usando software comercial off-the-shelf (COTS)
- Configuraciones y gestión de la configuración
- Líneas de productos

Evaluación

El curso posee 3 tipos de evaluación

- Interrogaciones y Examen
- Actividades
- Proyecto

Todas las notas se calculan con dos decimales, excepto la nota final que se entrega con un decimal.

Interrogaciones y examen

El curso contará con 2 interrogaciones a lo largo del semestre, se enviarán para ser realizadas en casa y tendrán 72 horas (3 días) para hacerlas, con un objetivo promedio de 3 horas de dedicación. Son estrictamente individuales y son revisadas por posible copia.

Las interrogaciones se realizarán en las siguientes fechas:

Interrogaciones	Fecha
1	29 de Septiembre
2	Jueves 16 de Noviembre
Examen	Lunes 4 de Diciembre

La falta a cualquier interrogación será calificada con nota 1. También el curso contará con un examen reprobatorio **NE** que cuenta dos veces en la nota. Se eliminará una de las peores notas (ya sea una interrogación o un **NE** de los dos considerados)

El examen podrá reemplazar una interrogación con nota 1, lo que implica que pueden faltar a una interrogación que deseen y recuperarla usando el examen. Por lo tanto, no necesitan justificación para faltar a una interrogación. Naturalmente, esto impide la eximición.

El examen será presencial y evaluará tópicos esenciales de todo el curso. En caso de faltar al examen con justificación via DIPRE, se le ofrecerá una instancia oral, con fecha a acordar según el caso.

Existirá una nota de eximición **NW = 5.00** y para eximirse debe cumplir los siguientes requisitos

- **NI** >= **NW**
- No tener notas bajo 4.00 en las interrogaciones
- Haber llenado correctamente el formulario de skills al inicio del curso
- Ética intachable en el curso (No haber plagado o violado el código de ética)

La nota de cátedra será:

$$NI = (I1 + I2 + 2*EX - MIN(I1,I2,EX))/3$$

En cambio si alcanza a la eximición, se calculará como:

$$NIE = AVG(I1,I2)$$

Actividades

A lo largo del semestre existirán 3 tipos de actividad

- Quizzes
- Actividades de modelación
- Debates

Se les dará un tiempo específico en clases (aprox. 20 min) para que puedan realizar las actividades del curso.

Quizzes

Se harán varios Quiz en horario de clases, que consisten en una pequeña batería de preguntas MCQ para monitorear su progreso en el curso. No hacer un quiz lo calificará con nota **1.00**. Se borrarán los 3 peores quiz que hayan rendido, por lo que no es necesario justificar el no rendimiento de estos. No es necesario estar presente para rendirlos, pero se recomienda fuertemente estar presente en la sala de clases para discutir las preguntas del quiz. Su duración es de 3 minutos, pudiendo variar.

Se calculará la nota **NQ** como el promedio simple de los quiz, descontando los 3 peores

Actividades de modelación

Se harán actividades de modelación y estimación en clases. Se harán 3 de estas actividades usando grupos formados ad-hoc, esto es, en la misma sala de clases. No asistir a estas actividades las calificará con nota **1.00**. Se borrará la peor actividad, por lo que no es necesario justificar la ausencia a estas.

Se calculará la nota **NM** como el promedio simple de las actividades, descontando la peor de ellas

Debate

En cada sección se construirán grupos de 5 personas, que se enfrentarán con otro grupo en un debate sobre dos problemas dados previamente de arquitectura de sistemas de software. Un grupo representará un equipo de arquitectura, mientras que el otro representará un equipo de stakeholders que hará preguntas y cuestionará la arquitectura propuesta. Posteriormente, se invertirán los roles con base a otro problema. La evaluación será in-situ con el equipo docente en el horario

de actividad, en base a una rúbrica que se publicará oportunamente.

Se publicarán los casos con anterioridad para que puedan ser estudiados por ambas partes antes de exponer/defender.

Esta actividad tendrá una evaluación de pares similar a la del proyecto, donde la nota en un 100% la determinará el grupo. Si un miembro no asiste al debate automáticamente les dará una nota de 1.00. Preguntas del público especialmente buenas a criterio del equipo docente podrán tener una bonificación adicional para el grupo que las haga.

Deberán subir el material requerido en formato PDF o PPTX a las 23:59 del día anterior de la presentación del grupo. El no subirlo anteriormente implicará que el grupo no se presenta al debate, y la posición de cuestionamiento para el otro grupo que participa será tomada por el equipo docente

Se calculará la nota **ND** como el promedio entre

- **Presentación arquitectura (20 ptos)**
- **Respuestas (20 ptos):** Respuestas del equipo a las preguntas de los stakeholders
- **Preguntas (20 ptos):** Preguntas del equipo al modelo del otro equipo

El balance de las notas es indicativo de que todas las partes deben ser de calidad y consideradas. Debido a que a medida que pase el semestre los estudiantes cuentan con más contenido para justificar, cada vez se hará una evaluación mas exhaustiva, donde deben usarse conceptos más justificados. Se realizará una simulación del debate antes de lo que se anuncie

Resultado

La nota de actividades **NA** se calcula como

$$NA = \text{AVG}(NQ + NM + ND)$$

Proyecto

El proyecto del curso girará en torno a una app inicial que deben crear. Cada entrega hará mejoras incrementales sobre esta app, hasta lograr una versión lista para cubrir la temática del proyecto.

Cada entrega podrá tener dos formatos

- Una seccion de requisitos funcionales de **3p** y otra de requisitos no funcionales de **3p**, que pueden contener variables, y de las que deben escoger un subconjunto. Pueden haber objetivos bonus por cada entrega
- Entrega con requisitos y diversos bonus opcionales

Cada entrega tendrá requisitos marcados como críticos. El no completar estos requisitos limitará la nota máxima a **4.00**

Va a ser posible obtener notas mayores a 7 en cada entrega de proyecto, con un máximo de 8. No obstante, la nota final **NP** calculada de todas las entregas no podrá ser mayor a 7.

Las entregas asumen que la parte mínima de la entrega anterior está **funcional** a la hora de iniciar la entrega. Si el grupo no logra conseguir la funcionalidad esencial del proyecto en la entrega (se les indicará que es esencial), se espera que la tengan preparada para la siguiente. Esto es por un tema práctico.

Se contará con una entrega introductoria individual (E_0) y varias entregas crecientes en grupos de 5 personas. Serán creados en base al sistema trio-par, que será explicado en el curso.

Entrega	Publicación	Entrega
E_0	Viernes 11 de Agosto	Domingo 3 de Septiembre

El proyecto grupal tiene 3 fases y éstas son:

Entrega	Publicación	Entrega
E_1	Lunes 4 de septiembre	Domingo 24 de septiembre
E_2	Lunes 25 de septiembre	Domingo 29 de octubre
E_3	Lunes 30 de octubre	Domingo 26 de noviembre

No se corregirá nada que no esté en un servidor en la nube pública de AWS, bajo ninguna excepcion. Todo proyecto que se presente con petición de corregir ejecutando cualquier clase de código en otro lugar tendrá automáticamente un **1.00**.

Las entregas son a las 23:00, y no se permitirá ningun commit a la rama master de su repo, modificar el software que esté en producción, así como modificaciones en la infraestructura a evaluar.

Se especificará por Canvas y Slack que se espera del proyecto y la estructuración de los equipos, además de los desafíos a resolver. Puede pasar que se les den más días para la entrega, pero las fechas de publicación se conservan.

Habrá un enunciado disponible para cada una de las entregas.

Cada entrega tiene un componente de coevaluación, evaluado en:

$$E_i = 1 + ((E_{i \text{ grupal}} - 1) * F_g)$$

Donde i es la entrega y F_g es un factor de coevaluación asignado por el grupo que va de 0 a 1.2 . Al momento de coevaluar, las asignaciones del equipo deben sumar n , siendo n el numero de miembros. Esto quiere decir que si alguien trabajó extra, otro debió trabajar menos en el equipo. Nótese que esto es un control absoluto del

equipo sobre la nota. No respetar esto invalidará su coevaluación.

La nota del proyecto total individual se calcula como:

$$NP = 0.25 * E_0 + 0.2 * E_1 + 0.25 * E_2 + 0.3 * E_3$$

Esta nota no podrá superar el 7.00 si se llegara a alcanzar una nota superior.

Criterio de aprobación

El curso se calificará con las siguientes ponderaciones

$$N_{calc} = 0.3 * NC + 0.3 * NA + 0.4 * NP$$

Cada nota a ponderar tendrá como máximo valor 7.00

Para aprobar el curso se requiere que

- Rinda el formulario de assessment inicial (también conocido como el de conocimientos) en el plazo estipulado
- $NE \geq 3.00$ (si se rindiera)
- $NC \geq 3.95$
- $NA \geq 3.95$
- $NP \geq 4.00$

La nota final, si se cumplen los criterios anteriores se calcula como:

$$NF = N_{calc}$$

Y aprueba el curso si y solo si

$$NF \geq 3.95$$

En otro caso

$$NF = \max(3.9, N_{calc})$$

Foro

Las dudas las pueden escribir en Slack, en los canales apropiados, o consultando directamente con los ayudantes o los profesores.

Bibliografía

- Ford & Richardson, "Fundamentals of software architecture: An engineering approach", O'Reilly
- Martin Fowler et al., "Patterns of Enterprise Application Architecture", Addison Wesley, 2002
- Richard N. Taylor, "Software Architecture, Foundations, Theory and Practice", Wiley, 2010.
- Spinellis & Gousios, "Beautiful Architecture", O'Reilly, 2009.
- Ian Gorton, "Essential Software Architecture", Springer, June, 2006.

Política de integridad académica DCC

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 Alumno de la Pontificia Universidad Católica de Chile](#). 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 [Codigo 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 [Codigo de honor](#)