

## **PROGRAMA de Estrategias de Persistencia**

**Carreras:** Tecnicatura Universitaria en Programación Informática - Licenciatura en Informática

**Asignatura:** Estrategias de Persistencia

**Núcleo al que pertenece:** Básico

**Profesores:** Cristian López y Juan Pablo Delpino

**Asignaturas Correlativas:** Bases de Datos - Programación con Objetos 2.

**Objetivos:**

Que el estudiantado:

- Conozca distintos mecanismos de persistencia, en particular persistencia en archivos y persistencia en bases de datos relacionales u orientadas a objetos; haga experiencia práctica con ellos y tenga elementos para realizar comparaciones.
- Entienda las ideas básicas para poder interactuar con un mecanismo de persistencia desde un programa externo, las problemáticas asociadas de transformación entre estructuras de persistencia y estructuras del programa, y la pertinencia de acotar el ámbito en el cual el mecanismo de persistencia toma preeminencia sobre las abstracciones encontradas en el programa.
- Comprenda la problemática específica de la persistencia de objetos y de su implementación en distintos mecanismos de persistencia, en particular la impedancia objetos/relacional.
- Entienda los problemas de concurrencia específicos que derivan del acceso a un mecanismo de persistencia, y tenga experiencia práctica en algunas técnicas para tratarlos, analizando tanto los mecanismos de concurrencia como su inserción en programas externos.
- Conozca algunas técnicas que permitan trabajar sobre la performance de mecanismos de concurrencia y entienda en qué casos es adecuado evaluarlas y aplicarlas, analizando tanto los mecanismos de concurrencia como su inserción en programas externos. En particular, en relación con bases de mediano o gran volumen de información.
- Entienda y pueda manejar en la práctica conceptos relacionados con la seguridad en el acceso a un mecanismo de persistencia.

### Contenidos mínimos:

- Nociones sobre los problemas que derivan del acceso concurrente a una base de datos. Algunas estrategias para mitigarlos, en particular lockeo y manejo adecuado de transacciones.
- Nociones sobre la problemática de performance en el acceso a una base de datos, relación con la escala, otros factores que influyen. Estrategias de acceso a los datos ante una consulta, concepto de índice.
- Conceptos de usuario y permiso en una base de datos, esquemas típicos de definición de usuarios y permisos.
- Bases de objetos: concepto, panorama, experimentación práctica, comparación con bases de datos relacionales.
- Bases de datos distribuidas para grandes volúmenes de datos, acceso a datos como un servicio, herramientas de programación asociadas.
- Interacción entre un programa y un mecanismo de persistencia: nociones básicas, problemáticas generales.
- Mecanismos de acceso y recuperación de objetos persistidos en bases de datos relacionales: mecanismos de recuperación de objetos (uso de lenguajes de consulta relacionales, lenguajes de consulta orientados a objetos, interfaz en objetos orientada al acceso, interfaces en términos del modelo de dominio). Actualización del estado persistente: reachability, cascada.
- ORM, conceptos básicos, alcances, cuestiones que resuelven, enfoque que toma respecto de la transformación de objetos. Problemas de mapeo: herencia, relaciones n-m, estrategias no standard.
- Transacciones a nivel aplicación, transacciones de negocio, reflejo de la transaccionalidad al acceder a un mecanismo de persistencia, concepto de unit of work.
- Reflejo de cuestiones de performance y concurrencia al acceder a un mecanismo de persistencia desde un programa, lazyness, cache, versionado, lockeo optimista y pesimista.

**Carga horaria semanal:** 6 hs

**Programa analítico:**

### **Unidad 1: Modelo Relacional**

Repaso Modelo Relacional, objetos de la base de datos. Transacciones / Herramientas de integridad. Repaso sentencias SQL. Manejo de Conexiones. Tipos de datos. Consultas, Prepared Statements, Resultsets. Seguridad / Manejo de Usuarios.

### **Unidad 2: Mapeo Objetos Relacional**

Concepto de mapeo objetos / relacional. Soluciones Ad Hoc vs Frameworks. Diferencias entre modelo de objetos y modelo relacional. Soluciones posibles. Mapeos. Relaciones entre entidades. Modelos de Herencia. Mapeos Avanzados. Relaciones con patrones de diseño. Consultas declarativas. Manejo de Sesiones. Concurrencia / Lockeos / Transacciones.

### **Unidad 3: Performance en Modelos Relacionales**

Problemas de performance. Optimización temprana vs Optimización bajo demanda. Fuentes de información. Diagnostico y reconocimiento de problemas. Análisis por casos de uso. Query Plan / Indices / Vistas Materializadas. Desnormalización. Optimización de consultas en ORM. Modelo Lazy vs Eager. Configuración general y por consultas. Análisis de casos de uso general y particulares. Decisiones por peso del caso de uso (frecuencia, valor de negocio, etc).

### **Unidad 4: Modelos alternativos al relacional.**

Características del modelo relacional como medio para comparación con otros modelos. Base de datos Objetos: Conceptos, Consultas, consecuencias de integración con programas. Base de datos orientada a Grafos: Conceptos, Consultas, consecuencias de integración con programas. Prevalencia: Conceptos, Consultas, consecuencias de integración con programas. Bases de datos distribuidas. Datos particionables. Transacciones distribuidas. Arquitecturas de distribución. Arquitecturas redundantes. Diseño guiado por casos de uso. Modelos de consistencia y transaccionalidad. Introducción a Bases de datos No-SQL.

### **Unidad 5: Base de datos Clave Valor.**

Conceptos. Modelo de datos. Consecuencias de la Estructura. Accesos posibles. Consultas Posibles. Algoritmo de Map / Reduce. Problemas. Ventajas y Desventajas. Casos de uso recomendados.

### **Unidad 6: Base de datos Multi Columnas.**

Conceptos. Modelo de datos. Consecuencias de la Estructura. Accesos posibles. Consultas Posibles. Problemas. Ventajas y Desventajas. Casos de uso recomendados.

### **Unidad 7: Base de datos orientadas a Documentos.**

Conceptos. Modelo de datos. Consecuencias de la Estructura. Accesos posibles. Consultas Posibles. Problemas. Ventajas y Desventajas. Casos de uso recomendados.

#### **Bibliografía obligatoria:**

- Erich Gamma, Richard Helm, Ralph Johnson y John M. Vlissides, Design Patterns: Elements of Reusable Object-Oriented Software (en inglés o traducción al español), Addison-Wesley Professional, Computing Series, 1995.
- Jim R. Wilson, Eric Redmond, Seven Databases in Seven Weeks: A Guide to Modern Databases and the NoSQL Movement, Pragmatic Bookshelf; first edition. 2012.
- Bauer, King. Hibernate in Action. Manning Publications. 2004.
- Sitio de la materia <https://sites.google.com/site/estrategiasdepersistencia/> con material desarrollado por la cátedra.

#### **Bibliografía de consulta:**

- Baron Schwartz, Peter Zaitsev, Vadim Tkachenko, High Performance MySQL: Optimization, Backups, and Replication, O'Reilly Media; Third Edition. 2012.
- Christopher Date, Database Design and Relational Theory: Normal Forms and All That Jazz (Theory in Practice). O'Reilly Media. 2012.
- The Language of SQL: How to Access Data in Relational Databases. Rockoff. Cengage Learning PTR. 2010.
- Pramod J. Sadalage, Martin Fowler. NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence. Addison-Wesley Professional. 2012.

#### **Organización de las clases:**

Las clases están organizadas en dos partes, teóricas y prácticas. Durante las teóricas se presenta el contenido formal del próximo tema práctico a realizar. Durante las clases prácticas se resuelven los distintos trabajos prácticos obligatorios de la materia.

#### **Los trabajos prácticos a desarrollar son:**

##### ***Trabajo Práctico 1. JDBC:***

Los objetivos de este trabajo son: (1) repasar el modelo relacional, (2) persistir de un modelo de objetos simple en una base relacional con JDBC.

##### ***Trabajo Práctico 2. Hibernate:***

Los objetivos de este trabajo son; (1) realizar un modelo de objetos más complejo, complementando el TP 1 y utilizar un ORM (Object Relational Mapper) como estrategia de persistencia, (2) introducir el concepto de transacción .

**Trabajo Práctico 3. Hibernate Performance:**

Los objetivos de este trabajo son: (1) familiarizarse con el concepto de índices, (2) realizar análisis, diagnósticos y optimizaciones sobre una base de datos.

**Trabajo Práctico 4. Base de datos de Grafos:**

Los objetivos de este trabajo son: (1) introducir modelos alternativos al relacional, (2) agregar un nuevo módulo al TP utilizando una base de datos de grafos como mecanismo de persistencia, (3) introducir conceptos básicos sobre teoría de grafos y algunos algoritmos conocidos como *shortest path*, (4) implementar transacciones distribuidas.

**Trabajo Práctico 5. Base de datos Documentales:**

Los objetivos de este trabajo son: (1) introducir el modelo Documental, (2) agregar módulo al TP para ser implementado utilizando una base de datos Documental, (3) crear un cluster con redundancia y particionado para la base de datos, (4) introducir el concepto de Map-Reduce.

**Trabajo Práctico 6. Investigación:**

Los objetivos de este trabajo son: (1) investigar y desarrollar alguna tecnología no vista en clase, (2) hacer una exposición explicando el tema elegido y mostrar una demo de integración con dicha tecnología, (3) comparar los distintos modelos de persistencia utilizados.

**Modalidad de evaluación:**

Los mecanismos de evaluación en modalidades libre y presencial de esta asignatura están reglamentados según los siguientes artículos del Régimen de estudios de la UNQ (Res. CS 201/18).

En la modalidad de libre, se evaluarán los contenidos de la asignatura con un examen escrito, un examen oral e instancias de evaluación similares a las realizadas en la modalidad presencial.

### CRONOGRAMA TENTATIVO

Semana	Tema/unidad	Actividad*				Evaluación
		Teórico	Práctico			
			Res Prob.	Lab.	Otros Especificar	
1	Conceptos preliminares / Inicio TP #1	x	x	x		
2	Práctica- Resolución del TP #1		x	x		
3	Hibernate - Manejo de Session / Entrega TP #1 / Inicio TP #2	x	x	x		x
4	Hibernate - Relaciones - Herencia	x	x	x		
5	Práctica- Resolución del TP #2		x	x		
6	Hibernate – Performance / Entrega TP #2 / Inicio TP #3	x	x	x		x
7	Práctica- Resolución del TP #3		x	x		
8	Intro NO SQL - Neo4j / Entrega TP #3 / Inicio TP #4	x	x	x		x
9	Práctica- Resolución del TP #4		x	x		
10	MongoDB – MapReduce / Entrega TP #4 / Inicio TP #5	x	x	x		x
11	Práctica- Resolución del TP #5		x	x		
12	Cassandra / Entrega TP #5 / Inicio TP #6	x	x	x		x
13	Práctica- Resolución del TP #6					
14	Defensa TPs / Entrega TP #6	x	x	x		x
15	Defensa TPs		x	x		x
16	Consultas		x	x		
17	Recuperatorios de TPs					x
18	Integrador					

\*INDIQUE CON UNA CRUZ LA MODALIDAD