Taller de Lenguajes II	**Carrera/ Plan:** *Ingeniería en Computación* Plan 2024
Año 2025	**Año:** 2 **Régimen de Cursada:** *Semestral* **Carácter (Obligatoria/Optativa):** Obligatorio **Correlativas**: Taller de Lenguajes I **Profesoras**: Claudia Queiruga y Laura Fava **Hs. semanales**:6

Fundamentación

Taller de Lenguajes 2 es una asignatura de carácter obligatorio y de régimen semestral correspondiente al 2do. año de la carrera de Ingeniería de Computación, del trayecto de algoritmos y programación. Articula con la asignatura “Taller de Lenguajes 1” también de segundo año (primer semestre); es en ésta en la que el estudiante pone en práctica los conceptos de resolución de problemas, adquiridos hasta el momento, utilizando un lenguaje de programación procedural. Estos conceptos son retomados en “Taller de Lenguajes 2” y reformulados utilizando el paradigma de programación orientada a objetos y aplicados al lenguaje de programación Java.
Taller de Lenguajes 2 es de gran relevancia en la formación del estudiante dado que en ella adquiere los conocimientos fundamentales del paradigma de orientación a objetos, que aplicará en el desarrollo proyectos de software y que profundizará en las asignaturas más avanzadas de la carrera como “Ingeniería de Software” y en las asignaturas optativas de tipo A (cercanas al software).
Asimismo, al ser una materia de tipo taller, permite al estudiante adquirir habilidades y destrezas en el uso de un lenguaje de programación orientado a objetos popularmente usado en la industria de desarrollo de software, como lo es Java. Herramientas utilizadas en entornos de trabajo real, son adoptadas en el taller y de esta manera se acerca al estudiante a un ambiente de trabajo profesional, real y cotidiano.

Objetivos Generales

Presentar formalmente el paradigma de objetos, sus características, ventajas y aplicaciones dentro del desarrollo de sistemas de software. Desarrollar prácticas concretas con lenguajes orientados a objetos (como por ejemplo Java o C++).

Ejes Transversales

03. Especificación, proyecto y Desarrollo de Software y Sistemas Conjuntos de Hardware y Software haciendo uso de conceptos, métodos y herramientas de gestión de proyectos, ingeniería de software, base de datos, experiencia del usuario, elicitación, análisis, especificación y validación de requerimientos. (2-Medio)
04. Desarrollo de Redes de Computadoras y de Redes de Computadoras de área amplia,locales, inalámbricas y móviles. (1-Bajo)
06. Proyecto, desarrollo, dirección, control, construcción, operación y mantenimiento de Sistemas de Procesamiento de Señales, Sistemas Embebidos y sus periféricos incluido en software de soporte, Sistemas Computarizados de automatización y control y Sistemas Conjuntos de Hardware y Software. (2-Medio)

09. Identificación, formulación y resolución de problemas de ingeniería en computación. (2-Medio)
10. Concepción, diseño y desarrollo de proyectos de ingeniería en computación. (1-Bajo)
12. Utilización de técnicas y herramientas de aplicación en la ingeniería en computación. (2-Medio)
15. Fundamentos para una comunicación efectiva. (1-Bajo)

Contenidos Mínimos

Objetos. Clases e instancias. Encapsulamiento. Jerarquías de clase. Herencia. Polimorfismo. Lenguajes y aplicaciones.

Programa Analítico

Unidad I: Conceptos de Programación Orientada a Objetos
Objetos, estado y comportamiento. Clases e Instancias. Interacción entre objetos mediante mensajes. Encapsulamiento y ocultamiento de información. Generalización y especialización. Herencia, Herencia Simple. Polimorfismo y binding dinámico. Conceptos básicos de modelado usando UML (Unified Modeling Language).

Unidad II: Java, un Lenguaje y una Plataforma
El desafío de la plataforma universal JAVA. La plataforma de desarrollo Java o JSE (Java Standard Edition)
La plataforma de ejecución Java o JRE (Java Runtime Environment). La máquina virtual JAVA.
Eclipse: un IDE (Integrated Development Environment) de la comunidad de Software Libre para Java.

Unidad III: Clases y Espacios de Nombres en JAVA
Definición de clases y espacios de nombres.
Miembros de una clase: atributos y métodos. Creación e inicialización de objetos: Constructores. La palabra clave this y this(). Especificadores de acceso para miembros: private, protected, default y public. Especificadores de calificación: static, final y abstract. Paquetes, como espacio de nombres: la palabra clave package. La variable de entorno CLASSPATH.
Empaquetado de archivos .class y recursos de la aplicación usando el formato estándar JAR (Java ARchive). Documentación de la API de Java: ¿dónde está disponible? y ¿cómo usarla?

Unidad IV: Herencia y Polimorfismo en JAVA
Creación de objetos. Encadenamiento de constructores. La palabra clave super y super(). La clase Object: los métodos toString(), equals() y hashCode(). Relación entre el especificador de acceso protected y la herencia. Sobrescritura de métodos y ocultamiento de atributos. Upcasting. Clases Abstractas. Interfaces. Comparación entre interfaces y clases abstractas. Conexión entre interfaces y herencia múltiple. Interfaces y polimorfismo.

Unidad V: El framework de Colecciones JAVA
Interfaces core: Collection, Set, List, SortedSet, Map, SortedMap. Interfaces para ordenación de objetos: Comparable y Comparator. Interfaces para iterar: Iterator, ListIterator. Implementaciones: HashSet, HashMap, HashTable, ArrayList, Vector, TreeSet, TreeMap, LinkedList. Algoritmos polimórficos para ordenación, búsqueda, manipulación de datos.

Unidad VI: Desarrollo aplicaciones de escritorio con interfaz de usuario gráfica
JFC (Java Foundation Classes): AWT (Abstract Window Toolkit), componentes de IU (Interfaces de Usuario), manejo de eventos; componentes Swing, gerenciadores de tamaño y posicionamiento.

Creación de aplicaciones de escritorio con interfaz de usuario. Las clases: java.awt.Frame y javax.swing.JFrame.

Unidad VII: Acceso a base de datos usando JDBC (Java Database Connectivity)
Persistencia de datos. Tipos de persistencia con JAVA: Serialización y JDBC & SQL. Drivers, definición y tipos. Sentencias simples SQL para operaciones CRUD (create-read-update-delete). Definición de capas de acceso a datos (DAO) por entidad y utilizando tipos de datos genéricos.

Unidad VIII: Manejo de errores en JAVA
Manejo de errores en Java. Tipos de errores. La clase Exception y sus subclases. Definición de excepciones propias. Las sentencias throw y throws y el bloque try/catch/finally.

Unidad IX: Concurrencia en JAVA
Threads. Creación y administración de threads.
La clase Thread y la interfaz Runnable. El método run().
Métodos de la clase Thread. El ciclo de vida de un thread.
El sistema de scheduling de threads en JAVA.

Bibliografía

Análisis y diseño orientado a objetos con aplicaciones (2da edición). Grady Booch Editorial: Addison-Wesley, 1998.

UML: arquitectura de aplicaciones en C++, Java y Python (2a ed). Jiménez de Parga, Carlos.RA-MA, 2020.

Aprende a programar con Java: un enfoque práctico partiendo de cero (2da edición). Jiménez Marín, Alfonso. Madrid: Paraninfo, 2021. ISBN: 9788428338578.

Effective Java (3ra edición). Boston: Addison-Wesley, 2018. ISBN: 9780134685991.

El Lenguaje Unificado de Modelado: guía del usuario (2da edición). Booch, Grady. Madrid: Pearson Educación, 2006. ISBN: 978-84-7829-076-5.

Head first design patterns (1ra edición). Freeman, Elisabeth. Sebastopol, 2004. ISBN: 978-0-596-00712-6.

The Java module system. Parlog, Nicolai. Shelter Island: Manning, 2019. ISBN: 9781617294280.

Thinking in Java (4ta edición). Eckel, Bruce. Upper Saddle River, 2006. ISBN: 0131872486.

UML gota a gota. Fowler, Martin. Pearson, 1999. ISBN: 968-444-364-1.

UML toolkit (1ra edición). Eriksson, Hans-Erik. Nueva York: Wiley, c. 1998. ISBN: 0-471-19161-2.

Metodología de Enseñanza

Las clases están organizadas en actividades teóricas y prácticas de carácter presencial. Las estrategias empleadas para dictar la asignatura combinan la exposición oral para el desarrollo de la teoría, la resolución de trabajos prácticos en las clases prácticas y el desarrollo de un proyecto de software integrador. Los contenidos de la teoría y la práctica se encuentran estrechamente vinculados.
En las clases teóricas se trabajan contenidos conceptuales de programación orientada a objetos y su implementación en Java, que son retomados en forma aplicada durante las prácticas.
La metodología de trabajo en las prácticas es de tipo taller.

En las clases prácticas se trabaja en la resolución de ejercicios vinculados a los temas desarrollados en los teóricos y en la implementación de un proyecto de software integrador con entregas parciales que son consideradas instancias de evaluación parcial. Todas las actividades prácticas de los estudiantes son supervisadas por los docentes.

Recursos y materiales didácticos

Para el dictado de las clases teóricas se utiliza una PC y un cañón óptico, disponibles en el aula, facilitando la muestra de ejemplos que pueden aplicarse a las clases prácticas.
Las explicaciones y guías se ponen a disposición de los estudiantes de la asignatura mediante su publicación en el aula virtual de la misma. Para ello se utiliza la plataforma de gestión de cursos virtuales del LINTI basada en la herramienta Moodle (https://catedras.linti.unlp.edu.ar/) como mecanismo de apoyo adicional a las clases presenciales.
Las clases prácticas se desarrollan en dos espacios áulicos: a) un aula laboratorio equipada con 30 computadoras de escritorio y b) un aula de clases en la que se dispone de un laboratorio móvil equipado con 30 notebooks de la Facultad de Informática. Por otra parte, es habitual que algunos estudiantes lleven sus computadoras personales a las clases prácticas.
Para la resolución de los trabajos prácticos se utilizan tecnologías y herramientas de desarrollo de software como UML para el modelado del sistema, IDEs para el desarrollo y administración del proyecto, clientes SQL para la definición y administración de la base de datos y repositorios de software para el trabajo colaborativo y gestión de versionado de código.

Evaluación

Para la aprobación de la asignatura se utiliza un sistema de promoción directa basado en la evaluación por módulos, en los que se evalúan contenidos teórico-prácticos.
Estas evaluaciones se realizan durante la cursada y se estructuran en tres módulos que constituyen las instancias de evaluación parcial teórico-práctica. Cada uno de los módulos se corresponde con las entregas parciales del proyecto de software integrador. Estas entregas cuentan con una instancia de recuperatorio, a la que se suma una instancia de evaluación flotante que puede ser utilizada por los estudiantes para alcanzar la aprobación de los módulos desaprobados o para alcanzar una nota que les permita promocionar la materia.

Condiciones de aprobación de la asignatura

Para aprobar los trabajos prácticos, los estudiantes deberán obtener al menos 4 puntos en cada uno de los módulos de la evaluación teórico-práctica.
Para aprobar la promoción se debe obtener en promedio 6 puntos entre los 3 módulos y al menos 4 puntos en cada uno de ellos.
Los estudiantes que no alcancen la nota de promoción, pero sí aprueben los tres módulos, aprobarán la cursada. Para aprobar la asignatura deben rendir un examen final en alguna de las mesas examinadoras del calendario académico.

Cronograma de clases y evaluaciones

Las clases teóricas y prácticas se desarrollan los días lunes (teoría y práctica) y los días martes y jueves (sólo práctica).

	SEMANA	ACTIVIDADES		
		LUNES		MARTES/JUEVES
		TEORÍA	PRÁCTICA	PRÁCTICA
1	11/08/2025			**Jueves 14/08/2025** (TEORÍA) Presentación del paradigma de OO e introducción al modelado OO
2	18/08/2024	**Feriado nacional - Paso a la Inmortalidad del Gral. José de San Martín (trasladado del 17 de Agosto)**		Práctica 1 - Modelado
3	25/08/2025	La Plataforma JAVA Definición de clases. Instanciación de Objetos. Constructores de clases	Explicación UML Práctica 2 - Plataforma Java **Explicación del Trabajo Entregable N°1**	Práctica 2 - Plataforma Java Consultas Entregable 1
4	01/09/2025	Herencia en JAVA Cadena de constructores	Práctica 3 - Herencia Consultas Entregable 1	Martes 09/09/2025 Práctica 3 - Herencia Consultas Entregable 1
				Jueves 11/09/2025 Asueto - Día del Trabajador Docente
5	08/09/2025	Clases abstractas y Polimorfismo Librería de componentes (paquetes: Creación de librerías, Nombres únicos de clases y Archivos JAR)	Práctica 4 - Cadena de constructores, Polimorfismo Consultas Entregable 1	Práctica 4 - Cadena de constructores, Polimorfismo Consultas Entregable 1
6	**Instancia de evaluación MÓDULO 1: ENTREGABLE 1**			
	15/09/2025	Ocultamiento de información: especificadores de acceso en JAVA. Tipos enumerativos. Singleton	**Entrega del Entregable 1** Práctica 5 - Encapsulamiento, Especificadores de acceso. Tipos enumerativos	Corrección (no hay clases prácticas)
7	22/09/2025	**Coloquio Entregable 1 (horario de teoría y práctica)**	Práctica 5 - Encapsulamiento, Especificadores de acceso. Tipos enumerativos	Práctica 5 - Encapsulamiento, Especificadores de acceso. Tipos enumerativos

8	29/09/2025	Interfaces	Práctica 6 - Interfaces **Explicación del Trabajo Entregable N°2**	Práctica 6 - Interfaces
9	06/10/2025	JDBC y SQL	Explicación - Uso de GIT Práctica 7 A- JDBC y SQL Consultas Entregable 2	Práctica 7 A -JDBC y SQL Consultas Entregable 2
10	13/10/2025	**Feriado nacional - Día del Respeto a la Diversidad Cultural (trasladado del 12/10)**		Consultas Entregable 2
11	20/10/2025	Data Access Object (DAO)	**Re-entrega Entregable 1** Práctica 7 B- DAO - Factory DAO Consultas Entregable 2	Práctica 7 B- DAO - Factory DAO Consultas Entregable 2
12	27/10/2025	Colecciones con tipos genéricos	Práctica 8 - Colecciones y tipos genéricos Consultas Entregable 2	Práctica 8 - Colecciones y tipos genéricos Consultas Entregable 2
13	03/11/2025	**Instancia de evaluación MÓDULO 2: ENTREGABLE 2**		
		AWT/JFC, Swing y Manejo de eventos	**Entrega del Entregable 2** Práctica 9 - GUI	Corrección (no hay clases prácticas)
14	10/11/2025	**Coloquio Entregable 2 (horario de teoría y práctica)**	Práctica 9 - GUI	Práctica 9 - GUI
15	17/11/2025	Entrada Salida - java.io Excepciones	**Explicación del Trabajo Entregable N° 3** Consultas Entregable 3	Consultas Entregable 3
16	24/11/2025	**Feriado nacional - Día de la Soberanía Nacional (traslado del 20 de Noviembre)**		Consultas Entregable 3
17	01/12/2025	Threads	**Re-entrega Entregable 2** Consultas Entregable 3	Consultas Entregable 3
18	08/12/2025	**Instancia de evaluación MÓDULO 3: ENTREGABLE 3 (martes 09/09/2025)** **Feriado nacional - Inmaculada Concepción de María**		Corrección del Entregable 3
19	15/12/2025	**Coloquio Entregable 3 (horario de teoría y práctica)**	No hay clase práctica	No hay clase práctica
	VACACIONES			
	09/02/2026	**Entregable 3 y Re-Entrega**		
	16/02/2026	**Coloquio Entregable 3 (horario de teoría y práctica) y Fecha Flotante**		

Evaluaciones previstas	Fecha
Entrega del MÓDULO 1: ENTREGABLE 1	15/09/2025
Coloquio ENTREGABLE 1	22/09/2025
Recuperatorio ENTREGABLE 1: Re-entrega	20/10/2025
Entrega del MÓDULO 2: ENTREGABLE 2	03/11/2025
Coloquio ENTREGABLE 2	10/11/2025
Recuperatorio Entregable 2: Re-entrega	01/12/2025
Entrega del MÓDULO 3: ENTREGABLE 3 (fecha desdoblada)	09/12/2025
Coloquio ENTREGABLE 3 (fecha desdoblada)	15/12/2025
Entrega del Módulo 3: ENTREGABLE 3 (fecha desdoblada) Recuperatorio ENTREGABLE 3: Re-entrega (destinada a estudiantes que entregaron en diciembre y no alcanzaron a aprobar)	09/02/2026
Coloquio ENTREGABLE 3 (fecha desdoblada)	16/02/2025
Fecha Flotante	

Contacto de la cátedra (mail, sitio WEB, plataforma virtual de gestión de cursos):

Profesora Titular: Claudia Queiruga (claudiaq@info.unlp.edu.ar)
Profesora Adjunta: Laura Fava (lfava@info.unlp.edu.ar)

Jefa de Trabajos Prácticos: Vanessa Aybar Rosales (vaybar@info.unlp.edu.ar)

Plataforma virtual: https://catedras.linti.unlp.edu.ar/ (categoría “Cursos 2024”)

Claudia Queiruga - Profesora Titular

Laura Fava - Profesora Adjunta