

UNIVERSIDAD DE BUENOS AIRES
FACULTAD DE CIENCIAS EXACTAS Y NATURALES

1. DEPARTAMENTO: Computación.
2. CUATRIMESTRE: Segundo 2003
3. ASIGNATURA: ARQUITECTURA DE SOFTWARE
4. CARRERA: Licenciatura en Ciencias de la Computación
5. CARÁCTER DE LA MATERIA: Optativa
6. NUMERO DE CÓDIGO DE CARRERA: 18
7. NUMERO DE CÓDIGO DE MATERIA: C
8. PUNTAJE: 1 p
9. PLAN DE ESTUDIOS AÑO: 1993
10. DURACIÓN DE LA MATERIA: semanal
11. HORAS DE CLASE SEMANAL:

a) TEÓRICAS/PRACTICAS: 15 hs	c) PROBLEMAS HS.
b) LABORATORIO:	d) SEMINARIOS HS
12. CARGA HORARIA TOTAL: 15 hs
13. ASIGNATURAS CORRELATIVAS: Es conveniente haberse graduado o estar en los últimos años de una carrera de grado en Informática (o área relacionada), tener al menos un año de experiencia en el desarrollo de sistemas, y poder leer material técnico en inglés. Además, es altamente recomendable el conocimientos de tecnologías de orientación a objetos y de UML (Unified Modeling Language).
14. FORMA DE EVALUACIÓN: trabajos prácticos. Final.
15. PROGRAMA: adjunto a esta hoja
16. BIBLIOGRAFÍA: se adjunta

FECHA: 5/05/03


Firma del Profesor
Hernán ASTUDILLO


Firma del Director

Dr. Marcelo Radian
Director
Depto. de Computación
F C E y N UBA

CURSO DE ARQUITECTURA DE SOFTWARE

Audiencia objetivo:

Profesionales de sistemas interesados en perfilarse como arquitectos de software o asumir roles de estrecha relación con ellos (jefes de proyecto, diseñadores, analistas)

Ejecutivos de áreas de TI interesados en mejorar sus procesos de desarrollo haciéndolos mas centrados en arquitectura.

Programa:

Día 1:

Presentación del Curso. Inserción de las actividades de arquitectura dentro del proceso de desarrollo de software y el ciclo de vida de los sistemas. Atributos funcionales y cualitativos de los sistemas. Nociones de vista y notación.

Ejemplos: aplicaciones tempranas sobre Internet. Diferencias entre las actividades de arquitectura y las de diseño. El concepto de "parte interesada" ("stakeholder"). Vistas y notaciones.

Día 2:

Elaboración de los ejemplos, enfatizando evaluación desde el punto de vista de las partes interesadas. Ejemplo: banca electrónica. Comparación de soluciones y discusión.

Escenarios para la definición de atributos cualitativos: disponibilidad, modificabilidad, desempeño, seguridad, posibilidad de prueba, usabilidad. Calidades de la especificación, la arquitectura, el sistema y el proceso. Tácticas para el cumplimiento de requisitos cualitativos.

Día 3:

Elaboración de Arquitecturas. Patrones de arquitectura. Estilos arquitecturales.

Documentación de Arquitecturas: normas, enfoques. Documentación de variabilidad, dinamismo, contexto, interfaces, comportamiento.

Día 4:

Evaluación de arquitecturas: rol de los "stakeholders". Métodos: ATAM, SAAD, ARID.

Diseño de arquitectura dirigido por atributos. Ejemplo: la problemática básica de los sistemas CRM.

Día 5:



Dr. Marcelo Fabián Friar
Director
Depto. de Computación
E. C. E. N. I. C. A.

Arquitectura y Middleware. Uso de Componentes. Desarrollo de Líneas de Productos.

Rol del arquitecto en la definición del proceso y el ciclo de vida. Estructura organizacional y arquitectura. Reflexión. Problemas abiertos en la disciplina.

Bibliografía:

Software Architecture in Practice, 2nd Edition
Len Bass, Paul Clements, Rick Kazman
Addison-Wesley (2003), ISBN: 0321154959 amazon

Design and Use of Software Architectures
Jan Bosch
Addison-Wesley (2000), ISBN: 0201674947 amazon

Software Architecture: Organizational Principles and Patterns
David M. Dikel, David Kane, James R. Wilson
Prentice Hall (2000), ISBN: 0130290327 amazon

Component Software: Beyond Object-Oriented Programming, 2nd Edition
Clemens Szyperski
Addison-Wesley, 2002. amazon

Software Reuse: Architecture, Process and Organization for Business Success
Ivar Jacobson, Martin Griss, Patrik Jonsson
Addison-Wesley, 1997.

Documenting Software Architectures: Views and Beyond
Paul Clements (Editor), Felix Bachmann, Len Bass, David Garlan, James Ivers, Reed Little, Robert Nord, Judith Stafford
Addison Wesley Professional (2002) amazon

Evaluating Software Architectures: Methods and Case Studies
Paul Clements, Rick Kazman, Mark Klein
Addison-Wesley (2002), ISBN: 020170482X amazon

Pattern-Oriented Software Architecture, Volume 1: A System of Patterns
Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, Michael Stal
Wiley (1996), ISBN: 0471958697 amazon

Software Architecture for Product Families: Principles and Practice
Mehdi Jazayeri, Alexander Ran, Frank van der Linden
Addison-Wesley (2000), ASIN: 0201699672 amazon

IT Architectures and Middleware: Strategies for Building Large, Integrated Systems
Chris Britton
Addison-Wesley (2000) amazon

Building Java Enterprise Applications, Vol. 1: Architecture
Brett McLaughlin
O'Reilly (2002) amazon



Dr. Marcelo Fabián Frias
Director
Depto. de Computación
F C E v N UBA