



UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA

Java y aplicaciones avanzadas sobre internet

Carrera: Ingeniería en Computación
Profesor Responsable: Queiruga, Claudia
Año: Optativa
Duración: Semestral
Carga Horaria Semanal: 6hs
Carga Horaria Total: 96hs

OBJETIVOS GENERALES

Construir aplicaciones empresariales server-side distribuidas, multiusuario, de alta disponibilidad, de acceso seguro, escalable e interoperable usando tecnologías JAVA. Consolidar la formación experimental del estudiante.

PROGRAMA

Unidad 1: Introducción a JEE

Objetivos:

- Aplicar el modelo de componentes multi-tier (múltiples capas físicas) promovido por JEE en el diseño de aplicaciones.
- Establecer qué funcionalidad debe proveer cada componente y cómo interactúan entre ellas.

Las componentes del modelo JEE: componentes clientes, componentes web y componentes empresariales.

La componente fundacional del módulo web de JEE: Servlets.

Páginas dinámicas escritas en JAVA: JSP (Java ServerPages).

Las componentes empresariales: EJB (Enterprise Java Beans) de Sesión, de Entidad y Orientados a Mensajes.

Interacción entre las diferentes componentes JEE.

Los contenedores JEE. Servicios provistos implícitamente a las componentes que gerencia.

Los archivos descriptores de la aplicación JEE: ejb-jar.xml, web.xml, applicationclient.xml.

El modelo de despliegue de una aplicación JEE.

Unidad 2: Servlets HTTP

Objetivos:

- Programar servlets HTTP como una extensión de un servidor HTTP.
- Relacionar los métodos del protocolo HTTP con las peticiones a un servlet.
- Analizar el funcionamiento de un servlet, su interacción con el cliente web y el manteniendo de sesiones con diferentes clientes web.



UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA

- Desplegar servlets en servidores de testeo y analizar el archivo descriptor de la aplicación web, web.xml.

El ciclo de vida de un Servlet: inicialización, pedido de servicio y destrucción.

La API (Application Programming Interface) de Servlets: la interface `javax.servlet.Servlet`. La anatomía de un Servlet.

La clase `GenericServlet` y la clase `HttpServlet`.

La interface `HttpServletRequest` y la interface `HttpServletResponse`.

Soporte de Sesiones con Servlets: la interface `HttpSession`.

Unidad 3: Servlet Listener y Servlet Filtros

Objetivos:

- Programar listener de contexto y de sesión.
- Analizar los beneficios que aportan las componentes listeners respecto del código producido.
- Programar requerimientos no funcionales con servlets filtros: seguridad, auditoria, logging, compresión de la respuesta, etc.

El contexto de servlets: la interface `ServletContext`.

Componentes web que atienden los eventos del ciclo de vida de la aplicación, de las sesiones y de las peticiones: `ServletListeners`. Las interfaces listeners.

Relación entre el `ServletContext` y los listeners del ciclo de vida de la aplicación.

Componente web que permiten realizar pre-procesamiento y post-procesamiento de peticiones HTTP: Servlets Filtros. Cómo usar servlets filtros para implementar requerimientos no-funcionales.

Unidad 4: Java ServerPages

Objetivos:

- Programar JSPs usando los diferentes elementos promovidos por la distribución de referencia de JEE.
- Desarrollar custom tags controlados por tag handlers y por tag files y distribuirlos en un archivo JAR.
- Programar descriptivamente los manejadores que atenderán las excepciones y errores.

El ciclo de vida de una JSP. El soporte de servlets provisto para JSP.

Diferentes elementos para programar JSP: scripting (scriptlets, expresiones, declaraciones); directivas (`@page`, `@include` y `@taglib`); acciones estándares (`<jsp:useBean>`, `<jsp:forward>`, `<jsp:include>`, `<jsp:param>`, `<jsp:plugin>`); acciones customizadas y el lenguaje de expresiones JSP EL.

JSP y Java Beans.

Desarrollar tags customizados en JSP usando el modelo clásico y el “nuevo y simple” de JSP 2.0. Ventajas de trabajar con librerías de tags customizados.



UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA

El archivo descriptor de la librería de custom tags, TLD.
Objetos que controlan los tags customizados: tag handlers y tag files.
Empaquetado de librerías de custom tags en un archivo JAR.
Manejo de Errores en JSP: tipos de Errores y excepciones.

Unidad 5: Conexión a Bases de Datos

Objetivos:

- Programar componentes que accedan a una base de datos.
- Analizar los beneficios de establecer conexiones usando DataSource.

La API JDBC (Java DataBase Connectivity). Tipos de Drivers JDBC

Establecimiento de una Conexión: la clase DriverManager y la interface DataSource.

Ejecución de Sentencias SQL. Las clases Statement, PreparedStatement y CallableStatement. Recuperación de resultados: la interface ResultSet.

Soporte de Transacciones. Manejo de excepciones.

JPA (Java Persistente API). Introducción. Beans de Identidad. Anotaciones. Anotaciones del ciclo de vida.

Unidad 6: El framework JSF (JavaServer Faces)

Objetivos:

- Construir una aplicación prototipo usando JSF.
- Comparar JSF con Swing.
- Analizar los eventos generados en el cliente web y consumidos en el servidor.
- Experimentar con las facilidades provistas por los IDEs para programar vistas (páginas) JSF.

El primer framework estándar JSF (JavaServer Faces) incorporado en la especificación JEE: JSF.

Construcción de aplicaciones web programando componentes de interfaz de usuario y manejadores de eventos.

Componentes fundamentales de JSF: la vista, componentes de IU, eventos, manejadores de eventos, convertidores, validadores, mensajes, renders, el archivo faces-config.xml. Fases del procesamiento de una petición HTTP. Facelets

Unidad 7: El lenguaje AspectJ

Objetivos:

- Programar aspectos que permitan securitizar y auditar las aplicaciones web

Conceptos básicos sobre los elementos que conforman el lenguaje AspectJ: Aspect, Join points, Pointcut, Advice y Weaving



UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA

BIBLIOGRAFIA

Título: Servlets and JavaServer Pages: The J2EE Technology Web Tier

Autores: Jayson Falkner and Kevin Jones

Editorial: Addison-Wesley Professional

Año de edición: 2003

Título: Java Platform, Enterprise Edition: The Java EE Tutorial

Autores: Eric Jendrock, Ricardo Cervera-Navarro, Ian Evans, Kim Haase, William Markito

Editorial: ---

Año de edición: 2014

Disponible on-line: <https://docs.oracle.com/javasee/7/JEETT.pdf>

Título: JavaServer Faces in action

Autores: Mann, Kito D.; Burns, Alan

Editorial: Manning

Año de edición: 2005

Título: Expert one-on-one J2EE design and development

Autores: Johnson, Rod.

Editorial:

Año de edición: 2003

Título: RESTful Java with JAX-RS 2.0, 2nd Edition. Designing and Developing Distributed Web Services

Autores: Burke, Bill

Editorial: O'Reilly Media

Año de edición: 2013

Título: Aspect J In Action

Autores: Laddad, R.

Editorial: Manning

Año de edición: 2009

DESCRIPCIÓN DE LAS ACTIVIDADES TEÓRICAS Y PRÁCTICAS

El desarrollo conceptual consiste en explicaciones que luego son retomadas para la resolución de las prácticas de laboratorio.

Las actividades prácticas son grupales, se desarrollan en laboratorios de PCs, se utilizan herramientas específicas de soporte y desarrollo típicas en la comunidad de software libre, entre ellas SVN, servidor web, motor de base de datos, máquinas



UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA

virtuales, IDE Eclipse y plugins. Se estipulan instancias de resolución de trabajos prácticos en clase y de carácter individual, permitiendo el seguimiento del proceso formativo.

METODOLOGÍA DE ENSEÑANZA Y EVALUACIÓN

METODOLOGÍA DE ENSEÑANZA

La asignatura consolida la formación experimental y profesional del alumno ubicándolo en un entorno de trabajo similar al real y cotidiano. Los contenidos de la asignatura se encuentran articulados y se organizan en instancias prácticas y teóricas semanales.

Se utilizan PCs y presentaciones proyectables. Se utiliza la plataforma virtual de la Facultad de Informática, <http://catedras.info.unlp.edu.ar>, basada en la herramienta opensource Moodle (<http://moodle.org>) para publicación de apuntes y guías elaborados por la cátedra y para comunicación entre docentes y alumnos.

A partir de 2013, se incorpora en esta asignatura la modalidad de cursada semi-presencial. Los alumnos que cursen en esta modalidad, utilizarán la plataforma virtual para mantenerse comunicados con la asignatura, se abrirá un foro de consulta especial para los alumnos de esta modalidad. En esta modalidad se contemplan actividades e instancias de evaluación presencial. Asimismo se recomienda que los estudiantes que opten por la modalidad semi-presencial, cuenten con buen manejo de Java Standar Edition.

METODOLOGÍA DE EVALUACIÓN

Para la aprobación de la asignatura, el alumno deberá desarrollar un proyecto a lo largo de la cursada que tendrá pautadas 6 entregas, las mismas son de seguimiento y de evaluación con calificación. La sexta entrega consiste en la entrega del proyecto con la funcionalidad completa.

Los alumnos que cursan en modalidad semi-presencial realizarán 4 entregas de las 6 en forma presencial. A su vez, esos mismos días serán evaluados acerca de los temas de las entregas (en forma de coloquio o escrito). Las restantes 2 entregas se realizarán en forma virtual.

La aprobación de la asignatura será por promoción directa: se deberá obtener al menos 6 puntos de promedio entre las 6 entregas y no menos de 4 puntos en cada entrega individual.

Los estudiantes que no alcancen los requisitos de la promoción directa y hayan obtenido al menos 4 puntos en cada una de las primeras 5 entregas, aprobarán la cursada. Para aprobar la asignatura deberán rendir un examen final escrito en alguna de las mesas de examen contemplada en el calendario académico.