Ministerio de Cultura y Educación **(Programa del año 2008)**
Universidad Nacional de San Luis **(Programa en trámite de aprobación)**
Facultad de Ciencias Físico Matemáticas y Naturales **(Presentado el 19/06/2008 12:08:53)**
Departamento: Informatica
Area: Area IV: Pr. y Met. de Des. del Soft.

I - Oferta Académica

Materia	Carrera	Plan	Año	Período
INGENIERIA WEB	TCO.UNIV.EN WEB		2008	1° cuatrimestre

II - Equipo Docente

Docente	Función	Cargo	Dedicación
ROQUÉ FOURCADE, LUIS ERNESTO	Prof. Responsable	P.Adj Simp	10 Hs
SALGADO, CARLOS HUMBERTO	Prof. Co-Responsable	P.Adj Exc	40 Hs

III - Características del Curso

Credito Horario Semanal				
Teórico/Práctico	**Teóricas**	**Prácticas de Aula**	**Práct. de lab/ camp/ Resid/ PIP, etc.**	**Total**
5 Hs	Hs	Hs	5 Hs	10 Hs

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

Duración			
Desde	**Hasta**	**Cantidad de Semanas**	**Cantidad de Horas**
10/03/2008	20/06/2008	15	150

IV - Fundamentación

El desarrollo de aplicaciones web se ha visto impactado por múltiples tecnologías que confluyen en un ambiente heterogéneo como Internet. Este nuevo panorama tecnológico obliga a reconsiderar el grado de importancia asignado al diseño arquitectural ubicándolo en el centro de la escena y convirtiéndolo en la médula de los procesos de desarrollo de software.
En ese sentido los métodos y técnicas de desarrollo de software recientes, asigan al diseño arquitectural un rol central, considerándolo desde etapas tempranas y ubicándolo en el eje que conduce la mayoría de las actividades de desarrollo.
En otro sentido también, la tecnología ha producido avances importantes en la identificación y especificación de una variedad de patrones arquitecturales, con diferentes grados de estandarización, que simplifican, aceleran y estandarizan el diseño arquitectural.
Los métodos de desarrollo, a pesar del rol central que asignan al diseño arquitectural, no incorporan de manera formal el uso de estos patrones arquitecturales (más allá de aconsejar su uso) por diferentes motivos, entre los cuales se puede mencionar: mantener el grado de generalidad del método, el ritmo aún lento con que los métodos se adecúan a los avances tecnológicos, etc.

V - Objetivos / Resultados de Aprendizaje

Formar a los alumnos en uso de metodologías de desarrollo incorporándo patrones arquitecturales ampliamente utilizados en el diseño de arquitecturas para aplicaciones web.

VI - Contenidos

Unidad 1: Introducción al desarrollo de aplicaciones web
Historia y evolución de Modelos de Desarrollo de Software. Modelos de desarrollo centrados en la arquitectura.

Arquitecturas Orientadas a Componentes. Arquitecturas Orientadas a Servicios. Patrones de Diseño. Patrones Arquitecturales. Patrones de Solución. Evolución tecnológica y patrones.
Internet como ambiente para la ejecución de aplicaciones distribuidas y orientadas a componentes: Componentes, Integración, Aplicaciones compuestas.

Unidad 2: Orientación a Objetos, Java, UML y XML
Revisión de Orientación a Objetos. Introducción. Clases y Objetos. Operaciones y Atributos. Abstracción y Encapsulamiento. Comunicación y Mensajes. Asociaciones y Composiciones. Herencia y Polimorfismo.
Revisión de Java. Java, un lenguaje Orientado a Objetos. JVM, Aplicaciones y Applets. Estructura del Lenguaje y principales construcciones. Tipos de Datos y Operadores, Herencia y Polimorfismo en Java. Swing y JDBC.
Revisión de UML. Principales características. Modelo de Ciclo de Vida. Modelo Iterativo. Entrega Incremental. Principales Actividades y Modelos. Principales características de las Actividades de Análisis y Diseño.
XML, Introducción. Validación e interpretación. Procesamiento. Transformaciones.

Unidad 3: Java 2 Enterprise Edition (J2EE)
Introducción. Beneficios y principales características. J2EE y MVC. Patrones de Diseño. Vista, Controlador y Modelo. Diseño de Aplicaciones J2EE usando el Patrón Arquitectural MVC.
Introducción a la Vista. Contenedor Web. Servlets y JSP's.
Servlets, principales características. Ciclo de Vida. Principales métodos. Parámetros HttpServletRequest y HttpServletResponse. Invocando un Servlet. Manejo de entrada y salida en Servlets. Mapping, Encabezados, Respuestas y Códigos de Estado. Desplegado. Acceso a Bases de Datos. JDBC. Datasources.
Técnicas avanzadas. Despacho y Filtros. Eventos de Ciclo de Vida. Sesiones
JavaServer Pages. Comparación con Servlets. Invocación. Elementos Básicos. Desarrollo centrado en componentes. JavaBeans. Custom Tags. Lenguaje de Expresiones
Comunicaciones. Remote Method Invocation (RMI). RMI y J2EE. Java Naming and Directory Interface (JNDI). Contenedor J2EE y JNDI. Contexto Inicial. Método lookup(). Obteniendo referencias a recursos locales y remotos. Propiedades de ambiente. Tunneling sobre http. Configuración de referencias a EJB's y Datasources.
Enterprise JavaBeans (EJB's). Introducción. Tipos. Session, Entity y Message-Driven EJB's. Arquitectura. Servidor y Contenedores EJB. Servicios provistos. Clientes de EJB's. Clases e Interfaces de EJB's. Interfase Componente y Home (Local y Remota). Objeto Componente y Home (Local y Remoto). Clase del Bean. Archivos de configuración. Desplegado y ejecución.
Enterprise JavaBeans. Session Bean (Stateful and Stateless). Entity Bean. Container Managed Persistence (CMP) y Bean Managed Persistence (BMP). Container Managed Relationship (CMR). Java Message Service (JMS). Message-Driven Bean (MDB). Desarrollo de MDB's usando JMS. Asociación de Recursos JMS con MDB.
Integrando componentes J2EE en una Aplicación J2EE.

Unidad 4: Tópicos Avanzados
Web Services. Introducción. Arquitecturas SOA (Service Oriented Architecture). SOAP (Simple Object Access Protocol), WSDL (Web Service Description Language) y UDDI (Universal Description, Discovery and Integration). Desarrollo.
Seguridad. Introducción. Arquitectura de Seguridad. Java Authentication and Authorization Service (JAAS). Definición de Usuarios, Grupos y Roles.

VII - Plan de Trabajos Prácticos

Trabajos Prácticos
Práctico 1: Ejercitación sobre Lenguaje de Programación Java
Práctico 2: Ejercitación sobre XML
Práctico 3: Ejercitación sobre Arquitecturas Web y Patrones Arquitecturales usando UML y J2EE

Laboratorio
Desarrollo de un caso hasta especificación de diseño usando UML y Modelo de Integración de Patrones

Los alumnos deben desarrollar un caso utilizando el enfoque dado en la materia de uso e integración de patrones

arquitecturales a efectos de obtener una arquitectura adecuada para productos de software que se desempeñarán en un ambiente web.
Del mismo modo deberán aplicar las técnicas proporcionadas en la materia a efectos de orientar actividades de diseño en UML al uso de los patrones arquitecturales mencionados.

Para aprobar el laboratorio, los alumnos deberán:

1)Seleccionar al menos 3 Escenarios de un Caso de Uso
El Caso de Uso puede ser tomado de casos proporcionados por la cátedra o aportados por el alumno y aprobados por la cátedra
2)Avanzar con los modelos UML de Casos de Uso, Modelo de Clases, Escenarios y Diagramas de Secuencia
Cada Diagrama de Secuencia debe avanzar hasta un punto en el que:
- Sea evidente la aplicación de las técnicas dadas en la materia
- Sea visible un diseño arquitectural que satisfaga el modelo de integración de patrones visto en la materia
- Sea visible de manera lógica y sin ambigüedades la posible implementación de un prototipo de los diagramas de secuencia
El alumno puede opcionalmente presentar una implementación del prototipo, aunque no es exigible
3)Entregar el formato impreso y digital del laboratorio

VIII - Regimen de Aprobación

Condiciones de regularización
1. Presentación y aprobación de los proyectos de práctico de máquina
2. Aprobación de 2 exámenes parciales teórico/prácticos con nota mayor o igual a 6 (seis) o su correspondiente recuperación.
3. Se otorga, tal como lo expresa la reglamentación vigente, un recuperatorio por trabajo solamnete a aquellos alumnos que hayan acreditado tal condición en Sección Alumnos

Condiciones de aprobación
1. Por promoción, para lo cual el alumno debe contar con las condiciones de regularización y con una nota de exámenes parciales mayor o igual a 8 (ocho), ó,
2. Examen final

Condiciones de alumnos libres
Solo se admiten exámenes libres de aquellos alumnos que hayan entregado en tiempo y forma los trabajos prácticos de máquina solicitados por la cátedra. Esta condición se mantendrá en vigencia hasta el inicio del dictado de la materia correspondiente al siguiente año lectivo al que el alumno haya aprobado el/los práctico/s de máquina.

Los alumnos libres deberán aprobar un examen escrito/oral sobre los aspectos teóricos de la materia

IX - Bibliografía Básica

[1] Unified Modeling Language User Guide - Grady Booch, James Rumbaugh and Ivar Jacobson
[2] The J2EE(TM) 1.4 Tutorial - http://java.sun.com/j2ee/1.4/docs/tutorial/doc/
[3] Apuntes de la materia

X - Bibliografia Complementaria

[1] Object-Oriented Software Engineering: A Use Case Driven Approach - Ivar Jacobson
[2] Unified Modeling Language - http://www.omg.org/technology/documents/formal/uml.htm

XI - Resumen de Objetivos

Utilizar patrones arquitecturales en el diseño de arquitecturas para aplicaciones web.

XII - Resumen del Programa

1. Introducción al desarrollo de aplicaciones web

2. Orientación a Objetos, Java, UML y XML
3. Java 2 Enterprise Edition (J2EE)
4. Tópicos Avanzados

XIII - Imprevistos

XIV - Otros

ELEVACIÓN y APROBACIÓN DE ESTE PROGRAMA	
	Profesor Responsable
Firma:	
Aclaración:	
Fecha:	