

<b>NOMBRE DE LA INSTITUCIÓN</b>	Universidad de Sonora
<b>DIVISIÓN ACADÉMICA</b>	División Ciencias Exactas y Naturales
<b>DEPARTAMENTO QUE IMPARTE LA MATERIA</b>	Departamento de Matemáticas
<b>LICENCIATURAS USUARIAS</b>	Ciencias de la Computación
<b>NOMBRE DE LA MATERIA</b>	<b>Bases de Datos II</b>
<b>CLAVE</b>	<b>9480</b>
<b>EJE FORMATIVO</b>	Especializante
<b>REQUISITOS</b>	Bases de Datos I
<b>CARÁCTER</b>	Optativa
<b>VALOR EN CRÉDITOS</b>	8 (3 hrs. teoría/2 hrs. laboratorio)

### Introducción

La sociedad moderna cada día depende más y más de sistemas informáticos eficientes y confiables. Las demandas por servicios de procesamiento de información que se adecuen a las necesidades de los usuarios de manera fiable, y que además se libere el software en corto tiempo y su desarrollo sea de bajo costo tienen un impacto profundo en las estrategias y procesos de desarrollo de software.

El desarrollo de sistemas de información de alta calidad requiere de recursos humanos que tengan conocimientos profundos de los distintos paradigmas de construcción de software así como de las diversas técnicas que se pueden aplicar para garantizar la calidad de los distintos productos generados y de los procesos utilizados. Además, requiere conocer y desarrollar las habilidades de comunicación y coordinación que le permitan integrarse y colaborar eficazmente en un equipo de desarrolladores de software y debe conocer los distintos componentes reutilizables del software de sistemas que pueden facilitar el desarrollo de aplicaciones de bases de datos.

### Objetivo General del Curso

El alumno será capaz de comprender los principios que permiten la utilización de sistemas administradores de bases de datos en la elaboración de sistemas de información. Además, será capaz de utilizar el lenguaje estándar de comunicación que se puede utilizar en sistemas de bases de datos relacionales para crear consultas especializadas, de acuerdo al proyecto de desarrollo de sistemas de software que realice.

### Objetivos Específicos del Curso

- Conocer los aspectos básicos del estándar SQL como sistema manejador de bases de datos.
- Conocer algún manejador de bases de datos para red en SQL.
- Desarrollar sistemas de bases de datos en red.
- Establecer con claridad el ámbito del software en el proyecto.
- Establecer y aplicar los lineamientos de aseguramiento de la calidad a cada uno de los productos generados durante el proceso de desarrollo del software.
- Aplicar las técnicas más adecuadas al proyecto para la construcción técnica de cada uno de los artefactos generados de acuerdo al proceso de desarrollo de software seleccionado.
- Verificar y validar la calidad del producto entregado al cliente.
- Organizar y distribuir las tareas entre los integrantes del equipo de desarrolladores.
- Establecer criterios específicos para evaluar el avance el proyecto y favorecer la comunicación y coordinación entre todos los miembros del equipo.

### Contenido

#### 1. Conceptos básicos de SQL.

- 1.1. Características generales de SQL.
- 1.2. Bases de datos relacionales.
- 1.3. Principales comandos de SQL.

#### 2. Manejo de una base de datos.

- 2.1. Recuperación de datos: Consulta simple y consulta multitabla.
- 2.2. Subconsultas.

- 2.3. Actualización de bases de datos.
- 2.4. Integridad de datos.
- 2.5. Procesamiento de una transacción.
- 3. Estructura de una base de datos.**
  - 3.1. Diseño de una base de datos.
  - 3.2. Vistas.
  - 3.3. Seguridad SQL.
  - 3.4. Catálogo del sistema.
- 4. Programación con SQL.**
  - 4.1. SQL incorporado.
  - 4.2. SQL dinámico
  - 4.3. API de SQL.
- 5. Conceptos básicos de un manejador de bases de datos.**
  - 5.1. Variables
  - 5.2. Operadores en variables.
  - 5.3. Instrucciones de control
  - 5.4. Funciones.
  - 5.5. Manejo de pantalla.
  - 5.6. Manejo de archivos.
- 6. Operaciones en Bases de Datos.**
  - 6.1. Apertura y cierre.
  - 6.2. Ordenamiento.
  - 6.3. Actualización de información.
  - 6.4. Programación de relaciones y filtros.
- 7. Operaciones en bases de datos en red.**
  - 7.1. Generalidades sobre redes.
  - 7.2. Compartición y bloqueos.
  - 7.3. Control en red.

### **Estrategias Didácticas**

- Propiciar la vinculación del alumno con el sector productivo al solicitarle que construya un sistema de software que sea utilizado por un cliente real.
- Promover la colaboración de los alumnos al permitirle la elección de distintas responsabilidades de trabajo, de acuerdo a las diversas tareas involucradas en el proyecto.
- Promover la participación oral y escrita de todos los alumnos al discutir el ámbito de los proyectos que aborden y especificar los requisitos de usuario en un documento.
- Implementar diversas técnicas de revisión, formales e informales, para evaluar el avance del proyecto y la calidad de los productos generados.
- Promover la investigación bibliográfica y en línea de información necesaria para comprender el dominio del problema abordado en el proyecto por el alumno así como de aspectos técnicos relacionado con la construcción del sistema de software a entregar.

### **Estrategias de Evaluación**

Para la evaluación de los estudiantes, el instructor tomará en cuenta:

- Resultados de los exámenes parciales aplicados (de acuerdo a la teoría abordada, el maestro decidirá la cantidad de éstos).
- La calidad de los productos generados en el proceso de desarrollo de software y que se entreguen de acuerdo al plan de desarrollo establecido.
- participación individual y colectiva en las revisiones.
- Proyecto completo terminado y aprobado por el cliente.
- Los porcentajes de cada uno de los indicadores serán acordados al inicio del semestre.

## **Bibliografía**

La bibliografía básica del área de ingeniería del software es la siguiente:

- Silberschatz, Abraham, Korth, H. y Sudarshan, S. Fundamentos de Bases de Datos. 4ta. Edición. McGraw-Hill. Madrid, 2002.
- Sommerville, Ian. Ingeniería de software. 6ta. ed. Pearson Educación, México, 2002.
- Pressman, Roger. Ingeniería del software: Un enfoque práctico. 5ta. ed. McGraw Hill, Madrid, 2002.

## **Perfil académico deseable del maestro**

Se recomienda que el profesor tenga las siguientes características:

- Formación sólida en ciencias de la computación o área afín y se haya especializado en ingeniería de software.