



**PROGRAMA PLANEAMIENTO EDUCATIVO  
DEPARTAMENTO DE DISEÑO Y DESARROLLO CURRICULAR**

<b>PROGRAMA</b>					
		<b>Código en SIPE</b>	<b>Descripción en SIPE</b>		
<b>TIPO DE CURSO</b>		049	Educación Media Tecnológica		
<b>PLAN</b>		2004	2004		
<b>SECTOR DE ESTUDIO</b>		620	Informática		
<b>ORIENTACIÓN</b>		481	Informática		
<b>MODALIDAD</b>		---	Presencial		
<b>AÑO</b>		2°	Segundo		
<b>TRAYECTO</b>		---	---		
<b>SEMESTRE</b>		---	---		
<b>MÓDULO</b>		---	---		
<b>ÁREA DE ASIGNATURA</b>		915	Técnicas Informáticas Nivel II		
<b>ASIGNATURA</b>		3502	Programación II		
<b>ESPACIO o COMPONENTE CURRICULAR</b>		Tecnológico			
<b>DURACIÓN DEL CURSO</b>		Horas totales: 96	Horas semanales: 3	Cantidad de semanas: 32	
Fecha de Presentación: 14/9/18	N° Resolución del CETP	Exp. N°	Res. N°	Acta N°	Fecha __/__/__

## FUNDAMENTACIÓN

Se debe orientar al estudiante a construir soluciones técnicas utilizando tecnologías Java de Sun Microsystems®. Es responsable de construir aplicaciones JEE en capas con acceso a bases de datos MySQL mediante JDBC, interfaces gráficas con Swing y uso de servlet.

Este curso pretende afianzar los conceptos de la programación orientada a objetos y su implementación práctica en Java. Se resaltarán la ventaja de utilizar el lenguaje Java por la portabilidad del código obtenido, el cual puede ejecutar tanto en plataformas Windows® como en Linux/Unix y Mac® así como el carácter gratuito de todas las herramientas utilizadas en el curso además de seguir siendo el lenguaje de programación más usado del mundo.

Se cubrirán también aspectos básicos del lenguaje gráfico de modelado UML (Unified Modeling Language) para la representación de las clases y objetos, la construcción de aplicaciones en 3 capas (Presentación, Lógica y Persistencia) utilizando para cada capa un subconjunto de tecnologías disponibles en Java.

En la actualidad los dos entornos dominantes de programación son, por un lado, .NET de Microsoft® y Java de Sun®. Si bien estas dos plataformas presentan importantes diferencias, también comparten varias características como ser la orientación a objetos, la construcción de interfaces gráficas basadas en eventos, el manejo estructurado de errores (excepciones), el uso de un lenguaje intermedio (MSIL y Bytecodes), librerías base, etc. Por tanto los asistentes al curso no sólo aprenderán las herramientas básicas para construir aplicaciones de escritorio en una de las plataformas más utilizadas (Java), sino que también aprenderán múltiples conceptos que le servirán para otras plataformas (como ser .NET) que la estudiarán en el próximo año (Programación III).

## OBJETIVOS

- ) Programar bajo el paradigma de Programación Orientada a Objetos (POO)
- ) Desarrollar interfaces visuales de programación
- ) Conectar los programas desarrollados con una Base de Datos relacional
- ) Producir productos dinámicos para Internet
- ) Desarrollar productos de software mediante el modelo a 3 capas
- ) Fomentar la creatividad, el respeto, la empatía y el trabajo en equipo

## CONTENIDOS

- 1) Repaso de Programación 1 (Marzo)
  - a) Tipos de datos
  - b) Estructuras de control
  - c) Estructuras iterativas
  
- 2) Programación Orientada a Objetos JAVA (Abril)
  - a) Conceptos básicos y avanzados orientados a objetos y su aplicación en Java
  - b) Construcción de programas en consola (usando por ejemplo: objeto, clase, atributos, métodos, operaciones, constructores, destructores, referencias, herencia, redefinición, sobrecarga, polimorfismo y otros)
  - c) Introducción a la notación en UML y aplicación de la misma
  
- 3) Desarrollo de interfaces gráficas (Mayo/Junio)
  - a) Introducción a WindowsBuilder
  - b) Introducción a Swing
  - c) Jerarquía de componentes gráficos en swing
  - d) Distribución de componentes mediante el uso de LayoutManagers
  - e) Manejo eventos en swing
  - f) Formularios, contenedores intermedios (paneles) y controles visuales (botones, listas, listas desplegables, botones, menús,...)
  - g) Construcción de interfaces gráficas de usuario (aplicaciones de escritorio) utilizando swing
  
- 4) Unificación de conceptos: (Julio)
  - a) Modelado con UML de una aplicación en 2 capas orientada a objetos y aplicación de los conocimientos adquiridos en los dos puntos anteriores (OOP y Swing) en la construcción de la Capa Lógica y de la Capa Presentación de dicha aplicación, manteniendo separadas las responsabilidades de cada capa (luego se extiende el Caso de Estudio a 3 capas).
  
  - b) Correspondencia entre aplicaciones orientadas a objetos (paradigma de objetos) y acceso a bases de datos (paradigma relacional): mapeo objeto/relacional. Discusión de alternativas para el desarrollo de la persistencia.

- 5) Arquitectura y diseño de aplicaciones Java Enterprise (Agosto)
- a) Plataforma Java EE.
  - b) Herramientas de desarrollo y servidor de aplicaciones.
  - c) JavaBeans, anotaciones y registro.
  - d) Modelo de componentes Web.
  - e) Desarrollo de Servlets.
  - f) Desarrollo con con tecnología Java Server Page (JSP).
  - g) Tecnología Web Service.
  - h) Implementar una política de seguridad.
- 6) Desarrollo de Componentes Web con Servlets (Agosto)
- a) Fundamentos de aplicaciones web.
  - b) Desarrollo de un Servlet.
  - c) Manejo de formularios con Servlets.
  - d) Configurar tu aplicación web.
  - e) Implementar un diseño MVC.
  - f) Usar filtros en aplicaciones web.
  - g) Otras funcionalidades Servlets.
  - h) Implementar políticas de seguridad.
  - i) Integrar aplicaciones web con bases de datos.
  - j) Construcción de aplicaciones en 3 capas y desarrollo de la tercera capa (Capa de Persistencia) utilizando los objetos provistos por JDBC. Revisión del programa realizado a 2 capas para agregar la capa de persistencia.
- 7) Conexión con MySQL (Setiembre)
- a) Conceptos cliente-servidor.
  - b) MySQL Clients.
  - c) Conectores y APIs MySQL.
  - d) Tipos de datos.
  - e) Expresiones SQL.
  - f) Obtener Metadatos.
  - g) Bases de datos.
  - h) Tablas.
  - i) Manipulación de datos de las tablas.
  - j) Sentencias.

- k) Rutinas de almacenamiento.
  - l) Gestión de errores y alertas.
  - m) Optimización.
- 8) Trabajo Obligatorio (Octubre)
- a) Realización de un trabajo obligatorio, en equipo, guiado y tutorado por el docente

### EVALUACIÓN

La evaluación educativa se concibe como un proceso enfocado a la valoración del grado de logro alcanzado de los objetivos planteados. Es entonces un proceso de valoración de cómo se van desarrollando los procesos de enseñanza y aprendizaje con el fin de mejorarlos en ambos sentidos: mejorar la tarea docente y facilitar el desarrollo de los aprendizajes.

Los criterios de evaluación se extraen de la justificación del curso, de los objetivos y contenidos del mismo. Es el profesor quien deberá concretarlos y adaptarlos, en función de las características de su alumnado, los medios disponibles y la metodología de enseñanza.

La evaluación del alumno debe tener las siguientes características:

- )] Integral: ya que se ha de tener en cuenta todos los aspectos de la personalidad del alumno (conceptos, procedimientos, actitudes, capacidades de relación y comunicación así como del desarrollo autónomo de cada estudiante.).
- )] Continua: durante el proceso. Se ha de hacer una evaluación:
  - o Inicial: para detectar los conocimientos previos de los alumnos
  - o Progresiva: para saber en qué situación se encuentra el alumno en cada momento.
  - o Final: como síntesis de todo lo conseguido.
- )] Formativa: los resultados de la evaluación deben servir al alumno para mejorar sus procesos de aprendizaje con la orientación del profesor.

Los instrumentos de evaluación tienen por objeto dar información sobre el grado de adquisición de los aprendizajes esperados. La información recogida debe ser válida y fiable.

La evaluación debe realizarse en múltiples ocasiones y a través de diferentes instrumentos como ser: observación, orales, trabajos domiciliarios (individuales o en grupo), trabajos en clase, escritos, prácticos, obligatorios, cuestionarios, exposiciones u otras.

Al final de curso los estudiantes deberán conformar equipos para realizar un trabajo obligatorio. La entrega, evaluación y defensa del trabajo obligatorio corresponden al 50% de la calificación anual final.

### BIBLIOGRAFÍA

- J Arnold, K., Gosling, J. y Holmes (2001) "El lenguaje de programación Java", D. Editorial Prentice-Hall. Tercera edición.
- J BRUNNER, J. (2003). Educación e Internet ¿la próxima revolución? Chile. Fondo de Cultura Económica.
- J COLL César (1994): Psicología y Curriculum. Paidós.
- J Flanagan, D. (2002) Java in a nutshell - A desktop quick reference", Editorial Prentice-Hall.. 609 páginas.
- J Froufe, A (2000) "Computing Concepts with Java 2 Essentials",. Editorial John Wiley & sons. Inc., 682 pp.  
*Comienza con una introducción al lenguaje Java para enseguida proporcionar información sobre los aspectos mas importantes de la plataforma Java 2: excepciones, multitarea, AWT, delegacion de eventos, swing, gráficos, servlets, JDBC, ... En español.*
- J García-Beltrán, A. y Arranz, J.M, (2007). "Programación Orientada a Objetos con Java" ETSII-UPM. 315 páginas.
- J GIMENO SACRISTAN, J. (1998). El currículo: una reflexión sobre la práctica. Madrid: Morata
- J Hortsmann, C. (2000) "Java 2 - Manual de usuario y tutorial", Editorial Ra-Ma.
- J Muñoz, C., Niño, A. y Vizcaíno (2002) "Introducción a la Programación con Orientación a Objetos", A. Editorial Prentice-Hall.
- J Perez Menor, Jose María y otros. (2003) "Problemas resueltos de Programación en lenguaje Java", Editorial Thomson.  
*Pertenece a la colección Paso a Paso. Enfoque eminentemente práctico y formativo, incluyendo numerosos ejemplos y ejercicios de programación resueltos. Presenta las estructuras de datos y algoritmos desde la perspectiva del pensamiento abstracto y la resolución de problemas. Contiene múltiples ejemplos de aplicación, de una cierta entidad, cuyo código original está disponible en Internet. Visión detallada de todas las características de Java 1.1. En inglés.*
- J Weiss, M.A. (2000) "Estructuras de datos en Java", Editorial Addison Wesley