

Bases de Datos 3

Docentes: Federico Gómez, Diego Siri

Carreras: Licenciatura & Ingeniería en Informática

Año de la carrera: 3º

Capítulo 0: Introducción

Objetivos del curso:

- Presentar mecanismos para acceder a Bases de Datos e interactuar con DBMS desde lenguajes de programación.
- Estudiar criterios para un correcto desarrollo de aplicaciones que utilizan Bases de Datos como mecanismo de persistencia.
- Programar aplicaciones de pequeño y mediano porte utilizando los mecanismos y criterios anteriores.
- Definir estrategias para migrar una aplicación de un DBMS a otro minimizando la cantidad de código fuente a modificar en la aplicación.
- Introducir frameworks de persistencia comerciales que faciliten la interacción de la aplicación con el DBMS utilizado para persistir los datos manejados por ella.

Características del curso:

- **Tercer** curso de Bases de Datos.
- A diferencia de los dos cursos anteriores, este curso se concentra en el desarrollo de aplicaciones que interactúan con Bases de Datos más que en el estudio concreto de las mismas.
- El curso de **Bases de Datos 1** ofrece un marco teórico y conceptual sobre las bases de datos relacionales (**MER, álgebra relacional, SQL básico, diseño relacional de BD**, etc.).
- El curso de **Bases de Datos 2** ofrece un marco práctico y detallado de trabajo concreto con bases de datos (**trabajo en máquina con un DBMS, creación de esquemas, seguridad, SQL avanzado, vistas, índices, stored procedures**, etc.).

Características del curso (continuación):

- El curso de **Bases de Datos 3** ofrece un marco teórico-práctico para el desarrollo de aplicaciones que utilizan Bases de Datos relacionales como mecanismo de persistencia de información.
- Se utilizarán conceptos aprendidos en los dos cursos anteriores para creación y manipulación de las Bases de Datos que almacenan la información usada por la aplicación a desarrollar.
- Se verán conceptos nuevos que permitirán desarrollar dichas aplicaciones de modo correcto y eficiente, aprovechando las ventajas que ofrecen los DBMS para la gestión de información.
- Algunos de los conceptos que se verán en este curso incluso serán de utilidad para aplicaciones que interactúen con otros mecanismos de persistencia, no sólo con Bases de Datos.

Temario del curso:

1. Arquitecturas.
 - Concepto de arquitectura de una aplicación.
 - Arquitectura física de una aplicación.
 - Arquitectura lógica de una aplicación.
 - Relación entre arquitectura y persistencia (Bases de Datos).
2. Acceso a Bases de Datos desde lenguajes de programación.
 - Interacción entre DBMS y lenguajes de programación.
 - Acceso a DBMS desde Java (JDBC).
 - Componentes de JDBC.
 - Primeros ejemplos de acceso a BD usando JDBC.
3. Acceso a Bases de Datos en arquitecturas de 1 capa.
 - Características, pros y contras de esta solución.
 - Ejemplos de aplicaciones que utilizan esta solución.

Temario del curso (continuación):

4. Acceso a Bases de Datos en arquitecturas de 2 capas.
 - Características, pros y contras de esta solución.
 - Patrones de arquitectura en 2 capas:
 - Facade, Value Object, MVC.
 - Ejemplos de aplicaciones que utilizan esta solución.
5. Acceso a Bases de Datos en arquitecturas de 3 capas.
 - Características, pros y contras de esta solución.
 - Mapeo OO – BD en sistemas de arquitectura en 3 capas.
 - Patrones de arquitectura en 3 capas: DAO, Abstract Factory.
 - Combinación de patrones para migración a otro DBMS.
 - Ejemplos de aplicaciones que utilizan esta solución.
6. Introducción a los Frameworks de Persistencia.
 - Qué son y para qué sirven, pros y contras de su utilización.
 - Interacción con DBMS usando frameworks de persistencia.
 - Ejemplos de frameworks comerciales (Hibernate, etc.).

Software utilizado:

- Java 1.7 o superior como lenguaje de programación.
- Eclipse Kepler o superior como entorno de desarrollo.
- MySQL 5.1 y SQLServer 2000 o superior como DBMS a utilizar.

Bibliografía:

- Fundamentals of Database Systems [Elmasri, Navathe]
Addison-Wesley.
- Thinking in Java [Eckel] Prentice Hall.
- Design Patterns. Elements of Reusable OO Software [Gamma]
Addison-Wesley

Régimen:

- Dos veces por semana (Martes y Jueves)

Evaluación:

- Dos pruebas escritas individuales (75% del puntaje final)
 - Primera prueba, mínimo 40 puntos en 100.
 - Segunda prueba, mínimo 60 puntos en 100.
 - El promedio de ambas pruebas debe superar los 60 pts.
- Trabajo obligatorio grupal (25% del puntaje final)
 - El trabajo obligatorio tiene como mínimo 60 puntos en 100.
 - Se desarrollará en la segunda mitad del curso.
- Cumplidos los mínimos anteriores, el estudiante exonera la asignatura, en otro caso deberá rendir un examen en los períodos habituales, siempre que llegue al 60% de asistencia.