Ministerio de Cultura y Educación **(Programa del año 2017)**
Universidad Nacional de San Luis
Facultad de Ciencias Físico Matemáticas y Naturales
Departamento: Informatica
Area: Area IV: Pr. y Met. de Des. del Soft.

I - Oferta Académica

Materia	Carrera	Plan	Año	Período
ARQUITECTURA DE SOFTWARE	ING. INFORM.	026/12-08/15	2017	1° cuatrimestre

II - Equipo Docente

Docente	Función	Cargo	Dedicación
RIESCO, DANIEL EDGARDO	Prof. Responsable	P.Asoc Exc	40 Hs

III - Características del Curso

Credito Horario Semanal				
Teórico/Práctico	Teóricas	Prácticas de Aula	Práct. de lab/ camp/ Resid/ PIP, etc.	Total
Hs	2 Hs	Hs	3 Hs	5 Hs

Tipificación	Periodo
B - Teoria con prácticas de aula y laboratorio	1º Cuatrimestre

Duración			
Desde	Hasta	Cantidad de Semanas	Cantidad de Horas
13/03/2017	23/06/2017	15	75

IV - Fundamentación

La arquitectura de software se ha vuelto indispensable con el crecimiento en complejidad y tamaño de los sistemas de software que se desarrollan. Esta describe la organización fundamental del sistema, y esta descripción ilumina decisiones de diseño de alto nivel: su composición y partes interactuantes, sus interacciones y patrones de comunicación presentes, y las propiedades claves de sus partes en las que la totalidad del sistema descansa o impone al sistema. Se convierte entonces en un modelo relativamente pequeño con respecto al sistema real, que permite a los distintos interesados comunicarse, y tratar decisiones de diseño que tendrán un profundo impacto en el trabajo subsiguiente, y en el éxito del sistema.

V - Objetivos / Resultados de Aprendizaje

Al finalizar el curso se espera que el alumno sea capaz de:
* Describir las arquitecturas de software con precisión y de manera efectiva.
* Desarrollar las habilidades para reconocer patrones arquitectónicos en sistemas de software existentes.
* Proponer arquitecturas alternativas para encarar un determinado problema.
* Desarrollar módulos de software correspondientes a una definición de arquitectura determinada.
* Desarrollar herramientas para definir arquitecturas.
* Utilizar el “dominio de conocimiento” para instanciar una arquitectura para una familia de aplicaciones en particular.

VI - Contenidos

Unidad 1 - Introducción
La Arquitectura del Software. Definición. Que no es Arquitectura. El papel de la arquitectura en proyectos de software de alta complejidad. Niveles de diseño de software y el nivel arquitectónico del diseño de software. Caracterización de la arquitectura de software.

Unidad 2 - Lenguajes de Descripción Arquitectónica
Lenguajes para el modelado, la descripción y prueba de la arquitectura. Representación de componentes, conectores, configuraciones y restricciones. Prueba de consistencia arquitectónica. Soporte de evolución

Unidad 3 – Análisis y Diseño Arquitectónico
Arquitectura lógica: conjunto de patrones y abstracciones coherentes que proporcionan el marco de referencia necesario para guiar la construcción del software para un sistema de información. La Arquitectura de Software como fundamento para que analistas, diseñadores, programadores trabajen coherentemente. Arquitecturas y atributos de Calidad. Alcanzando la Calidad. Criterios de análisis y diseño. El rol del dominio del espacio de problema en la especialización de una arquitectura general a una familia de aplicaciones en particular. Métodos de análisis, diseño y evaluación de arquitecturas. MDA (Model Driven Architecture). SOA (Service Oriented Architecture). Herramientas de análisis, diseño e instanciación.

Unidad 4 – Patrones Arquitectónicos
Caracterización de patrón arquitectónico. Patrones, Modelos de Referencia y Arquitecturas de Referencia. Catálogos de patrones arquitectónicos. Patrones de arquitecturas de sistemas de software de mayor difusión. Catálogos conocidos: Shaw-Garlan, Buschmann, Fowler.

Unidad 5 – Modelos y Métodos Formales
Modelos y métodos formales para arquitecturas de software: Fundamentos formales de la Arquitectura de Software. Bases matemáticas. Caracterizaciones formales de propiedades extra-funcionales; ejemplos: mantenibilidad, calidad de servicio.

Unidad 6 - Temas pendientes en el ámbito de las Arquitecturas de Software
Análisis. Falta de criterios unificados. Ausencia de un modelo de proceso de “punta a punta” de amplia aceptación. Desarrollo en paralelo de conceptos antagónicos o no coordinados. Vínculo entre los enfoques de Arquitecturas de Software y Métodos ágiles. Arquitecturas de Software y Ciclos de vida. Patrones, estilos y tácticas. Definición precisa del ámbito de incumbencias de Arquitecturas de Software: ¿Existen estrategias que no implementan principios arquitectónicos pero que suelen incluirse en “Arquitecturas de Software?. Falta de “masa crítica” en las herramientas y lenguajes de modelado arquitectónico: conectores abstractos de primera clase.

VII - Plan de Trabajos Prácticos

1. Lenguajes de Descripción de Arquitecturas
2. Análisis y Diseño Arquitectónico
3. Estilos Arquitectónicos
4. Modelos y Métodos Formales

Laboratorio: Aplicar los conceptos aprendidos para instanciar una arquitectura de una familia de aplicaciones y construir software para la definición de dicha arquitectura. (40 hs).

VIII - Regimen de Aprobación

La materia se desarrolla con la modalidad de promoción sin examen final. Existen dos niveles:

a) Regularización solamente: Para regularizar la materia se deberá:
1.- Tener como mínimo un 80% de asistencia a clases prácticas.
2.- Tener los trabajos solicitados por la cátedra aprobados.
3.- Presentación y aprobación del proyecto integrador de laboratorio con nota mayor o igual a 7 (siete).
4.- Aprobar un parcial con una nota mayor o igual a 6 (seis). El parcial tiene dos recuperaciones.

b) Promoción sin examen final: Para regularizar y aprobar la materia se deberá:

1.- Cumplir con los requisitos a.1, a.2 y a.3.
2.- Aprobar un parcial o su recuperacion con una nota mayor o igual a 7 (siete).
3.- Aprobar una prueba final integradora con una nota mayor o igual a 7 (siete).

Aquellos alumnos que sólo regularicen la materia deberán rendir un examen final, en los turnos establecidos.

Alumnos Libres: Por las características propias del proyecto de laboratorio a desarrollarse durante todo el cuatrimestre, no se aceptan alumnos libres.

IX - Bibliografía Básica

[1] Software Architecture in Practice, 3rd Edition. Len Bass, Paul Clements, Rick Kazman (2012). Addison Wesley
[2] Pattern-Oriented Software Architecture For Dummies. Robert Hanmer. Wiley. 2012.
[3] MDA Explained: The Model Driven Architecture(TM): Practice and Promise, Anneke G. Kleppe, Jos Warmer, and Wim Bast (2003), Addison-Wesley Professional.
[4] SOA Design Patterns, Thomas Erl (2009), Prentice Hall.
[5] SOA Open Source, Jeff Davis. Anaya Multimedia. 2009.
[6] Software Architecture: Foundations, Theory, and Practice. R. N. Taylor, N. Medvidovic, E. M. Dashofy (2009). Wiley.
[7] Documenting Software Architectures: Views and Beyond (2nd Edition), Paul Clements, Felix Bachmann, Len Bass, David Garlan, James Ivers, Reed Little, Paulo Merson, Robert Nord, Judith Stafford (2010). Addison-Wesley
[8] Designing Software Architectures: A Practical Approach (SEI Series in Software Engineering). Humberto Cervantes, Rick Kazman. Addison-Wesley. 2016.
[9] Service-Oriented Architecture: Analysis and Design for Services and Microservices. Thomas Erl. Prentice Hall. 2016

X - Bibliografia Complementaria

[1] Evaluating Software Architectures: Methods and Case Studies. Paul Clements, Rick Kazman, Mark Klein (2005). Addison-Wesley
[2] Software Architecture: Perspectives on an Emerging Discipline. Mary Shaw, David Garland (1996). Prentice Hall
[3] Models for Evaluating and Improving Architecture Competence, Len Bass, Paul Clements, Rick Kazman, Mark Klein (2008). http://resources.sei.cmu.edu/asset_files/TechnicalReport/2008_005_001_14972.pdf
[4] Pattern-Oriented Software Architecture Volume 1: A System of Patterns. Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, Michael Stal (1996). Wiley.
[5] Patterns of Enterprise Application Architecture. Martin Fowler (2002). Addison-Wesley Professional.

XI - Resumen de Objetivos

Reconocer, describir, y desarrollar arquitecturas de software

XII - Resumen del Programa

Diseño Arquitectónico
Patrones Arquitectónicos
Modelos y Métodos Formales

XIII - Imprevistos

.

XIV - Otros