

1. Objetivos

Los objetivos de la asignatura de Sistemas Operativos Avanzados pueden agruparse de la siguiente manera:

a- De la cátedra

Orientar y coordinar el Aprendizaje por parte del alumno que curse la materia en los conceptos fundamentales que le permitan entender y participar del desarrollo de módulos que conforman la arquitectura y diseño de los Sistemas Operativos (SOp) vigentes en el mercado.

Introducirlo en el desarrollo de Programas y Aplicación de Utilidades que conforman los Sistemas Operativos, en los ambientes de trabajo del futuro profesional de la Informática.

b- Del alumno

Conformar un Aprendizaje teórico-práctico, sobre los SOp no convencionales 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, de desarrollo de programas y aplicación de métodos de solución, de los diferentes problemas que se presentan entre la interacción del Hardware y los SOp, con la orientación de solución de problemas de instalación, Configuración y Explotación de Sistemas Informáticos y Computacionales.

c- Según Contenidos Mínimos

Deberá cumplirse con los siguientes CMs:

Sistemas operativos de tiempo real y sistemas distribuidos, conceptos y arquitecturas.

Sistemas empotrados. Sistemas operativos no convencionales. Concurrencia. Modelo cliente-servidor. Gestión de procesos distribuidos. Planificación de multiprocesamiento distribuido. Manejo de archivos en sistemas distribuidos. Transacciones distribuidas. Servicios web, SOA, XML, cloud computing. Tópicos de seguridad en sistemas distribuidos.

2. Contenidos

La Cátedra de Sistemas Operativos se encuentra orientada a que los Alumnos desarrollen las siguientes bases conceptuales y prácticas.

Unidad I: Introducción y Contexto

1. Repaso de los tipos de SOp más usados del mercado. Arquitectura y funcionalidad.
2. Sistemas Operativos de Tiempo Real. Estructura y funcionalidad.
3. Sistemas Operativos empotrados o embebidos. Estructura y funcionalidad.
4. Sistemas Operativos Distribuidos, conceptos y arquitectura. Sistemas Cliente/Servidor.
5. Sistemas Operativos no convencionales. Comunicación. Sincronización.

Unidad II: Sistemas monofuncionales – Empotrados o Embebidos (SSEE)

1. Desarrollo de módulos.
2. Herramientas de programación.
3. Entornos de aplicación.
4. Drivers para SSEE.

Unidad III: Sistemas Distribuidos y Cliente/Servidor

1. Concurrencia. Conceptos generales.
2. Gestión de procesos distribuidos.
3. Planificación de Multiprocesamiento bajo SSDD.
4. Problemas de diseño de SSDD.
5. Diferencias de campo entre Clientes y Servidores.

Unidad IV: Sistema de Archivos Distribuidos

1. Introducción. Características, requisitos, casos de estudio y desarrollo.
2. Arquitectura de un Sistema de Archivo Distribuido.
3. Sistema de Archivos de Red. Middleware.
4. Sistemas de archivos Andrew.

Unidad V: Transacciones Distribuidas

1. Introducción. Transacciones distribuidas planas y anidadas. Coordinación.
2. Protocolos de consumación atómica.
3. Concurrencia en transacciones distribuidas.
4. Interbloqueo distribuido.
5. Recuperación de transacciones.
6. Memoria compartida distribuida.

Unidad VI: SOp no Convencionales

1. Servicios web, XML Complejidad de entorno.
2. SOA como estructura de servicio general. XML – HTTP – SOAP – WSDL – UDDI.
3. Cloud Computing. Como servicio integrador.
4. Otros SOp no convencionales y el futuro.

Unidad VII: Seguridad Distribuida

1. Amenazas y ataques.
2. Seguridad en transacciones y diseño de sistemas seguros.
3. Criptografía moderna, usada en transacciones.
4. Firma digital en transacciones.
5. Casos de estudio. Needham-Schroeder, Kerberos, SSL y Millicent.

3. BIBLIOGRAFIA

3.1. Principal - Primaria

Sistemas Operativos, principios de diseño e interioridades. Stallings William. 5ta Edición (Esp). Prentice Hall. Madrid. 2006.

Sistemas Distribuidos. Conceptos y Diseño. 3era Edición. George Coulouris, Jean Dollimore, Tim Kindberg. Ed. Pearson, Addison Wesley. Ed. 2001, reimpresión 2005.

Sistemas de Tiempo Real y Lenguajes de Programación. Alan Burns y Andy Wellings. 3era Ed. Addison Wesley. Madrid 2003.

Sistemas Operativos, un enfoque basado en conceptos. D. M. Dhamdhere. Ed. McGraw Hill. 2008.

3.2. Consulta - Secundaria

Embedded System Design. Steve Headth. 2da edición. EDN Series for design engineer. Newnes. 2003.

Service –Oriented Architecture (SOA). Norbert Bieberstein. IBM press. Oct. 2005.

<http://eyeos.org/es/sendForm.php>. Sitio de la comunidad de EyeOS, SOp no convencional tipo Cloud Computing. Oct. 2010.

Producción de Software de Código abierto. Karl Fogel. 2007.

<http://www.producingoss.com/es/>

Por una cultura libre. Lawrence Lessing. <http://www.free-culture.cc/freecontent/>

4. METODOLOGIA DE LA ENSEÑANZA

El proceso educativo parte de la descripción y análisis de situaciones y problemas, introduciendo métodos útiles para resolverlos y aplicarlos utilizando herramientas computacionales que aprenden a utilizar los estudiantes.

El enfoque es un Taller que se dicta en un Laboratorio de Informática. La actividad comienza con el desarrollo de los contenidos conceptuales con ayuda de elementos audiovisuales y la descripción de ejemplos sencillos. Sobre la base de los problemas se presentan los modelos que los representan y desarrollan los métodos de resolución.

Una vez captado el método por parte de los estudiantes, se utilizan Sistemas Operativos usados en el mercado de trabajo para resolver casos de estudio, realizar prácticas supervisadas, por parte de los estudiantes. A dicho trabajo se agrega la discusión sobre material obtenido de Internet y una relación frecuente mediante correo electrónico con el profesor para efectuar consultas y recibir correcciones.

La Cátedra ha desarrollado problemas y casos de estudio para los distintos temas, los que sirven para ejemplificar los métodos de operación y diseño que se utilizan en los Sistemas Operativos.

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:

Actividades de Enseñanza en el Aula. Clases Grupales de tipo Teórico.

Actividades de Práctica en Laboratorios de Computadoras, que le permitirán familiarizarse con el ambiente de trabajo, y desde allí construir en la operación su Aprendizaje.

Actividades de Investigación aplicada en los Trabajos Prácticos, de tipo Grupal. Cada Investigación deberá concluir con la correspondiente **presentación de la documentación** por mail y con el correspondiente medio de almacenamiento (disquette – cd – dvd). En todos los casos la correcta 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.

Iniciarse en las actividades de trabajo que conlleva el desarrollo colaborativo con el propio grupo o con grupos más extensos, a nivel local y remoto.

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:

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 Sistema Operativo 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.

Resolución Práctica del Mundo Real (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 Sistemas Operativos. 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 Sistemas Operativos.

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.

Lista de Trabajos Prácticos propuestos:

- TP 1. Repaso general de SOp.
- TP 2. Solución a SSEE
- TP 3. Desarrollo Cooperativo, SSDD.
- TP 4. Cloud Computing. SSDD
- TP 5. Seguridad Distribuida.

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 desaprobar o no asistir a la recuperación (teniendo el parcial desaprobado) tiene como consecuencia desaprobar 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.