

### **1. Objetivos**

Dominar los conceptos de Arquitectura de Red y sus niveles, reconociendo los elementos computacionales que aparecen en los "headers" de los distintos niveles.

Introducción al concepto de Computación Distribuida, utilizando los elementos básicos de Programación Concurrente -p.ej. los "Threads" de Java-.

Navegar en la API de Java para encontrar las clases del package java.net y reconocer las más usadas. Asociar el manejo de archivos en Java con el manejo de "sockets" para la programación de aplicaciones en redes, entendiendo el manejo de InputStream y OutputStream.

### **2. Contenidos Mínimos**

Antecedentes de comunicación entre computadores y desarrollo de Internet; principales servicios que brinda y concepto de protocolo TC/IP. Panorama de arquitecturas y topologías de redes, evolución, antecedentes de sistema operativo de red. La web como un ejemplo de computación cliente servidor, protocolos HTML, XML y formato de páginas web. Compresión de datos. Redes y servicios: contraseñas y mecanismos para control de acceso, nombres de dominios y nombres de servicios. Desarrollo de software en ambientes web.

### **3. Contenidos**

#### **Unidad 1: Arquitectura de Redes TCP/IP. Concepto de Sockets.**

1. Conceptos sobre Arquitectura de Red.
2. Análisis desde la capa de Aplicación hasta la placa de Red. Aplicaciones: mail, telnet y FTP. Elementos computacionales de las distintas capas.
3. Caso concreto aplicado al TCP/IP. Funcionalidades de ambos Protocolos.
4. Concepto de Sockets y Circuito Virtual.
5. Escenario Cliente-Servidor. Acción del inetd(internetworking daemon).
6. Análisis de traces para descubrir funcionalidades del los niveles 2,3 y 4 de la Arquitectura de Red y su relación con los Headers y sus elementos computacionales.

#### **Unidad 2: Conceptos Fundamentales de Programación.**

1. Introducción. Conceptos de Procesos. Paralelismo. La Programación Concurrente en lenguaje Java.
2. Multithreads. Runnable. Programas Java

---

implementando

estos conceptos.

3. Programación de puertos IRQ para Serie y Paralelo.

#### **Unidad 3: Conexión a Base de Datos a través de las redes usando Java.**

1. JDBC(Java Data Base Connectivity).
2. Ejemplo para la conexión con Bases de Datos Oracle.

#### **Unidad 4: Computación bajo la forma Cliente-Servidor, aplicado a la web.**

1. La WEB como un ejemplo de computación cliente-servidor, protocolos HTML, XML y formato de páginas Web. Compresión de datos.
2. Redes y Servicios: contraseñas y mecanismos para control de acceso, nombres de dominios y nombres de servicios. Desarrollo de software en ambientes Web.

### **4. Bibliografía**

#### *4.1. Obligatoria*

Comer, D.: TCP/IP . Tomo 2, Prentice Hall.

Palma Mendez, J. T. y otros: Programación Concurrente, Ed Thomson, España,2003.

Bruce Eckel: Thinking in Java 2nd Edition,Ed Prentice Hall 2000.

Elliott Rusty Harold (First Edition): Java Networking Programming, O'Reilly 1997.

Elliott Rusty Harold (Second Edition): Java Networking Programming, O'Reilly. 2001.

#### *4.2. Complementaria*

Comer, D.: Tep/IP . Tomo 3, Prentice Hall

Deitel; H. M. - Deitel, P. J : Como Programar en Java , Ed Prentice Hall, 2005.

Deitel; H. M. - Deitel, P. J : Como Programar en C++ , Ed Prentice Hall, 2003.

Orfali, R.;Harkey; D.;Edwards, J.: Cliente/ Servidor. Guia de Supervivencia, McGraw Hill Interamericana(2da Ed), México, 1998.

### **5. Metodología de trabajo**

Clases explicativas de los conceptos teóricos fundamentales para la comprensión de la programación de las aplicaciones en Redes.

Análisis de trances TCP/IP para descubrir los elementos computacionales de estos protocolos presentes en los Headers de los mismos y comprender en profundidad su funcionalidad.

Uso del lenguaje Java para la implementación de la programación concurrente (Multithreading) y el manejo de sockets en el entorno "Cliente-Servidor" y clases del package java.net.

En los Trabajos Prácticos de laboratorio usar productos integrados de desarrollo para Java como el BlueJ de la Universidad de Kent( Deakin University) y el JDeveloper versión 10g de Oracle.

### **6. Criterios de evaluación**

Realización de uno o dos Parciales. Se exigirá obtener calificación 6(seis) para aprobar. Realización de Trabajos Prácticos, programas que se deberán mostrar funcionando en el laboratorio. Examen Final, para poder darlo es requisito tener aprobados Parciales y Trabajos Prácticos.

### **7. Práctica Propuesta**

1. Ejercicios de Análisis de Traces.
2. Programación Java para Manejo de Threads y Prioridades de los mismos. Programación Java, con el uso de la librería java.net para comunicaciones y redes. Manejo de Sockets Threading y Multithreading.
3. Conexión a Base de Datos desde la programación Java, JDBC(Java Data Base Connectivity). Uso del Jdeveloper de Oracle y ADF(Application Development Framework).
4. Aplicación integradora en ambiente Cliente-Servidor.

**Nota:** Se hace una mínima introducción conceptual en algunos temas que luego serán desarrollados con mayor profundidad y conocimientos en materias de Sistemas Distribuidos y Sistemas Concurrentes.