

**Carrera/ Plan:**

*Licenciatura en Informática Planes 2021/2015/2012/  
Licenciatura en Sistemas Planes 2021/2015/2012/*

**Año:** 4to/5to

**Régimen de Cursada:** Semestral

**Carácter (Obligatoria/Optativa):** Optativa

**Correlativas:** Proyecto de Software

**Profesor/es:** Claudia Queiruga y Jorge Rosso

**Hs. semanales teoría:** 2 hs.

**Hs. semanales teoría:** 4 hs.

**Año 2023**

**FUNDAMENTACIÓN**

**Java y Aplicaciones Avanzadas sobre Internet** es una materia optativa de cuarto y quinto año de las carreras Licenciatura en Informática y Licenciatura en Sistemas.

Las y los estudiantes que cursan Java y Aplicaciones Avanzadas sobre Internet manejan el lenguaje JAVA y están entrenados en el uso de algún IDE (Integrated Development Environment) relacionado a tecnologías JAVA y en herramientas de desarrollo.

Los desarrolladores de hoy reconocen cada vez más la necesidad de aplicaciones distribuidas, transaccionales y portables que aprovechan la velocidad, seguridad y fiabilidad de la tecnología del lado del servidor. **Java y Aplicaciones Avanzadas sobre Internet** le provee al estudiante de los fundamentos para la construcción de aplicaciones de gran envergadura, server-side, en tecnologías JAVA, caracterizadas por ser distribuidas, multiusuario, de alta disponibilidad, de acceso seguro, portables, escalables e interoperables.

Asimismo la tecnología elegida en esta asignatura le proporciona al estudiante de habilidades para manejar un potente conjunto de APIs, que a la vez le permiten acortar el tiempo de desarrollo, reducir la complejidad de la aplicación y mejorar el rendimiento de las mismas.

Las habilidades que el estudiante adquiere en esta asignatura le permitirán programar y desplegar aplicaciones web server-side programadas en JAVA, usando las componentes estándares que provee la especificación JEE (Java Enterprise Edition) en las diferentes capas físicas del modelo multi-capas y también usando frameworks de nivel aplicación, que facilitan y mejoran la calidad de las aplicaciones desarrolladas y desplegadas. Pondrán en práctica plataformas de despliegue y ejecución de aplicaciones en contenedores, al estilo Docker.

**OBJETIVOS GENERALES**

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

**COMPETENCIAS**

LI-CE4- Planificar, dirigir, realizar y/o evaluar proyectos de relevamiento de problemas del mundo real, especificación formal de los mismos, diseño, implementación, prueba, verificación, validación, mantenimiento y control de calidad de sistemas de software/sistemas de información que se ejecuten sobre equipos de procesamiento de datos, con capacidad de incorporación de tecnologías emergentes del cambio tecnológico. Capacidad de análisis, diseño y evaluación de interfases humano computador y computador-computador.

LI-CE5- Planificar, dirigir, realizar y/o evaluar proyectos de sistemas de software de base: Sistemas Operativos, Sistemas Operativos Distribuidos, Sistemas Operativos Dedicados. Especificación, diseño, implementación, prueba, verificación, validación, mantenimiento y control de eficiencia de los sistemas de administración de recursos que se implanten como software de base de datos sobre sistemas de procesamiento de datos, incluyendo la virtualización de recursos físicos y lógicos.

LS-CE1- Planificar, dirigir, realizar y/o evaluar proyectos de relevamiento de problemas del mundo real. Especificación formal, diseño, implementación, prueba, verificación, validación, mantenimiento y control de calidad de sistemas de software que se ejecuten sobre sistemas de procesamiento de datos, con capacidad de incorporación de tecnologías emergentes del cambio tecnológico. Capacidad de análisis, diseño y evaluación de interfases humano computador y computador-computador.

LS-CE8- Planificar, dirigir, realizar y/o evaluar proyectos de sistemas de administración de recursos. Especificación formal de los mismos, diseño, implementación, prueba, verificación, validación, mantenimiento y control de eficiencia/calidad de los sistemas de administración de recursos que se implanten como software sobre sistemas de procesamiento de datos.

### **CONTENIDOS MÍNIMOS**

La asignatura tiene un enfoque de una temática experimental sobre tecnologías de desarrollo para aplicaciones de gran envergadura.

### **PROGRAMA ANALÍTICO**

#### **Unidad 1: Introducción a JEE**

Objetivos de aprendizaje:

- Aplicar el modelo de componentes multi-tier (múltiples capas físicas) promovido por JEE en el diseño de aplicaciones.
- Identificar las funcionalidades que provee 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.

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`, `application-client.xml`.

El modelo de despliegue de una aplicación JEE.

## Unidad 2: Servlets HTTP

### Objetivos de aprendizaje:

- Programar servlets 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.
- 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 de aprendizaje:

- 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: Conexión a Bases de Datos

### Objetivos de aprendizaje:

- 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 5: Arquitectura de Aplicaciones Web: REST y SPA**

### **Objetivos de aprendizaje:**

- Introducir los conceptos del desarrollo de aplicaciones basadas en servicios REST y clientes web SPA (Single Page Application).
- Desarrollar aplicaciones bajo los lineamientos de REST y SPA.

Conceptos básicos de la Arquitectura REST. Web Services del tipo RESTful.

Desarrollar APIs utilizando la API de Java, JAX-RS (JSR 311 - Java API para RESTful Web Services).

Técnica de Inyección de Dependencias (JSR 330 - Dependency Injection for Java) implementada por HK2

El framework SPA: Angular.

Tecnologías de Seguridad de aplicaciones Angular-REST: CORS y JWT.

## **Unidad 6: Integración y Despliegue continuo de Aplicaciones**

### **Objetivos de aprendizaje:**

- Introducir el concepto de contenedores Docker y su utilización para el despliegue de aplicaciones.
- Introducir los conceptos de CI/CD - integración continua y entrega continua.
- Desarrollar aplicaciones utilizando imágenes Docker y CI/CD

Conceptos claves de Docker: contenedores, imágenes, dockerfile y registro de docker.

Volúmenes y redes de Docker. Docker compose.

Cómo construir, probar y desplegar aplicaciones de manera rápida y eficiente utilizando Gitlab y Docker.

## **BIBLIOGRAFÍA**

Servlets and JavaServer Pages: The J2EE Technology Web Tier. Jayson Falkner and Kevin Jones.

Addison-Wesley Professional, 2003. Disponible para su descarga.

Professional JSP, 2nd Edition. Simon Brown, Robert Burdick, Jayson Falkner, Ben Galbraith, Rod Johnson, Larry Kim, Casey Kochmer, Thor Kristmundsson, Sing Li. Editorial Apress, 2001. ISBN: 1861004958

Java Platform, Enterprise Edition: The Java EE Tutorial. Release 8. Disponible en <https://javaee.github.io/tutorial/>

Expert one-on-one J2EE design and development. Johnson, Rod. Editorial Wrox Press, 2003. ISBN: 0764543857.

Java Persistence with Hibernate. Christian Bauer, Gavin King. Editorial Manning Publications, 2007. ISBN: 1932394885

RESTful Java with JAX-RS 2.0, 2nd Edition. Designing and Developing Distributed Web Services. Editorial O'Reilly Media, 2013. ISBN: 978-1-44936-134-1.

---

RESTful web APIs, 1st Edition. Leonard Richardson, Sam Ruby, Mike Amundsen. Editorial O'Reilly Media, 2013. ISBN: 9781449358068.

Microservicios: un enfoque integrado. Ra-Ma, 2018. Roldán Martínez, David; Valderas Aranda, Pedro J y Torres Bosch, Victoria.

Sitio oficial de Angular: <https://angular.io>

JSON Web Token (JWT) - RCF 7519: <https://tools.ietf.org/html/rfc7519>.

JWT.IO: <https://jwt.io>

Jersey RESTful Web Services: <https://eclipse-ee4j.github.io/jersey/>

## **METODOLOGÍA DE ENSEÑANZA**

La asignatura consolida la formación experimental y profesional del estudiante 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.

El desarrollo conceptual consiste en explicaciones orales que luego son retomadas para la resolución de los trabajos prácticos y del trabajo final integrador.

Las actividades prácticas se trabajan en grupos de estudiantes, se utilizan herramientas específicas de soporte y desarrollo típicas en la comunidad de software libre, entre ellas, servidor web, motor de base de datos, máquinas virtuales, IDEs para desarrollo, repositorios de software, tecnologías Docker, de licencias libres.

Se prevén instancias de resolución de trabajos prácticos en clase, de carácter individual y grupal, permitiendo el seguimiento del proceso formativo. Durante la cursada las y los estudiantes desarrollan un **proyecto final integrador**, organizado en diferentes entregables.

En síntesis, **las estrategias empleadas para el dictado de la asignatura** combinan:

- la exposición oral para el desarrollo de los contenidos teóricos,
- explicaciones de herramientas y/o tecnologías necesarias para el desarrollo de determinadas actividades,
- la resolución de trabajos prácticos, con entregas pautadas,
- muestras y coloquios de las producciones para el seguimiento del proceso formativo,
- el desarrollo de un proyecto de software final integrador, organizado en entregables.

Las y los estudiantes desarrollarán el proyecto final en equipos de 2 estudiantes, en algunos casos se admitirán grupos de 3. Cada grupo tendrá asignado un docente que acompañará la evolución de los aprendizajes y el desarrollo del proyecto final. Esta asignación se realiza al comenzar la cursada y es el mismo docente el que acompaña todas las actividades desarrolladas, tanto individuales como grupales, favoreciendo el seguimiento del proceso formativo.

La formulación del **proyecto final integrador**, en lo posible, se articula con la Secretaría de Extensión de la Facultad y dará respuesta a problemáticas de organizaciones de la sociedad civil y/o instituciones de

---

gestión pública con las que se trabaja desde dicha Secretaría y se intentará articular con proyectos o equipos de extensión.

Se **utilizará el EVEA** (Entorno Virtual de Enseñanza y Aprendizaje) de la Facultad de Informática <http://catedras.info.unlp.edu.ar> como apoyo adicional a las actividades presenciales y como herramienta fundamental para la modalidad de cursada semipresencial. Los materiales con los que se trabaja serán puestos a disposición a través de dicha plataforma y se utilizará la facilidad de tareas programadas para completar las actividades entregables y las evaluaciones.

Se **utilizará en repositorio de software GitLab** disponible en el LINTI para los entregables del proyecto final (<https://gitlab.linti.unlp.edu.ar>).

La asignatura ofrece dos modalidades de cursada: **presencial** y **semipresencial**<sup>1</sup>, que el estudiante puede optar al comenzar la cursada.

### **Modalidad Presencial**

Para el desarrollo de las clases teóricas, en esta modalidad, se utiliza una PC y un cañón óptico dispuesto en el aula, asimismo se dispone de acceso a Internet, posibilitando mostrar on-line, durante las clases, ejemplos que pueden aplicarse a las clases prácticas. Estas explicaciones y guías se ponen a disposición de las y los estudiantes mediante su publicación en el EVEA.

Las actividades prácticas y las evaluaciones se desarrollan en una de las salas de PC de la Facultad, donde se dispone de computadoras con acceso a Internet y doble booteo.

### **Modalidad semipresencial**

Para el desarrollo de esta modalidad se utilizará el EVEA mencionado previamente, para mantener la comunicación con los docentes y acceder a los materiales.

Se abrirá un foro de consulta especial destinado a las y los estudiantes que cursan en esta modalidad.

Se pondrá a disposición los materiales generados en la cursada 2021, entre ellos, grabaciones de clases teóricas y explicaciones.

Se contemplan actividades e instancias de evaluación presencial.

Si bien no constituye un requisito, se recomienda a las y los estudiantes que opten por la modalidad semipresencial cuenten con manejo de Java Standard Edition.

### **EVALUACIÓN**

Para la aprobación de la asignatura las y los estudiantes deberán desarrollar el proyecto final a lo largo de la cursada, organizado en 6 etapas entregables. Las mismas son de seguimiento y evaluación con calificación. La última de estas entregas consiste en el proyecto completo con toda la funcionalidad implementada, y en funcionamiento en el servidor de prueba provisto por la cátedra.

**La aprobación de la asignatura es mediante 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.

---

<sup>1</sup>A partir de 2013 se incorpora a la **cursada presencial la modalidad semi-presencial**. Esta modalidad puede ser elegida por las y los estudiantes.

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 contempladas en el calendario académico. Durante estas instancias de entregables se realizará un breve coloquio sobre la entrega desarrollada, con la intención que las y los estudiantes puedan explicar la solución planteada y las decisiones tomadas.

En la modalidad presencial las evaluaciones se realizan en la sala de PC asignada, utilizando el equipamiento y el software provisto por la facultad, en los horarios de práctica, de manera de permitir a los estudiantes recibir orientaciones sobre el desarrollo de las mismas.

En la modalidad semipresencial, 3 de los 6 entregables parciales serán presenciales, 2 virtuales y la entrega final puede ser virtual o presencial. A su vez, en estas mismas entregas se realizarán dos coloquios: uno sobre los temas teóricos abordados en el entregable y otro sobre la solución planteada y decisiones tomadas.

### **CRONOGRAMA DE CLASES Y EVALUACIONES**

Clase	Fecha	Contenidos	Actividades
Clase 1	Miércoles 15/3	Arquitectura JEE Desarrollo de Servlets HTTP	Presentación de la materia.
			Presentación de la arquitectura JEE y sus componentes. La componente fundacional: Servlets.
			Presentación del proyecto final
			Práctica de servlets http.
			Presentación de las herramientas de desarrollo (IDE, github, servidor Tomcat, servidor de BD)
Clase 2	Miércoles 22/3	Soporte de Sesiones HTTP	Presentación de Sesiones HTTP en la API de Servlets.
			Práctica de sesiones HTTP.
Clase 3	Miércoles 29/3	Servlet Listeners	Presentación de objetos listeners y su aplicación en desarrollo de aplicaciones web. Problemas de concurrencia en aplicaciones web java. Delegación de peticiones HTTP y redirecciones respuestas HTTP.

			Práctica sobre desarrollo de servlets listeners, alcances y redireccionamiento HTTP.
Clase 4	Miércoles 5/4	Persistencia en Aplicaciones JAVA server-side.	Presentación de acceso a bases de datos a través de JDBC (Java DataBase Connectivity) y DataSource.
		Persistencia de objetos mediante un ORM: JPA & Hibernate	Ejemplos de DAOs (Data Access Objects). Presentación de JPA (Java Persistence API). Hibernate.
			<b>Publicación de la etapa 1 del proyecto final: “Análisis, diseño y maquetado del sistema”</b>
			Se trabaja en la etapa 1 del proyecto final. Consultas y guías para el desarrollo de la primera entrega del proyecto final.
<b>Clase 5</b>	<b>Miércoles 12/4</b>	<b>Primera instancia de evaluación</b>	
			Consultas y entrega del prototipo del proyecto final
			<b>Entrega y presentación de la etapa 1 del proyecto final: “Análisis, diseño y maquetado del sistema”</b>
Clase 6	Miércoles 19/4		<b>Puesta en común acerca del alcance del proyecto final:</b> se harán las devoluciones de las entregas y se <b>definirá el alcance del proyecto final</b> . Se solicitará a cada grupo que adecúe el documento de análisis y diseño a lo establecido y que lo <b>entregue como parte de la segunda instancia de evaluación</b> .
		Servlet Filter	Presentación de objetos filtros y su aplicación a requerimientos no funcionales.  Práctica sobre desarrollo de servlets filtros, uso de conexiones JDBC y DATASOURCE para conexiones a bases de datos y del patrón DAO.
Clase 7	Miércoles 26/4	Integración con Maven	<b>Publicación del alcance de la segunda etapa del trabajo final: “Definición de objetos del modelo”</b>

			Explicación de Maven y su aplicación en el trabajo final.
			Práctica sobre persistencia usando JPA.
<b>Clase 8</b>	<b>Miércoles 3/5</b>	<b>Segunda instancia de evaluación</b>	
			Consultas sobre la etapa 2 del proyecto final. <b>Entrega y presentación de la etapa 2 del trabajo final “Definición de los objetos del modelo” y de las adecuaciones del documento de análisis y diseño.</b>
Clase 9	Miércoles 10/5	Inyección de dependencias	<b>Explicación y publicación de los alcances de la etapa 3 del trabajo final: “Desarrollo de la capa de persistencia”</b>
			Se presenta la Inyección de dependencias y su implementación con HK2
Clase 10	Miércoles 17/5	Webservices tipo RESTful	Presentación de Arquitectura REST, Web Services de tipo RESTful.
			Se trabaja en la etapa 3 del proyecto final “Desarrollo de la capa de persistencia”.
			Comienzo del desarrollo de APIs utilizando la API de Java JAX-RS (JSR 311 - Java API para RESTful Web Services).
<b>Clase 11</b>	<b>Miércoles 24/5</b>	<b>Tercera instancia de evaluación</b>	
			<b>Entrega y presentación de la etapa 3 del trabajo final: “Desarrollo de la capa de persistencia”</b>
Clase 12	Miércoles 31/5	Aplicaciones SPA: Angular - Parte 1	<b>Explicación sobre los alcances de la etapa 4 del trabajo final: “Desarrollo de una API REST”</b>
			Presentación del framework Angular. Elementos principales de un proyecto Angular. Binding, formularios y ruteo.  Consultas de la etapa 4 del trabajo final “Desarrollo de una API REST”
Clase 13	Miércoles 7/6	Aplicaciones SPA:	Presentación de ruteo avanzado de Angular,

		Angular - Parte 2  Swagger	servicios e inyección de dependencias y comunicaciones con servidor remoto utilizando HTTP.
			Práctica de Angular.
			Explicación de Swagger.
			Consultas de la etapa 4 del trabajo final "Desarrollo de una API REST"
<b>Clase 14</b>	<b>Miércoles 14/6</b>	Aplicaciones SPA: Angular - Parte 3	<b>Cuarta instancia de evaluación</b>
			<b>Entrega y presentación de la etapa 4 del trabajo final: "Desarrollo de una API REST"</b>
			Presentación de comunicación entre componentes Angular, configuración de ambientes y uso de interceptores.
			<b>Explicación y publicación de los alcances de la etapa 5 del trabajo final: "Desarrollo de la vista del proyecto usando Angular e integración con la capa de servicios"</b>
Clase 15	Miércoles 21/6		<b>Consultas de la etapa 5 del trabajo final: "Desarrollo de la vista del proyecto usando Angular e integración con la capa de servicios"</b>
<b>Clase 16</b>	<b>Miércoles 28/6</b>	Integración y Despliegue continuo de Aplicaciones	<b>Entrega Etapa 5</b>
			<b>Entrega Etapa 5, con capa de servicios y conexión a servicios REST.</b>
			Explicación sobre despliegue de aplicaciones usando Docker.
<b>Clase 17</b>	<b>Miércoles 5/7</b>		<b>Re-Entrega Etapa 5</b>
			<b>Consultas de la etapa 5 del trabajo final: "Desarrollo de la vista del proyecto usando Angular e integración con la capa de servicios".</b>
			<b>Entrega Final del proyecto (6ta entrega)</b>

<b>Clase 18</b>	<b>Miércoles 12/7</b>		<b>Desarrollo de la vista del proyecto usando Angular e integración con la capa de servicios</b>
<b>VACACIONES DE INVIERNO del 17/07 al 30/07</b>			

**Cronograma de evaluaciones**

<b>Evaluaciones previstas</b>	<b>Fecha</b>
<b>Entrega etapa 1 del proyecto final</b>	<b>Miércoles 12/4</b>
<b>Re-entrega de la etapa 1 del proyecto final</b> Para las y los estudiantes que no alcanzaron los objetivos de la entrega 1.	Miércoles 3/5
<b>Entrega etapa 2 del proyecto final.</b>	<b>Miércoles 3/5</b>
<b>Re-entrega de la etapa 2 del proyecto</b> Para las y los estudiantes que no alcanzaron los objetivos de esta etapa.	Miércoles 24/5
<b>Entrega etapa 3 del proyecto final.</b>	<b>Miércoles 24/5</b>
<b>Re-entrega de la etapa 3 del proyecto</b> Para las y los estudiantes que no alcanzaron los objetivos de esta etapa	Miércoles 14/6
<b>Entrega etapa 4 del proyecto final.</b>	<b>Miércoles 14/6</b>
<b>Re-entrega etapa 4 del proyecto final.</b>	Miércoles 26/6
<b>Entrega etapa 5 del proyecto final (nota de cursada)</b>	<b>Miércoles 28/6</b>
<b>Re-entrega etapa 5 del proyecto final (nota de cursada)</b>	Miércoles 5/7
<b>Entrega entrega final del proyecto - 6ta entrega (nota de promoción)</b>	<b>Miércoles 12/7</b>

**Contacto de la cátedra (mail, sitio WEB, plataforma virtual de gestión de cursos):**Profesora Titular: Claudia Queiruga: [claudiaq@info.unlp.edu.ar](mailto:claudiaq@info.unlp.edu.ar)Profesor Adjunto: Jorge Rosso: [jrosso@info.unlp.edu.ar](mailto:jrosso@info.unlp.edu.ar)Jefe de Trabajos Prácticos: Diego Bellante: [diegobellante@gmail.com](mailto:diegobellante@gmail.com)Plataforma virtual: <https://catedras.info.unlp.edu.ar/> (categoría "Cursos 2023")

Firma del/los profesor/es



Claudia Queiruga



Jorge Rosso