



000027

Estrategias de persistencia

INSTITUTOS/S: INSTITUTO DE TECNOLOGÍA E INGENIERÍA

CARRERA/S: Licenciatura en Informática.

RESPONSABLE DE LA ASIGNATURA Y EQUIPO DOCENTE:
Gerardo Gonzalez Tulián

AÑO: 2025

CRÉDITOS: 7

CARGA HORARIA DE INTERACCIÓN PEDAGÓGICA: 64

CARGA HORARIA TOTAL: 175

CÓDIGO DE LA MATERIA EN SIU: 759

000027

1. Fundamentación

Se entiende como persistencia a la acción de preservar la información de un objeto de forma permanente (guardado), pero a su vez también se refiere a poder recuperar la información de este (leerlo) para que pueda ser nuevamente utilizado.

Esta materia plantea abordar las diferentes alternativas de bases de datos y la correcta implementación de estas a la hora de definir la arquitectura de un sistema de software. (Bases de datos relacionales, Bases de datos no relacionales orientadas a grafos, Bases de datos no relacionales orientadas a documentos, Bases de datos no relacionales orientadas a objetos, Bases de datos distribuidas, Bases de datos centralizadas).

Como aporte fundamental, la materia otorga conocimientos generales de diferentes tipos de base de datos como así también el conocimiento en tecnología Nodejs y ORM Sequelize, contenido con el cual se intenta abordar las diferentes situaciones suscitadas.

2. Propósitos y/u objetivos

Que el alumno pueda:

- *Experimentar distintos mecanismos de persistencia, en particular persistencia en archivos, bases de datos relacionales u orientadas a objetos.*
- *Comprender la mecánica y la problemática de la persistencia de objetos y de su implementación.*
- *Conocer algunas técnicas de trabajo de la performance y entender en qué casos es adecuado evaluarlas y aplicarlas.*
- *Explorar casos de usos y posibles soluciones.*
- *Comprender la importancia de la seguridad en el acceso a un mecanismo de persistencia y experimentar este concepto.*
- *Conocer diferentes modelos alternativos al modelo relacional o de objetos.*

000027


3. Contenidos mínimos

Problemas vinculados al acceso concurrente a una base de datos (BD). Performance en el acceso a una BD. Modelado y calidad de datos. Concepto de índice. Conceptos de usuario y permiso en una base de datos. Bases de objetos: concepto, panorama, experimentación práctica, comparación con bases de datos relacionales. BD distribuidas para grandes volúmenes de datos, acceso a datos. Transacciones distribuidas. 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. Actualización del estado persistente: reachability, cascada. ORM y problemas de mapeo: herencia, relaciones n-m, estrategias no estándar. Transacciones a nivel aplicación y de negocio, concepto de unit of work. Cuestiones de performance y concurrencia al acceder a un mecanismo de persistencia desde un programa, lazyness, cache, versionado, lockeo optimista y pesimista.

4. Carga horaria

Créditos	Interacción pedagógica		Trabajo Autónomo	Total
7	Teoría: 30	Práctica: 34	111	175

4.1 Trabajo autónomo del/la estudiante

Actividad	Carga horaria
Lectura de contenidos teóricos apuntes o bibliografía.	25
Desarrollo de los ejercicios prácticos propuestos.	36
Investigar los diferentes enfoques	20
Estudio para los parciales	30

5. Programa analítico

UNIDAD 1: Modelo Relacional.

Repaso Modelo Relacional. Transacciones / Herramientas de integridad. Repaso SQL. Manejo de Conexiones. Tipos de datos. Prepared Statements, Resultsets. Gestión de Usuarios. Seguridad.

000027**UNIDAD 2: Mapeo Objeto Relacional.**

Mapeo objetos / relacional. 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 / Locks / Transacciones.

UNIDAD 3: Performance.

Problemas de performance. Optimización. Diagnóstico y reconocimiento de problemas. Análisis por casos de uso. Query Plan / Índices / Vistas Materializadas. Desnormalización. Optimización de consultas en ORM. Modelo Lazy vs Eager.

UNIDAD 4: Bases de Datos Documentales: Modelo, Consultas e Integración con Aplicaciones.

Almacenamiento de documentos estructurados en formatos JSON O BSON. Características, ventajas, desventajas y análisis de la flexibilidad en la gestión de datos frente a los modelos relacionales dependiendo del caso de uso. Lenguaje de consultas e integración con mediante drivers y frameworks en distintos lenguajes de programación.

UNIDAD 5: Almacenamiento Clave-Valor y su Uso en Caché: Integración con Aplicaciones.

*Bases de datos **clave-valor**, su modelo de datos y las formas de acceso y consulta. Se analizarán sus ventajas y desventajas en comparación con otros modelos. Caso de uso de aplicaciones que implementen caché en memoria para mejorar el rendimiento y la escalabilidad. Estrategias de **expiración de claves**, **políticas de reemplazo de caché** (LRU, LFU, FIFO)*

UNIDAD 6: Otros Modelos de Bases de Datos No Relacionales y Alternativas al Modelo Relacional.

*Otros sistemas de persistencia de datos **NoSQL** (Not Only SQL) diseñados para manejar grandes volúmenes de información de forma flexible. Bases de datos orientada a Grafos y Objetos. Conceptos, Consultas, e integración con programas. Prevalencia: Conceptos, Consultas, integración con programas. Bases de datos distribuidas. Datos particionables. Arquitecturas de distribución. Arquitecturas redundantes.*

000027

6. Ejes y enunciados Multidimensionales y transversales

<i>Eje</i>	<i>Nivel de logro de Aprendizaje</i>	<i>Acciones de enseñanza</i>
<i>C1. Identificación, formulación y resolución de problemas de informática</i>	<i>Alto</i>	<i>Presentación de estudios de caso donde se analicen problemas de persistencia y rendimiento en bases de datos. Resolución de ejercicios prácticos con problemas reales.</i> <i>Aprendizaje Basado en Problemas.</i>
<i>C2. Concepción, diseño y desarrollo de proyectos de informática</i>	<i>Alto</i>	<i>Aprendizaje orientado a Proyectos.</i> <i>Clases prácticas.</i>
<i>C3. Gestión, planificación, ejecución y control de proyectos de informática</i>	<i>Bajo</i>	<i>Método de Casos.</i> <i>Clases expositivas dialogadas.</i>
<i>C4. Utilización de técnicas y herramientas de aplicación en la informática</i>	<i>Alto</i>	<i>Clases prácticas con herramientas de bases de datos relacionales y NoSQL. Implementación de consultas optimizadas y análisis de performance.</i> <i>Ilustraciones.</i>
<i>C5. Generación de desarrollos tecnológicos y/o innovaciones tecnológicas</i>	<i>Alto</i>	<i>Aprendizaje orientado a Proyectos.</i> <i>Lluvia de ideas.</i>
<i>C6. Fundamentos para el desempeño en equipos de trabajo</i>	<i>Medio</i>	<i>Aprendizaje cooperativo.</i> <i>Tutoría.</i>
<i>C7. Fundamentos para la comunicación efectiva</i>	<i>No Aporta</i>	-
<i>C8. Fundamentos para la acción ética y responsable</i>	<i>No Aporta</i>	-
<i>C9. Fundamentos para evaluar y actuar en relación con el impacto social de su actividad en el contexto global y local</i>	<i>No Aporta</i>	-

000027



C10.Fundamentos para el aprendizaje continuo	Medio	Aula invertida. Tutoría.
C11.Fundamentos para la acción emprendedora	No Aporta	-

7. Bibliografía y recursos

7.1 Bibliografía obligatoria

Silberschatz, A., Korth, H. F., & Sudarshan, S. (2014). "Fundamentos de bases de datos" (6th ed.). McGraw-Hill/Interamericana de España, S.L.

Wilson, Redmon, Perkins (2018). "Seven Databases in Seven Weeks: A Guide to Modern Databases and the NoSQL Movement", 2da edición Pragmatic Bookshelf.

Antonio Sarasa (2019), "Introducción a las bases de datos NSQL clave-valor usando Redis", Editorial UOC, S.L.

Gonzalez Mallols J. (2023), "MongoDB. Modelado de datos (Bases de datos | MongoDB)". ISBN-13 979-8871333488

Gamma, E.; Helm, R.; Johnson, R.; Vlissides, J.M. (2003). "Patrones de Diseño: Elementos de Software orientado a objetos reutilizables". Editorial Addison Wesley.

7.2 Bibliografía optativa

García-Molina, H., Ullman, J. D., & Widom, J. (2008). "Sistemas de bases de datos: El libro completo" (2nd ed.). Pearson Educación. disponible en:
<https://github.com/varungupte/DATABASE/blob/master/Molina%2CUllman%20-%20Database%20Systems%20The%20Complete%20Book.pdf>

Justo Gonzalez Mallols, MongoDB. (2024) "Agregación de datos: Agrupación, transformación, resumen y análisis de datos" (Bases de datos | MongoDB) JSNADN y JSNSD.

Pragmatic Bookshelf. Bauer, K. (2004). "Hibernate in Action". Manning Publications.

000027

Date, C.J. (2012). "High Performance MySQL: Optimization, Backups, and Replication". Third Edition. O'Reilly Media.

Rockoff, L. (2010). "The Language of SQL: How to Access Data in Relational Databases." Delmar Cengage Learning. Sadalage, P. y Fowler, M. (2012).

Sadalage, P. J., & Fowler, M. (2012). "NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence". Addison-Wesley.

7.3 Recursos

Todo el material usado durante la cursada estará disponible en el campus virtual. Incluidas las grabaciones de las clases cuando la modalidad sea virtual.

Para tener una comunicación más fluida vamos a utilizar un canal de Discord de la materia dentro del servidor de la Universidad.

Se utilizarán diapositivas e ilustraciones para desarrollar los contenidos teóricos de cada unidad proyectando cuando el encuentro sea presencial o compartiendo pantalla cuando el encuentro sea virtual. Se focalizará el tiempo de clase en la comprensión conceptual y desarrollo de las capacidades de aprendizaje de la temática, guiando al desarrollo de habilidades prácticas, y fomentando la lectura domiciliaria compartiendo recursos adicionales de lectura y compartiendo las prácticas resueltas.

Recursos Prácticos y Herramientas

1. Entornos de Desarrollo y Bases de Datos:

- Visual Studio Code como IDE.
- Node.js: Entorno de ejecución para JavaScript utilizado en el curso.
- Sequelize: ORM para Node.js que se utilizará en las prácticas.
- Bases de Datos Relacionales: MySQL, PostgreSQL, o SQLite para prácticas con bases de datos relacionales.
- Bases de Datos No Relacionales: MongoDB (para bases de datos documentales) y Redis (para bases de datos clave-valor).
- Docker: Para crear contenedores con bases de datos y facilitar la configuración del entorno de desarrollo.

2. Herramientas de Visualización y Análisis:

000027

- *DBeaver: Herramienta gratuita para gestionar y visualizar bases de datos relacionales.*
 - *MongoDB Compass: Interfaz gráfica para MongoDB.*
 - *RedisInsight: Herramienta para visualizar y gestionar bases de datos Redis.*
 - *Neo4j Browser: Interfaz gráfica para bases de datos orientadas a grafos.*
3. *Repositorios de Código y Control de Versiones*
- *GitHub: Para compartir código, prácticas resueltas, y proyectos.*
4. *Aprendizaje Adicional / Recursos de Documentación Oficial:*
- *Documentación de Sequelize: <https://sequelize.org/>*
 - *Documentación de Node.js: <https://nodejs.org/es/docs/>*
 - *Documentación de MongoDB: <https://docs.mongodb.com/>*
 - *Documentación de Redis: <https://redis.io/documentation>*
 - *Documentación de Neo4j: <https://neo4j.com/docs/>*

8. Metodología de enseñanza

8.1 Modalidades u opciones pedagógicas

La asignatura está constituida por 6 unidades temáticas, las cuales se dictarán durante un cuatrimestre en clases teóricas y prácticas ,50% en formato presencial y 50% en formato virtual de clase sincrónica, con resolución de problemas a cargo de los/las alumnos/as supervisados por los/las docentes.

Las clases están organizadas para integrar los contenidos teóricos en la solución de problemas prácticos presentados a los/las estudiantes. El nivel de complejidad de los problemas aumenta en función a la presentación de los contenidos teóricos de cada unidad.

El objetivo de esta metodología busca que el saber de los conceptos introducidos en cada clase se refleje en el saber y hacer práctico de la solución de un problema planteado. Se presentan los conceptos teóricos, se complementa con videos teórico/práctico y lecturas adicionales recomendadas, y se verifica en el hacer práctico que se apliquen los conceptos presentados.

000027

Cada problema es presentado y discutido teóricamente, y luego es resuelto por los/las estudiantes directamente mediante la implementación de programas. Esta estrategia unifica los conceptos, el modelado y solución del problema, y la implementación como clase de laboratorio.

8.2 Formación Práctica

La organización propuesta para la cursada está fuertemente basada en la existencia de una serie de guías de ejercicios, que están graduados de forma de acompañar la evolución de los/las estudiantes y la secuencia pedagógica propuesta.

9. Evaluación y régimen de aprobación

9.1 Modalidad de evaluación

El sistema normal de evaluación consistirá en 2 Exámenes Parciales 100% prácticos con un recuperatorio por cada uno de ellos, según el cronograma previsto. Tanto en los parciales como en los recuperatorios, los contenidos a evaluar y los ejercicios a resolver representan la totalidad de la materia descrita en el programa. Los mismos se realizarán en las fechas que, a tal efecto, se establezcan en el cronograma correspondiente.

9.2 Aprobación de la cursada

Para aprobar la cursada y obtener la condición de regular, el régimen académico establece que debe obtenerse una nota no inferior a cuatro (4) puntos. Todas las instancias evaluativas deberán tener una instancia de recuperatorio. Podrán acceder a la administración de esta modalidad solo aquellos y aquellas estudiantes que hayan obtenido una nota inferior o igual a 6 (seis) puntos en el examen parcial.

Siempre que se realice una evaluación de carácter recuperatorio, la calificación que los/as estudiantes obtengan reemplazará la calificación obtenida en el examen que se ha recuperado y será la considerada definitiva a los efectos de la aprobación.

9.3 Acreditación de la materia

La materia puede aprobarse por promoción, evaluación integradora, examen final o libre.

000027

Promoción directa: tal como lo establece el art°17 del Régimen Académico, para acceder a esta modalidad, el/la estudiante deberá aprobar la cursada de la materia con una nota no inferior a siete (7) puntos, no obteniendo en ninguna de las instancias de evaluación parcial menos de seis (6) puntos, sean evaluaciones parciales o recuperatorios. El promedio estricto resultante deberá ser una nota igual o superior a siete (7) sin mediar ningún redondeo.

Evaluación integradora: tal como lo establece el art°18 del Régimen Académico, podrán acceder a esta evaluación aquellos estudiantes que hayan aprobado lo cursado con una nota de entre cuatro (4) y seis (6) puntos.

La evaluación integradora tendrá lugar por única vez en el primer llamado a exámenes finales posterior al término de la cursada. Deberá tener lugar en el mismo día y horario de la cursada y será administrada, preferentemente, por el/la docente a cargo de la comisión. Se aprobará tal instancia con una nota igual o superior a cuatro (4) puntos, significando la aprobación de la materia.

La nota obtenida se promediará con la nota de la cursada.

Examen final: Instancia destinada a quienes opten por no rendir la evaluación integradora o hayan regularizado la materia en cuatrimestres anteriores. Se evalúa la totalidad de los contenidos del programa de la materia y se aprueba con una calificación igual o superior a cuatro (4) puntos. Esta nota no se promedia con la cursada.