



PROGRAMA DE ASIGNATURA

I. IDENTIFICACIÓN				
Carrera o programa: Ingeniería en Tecnologías de Información				
Unidad responsable: Escuela de Ingeniería				
Nombre de la asignatura: Bases de datos				
Código: ECIN-00418				
Semestre en la malla¹: 5				
Créditos SCT – Chile: 5				
Ciclo de Formación	Básico		Profesional	X
Tipo de Asignatura	Obligatoria	X	Electiva	
Clasificación de área de conocimiento²				
Área: Ingeniería y Tecnología			Subárea: Ingeniería Informática	
Requisitos:				
Prerrequisitos:			Requisitos para:	
<ul style="list-style-type: none">Estructuras de datos			<ul style="list-style-type: none">Desarrollo de Soluciones Web/MóvilPráctica Profesional	

II. ORGANIZACIÓN SEMESTRAL							
Horas Dedicación Semanal (Cronológicas)	Docencia Directa	4.5		Trabajo Autónomo	3,5	Total	8
Detalle Horas Directas	Cátedra	Ayudantía	Laboratorio	Taller	Terreno	Exp. Clínica	Supervisión
	3	1.5					

III. APORTE AL PERFIL DE EGRESO
La asignatura contribuye al dominio 1 del perfil de egreso, “Conocimiento científico y disciplinario”. Además, contribuye al dominio 2 “Habilidades y Actitudes Personales y Profesionales”. También contribuye al dominio 3 “Habilidades Interpersonales”. Por último, contribuye al dominio 4 “Habilidades para la Práctica de la Ingeniería”. Al finalizar la asignatura, los estudiantes serán capaces de modelar un esquema conceptual de base de datos para un problema y aplicar el diseño, implementación y programación de una base de datos relacional utilizando la tecnología de un sistema de gestión de bases de datos.

IV. COMPETENCIAS
La carrera declara las siguientes habilidades: 1.3. Aplicar conocimientos, métodos y herramientas con un enfoque sistémico principalmente en la ejecución de proyectos TI en el ámbito de las ciencias de la computación, infraestructura TI e ingeniería de software. 2.1. Identificar y resolver problemas con un razonamiento analítico. 3.1. Liderar y trabajar en equipos multidisciplinarios y multiculturales.

¹ Este campo sólo se completa en caso de carreras con programas semestrales.

² Clasificación del curso de acuerdo a la OCDE



- 3.2. Comunicar comprensivamente información técnica en español, en forma oral, escrita, y gráfica, a nivel avanzado.
- 4.4. Diseñar soluciones que involucren, por ejemplo, aplicaciones TI, infraestructura TI, toma de decisiones, gestión de datos y gestión de proyectos.
- 4.5. Implementar soluciones que involucren, por ejemplo, aplicaciones TI, infraestructura TI, toma de decisiones, gestión de datos y gestión de proyectos.

V. RESULTADOS DE APRENDIZAJE

1. Identificar los conceptos básicos asociados a las bases de datos.
2. Diseñar bases de datos a través del modelo entidad-relación y su posterior transformación al modelo relacional.
3. Construir consultas a una base de datos relacional utilizando Lenguajes Relacionales Formales y SQL.
4. Crear bloques, subprogramas, subprogramas almacenados, paquetes y triggers en una base de datos relacional utilizando el lenguaje SQL y PL/SQL.
5. Realizar el rediseño de una base de datos relacional utilizando la teoría de normalización.
6. Administrar un sistema de gestión de bases de datos relacionales.
7. Analizar las relaciones de causa y efecto de los procesos en estudio.
8. Realizar el modelado del diseño de la solución.
9. Desarrollar la solución tecnológica más adecuada basada en las características del problema y los recursos disponibles.

VI. ÁREAS TEMÁTICAS

1. Conceptos de sistemas de bases de datos
 - 1.1 BD, SABD, esquema, instancia
 - 1.2 Bases de datos versus sistema de archivos
 - 1.3 Funciones críticas de un SABD
 - 1.4 Arquitectura tres niveles de un SABD.
 - 1.5 Componentes de un SABD
2. Diseño conceptual de bases de datos utilizando el modelo entidad-relacionamiento (E-R)
 - 2.1 Fases en el diseño de una base de datos y su relación con el desarrollo de aplicaciones
 - 2.2 Modelo Entidad-Relacionamiento (MER)
 - 2.3 Documentación del MER
3. Modelo y lenguajes relacionales
 - 3.1 Modelo Relacional
 - 3.2 Diseño de bases de datos relacionales utilizando una transformación del MER
 - 3.3 Documentación modelo relacional
 - 3.4 Álgebra relacional
 - 3.5 Cálculo Relacional de tuplas
4. SQL: Un lenguaje de bases de datos relacional
 - 4.1 Select
 - 4.2 Funciones: aritméticas, de conjuntos, de caracteres, relacionadas con fechas
 - 4.3 Operadores unión, intersect y minus



- 4.4 Joins y outerjoin
 - 4.5 Subconsultas
 - 4.6 Exists, el cuantificador existencial de SQL
 - 4.7 Inserción, eliminación y actualización de datos
 - 4.8 Transacciones: Conceptos, Commit y Rollback
 - 4.9 Creación, modificación y eliminación de tablas
 - 4.10 Vistas y vistas en línea.
 - 4.11 Creación de índices y secuencias
 - 4.12 Diccionario de datos
 - 4.13 Control discrecional de acceso a la base de datos basado en privilegios
 - 4.14 Concurrencia en base de datos.
 - 4.15 Indexación.
5. Lenguaje de programación para bases de datos relacionales PL
- 5.1 Características y funcionamiento
 - 5.2 Bloque
 - 5.3 Declaración de tipos de datos y variables
 - 5.4 Asignación y conversión de tipos
 - 5.5 If y ciclos
 - 5.6 Sentencias SQL dentro del lenguaje de programación
 - 5.7 Cursores
 - 5.8 Manejo de excepciones
 - 5.9 subprogramas locales
 - 5.10 Subprogramas almacenados
 - 5.11 Triggers
6. Diseño de bases de datos relacionales utilizando la teoría de la normalización
- 6.1 Evaluación del diseño de una base de datos relacional
 - 6.2 Dependencias funcionales
 - 6.3 Normalización de relaciones
 - 6.4 Formas normales (1FN, 2FN, 3FN, BCNF)
7. Administración de sistemas de gestión de bases de datos relacionales
- 7.1 Funciones del Administrador de la Base de Datos
 - 7.2 Creación de bases de datos y tablas
 - 7.3 Especificación del restricciones de integridad de los datos
 - 7.4 Administración de la concurrencia
 - 7.5 Optimización del acceso de datos
 - 7.6 Respaldo y recuperación
 - 7.7 Autorización para el acceso a los datos

VII. ORIENTACIONES METODOLÓGICAS

1. La metodología a desarrollar en esta asignatura es principalmente práctica, por lo que se debe fomentar la interacción entre los estudiantes a través de trabajos prácticos colaborativos.



2. Se sugiere abordar la teoría mediante metodologías activas como la clase invertida, el aprendizaje basado en problemas (ABP), entre otras, creando oportunidades para presentaciones orales individuales y/o grupales, lo que favorece el aprendizaje contextualizado.
3. Las experiencias en la cátedra y el taller deben llevarse a cabo utilizando software moderno aplicable a la asignatura.
4. Se recomienda que los estudiantes realicen presentaciones periódicas sobre el trabajo realizado en el taller.

VIII. ORIENTACIONES Y CRITERIOS PARA EVALUACIÓN

1. Se recomienda la aplicación de una evaluación diagnóstica al inicio de la asignatura.
2. En la cátedra, se sugiere implementar estrategias de evaluación sumativa, con un peso del 70% en la nota final de la asignatura.
3. En el taller, se sugiere implementar estrategias de evaluación sumativa, con un peso del 30% en la nota final de la asignatura.
4. Las actividades pueden ser individuales o grupales.
5. La asistencia mínima requerida para las actividades de cátedra es del 70%.
6. Se requerirá un logro del 60% de los objetivos para aprobar las actividades de evaluación.
7. Se recomienda llevar a cabo evaluaciones de carácter formativo con retroalimentación de carácter personal.

IX. RECURSOS BIBLIOGRÁFICOS

Bibliografía Mínima

– Elmasri, R., Navathe, S. B. (2007). Fundamentos de sistemas de bases de datos, 5a ed. España: Pearson Educación.

Bibliografía Complementaria

– Teorey, T. J., Lightstone, S. S., Nadeau, T., Jagadish, H. (2011). Database Modeling and Design: Logical Design. Países Bajos: Elsevier Science.

– Date, C. J., Ruiz Faudón, S. L. M. (2001). Introducción a los sistemas de bases de datos. México: Pearson Educación.