



Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ciencia de la Computación
IIC2026 – Visualización de Información

IIC2026 – Visualización de Información

Programa de curso

Curso	: Visualización de información
Traducción	: <i>Information Visualization</i>
Sigla	: IIC2026
Creditos	: 10
Formato	: Presencial
Docente	: Hernán Valdivieso (hfvaldiviso@uc.cl)
Clases	: martes y jueves, módulo 2 (9:40 -10:50)
Ayudantías	: viernes, módulo 4 (12:20 -13:30)
Requisitos	: IIC1103 - Introducción a la programación
Sitio Web	: puc-infovis.github.io y Canvas

1. Descripción

En la época actual donde los datos e información abundan y sobran, las habilidades de comunicación de la información son clave. El área de visualización de información es una intersección no trivial entre disciplinas computacionales, de diseño gráfico y de psicología perceptiva. Este curso revisa cómo explorar este espacio común y muestra tanto técnicas computacionales como de diseño gráfico para guiar la creación de visualizaciones de información.

El curso estará orientado a estudiantes con experiencia básica en programación, y provee experiencias prácticas en el análisis, diseño, y creación de herramientas en visualización de información, que son habilidades esenciales para el perfil de científico de datos o perfiles relacionados. También como parte del perfil UC, este curso promueve una mirada interdisciplinaria de resolver problemas, ya que incorpora contenidos y actividades relacionadas a tres mundos distintos.

2. Objetivo General

Este curso busca que sus estudiantes conozcan y utilicen un modelo de trabajo complejo para guiar el diseño y la implementación (programación o uso de *software*) de herramientas en el área de visualización de información.

3. Resultados de aprendizaje

Al finalizar el curso, sus estudiantes serán capaces de:

- **Reconocer** etapas y conceptos básicos involucrados en un proceso de diseño de una visualización de información.

- **Analizar** visualizaciones de información mediante una deconstrucción de las codificaciones visuales que le componen.
- **Clasificar** los conjuntos de datos reflejados en herramientas de visualización según su naturaleza y contexto.
- **Evaluar** herramientas de visualización según principios perceptivos, principios de diseño gráfico y la efectividad de sus componentes.
- **Conocer** un *software* especializado en construir herramientas de visualización estáticas e interactivas.
- **Resolver** necesidades de comunicación de información mediante la programación de herramientas de visualización interactivas.
- **Diseñar** una herramienta de visualización de información mediante un modelo de trabajo, en una situación específica y escogida por el alumnado.

4. Contenidos específicos

El siguiente listado son las temáticas que se revisarán en el curso:

- **Unidad 1:** Implementación de visualizaciones
 1. **Tecnologías web:** Introducción práctica a las tecnologías web HTML, CSS y JavaScript.
 2. **SVG:** Formato de gráfico vectorial y bidimensionales para diseñar las visualizaciones.
 3. **Librería D3.js:** Librería de bajo nivel especializada para la creación de visualizaciones de información en conjunto a tecnologías web.
 4. **Looker Studio:** Software especializado para la creación de visualizaciones de información y *dashboards*.
- **Unidad 2:** Análisis y diseño de visualizaciones
 1. **Modelo anidado de análisis y validación:** Cuadro de trabajo propuesto por Tamara Munzner que guía el proceso de análisis, diseño y de validación de visualizaciones de información.
 2. **Codificaciones visuales elementales:** Los conceptos de marcas y canales como codificaciones visuales básicas.
 3. **Fundamentos de percepción de codificaciones:** Principios de percepción humana que explican los niveles de efectividad de codificaciones visuales.
 4. **Abstracción de datos y tareas:** Proceso de análisis y categorización de los datos que una visualización mostrará a sus usuarios, y de las tareas que esos usuarios buscan realizar.
 5. **Codificaciones visuales específicas:** Codificaciones gráficas propias a distintos tipos de conjuntos de datos. Particularmente se revisarán casos para datos tabulares, datos de red y datos georreferenciados.
 6. **Codificaciones de interacción:** Posibilidades de codificaciones utilizando la interacción de usuarios con herramientas computacionales.

5. Metodología

La metodología del curso se basa en clases expositivas, discusiones de ejemplos específicos, evaluaciones formativas y la realización de ejercicios teóricos y prácticos. Adicionalmente, el curso dispondrá de clases prácticas, donde se verán contenidos de programación o uso de *software* especializado en la confección de visualización y *dashboards*. Se recomienda la asistencia a las clases prácticas con un *notebook* para poder acceder, replicar, modificar y experimentar con los códigos o *software* presentado durante dichas clases.

El curso contará con ayudantías semanales con el fin de brindar otro espacio de estudio para las y los estudiantes del curso. Para cada sesión, ayudantes prepararán ejemplos de aplicación para reforzar contenidos previamente revisados en clase, prepararan material complementario para apoyar el avance del curso o las evaluaciones, o dedicarán el módulo para resolver dudas sobre los contenidos o evaluaciones del curso. Todo material utilizado en la ayudantía siempre estará disponible al final del día correspondiente al módulo de ayudantía.

6. Evaluaciones

La evaluación de cada estudiante será efectuada mediante cuatro tipos de evaluaciones, algunas opcionales y otras obligatorias: una de naturaleza continua durante el semestre, **la revisión de contenidos**; tres del tipo práctico, **tareas, control** y un **proyecto**; y un **examen bonus final**.

Todas estas evaluaciones buscan proponer instancias de práctica o evidencia de las habilidades indicadas al comienzo del documento como resultados de aprendizaje de los y las estudiantes del curso. Hacen esto al proponer situaciones de aplicación de los contenidos del curso ya sea aplicando el proceso de diseño presentado o implementando soluciones mediante programación.

6.1. Revisión de contenidos (opcional)

Para la revisión de los contenidos específicos a ver en el curso, se dispondrán de diferentes instancias/actividades que buscan levantar evidencia de que el estudiantado revisó y entendió los contenidos expuestos en el curso. Estas instancias/actividades podrán ser mini controles de alternativas sobre la materia, participar durante la clases en plataformas de aprendizaje basadas en juego como Menti o Kahoo, buscar visualizaciones que cumplan ciertos criterios, entre otros.

Este tipo de evaluación **es opcional** para el estudiantado. Por cada instancia/actividad lograda al 100%, se otorgará **un punto RC**. Cada **cuatro puntos RC** reunidos por el estudiantado, se otorgará 1 décima al promedio final del curso.

6.2. Tareas y Control (obligatorio)

Las tareas son evaluaciones sumativas cuyo fin es evaluar y buscar evidencia de ciertas habilidades desarrolladas por las y los estudiantes durante el curso. Este tipo de evaluación **no es opcional** para cada estudiante y su desarrollo es **individual**. En caso de no entregar una tarea, esta será evaluada con la nota mínima.

El control corresponde a otra evaluación sumativa que evaluará contenidos teóricos del curso. Este control corresponderá a un cuestionario en Canvas con diferentes preguntas de alternativas, verdadero/falso, selección múltiples, entre otros. Este tipo de evaluación **no es opcional** para cada estudiante y su desarrollo es **individual**. En caso de no responder el control, este será evaluada con la nota mínima.

Durante el semestre, el curso contará con tres instancias de tareas prácticas que otorgarían a cada estudiante notas: **T1, T2, T3** y de un control: **C1**.

Respecto los plazos, la T1, T3 y el control dispondrá de una semana para su confección, mientras que la T2 contará con 2 semanas para su realización. Además, todas las entregas serán **siempre** a las 20:00 hrs.

La nota final de los tareas y controles (**NTC**) se calcula como:

$$\text{NTC} = \frac{(\text{T1} + 2 \times \text{T2} + \text{T3} + \text{C1}) - \text{Min}(\text{T1}, \text{T2}, \text{T3}, \text{C1})}{4}$$

6.3. Proyecto (obligatorio)

Durante la segunda mitad del curso, se presentará una última evaluación práctica **extensa, con posibilidad de hacer de a pares y no opcional**.

Esta busca evaluar el proceso de diseño e implementación aprendido a lo largo del curso en un caso más extenso. Esta evaluación contará con **dos entregas**, la primera consistirá en un informe donde se diseñará una herramienta de visualización, mientras que la segunda entrega consistirá en la herramienta desarrollada/programada en su completitud. Se dispondrán de módulos de clases y/o de ayudantía para trabajar en esta evaluación de forma presencial.

En caso de realizar el proyecto en parejas, **ambas entregas deberán ser con los mismos integrantes o de forma individual**, es decir, una vez formada una pareja, esta no se puede cambiar, solo se permitirá disolver las parejas para que continúen su trabajo de forma individual.

6.4. Examen Bonus (opcional)

A final del curso, se presentará una última evaluación **presencial, extensa y opcional**, la cual se permitirá hacer de forma **individual o en parejas**.

Esta busca evaluar los aspectos teóricos del curso mediante un conjunto de preguntas de alternativas, verdadero/falso, entre otros. Esta evaluación será en el horario de examen definida por la Dirección de Pregrado y otorgará una bonificación **al promedio final del curso**.

6.5. Recorrección

Luego de publicadas las notas de una evaluación, se dará un periodo de una semana para recibir solicitudes de corrección. En esta solicitud los y las estudiantes tendrán la oportunidad de solicitar mayor explicación de aspectos en la retroalimentación que no estén de acuerdo o no entiendan completamente. Para los casos donde la solicitud es respecto a un desacuerdo con la retroalimentación. Esta solicitud deberá estar debidamente justificada. Solo se aceptarán solicitudes que sean enviadas dentro del periodo de una semana y por los canales que el curso disponga para este propósito.

En caso de que la respuesta a la solicitud de corrección no sea satisfactoria, se deberá llenar un formulario —dentro de una semana de publicada la corrección— para solicitar que el o la docente revise el caso. En esta instancia, el o la docente puede revisar el aspecto puntual por el cual se recorre 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.

6.6. Flexibilidad en las Tareas y Proyecto

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 hasta **3 días de atraso**, pero se aplicará un descuento en función del tiempo de atraso. Este descuento consiste en **5 décimas menos por día a la nota máxima**. En términos matemáticos, la fórmula para calcular la nota final de una evaluación es:

$$\text{nota_final_tarea} = \text{Min}(7 - 0,5 \times \text{días_de_atraso}, \text{nota_obtenida}) + \text{bonus}$$

En otras palabras, este descuento solo acota la nota que puede aspirar el estudiantado. En caso que la nota obtenida en la evaluación sea menor a la nota máxima a aspirar, la evaluación no presentará ningún descuento. Luego, una vez aplicado el descuento por atraso, se aplicará cualquier bonus que presente la evaluación.

Aún así, en caso de que un o una estudiante presente problemas personales o problemas de fuerza mayor, la persona podrá solicitar una extensión mayor a los 3 días de atraso permitidos en la tarea. Dicha extensión incluirá una penalización a la nota y la extensión a otorgar se definirá para cada solicitud. También, se confiará en el criterio personal de cada estudiante en hacer este tipo de solicitudes y la carga que significa en el equipo docente.

Para el **proyecto**, cada entrega contará con hasta **1 día de atraso**, y se aplicará un descuento de **10 décimas a la nota máxima**. Además, dado el extenso plazo para realizar cada entrega, esta evaluación no presentará flexibilidad para extender más la fecha de entrega. Solo se aceptarán justificaciones aceptadas previamente por la Unidad Académica del estudiante.

6.7. Calificaciones y aprobación

A final del semestre, cada estudiante contará con dos notas finales: una correspondiente a la ponderación de las tareas y control: **NTC**; y una correspondiente a la nota del proyecto: **PR**. Dada estas notas, se calcula la nota de presentación **NP** como:

$$\text{NP} = 0,6 \times \text{NTC} + 0,4 \times \text{PR}$$

Cada estudiante aprobará si: 1. Su nota de presentación **NP** es mayor o igual a 3.95. 2. Su nota de tareas y control **NTC** es mayor o igual a 3.95. 3. Su nota del proyecto **PR** debe ser mayor o igual a **MIN_PRO**.

MIN_PRO corresponde a la nota mínima que deben obtener en el proyecto, y esta dependerá de la nota **NTC** de cada estudiante. Esta nota mínima se calcula como:

1. Si **NTC** es mayor o igual a 5.95, **MIN_PRO** será 2,95.
2. Si **NTC** es mayor o igual a 4.95, **MIN_PRO** será 3,45.
3. En otro caso, **MIN_PRO** será 3,95.

El objetivo de esta política es relajar la exigencia de aprobación del proyecto para todo estudiante que logró demostrar, adecuadamente, su aprendizaje en las tareas, pero seguir exigiendo un desarrollo mínimo de la última evaluación integradora.

En caso de cumplir todos los criterios, la nota final **NF** de cada estudiante será igual a la nota de presentación bonificada con las décimas otorgadas por la revisión de contenidos **RC** y el examen bonus (**NP + bono décimas**). En otro caso, la nota final **NF** será al mínimo entre **NP + bono décimas** 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**.

7. 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** 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 está permitido, siempre y cuando se referencie correctamente cuando es utilizado y quedará a responsabilidad del estudiante utilizarlo de forma responsable.

8. Bibliografía

- T. Munzner. Visualization Analysis and Design, CRC Press, 2014.
- S. Murray, Interactive Data Visualization for the Web, O'Reilly, 2017.
- C. Ware, Information Visualization - Perception for Design, Morgan Kaufmann Publishers, 2004.
- E. Tufte, The Visual Display of Quantitative Information, Graphics Press, 2001.
- A. Kirk, Data visualisation: A handbook for data driven design. Sage. 2016.
- A. Cairo. How charts lie: Getting smarter about visual information. WW Norton & Company, 2019.