

Programa de Curso

Marzo de 2021

Profesores:	Ándrés Cádiz afcadiz@uc.cl Raimundo Herrera rjherrera@uc.cl Matías toro matias.toro@uc.cl
Ayudante jefe cátedra	Valentina Álvarez vjalvarez@uc.cl
Ayudante jefe proyecto	Andrés Pincheira arpincheira@uc.cl
Clases:	miércoles, módulos 5 y 6. Por Zoom.
Horario de Ayudantías:	viernes, módulo 5. Por Zoom.

Descripción

Las bases de datos forman parte del núcleo del desarrollo de aplicaciones comerciales modernas, y son indispensables para cualquier aplicación que requiera almacenar grandes volúmenes de datos, actualizarlos y consultarlos de manera eficiente. El propósito de este curso es introducir al alumno en el diseño y uso de los sistemas de bases de datos, con especial atención en bases de datos relacionales.

Objetivo General

Durante el curso el alumno aprenderá la teoría detrás del modelo relacional, incluyendo el lenguaje SQL (hoy estándar para consultar bases de datos relacionales). Sabrá aplicar esta teoría al diseño y uso de bases de datos relacionales. Además, conocerá como interpretar y consultar bases de datos noSQL. Al final del curso el alumno podrá diseñar y manejar una base de datos relacional en un ambiente real. También conocerá los modelos teóricos usados en el estudio de las bases de datos, podrá analizar y comparar distintos esquemas y consultas de acuerdo a su utilidad práctica, e interpretar las ventajas y desventajas de los modelos de datos relacionales en comparación con modelos noSQL.

Competencias e indicadores de desempeño

Competencia	Indicadores de Desempeño
1. Entender qué es un sistema de bases de datos, y conocer las ventajas y desventajas de los modelos de datos relacional y noSQL.	<ol style="list-style-type: none">1. El alumno conoce la necesidad de implementar y usar sistemas de bases de datos.2. Entiende la diferencia entre modelos de datos, esquema de datos e instancia de un esquema.3. Distingue entre distintos modelos de datos, conoce que hay distintas alternativas, sus ventajas y desventajas.

2. Diseñar bases de datos relacionales para aplicaciones basadas en casos reales	<ol style="list-style-type: none"> 1. Crea un esquema de datos a partir de una descripción verbal. 2. Abstrae y optimiza requerimientos mediante un modelo ER. 3. Conoce como generar un esquema relacional a través de un modelo ER. 4. Compara distintos esquemas de acuerdo a los requerimientos. 5. Aplica la teoría de formas normales a la comparación y/o optimización de esquemas.
3. Usar el lenguaje de consulta SQL para bases de datos relacionales	<ol style="list-style-type: none"> 1. Conoce la sintaxis y la semántica de SQL. 2. Escribe consultas simples. 3. Establece un plan lógico frente a una consulta, y lo expresa en SQL. 4. Puede comprender el significado de una consulta compleja y establecer su plan lógico.
4. Usar bases de datos para trabajar en problemas de ciencia de datos en entorno realista	<ol style="list-style-type: none"> 1. Importa y exporta datos desde y hacia distintos formatos. 2. Usa consultas SQL para obtener transformaciones de datos que puedan responder hipótesis simples sobre datos. 3. Usa consultas SQL para transformar datos de manera de poder usar técnicas estadísticas básicas.
5. Comprender conceptos esenciales en teoría de bases de datos, especialmente el Álgebra Relacional (AR), y conoce como se usa para estudiar las limitaciones de las bases de datos	<ol style="list-style-type: none"> 1. Conoce la sintaxis y la semántica del Álgebra Relacional. 2. Puede transformar consultas de SQL a AR, y vice-versa. 3. Entiende el concepto consultas no expresables en SQL o AR, y puede dar ejemplos. 4. Conoce el concepto de plan de consultas, sabe el papel del AR en la optimización de consultas.
6. Conocer nociones básicas del paradigma noSQL, búsqueda de texto y herramientas de análisis de datos	<ol style="list-style-type: none"> 1. Conoce el paradigma key-value (JSON) el sistema MongoDB. 2. Conoce los desafíos detrás de la búsqueda en textos. 3. Conoce herramientas clásicas de análisis de datos.

7. Comprender los desafíos y problemas más importantes de la industria de bases de datos actuales. Discutir y saber plantear soluciones de diseño e implementación de bases de datos.	<ol style="list-style-type: none"> 1. Describe la arquitectura de un sistema de bases de datos. 2. Conoce y usa conceptos avanzados del lenguaje SQL, como vistas, transacciones, procedimientos almacenados, cursores y serialización. 3. Implementa una plataforma Web para acceder a una base de datos, y mostrar el resultado de las consultas. 4. Diseña soluciones para un caso real del problema de integración de datos. 5. Entiende cómo un buen sistema de bases de datos puede ayudar a tomar decisiones estratégicas a nivel industrial.
---	---

Contenidos

Por qué un sistema de bases de datos

1. Importancia de las bases de datos.
2. Sistemas de bases de datos versus almacenamiento ad-hoc.
3. Arquitectura básica de los sistemas de bases de datos.
4. Modelos de datos: XML, key-value, relacional.

Introducción al modelo relacional

7. Conceptos básicos.
8. SQL básico.

Modelamiento (diseño)

9. Creación de Tablas.
10. Lenguajes de modelamiento; diagramas E/R.
11. Principios de diseño.
12. Desde el diagramas E/R a esquemas relacionales.
13. Modelado de restricciones, formas normales.

SQL avanzado

14. Outerjoins y valores nulos.
15. Vistas.

Metodología

Las clases se dividen en submódulos expositivos, de discusión activa entre pares, o de desarrollo de guías, o de trabajo en el proyecto. Un proyecto semestral de desarrollo de una base de datos cuya información se

16. Procedimientos almacenados y cursores.

17. Transacciones y control de concurrencia.

Conceptos teóricos

20. El Álgebra relacional.
21. Limitaciones de SQL: valores nulos, clausura transitiva, etc.

Indexación y optimización

22. Procesamiento de operadores relacionales.
23. Optimización de consultas relacionales.
24. Índices en memoria secundaria.
25. Índices invertidos y búsqueda en texto

Otros modelos de datos

25. Tipos de bases de datos noSQL.
26. Paradigma JSON y MongoDB.

Bases de datos en la industria

28. Análisis de datos con Pandas.
29. Introducción a Data engineering.
30. Privacidad en las bases de datos.

revelará a medida que avanza el curso. Cada clase se estructura de la siguiente manera:

- **Lunes:** se sube el material complementario a la página.
- **Miércoles:** un módulo de clase expositiva y un módulo de taller para trabajar en actividad, tarea, evaluación o proyecto.
- **Viernes:** se entrega la actividad en caso de que sea evaluada.

Calendario de semanas (puede sufrir cambios, se avisará con al menos una semana de anticipación).

Numero	Semana	Actividad
1	Introducción y motivación	-
2	Modelo Relacional y álgebra relacional	Guía de ejercicios
3	SQL 1	Control evaluado 1
4	SQL 2	Guía de ejercicios
5	Diseño de BD	Proyecto
6	Dependencias	Guía de ejercicios
7	Almacenamiento e índices	Guía de ejercicios
8	Evaluación de consultas	Control evaluado 2
9	Lógica en la BD	Proyecto
10	Programación con SQL	Guía de ejercicios
11	ACID	Control Bonus
12	SQL para data science 1	Guía de ejercicios
13	NOSQL y JSON/MongoDB	Guía de ejercicios
14	Privacidad	Control evaluado 3
15	Data Engineering	Proyecto

Evaluación

La evaluación se realizará en base a:

- 3 Controles evaluados y un control bonus, a realizarse principalmente en clases con plazo hasta el viernes de la misma semana.
- Examen, con el objetivo de evaluar conocimientos teóricos.
- Un proyecto semestral con 3 evaluaciones parciales. El objetivo del proyecto es que el alumno desarrolle una base de datos para una aplicación basada en un caso real. Las etapas avanzadas del proyecto introducirán al alumno otras problemáticas reales de las bases de datos.

La **nota de controles y exámenes** (NCE) corresponde al promedio de los controles y el examen. En otras palabras, $NCE = (C1 + C2 + C3 + Ex) / 4$. El control bonus puede reemplazar su peor control, pero no el examen.

Se podrán eximir del examen los alumnos que tengan un promedio entre los 3 controles $> 5,5$.

La **nota del proyecto** (NP) corresponde al promedio ponderado de todas las entregas del proyecto. La ponderación es: Entrega 1: 20 %, Entrega 2: 40 %, Entrega 3: 40 %.

Importante: La nota de las entregas del proyecto estarán sujetas a una evaluación de pares que puede alterar la nota del alumno. Si el profesor lo considera necesario, citará a los alumnos cuyas evaluaciones de pares indican un comportamiento irregular en las entregas.

Para aprobar el ramo, el alumno debe cumplir que NCE y NP sean $\geq 3,95$. En ese caso, la **nota final** se calcula como $NF = (0,5NCE + 0,5NP)$. En caso contrario, $NF = \min\{ NCE , NP \}$. La fecha y forma de entrega de las entregas del proyecto será informada a medida que avanza el curso. El plazo final siempre es a las 20:00 del día de entrega.

Bibliografía

- Ramakrishnan, Gehrke. Database Management Systems (3ed).
- Garcia-Molina, Ullman, Widom. Database Systems: The complete book (2ed).
- Date, C. J.. Introducción a los sistemas de bases de datos (7ed).
- Elmasri, Ramez. Fundamentos de sistemas de bases de datos (5ed).
- Abiteboul, Hull, Vianu. Foundations of databases. (sobretudo para álgebra relacional y optimización).
- Phillip Greenspun. SQL for Web Nerds. <http://philip.greenspun.com/sql/>

Otros

El Departamento de Ciencias de la Computación adopta una política de tolerancia-cero frente a copias o plagios. Se sugiere revisar las políticas y penalidades que el departamento establece ante estas acciones.

El curso tiene dos canales de comunicación oficiales: Las clases y los avisos publicados en el sistema Canvas. Se asume que que toda la información que es entregada por ambos canales llega a todos los alumnos.

Es responsabilidad del alumno ponerse al día con los contenidos en caso de inasistencia a clases.