



Universidad Nacional de Córdoba

FACULTAD DE MATEMÁTICA ASTRONOMÍA Y FÍSICA

Expte. 03-07-02837

RESOLUCION HCD N° 94/07

VISTO

La solicitud del Dr. Juan Durán para que se incorpore la materia "Transformación de Modelos" como Optativa de la Licenciatura en Ciencias de la Computación; y

CONSIDERANDO

Que se cuenta con el acuerdo de la Comisión Asesora de Computación;

Que es conveniente agregar a la nómina de materias optativas, aprobada por Res. HCD 207/02, la asignatura que se propone;

Que mediante Resolución HCS n° 122/02 se ha delegado en este Cuerpo la facultad de modificar la nómina de materias optativas del Plan de Estudios de la Licenciatura en Ciencias de la Computación;

EL HONORABLE CONSEJO DIRECTIVO DE LA
FACULTAD DE MATEMÁTICA, ASTRONOMÍA Y FÍSICA

R E S U E L V E :

ARTÍCULO 1°: Hacer lugar a lo solicitado por el Dr. Juan Durán y, en consecuencia, modificar la nómina de materias optativas del Plan de Estudios de la Licenciatura en Ciencias de la Computación, incorporando a la misma la materia "Transformación de Modelos".

ARTÍCULO 2°: Fijar como programa, correlativas y carga horaria de la materia, los detallados en el Anexo que forma parte de la presente Resolución.

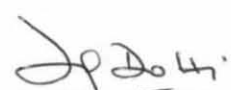
ARTÍCULO 3°: En cumplimiento con lo establecido en el artículo 2° de la Res. HCS n° 122/02, remítase a la Secretaría de Asuntos Académicos de la Universidad la presente resolución para su conocimiento y efectos.

ARTÍCULO 4°: Comuníquese y archívese.

DADA EN LA SALA DE SESIONES DEL HONORABLE CONSEJO DIRECTIVO DE LA FACULTAD DE MATEMÁTICA, ASTRONOMÍA Y FÍSICA, A ONCE DIAS DEL MES DE JUNIO DE DOS MIL SIETE.

npk


Dr. WALTER N. DAL LAGO
Secretario General Fa. M. A. F.


Dra. ISABEL DOTTI
VICE DECANO
Fa. M. A. F.



Expte. 03-07-02837

ANEXO A RESOLUCIÓN HCD N° 94/07

MATERIA OPTATIVA	CORRELATIVAS – PLAN 2002			CARGA HORARIA
	PARA CURSAR		PARA RENDIR	
	REGULARIZADA	APROBADA	APROBADA	
Transformación de Modelos	Ingeniería del Software I Base de Datos	Algoritmos y Estructuras de Datos II Matemática Discreta II	Ingeniería del Software I Base de Datos	120 hs.

MATERIA OPTATIVA	CORRELATIVAS – PLAN 1998			CARGA HORARIA
	PARA CURSAR		PARA RENDIR	
	REGULARIZADA	APROBADA	APROBADA	
Transformación de Modelos	Ingeniería del Software I Base de Datos	Algoritmos y Estructuras de Datos II	Ingeniería del Software I Base de Datos	120 hs.

Régimen de Cursado: Semestral.

Introducción

Las técnicas de desarrollo de software están evolucionando constantemente con la meta de resolver los problemas principales que todavía afectan a la construcción y mantenimiento de aplicaciones de software como son: tiempos, costos y propensión a errores. El desarrollo de software dirigido por modelos intenta reducir al menos algunos de esos problemas, enfocándose en la construcción de modelos, especificación de reglas de transformación, soporte de herramientas y generación automática de código y documentación.



La transformación de modelos juega un papel clave en el desarrollo de software dirigido por modelos. Sus aplicaciones incluyen las siguientes:

Generar modelos de más bajo nivel y eventualmente código, a partir de modelos de más alto nivel.

Mapeo y sincronización entre modelos en el mismo nivel o en diferentes niveles de abstracción.

Crear visiones basadas en consultas de un sistema.

Obtención de modelos de nivel más alto a partir de modelos de nivel más bajo o código (Ingeniería inversa).

Automatizar la parte laboriosa y pesada del desarrollo de software. Así se puede reducir el tiempo que se ocupa en tareas más o menos rutinarias.

Refactorización de código. Refactorización es la modificación del código fuente sin cambiar su comportamiento, lo que se conoce informalmente por limpiar el código. El objetivo es mejorar la facilidad de comprensión del código o cambiar su estructura o diseño y eliminar código muerto para facilitar el mantenimiento en el futuro.

Uno de los enfoques más importantes dentro de la transformación de modelos es MDA; que se basa en el lenguaje de modelado UML y en otros estándares. MDA promueve la creación de modelos muy abstractos que son desarrollados independientemente de la tecnología de implementación y son almacenados en repositorios estandarizados. Allí, esos modelos pueden ser accedidos repetidas veces y automáticamente transformados por herramientas en esquemas, esqueletos de código, código de integración y scripts de despliegue para varias plataformas. Otros dos enfoques muy importantes son los de transformaciones de grafos y el espacio tecnológico XML.

Programa

Capítulo 1. Introducción a la transformación de modelos. (2 clases)

Reglas de transformación. Control de aplicación de reglas. Organización de las Reglas. Incrementalidad. Direccionalidad. Traza. Transformaciones endógenas y transformaciones exógenas. Herramientas y lenguajes de transformaciones. Criterios de éxito para un lenguaje o herramienta de transformaciones. Requisitos de calidad para un lenguaje o herramienta de transformaciones. Clasificación de enfoques de transformación de modelos.

Capítulo 2. UML. (5 clases)

UML. Modelando estructuras de objetos. Modelando interacciones de objetos. Modelando comportamiento de objetos. Modelando la arquitectura de la aplicación. Arquitectura de UML. Metamodelo de UML. Mecanismos de Extensión. Perfiles UML.

Capítulo 3. OCL. (4 clases)



Universidad Nacional de Córdoba

FACULTAD DE MATEMÁTICA ASTRONOMÍA Y FÍSICA

3. Jos Warmer, Anneke Kleppe. *The Object Constraint Language: Getting Your Models Ready for MDA, Second Edition*. Addison-Wesley Professional; 2 edition (August 29, 2003)
4. Anneke Kleppe, Jos Warmer, Wim Bast. *MDA Explained: The Model Driven Architecture--Practice and Promise*. Addison-Wesley Pub Co; 1st edition (April 25, 2003)
5. Stephen J. Mellor, Kendall Scott, Axel Uhl, Dirk Weise. *MDA Distilled: Principles of Model-Driven Architecture*. Addison Wesley, 2004.
6. Gardner J. R., Rendon Z. L. *XSLT and XPATH: A Guide to XML Transformations*. Prentice Hall, 2001.
7. Michael Brundage. *XQuery: the XML Query Language*. Addison Wesley 2004.
8. Reiko Heckel. *Graph Transformation in a Nutshell*. *Electronic Notes in Theoretical Computer Science*. 148, pp 187-198. Elsevier 2006.
9. Andries M. et al. *Graph Transformation for Specification and Programming*. Technical Report 7/96, Universitat Bremen, 1996.
10. Luciano Baresi and Reiko Heckel. *Tutorial Introduction to Graph Transformation: A Software Engineering Perspective*. In *Proceedings of the First International Conference on Graph Transformation (ICGT 2002)*. October, 2002, Barcelona, Spain. *Lecture Notes in Computer Science*, volume 2505, pages 402-429.
11. Martin Fowler, Kent Beck, John Brant, William Opdyke, Don Roberts. *Refactoring: Improving the Design of Existing Code*. Addison Wesley 1999.
12. William C. Wake. *Refactoring Workbook*. Addison Wesley 2003.

Dr. WALTER N. DAL LAGO
Secretario General Fa. M. A. F.

Dra. ISABEL DOTTI
VICE DECANA
Fa.M.A.F.