

1. OBJETIVOS:

Los objetivos de la asignatura de Sistemas Distribuidos pueden agruparse de la siguiente manera:

a. De la cátedra

1. Orientar y coordinar el Aprendizaje por parte del alumno que curse la materia en los conceptos fundamentales de la arquitectura y diseño de los Sistemas Distribuidos (SSDD) vigentes en el mercado.
2. Introducirlo en el desarrollo de Programas que conforman los SSDD.

b. Del alumno

- ❖ Conformar un Aprendizaje teórico-práctico, sobre los SSDD en uso, llevado de una intensa actividad de cursado e investigación que volcará luego en la confección y aprobación de un conjunto de trabajos prácticos y evaluaciones parciales teórico-prácticas.
- ❖ Lograr una concepción global y un enfoque selectivo, para realizar soluciones algorítmicas y de desarrollo de programas, de los diferentes problemas que se presentan entre la interacción del Hardware y los SSDD.

c. Según Contenidos Mínimos

Comunicación a través de sockets. Serialización. Diseño de arquitecturas cliente/servidor N-tier. Programación de la capa cliente (aplicaciones que corren en los navegadores), la capa intermedia (generación dinámica de páginas Web, sistemas complejos transaccionales) y la capa de datos (acceso a bases de datos). Concurrencia con uso de threads. Sistemas basados en componentes: CORBA. Web Services. Taller de aplicación con un lenguaje orientado a objeto.

2. Contenidos:

La cátedra del Sistemas Distribuidos, discurrirá durante el ciclo de cursado sobre tres módulos diferenciados de trabajo:

- Asistencia y estudio del programa temático de la materia (ver "programa temático").
- Módulo de Trabajos Prácticos (ver "proceso de evaluación").
- Módulo de evaluación de parciales (ver "proceso de evaluación").

Esto permitirá al alumno:

1. Evaluar, a nivel de implementación, cualquier SSDD, sobre equipos existentes en plaza.
2. Desarrollar en el alumno el interés por la investigación; utilizando publicaciones, libros y Adquirir vocabulario técnico-informático y utilizarlo con precisión
3. Conocer en forma amplia y general el funcionamiento de las partes de un SSDD.
4. sistemas reales que sean propuestos por el Profesor.
5. Ayudar a desarrollar en el alumno la actitud de detector de posibles soluciones de problemas, realizando una adecuada ejercitación práctica.

Unidad 1: Conceptos de Sistemas Distribuidos (SSDD)

1. Que es un SSDD?. Inserción de Contexto. Arquitecturas de SSDD.
2. Comunicaciones a través de sockets, en SSDD - Redes. Concepto de Sockets. Conceptos sobre Arquitectura de Red. Análisis de la capa de Aplicación hasta la capa de Red. Elementos computacionales de las distintas capas. Caso concreto aplicado al TCP/IP. Funcionalidades de ambos Protocolos.
3. Circuito Virtual y su relación con los Sockets.
4. Escenario Cliente-Servidor. Acción del inetd (internetworking daemon).
5. Análisis de traces para descubrir funcionalidades de los niveles 2,3 y 4 de la Arquitectura de Red. Relación con los Headers y sus elementos computacionales.

Unidad 2: Sockets y Paso de Mensajes. Programación Concurrente. Implementación Modelo Cliente/Servidor.

1. Tipos de Sockets. Programación de Sockets.
2. Paso de Mensajes en Cliente/Servidor. Funcionalidad.
3. Conceptos Fundamentales de Programación Concurrente. Introducción. Conceptos de Procesos y Threads (Hilos). Paralelismo.
4. La Programación Concurrente en Lenguaje Java. Threads (Hilos), Multithreads. Clase Thread e Interface Runnable.
5. Elementos del Lenguaje Java para la Programación en Redes: packages java.net y java.io, reutilización de sus clases y métodos, objetos InputStream y OutputStream, uso de buffers, su relación con los Sockets.
6. Programación en Java de Cliente/Servidor, aplicación de Multithreading.
7. Uso del "local_host" 127.0.0.1 para realizar la práctica en un solo equipo. Uso de varios equipos: Un server y uno o varios clientes.

Unidad 3: Gestión de Procesos Distribuidos

1. Arquitectura Cliente/Servidor. Paso de mensajes. RPC. Cluster - Windows Middleware.
2. Migración de procesos. Estados Globales.
3. Exclusión Mútua Distribuida. Interbloqueo distribuido. Hilos - SMP - Micronúcleos.

Unidad 4: Planificación Multiprocesador distribuido

1. Planificación Multiprocesamiento.
2. Planificación en tiempo real.
3. Planificación en Linux-Unix.
4. Planificación Windows.

Unidad 5. Sistemas basados en componentes.

1. CORBA.
2. Match.
3. WebServices.
4. Taller de aplicación con un lenguaje orientado a objetos.

3. BIBLIOGRAFIA

3.1. Principal - Primaria

- ❖ Sistemas Distribuidos. Conceptos y Diseño. 3era Edición. George Coulouris, Jean Dollimore, Tim Kindberg. Ed. Pearson, Addison Wesley. Ed. 2001, reimpresión 2005.
- ❖ Java – Soluciones de Programación. Schild Herb. 1er Edición. Ed. McGraw-Hill. 2009.
- ❖ Sistemas Operativos una Visión Aplicada. Jesús Carretero Pérez, Pedro de Miguel Anasagasti, Félix García Carballeira y Fernando Pérez Costoya. McGraw-Hill. 2001.

3.2. Consulta - Secundaria

- ❖ Sistemas Distribuidos, principios y paradigmas. Andrew Tanenbaum y Maarten Van Steen. Editorial Pearson Addison Wesley. 2008.
- ❖ Redes de Computadoras. Andrew Tanenbaum. Ed. Pearson - Addison Wesley. 2003.
- ❖ Redes de Computadoras, un enfoque descendente desde Internet. James Kurose, Keith Ross. Ed. Pearson Educación. 2004.
- ❖ Como programar en Java. Harvey Deitel. Ed. Pearson – Addison Wesley. 2008.

4. METODOLOGIA DE LA ENSEÑANZA

El alumno tendrá acceso a un conjunto de Actividades que le permitirá conformar su entorno de Aprendizaje, los cuales se desarrollarán en los siguientes lugares:

- a. Actividades de Enseñanza en el Aula. Clases Grupales de tipo Teórico.
- b. Actividades de Práctica en Laboratorios de Computadoras, que le permitirán familiarizarse con el ambiente de trabajo, y de los SSDD y de allí construir en la operación su Aprendizaje.
- c. Actividades de Investigación aplicada en los Trabajos Prácticos, de tipo Grupal. Cada Investigación deberá concluir con la correspondiente **presentación de la documentación** escrita y con el correspondiente medio de almacenamiento (disquette – cd – dvd). En todos los casos la funcionalidad es la base de la corrección y aprobación de la asignatura.

Esto permitirá al alumno:

- ❖ Adquirir vocabulario técnico-informático y utilizarlo con precisión
- ❖ Conocer en forma amplia y general el funcionamiento de las partes de un SOP.
- ❖ Evaluar, a nivel de implementación, cualquier SOP. sobre equipos existentes en plaza.
- ❖ Desarrollar en el alumno el interés por la investigación; utilizando publicaciones, libros y sistemas reales que sean propuestos por el Profesor.
- ❖ Ayudar a desarrollar en el alumno la actitud de detector de posibles soluciones de problemas, realizando una adecuada ejercitación práctica.

- ❖ Conformación de los Trabajos Prácticos y Ensayos a realizar por los Alumnos a lo largo de su cursación.

Detalle de Actividades prácticas

Lo anterior será posible materializarlo con trabajos de investigación, desarrollo de aplicaciones y evaluaciones de tipo Individual y Grupal. Estas pueden listarse de la siguiente forma:

1. *Formación experimental (P1)*

Se resuelven problemas que ilustran la teoría mediante ejemplos que se plantean en el pizarrón y luego se resuelven mediante las herramientas del SSDD elegido. Los problemas ofrecen dificultades crecientes y en algunos casos son versiones simplificadas de problemáticas reales. Trabajamos con Sistema Operativos de la familia Unix-Linux y Windows.

2. *Resolución de Problemas Abiertos de Ingeniería (P2)*

Son problemas que corresponden a situaciones reales o hipotéticas cuya solución requiera la aplicación de los conocimientos de las ciencias básicas y de las tecnologías utilizadas en SSDD. En general no tienen un planteo matemático único, sino que dependerá de los requerimientos de los que toman las decisiones y los límites que pueden plantearse a la complejidad. Las conclusiones deben presentarse en informes grupales, que deben resultar útiles a quien tome decisiones en el diseño e implementación de los SSDD.

3. *Prácticas de proyecto y diseño de sistemas informáticos(P3)*

Se entiende por tales a las actividades que empleando ciencias básicas y de la ingeniería llevan al desarrollo de un sistema, componente o proceso, satisfaciendo una determinada necesidad y optimizando el uso de los recursos disponibles. Corresponde a los casos más complejos planteados, donde los alumnos deben relacionar conceptos de matemática, economía, sistemas y toma de decisiones. Las conclusiones deben presentarse en informes grupales, que deben resultar útiles a quien tome decisiones.

❖ *Instrucción Supervisada de Formación Práctica (P4)*

Se entiende por tales a las actividades que empleando diversas herramientas de software y hardware permiten conformar un conocimiento práctico, aplicable al ámbito profesional. Son actividades grupales y se realizan en forma concentrada en los Laboratorios, guiados por el Docente.

Unidad Temática	TP – Aplicativo	Comentarios
I. Conceptos de Sistemas Distribuidos	TP1. Programación de Puertos	
II. Sockets y paso de mensajes.	TP 2. Programación de Sockets y paso de mensajes.	
III. Gestión de Procesos distribuidos	TP 3. Gestión de Procesos Distribuidos.	
IV: Planificación Multiprocesador Distribuido	TP 4. Planificación Distribuida.	
V. Sistemas basados en componentes.	TP 5. Proyecto de Sistemas Distribuidos	

5. CRITERIOS DE EVALUACION

La evaluación de los alumnos se realiza a través de Trabajos Prácticos (TPs), participación en clases, evaluaciones parciales y el Examen Final.

- ❖ **En los TPs:** los alumnos deberán poner en juego las competencias desarrolladas y los conocimientos adquiridos mediante la resolución de problemas. Se tomarán en cuenta el contenido, el cumplimiento de objetivo y consignas y la calidad de la presentación (prolijidad, ortografía, comunicación).
- ❖ **En la participación en clase:** Los alumnos serán evaluados en forma permanente a través de la calidad y oportunidad de sus intervenciones.
- ❖ **En los Parciales:** la evaluación parcial tiene como objetivo corroborar el aprendizaje realizado por los alumnos durante el curso y su evolución. Se verificará el nivel de cumplimiento de los objetivos pedagógicos del curso.
- ❖ **En el Examen Final:** La evaluación final estará basada sobre la examinación del conocimientos vistos en la materia y resolver problemas reales que permitan poner en evidencia la integración de conocimientos. Se verificará la capacidad de los alumnos en la utilización de los conceptos fundamentales de la asignatura para la organización de su trabajo, así como el nivel de análisis desarrollado y la calidad de la solución propuesta.

5.2 Requisitos para la aprobación

Aprobación del cursado de la asignatura. Para aprobar es necesario cumplir con:

- ❖ Asistencia mínima del 50%
- ❖ Aprobación del examen parcial con nota igual o superior a cuatro puntos:
- ❖ Los parciales deben rendirse en las fechas estipuladas por la Facultad, según cronograma general de la Universidad.
- ❖ En el caso de que el alumno desaprobe el examen parcial cuenta con una instancia de recuperación.
- ❖ El desaprobado o no asistir a la recuperación (teniendo el parcial desaprobado) tiene como consecuencia desaprobado el curso de la materia.
- ❖ Aprobación de los Trabajos prácticos con nota igual o superior a cuatro puntos:

- ❖ En el caso de esta materia la nota final de los trabajos prácticos se calcula como una nota promedio de los trabajos requeridos que equivale al 75% del número de TPs obligatorios.

Aprobación de la asignatura. Para aprobar la materia es necesario aprobar el cursado y el Examen Final

- ❖ Para aquellos alumnos que no alcanzaran el 75% de asistencias deberán rendir un Examen Final Escrito y luego un Examen Final Oral.
- ❖ Para los alumnos que alcancen o superen el 75% el Examen Final será sólo de tipo Oral.