



Pontificia Universidad Católica de Chile  
Escuela de Ingeniería  
Departamento de Ciencia de la Computación  
IIC2523 – Sistemas Distribuidos

## IIC2523 – Sistemas Distribuidos

### Programa de curso

|                   |  |
|-------------------|--|
| <b>Curso</b>      | : Sistemas Distribuidos  |
| <b>Semestre</b>   | : 2026-1   |
| <b>Traducción</b> | : <i>Distributed System</i>  |
| <b>Sigla</b>      | : IIC2523  |
| <b>Creditos</b>   | : 10   |
| <b>Formato</b>    | : Presencial   |
| <b>Docente</b>    | : Hernán Valdivieso ( <a href="mailto:hvaldiviso@uc.cl">hvaldiviso@uc.cl</a> ) |
| <b>Clases</b>     | : lunes y miércoles, módulo 3 (11:00 - 12:10)                                  |
| <b>Ayudantías</b> | : viernes, módulo 3 (11:00 - 12:10)  |
| <b>Requisitos</b> | : IIC2333 o (IIC2233 y IIC2343) o ICS2122 o IRB2002 o IDI2025 o IBM2123        |

### 1. Descripción

Un sistema distribuido es una colección de sistemas computacionales interconectados en que procesos y recursos están repartidos entre varios computadores. Se implementa para compartir recursos y mejorar la eficiencia y confiabilidad de una solución computacional, y su diseño debe ser seguro y esconder los detalles de la distribución de procesos, datos y control. Lograr todos estos objetivos simultáneamente es desafiante.

El curso estará orientado a estudiantes con experiencia en programación y con una capacidad para estudiar e investigar bajo sus propios recursos.

### 2. Resultados de aprendizaje

Al finalizar el curso, los estudiantes serán capaces de:

- **Comprender** los principios y desafíos de los sistemas distribuidos.
- **Comprender** el desafío de lograr tolerancia a fallas, **explicar** las limitaciones y los compromisos involucrados, y **aplicar** técnicas y algoritmos distribuidos en la creación de sistemas tolerantes a fallos y seguros.
- **Aplicar** los diversos tipos de procesos que forman un sistema distribuido, **analizando** sus características distintivas y **explicando** el rol de cada uno.
- **Diseñar, explicar y aplicar** distintos algoritmos necesarios para coordinar el trabajo de los procesos, y para mantener la consistencia de información en presencia de la replicación de recursos.
- **Evaluuar** el diseño de un sistema distribuido considerando su capacidad de compartir recursos, su confiabilidad y su eficiencia, utilizando criterios técnicos y funcionales.

### **3. Contenidos específicos**

El siguiente listado son las temáticas mínimas que se revisarán en el curso. El orden de los contenidos puede variar durante el transcurso del semestre.

- **Sistemas Distribuidos y la comunicación:** Definiciones, desafíos, procesos vs *thread*, tipo de comunicación (directa, indirecta, remota), protocolos de comunicación (TCP y UDP), *Remote Procedure Call* (RPC).
- **Coordinación en un Sistema Distribuido:** Sincronización de relojes, algoritmos de consenso, ataque bizantino, elección de líder y exclusión mutua distribuida.
- **Fiabilidad en Sistemas Distribuidos:** tolerancia a fallas, replicación y consistencia de datos, transacciones distribuidas, control de concurrencia, teorema PAC y PACELC.
- **Seguridad y Estructuras Resilientes:** Mecanismos de seguridad (Autenticación, Autorización, Criptografía y Monitoreo), Redes Peer-to-Peer (P2P).
- **Tópicos Avanzados en sistemas Distribuidos:** Revisión de tecnologías y aplicaciones modernas.

### **4. Metodología**

La metodología del curso se basa en clases expositivas, el análisis y discusión de ejemplos específicos, y la realización de ejercicios tanto teóricos como prácticos.

El curso contempla un horario de ayudantías, el cual estará destinado principalmente a que el estudiantado utilice dicho módulo en la realización de actividades para la casa. Por defecto, no se utilizará el horario de ayudantía, salvo que se informe previamente a través de la plataforma del curso.

Adicionalmente, se publicará un aviso, en Canvas, el último día hábil de cada semana. En caso de que dicho día sea feriado, el aviso se enviará el día hábil inmediatamente anterior. Este aviso incluirá un resumen de los eventos relevantes de la semana siguiente, tales como los contenidos a tratar en clases y las evaluaciones programadas. Es responsabilidad del estudiantado revisar este aviso durante la noche del último día hábil o durante la mañana del primer día hábil de la semana siguiente.

### **5. Evaluaciones**

La evaluación de cada estudiante será efectuada mediante diversos tipos: evaluaciones escritas, actividades, controles periódicos, tareas a lo largo del curso y un proyecto al final del semestre. Todas estas evaluaciones buscan proponer instancias de práctica o evidencia de las habilidades indicadas al comienzo del documento como resultados de aprendizaje del estudiantado del curso. Hacen esto al proponer situaciones de aplicación de los contenidos del curso, ya sea reconociendo y aplicando diversos conceptos de sistemas distribuidos, o implementando algoritmos y arquitecturas distribuidas.

#### **5.1. Evaluaciones Escritas y actividades (EA)**

El curso contempla una serie de actividades, dos interrogaciones (I1, I2) y un examen final (EX). Las interrogaciones y examen son de carácter presencial y en las fechas definidas por la Dirección de Pregrado.

Sea **NA** las notas de actividades y **EA** la nota final correspondiente a las evaluaciones escritas y actividades. **EA** se calculará como:

$$\mathbf{EA} = 25\% \times \mathbf{NA} + 35\% \times \mathbf{MAX(I1, I2)} + 40\% \times \mathbf{Examen}$$

### **5.1.1. Interrogaciones y examen**

Respecto de estas evaluaciones, cada interrogación evaluará un subconjunto de los contenidos del curso, mientras que el examen abarcará la **totalidad de la materia**. Las fechas de estas evaluaciones son:

- Interrogación 1: Miércoles 29 de Abril.
- Interrogación 2: Jueves 18 de Junio.
- Examen: Lunes 6 de Julio a las 13:30.

En caso de ausentarse a alguna de estas instancias, la evaluación correspondiente será calificada con la nota mínima (1.0), y su inasistencia solo podrá ser justificada ante la Unidad Académica. Además, debido a la fórmula utilizada para el cálculo de la nota final, solo se realizará una evaluación recuperativa de las interrogaciones en el caso de ausentarse justificadamente a ambas interrogaciones. El formato de dicha evaluación recuperativa quedará a criterio del cuerpo docente y se llevará a cabo durante el período de exámenes.

### **5.1.2. Actividades**

Las actividades del curso consistirán principalmente en pequeñas evaluaciones cuyo objetivo es asegurar que el estudiante se mantenga al día con lo expuesto en el curso. Estas actividades corresponderán principalmente a **controles** de los contenidos, pero se podrán contemplar otros tipos de actividades.

Las actividades podrán ser de carácter **obligatorio u opcional**. Las actividades obligatorias otorgarán **puntos obligatorios**, mientras que las actividades optionales otorgarán **puntos adicionales**. Ambos tipos de puntos serán utilizados para calcular la nota final de actividades. Por otro lado, la naturaleza de cada actividad y las reglas para obtener los puntos obligatorios u optionales serán informadas explícitamente al momento de su publicación.

Respecto a los **controles** de contenidos:

- Estarán disponibles en la plataforma del curso y contarán con retroalimentación automática (visualización de las preguntas junto con sus respuestas correctas). Dicha retroalimentación solo estará habilitada para aquellos estudiantes que respondan la evaluación dentro de los plazos establecidos.
- Generalmente serán de un máximo de **dos preguntas de corrección automática**, tales como selección múltiple, verdadero/falso o asociación de conceptos, entre otras.
- Deberá ser respondido, como máximo, hasta el **viernes de la semana en que sea publicado, a las 20:00 horas**. En caso de que dicho viernes sea feriado, el plazo se extenderá hasta el siguiente día hábil, manteniéndose como hora límite las 20:00 horas.

En el caso de **otro tipo de actividades**, la modalidad y extensión de dicha actividad será calibrada durante el transcurso del semestre.

Finalmente, sea **N** la cantidad total de puntos obligatorios que se puede lograr en el semestre, y **PA** y **PO** la cantidad de puntos adicional y obligatorios obtenidos por el estudiante durante el semestre, la nota de actividades (**NA**) se calculará como:

$$\mathbf{NA} = \min \left( \frac{PA + PO}{N - 6} \times 6 + 1, 7,0 \right)$$

El propósito de esta evaluación es fomentar un estudio constante de la materia impartida y proporcionar

material de preparación para cada evaluación escrita. Además, dado que la nota máxima (7.0) puede alcanzarse aun sin responder 3 controles obligatorios, no se aceptarán justificativos para extender los plazos ni instancias recuperativas para este tipo de evaluación.

## 5.2. Tareas y Proyecto (NTP)

El curso contará con tres instancias de tareas prácticas y una nota por proyecto (**Pr**). La nota final de las tareas y proyecto (**NTP**) se calcula como:

$$NTP = 20\% \times T1 + 25\% \times T2 + 25\% \times T3 + 30\% \times Pr$$

La entregas de cada evaluación será **siempre** a las 20:00 horas. No caso de no entregar una evaluación en los tiempos informados en la plataforma implicará que dicha evaluación será calificada con la nota mínima. No se aceptará, bajo ninguna circunstancia, entregas que no sean por la plataforma del curso.

### 5.2.1. Tareas

El desarrollo de las tareas será, por defecto, **estrictamente individual**. No obstante, se permitirá el desarrollo en **parejas** únicamente a quienes se inscriban previamente en el formulario correspondiente, el cual será único para cada evaluación y se informará en el enunciado de la respectiva tarea. El desarrollo de una tarea en parejas sin una inscripción oportuna será considerado una falta al reglamento del curso y será sancionado severamente, pudiendo implicar desde un descuento significativo en la nota de ambos integrantes hasta la asignación de la nota mínima.

Las tareas con corrección automática dispondrán de un conjunto muy limitado de tests públicos, los cuales ayudarán a verificar que el código pueda ejecutarse. No obstante, será parte de la evaluación que el estudiante diseñe nuevos tests para validar el correcto funcionamiento de la tarea en distintos casos borde. Además, una vez finalizado el plazo de entrega, no se aceptarán solicitudes para corregir o ajustar el código; por lo tanto, será responsabilidad del estudiante asegurarse de enviar una versión que se ejecute correctamente. En caso contrario, la tarea será evaluada con la nota mínima.

Respecto a los plazos de las tareas, cada una contará con un plazo de entre 7 y 10 días hábiles, dependiendo de la evaluación, para su realización.

Finalmente, para el desarrollo de las tareas, se permite el uso de material externo o el apoyo de IA. No obstante, su uso e integración debe ser debidamente citado incluyendo fuentes y *prompt* (en el caso de IA) y además ser producto de un proceso de investigación, integración y adaptación. El cuerpo docente puede interrogar al estudiante respecto al trabajo entregado, y si se evidencia que no hubo un correcto proceso de investigación, integración y adaptación, se considerará una falta a la integridad académica.

### 5.2.2. Proyecto

El proyecto corresponde a una evaluación realizada al final del semestre, en la que **grupos de hasta cuatro integrantes** deberán investigar sobre un tema asignado por el profesor en el contexto de los Sistemas Distribuidos. Durante el tercer mes de clases se publicará un enunciado con el detalle de esta evaluación.

Posteriormente, cada grupo deberá exponer su investigación en algún formato por definir durante el semestre (presentación, página web, informe, video, entre otros) y, además, evaluar las investigaciones de sus compañeros.

### **5.3. Recorrección**

Luego de publicadas las notas de una evaluación, se dispondrá de un periodo de **1 semana** para recibir solicitudes de recorrección. En esta instancia, los estudiantes podrán:

- Solicitar una aclaración adicional respecto a aspectos de la retroalimentación que no hayan quedado completamente claros.
- Notificar sobre algún error o discrepancia en la corrección realizada por el cuerpo docente. Esta notificación debe incluir una justificación adecuada que demuestre el error identificado.

Solo se aceptarán solicitudes enviadas **dentro del periodo establecido y mediante los canales oficiales** definidos por el curso.

En esta instancia, el docente puede revisar el aspecto puntual por el cual se recorreja o toda la evaluación. Por lo tanto, la nota de dicha evaluación puede subir, mantenerse o bajar. La decisión que se tome en esta instancia es inapelable.

Es importante destacar que la recorrección no constituye una instancia para corregir errores presentes en la evaluación entregada, sino un espacio destinado a solicitar una retroalimentación más detallada o a informar sobre un posible error en el proceso de corrección, con el fin de asegurar que la nota asignada se ajuste a lo solicitado originalmente en el enunciado correspondiente y acorde a los contenidos evaluados.

Finalmente, toda solicitud de recorrección deberá ser confeccionada y redactada íntegramente por el estudiante, utilizando información fidedigna y verificable. En caso de presentar una recorrección con información falsa, de fuentes no confiables, como sistemas de Inteligencia Artificial (IA), o que haya sido explícitamente redactada por una IA, esta no será aceptada y podrá implicar una sanción grave, ya sea en la evaluación objeto de la recorrección o en la nota final del curso.

### **5.4. Atraso en las tareas**

Todas las tareas contarán con fechas y plazos fijos de publicación y entrega, con el fin de marcar el flujo de revisión de contenidos en el curso y organizar la carga que implica entregar retroalimentación oportuna a cada estudiante.

Para las **tareas** se permitirá realizar entregas con máximo **2 días de atraso**, pero se aplicará un descuento a la nota máxima que se puede obtener en función del tiempo de atraso. Luego, toda entrega realizada posterior a los 2 días de atraso será evaluada con la nota mínima. El descuento por atraso consiste en **10 décimas menos por día a la nota máxima**, donde cada día se comienza a considerar desde el tiempo 00:00:01 de atraso hasta las 24:00:00 de atraso. En términos matemáticos, la fórmula para calcular la nota final de una evaluación es:

$$\text{nota\_final\_tarea} = \text{Max}[\text{Min}(7 - 1,0 \times \text{días\_de\_atraso}, \text{nota\_obtenida}) - \text{descuento\_total}, 1]$$

En otras palabras, este descuento solo acota la nota que puede aspirar el estudiantado. En caso de que la nota obtenida en la evaluación sea menor a la nota máxima a aspirar por el atraso, la evaluación no presentará ningún descuento.

Finalmente, en caso de que ocurran situaciones de fuerza mayor que impidan al estudiante disponer de al menos el 50 % de la duración oficial de la tarea (considerando únicamente días hábiles y sin contabilizar el periodo de atraso), y siempre que dichas situaciones se encuentren debidamente justificadas ante la Unidad Académica, se podrá evaluar un eventual cambio en la fecha de entrega. Dicho cambio quedará a criterio del equipo docente y no necesariamente corresponderá a la totalidad de los días justificados.

En cambio, si la justificación corresponde a un período inferior al 50 % de la duración oficial de la tarea, únicamente se evaluará una eventual reducción en la penalización por atraso, sin que la fecha de entrega sea modificada.

## 5.5. Calificaciones y aprobación

A final del semestre, cada estudiante contará con dos notas finales: una correspondiente a la ponderación de las tareas y proyecto: **NTP**; y una correspondiente a la nota de evaluaciones escritas y actividades: **EA**. Dadas estas dos notas, se calcula la nota de presentación **NP** como:

$$\mathbf{NP} = 0,5 \times \mathbf{NTP} + 0,5 \times \mathbf{EA}$$

Cada estudiante aprobará si:

1. Su nota de presentación (**NP**) es mayor o igual a 3.95.
2. Su nota de tareas y proyecto (**NTP**) es mayor o igual a 3.95.
3. Su nota de evaluaciones escritas y controles (**EA**) debe ser mayor o igual a 3.95.

En caso de cumplir todos los criterios, la nota final **NF** de cada estudiante será igual a la nota de presentación. En otro caso, la nota final **NF** será al mínimo entre **NP** y **3.9**.

Todas las notas, a excepción de la nota final del curso **NF**, serán calculadas con un redondeo a **dos decimales**.

La nota final del curso **NF** que se calculará con redondeo a **un decimal**.

## 6. Integridad académica

Este curso busca formar personas y profesionales con integridad y ética, y siempre comenzará del supuesto de que el trabajo de sus estudiantes refleja estos principios.

Pero en situaciones donde lo contrario se ponga en evidencia, se tomarán pasos para identificar la verdad y eventualmente aplicar medidas de corrección. En aspectos formales, se rige para este curso tanto la política de integridad académica del Departamento de Ciencia de la Computación como el [Código de honor de la Escuela de Ingeniería](#). Luego, cualquier situación de **falta a la ética o integridad académica** detectada en alguna evaluación tendrá como **sanción un 1.1 final en el curso**. Esto sin perjuicio de sanciones posteriores que estén de acuerdo a la Política de Integridad Académica de la Escuela de Ingeniería y de la Universidad, que sean aplicables al caso.

Debido a la naturaleza de la disciplina en la que se enmarca el curso, está permitido el uso de código escrito por un tercero, pero solo bajo ciertas condiciones. Primero que todo, el uso de código ajeno **siempre debe** estar correctamente referenciado, indicando la fuente de donde se obtuvo. Y por otro lado, se permite el uso de código encontrado en internet u otra fuente de información similar, siempre y cuando su autor sea **externo al curso**, o en su defecto, sea parte del **equipo docente** del curso. Es decir, se puede hacer referencia a código ajeno al curso y código perteneciente al curso pero solo aquel escrito por el equipo docente, como material o ayudantías. Luego, compartir o usar código usado como entrega de una evaluación **actual o pasada** del curso se considera falta a la ética.

Finalmente, en este curso, el uso de herramientas generadoras de código o de IA está permitido como una herramienta de apoyo, siempre y cuando se refuerce correctamente cuándo es utilizado, y su uso sea producto de un proceso de investigación, integración y adaptación. En este sentido, se espera que toda respuesta entregada por una herramienta generadora de código o de IA pase por un proceso de análisis

crítico y modificación antes de ser incluida en una evaluación. En caso de entregar una tarea donde se identifique que solo hubo un proceso de copiar la respuesta dada por una herramienta generadora de código o de IA, esto no será aceptado y será considerado como una falta a la integridad académica.

## 7. Bibliografía

- Van Steen, M., & Tanenbaum, A. S. (2023). *Distributed systems\**(4th ed.). distributed-systems.net.
- Anthony, R. (2016). *Systems programming—Designing and developing distributed applications*. Morgan Kaufmann.
- Coulouris, G., Dollimore, J., & Kindberg, T. (2005). *Distributed systems: Concepts and design* (International Computer Science). Addison-Wesley Longman.