

---

**Formulario de Aprobación Curso de Actualización 2016**

**Asignatura:** Introducción a la Arquitectura de Software

---

**Profesor de la asignatura:** Dr. Ing. Daniel Calegari, Prof. Adjunto, Gr. 3, InCo

**Otros docentes de la Facultad:** MSc. Ing. Laura González, Prof. Adjunto, Gr. 3, InCo

**Instituto ó Unidad:** Instituto de Computación

---

**Fecha de inicio y finalización:** 5 de julio a 21 de julio de 2016

**Horario y Salón:** martes y jueves de 18 a 21, salón a confirmar

**Horas Presenciales:** 18

**Arancel:** \$4.900

**Público objetivo y Cupos:** El público objetivo son profesionales que trabajen en desarrollo de software. Tendrán preferencia aquellas personas inscriptas en programas de posgrado del Instituto.  
Sin cupo.

---

**Objetivos:**

En la actualidad, las funcionalidades esperadas de sistemas de software han crecido considerablemente en cantidad y complejidad. Un problema crucial que enfrenta el desarrollo de este tipo de sistemas de gran porte es definir una estructura manejable y controlable. En este contexto, la Arquitectura de Software ha emergido como disciplina cumpliendo ya un rol significativo en el desarrollo de software.

Los objetivos de este curso son introducir al estudiante en los conceptos básicos de la Arquitectura de Software, transmitir su importancia y su rol dentro del proceso de desarrollo, brindar los conceptos técnicos más relevantes e identificar el rol del Arquitecto en el proceso de desarrollo.

---

**Conocimientos previos exigidos:**

Conocimientos básicos de desarrollo de sistemas orientados a objetos y de bases de datos.

**Conocimientos previos recomendados:**

Experiencia o conocimientos en procesos de desarrollo e ingeniería de software, así como en modelado con UML

---

**Metodología de enseñanza:**

El curso consta de dos partes. La primera parte corresponde a exposición teórica-práctica que contiene controles de lectura que servirán como evaluación de los contenidos teóricos. La segunda parte está dedicada a un trabajo de laboratorio grupal que será realizado a lo largo del curso, en paralelo con la exposición teórico-práctica.

- Horas clase (teórico): 18
- Horas clase (práctico): incluidas en las horas de teórico
- Horas clase (laboratorio): 0

- Horas consulta: 0
  - Horas evaluación: incluidas en las horas de teórico
    - Subtotal horas presenciales: 18
  - Horas estudio: 18
  - Horas resolución ejercicios/prácticos:
  - Horas proyecto final/monografía: 0
    - Total de horas de dedicación del estudiante: 36
- 

**Forma de evaluación:**

Controles de lectura (individuales).

---

**Temario:**

- 1 Introducción
    - 1.1 Definición del concepto Arquitectura de Software.
    - 1.2 Importancia del mismo dentro de un proceso de desarrollo.
    - 1.3 Motivación para la adopción de un enfoque centrado en la arquitectura, características y beneficios.
  - 2 Representación de la Arquitectura
    - 2.1 Modelos de representación de la arquitectura.
    - 2.2 Vistas para representar diferentes aspectos de la arquitectura.
    - 2.3 Documento de especificación de la arquitectura (SAD).
  - 3 Perspectiva de Calidad
    - 3.1 Atributos de calidad (requerimientos no funcionales) como influenciadores de la arquitectura
    - 3.2 Tácticas para su resolución.
  - 4 Evaluación de Arquitecturas
    - 4.1 Mecanismos de evaluación de arquitecturas de software.
    - 4.2 Aseguramiento de requerimientos funcionales y no funcionales.
  - 5 Tecnología y Arquitecturas
    - 5.1 Relación entre tecnologías y arquitecturas.
    - 5.2 Casos de estudio.
- 

**Bibliografía:**

- [1] Software Architecture in Practice, Second Edition.  
Len Bass, Paul Clements, Rick Kazman. Addison Wesley, 2003, ISBN 0321154959
  - [2] Software System Architecture: Working With Stakeholders Using Viewpoints and Perspectives.  
Nick Rozanski, Eoin Woods. Addison Wesley, 2005, ISBN 0321112296
  - [3] Documenting Software Architectures: Views and Beyond.  
Paul Clements, Felix Bachmann, Len Bass, et al. Addison Wesley, 2002, ISBN 020170482X
  - [4] Evaluating Software Architectures: Methods and Case Studies.  
Len Bass, Paul Clements, Rick Kazman. Addison Wesley, 2001, ISBN 020170482X
  - [5] Pattern-Oriented Software Architecture: A System of Patterns  
F. Buschman et al. John Wiley & Sons, 1996, ISBN 0471958697
-