

Ficha Técnica

| | | | |
|----------------------------|--|---------------------------|--------------------|
| Titulación: | Grado en Ingeniería de Tecnologías y Servicios de Telecomunicación | | |
| Plan BOE: | BOE número 108 de 6 de mayo de 2015 | | |
| Asignatura: | Sistemas Distribuidos | | |
| Módulo: | Redes y Servicios | | |
| Curso: | 3º | Créditos ECTS: | 6 |
| Tipo de asignatura: | Obligatoria | Tipo de formación: | Teórica y Práctica |

Presentación

Los últimos avances en las redes de comunicaciones han incrementado la importancia de la computación distribuida en las disciplinas de las ciencias de la computación. La computación distribuida es aquella que se lleva a cabo en un sistema distribuido, que se define como un conjunto de computadores interconectados a través de una red y que son capaces de colaborar entre sí para lograr un fin común.

Esta asignatura explora las formas en que los programas, ejecutándose en computadores independientes, colaboran con otros con el fin de llevar a cabo una determinada tarea de computación, tal como los servicios de red o las aplicaciones basadas en la Web. En la asignatura se estudia la evolución de los sistemas distribuidos, los paradigmas de programación distribuida existentes y las APIs, conjuntos de herramientas, protocolos y estándares necesarios para desarrollar un sistema distribuido.

Es altamente recomendable para el aprovechamiento de esta asignatura tener conocimientos de redes de computadores, de sistemas operativos y de programación.

Antes de matricular la asignatura, verifique los posibles requisitos que pueda tener dentro de su plan. Esta información la encontrará en la pestaña "Plan de estudios" del plan correspondiente.

Competencias y/o resultados del aprendizaje

- Conocimiento y utilización de los fundamentos de programación en redes, sistemas y servicios de telecomunicación.
- Conocimiento de las características, funcionalidades y estructura de los sistemas distribuidos y capacidad para participar activamente en su especificación, diseño, implementación y mantenimiento.

Contenidos Didácticos

- 1 Introducción a la computación distribuida
 - 1.1 Conceptos básicos
 - 1.2 Computación monolítica, distribuida, paralela y cooperativa
 - 1.3 Ventajas, inconvenientes y desafíos de los sistemas distribuidos
 - 1.4 Conceptos previos sobre sistemas operativos: procesos y concurrencia
 - 1.5 Conceptos previos sobre redes
 - 1.6 Conceptos previos sobre Ingeniería del Software
 - 1.7 Organización de este manual

- 1.8 Conclusiones
- 2 Paradigmas de computación distribuida y comunicación entre procesos
 - 2.1 Comunicación entre procesos
 - 2.2 Sincronización de eventos
 - 2.2.1 Supuesto 1: envío síncrono y recepción síncrona
 - 2.2.2 Supuesto 2: envío asíncrono y recepción síncrona
 - 2.2.3 Supuesto 3: envío síncrono y recepción asíncrona
 - 2.2.4 Supuesto 4: envío asíncrono y recepción asíncrona
 - 2.3 Interbloqueos y temporizadores
 - 2.4 Representación y codificación de datos
 - 2.5 Clasificación y descripción de los paradigmas de computación distribuida
 - 2.6 Java como entorno de desarrollo de sistemas distribuidos
- 3 API de sockets.
 - 3.1 API de sockets, tipos y diferencias
 - 3.2 Sockets de tipo datagrama
 - 3.3 Sockets de tipo stream
 - 3.4 Solución a interbloqueos: temporizadores y eventos no bloqueantes
 - 3.5 Seguridad en sockets
 - 3.6 Conclusiones
- 4 Paradigma de comunicaciones Cliente-Servidor
 - 4.1 Características y conceptos fundamentales de los sistemas distribuidos de tipo cliente-servidor
 - 4.2 Proceso de diseño e implementación de un sistema cliente-servidor
 - 4.3 Problemas de direccionamiento no orientado a conexión con clientes anónimos
 - 4.4 Servidores iterativos y concurrentes
 - 4.5 Información de estado y de sesión
 - 4.5.1 Información de sesión
 - 4.5.2 Información de estado global
 - 4.6 Clientes complejos recibiendo respuestas asíncronas desde el lado servidor
 - 4.7 Servidores complejos actuando como intermediadores entre varios clientes
- 5 Comunicación de grupo
 - 5.1 Introducción a la multidifusión y usos comunes
 - 5.2 Fiabilidad y ordenación en los sistemas multidifusión
 - 5.3 Implementación Java de sistemas de multidifusión
 - 5.4 Ejemplo de uso de la comunicación en grupo entre iguales
 - 5.5 Implementaciones de multidifusión fiable
 - 5.6 Multitransmisión a nivel de aplicación
 - 5.7 Conclusiones
- 6 Objetos Distribuidos
 - 6.1 Introducción
 - 6.2 Arquitectura de una aplicación basada en objetos distribuidos
 - 6.3 Tecnologías de sistemas de objetos distribuidos
 - 6.4 Capas software de Java RMI en el lado cliente y en el lado servidor
 - 6.5 API Java RMI de objetos distribuidos
 - 6.6 Pasos para construir una aplicación RMI
 - 6.7 Uso de callback en RMI
 - 6.8 Descarga dinámica de resguardos de objetos remotos y gestor de seguridad RMI
 - 6.9 Conclusiones
- 7 Aplicaciones de Internet I: HTML, XML, HTTP.
 - 7.1 Introducción
 - 7.2 Lenguaje HTML
 - 7.3 Lenguaje XML
 - 7.4 Protocolo de Internet: HTTP

- 7.5 Uso de contenidos dinámicos: manejo de formularios y CGI
- 7.6 Manejo de datos de estado y sesión en Internet
- 7.7 Conclusiones
- 8 CORBA
 - 8.1 Introducción
 - 8.2 Arquitectura CORBA
 - 8.3 Lenguaje de descripción de interfaz en CORBA
 - 8.4 Protocolos de interoperabilidad GIOP
 - 8.5 Referencias a objeto remoto IOR
 - 8.6 Servicio de nombrado CORBA
 - 8.7 Ejemplo en IDL Java
 - 8.8 Pasos de diseño, compilación y ejecución en IDL Java
 - 8.9 Conclusiones
- 9 Aplicaciones de Internet II: Applets, Servlets y SOA.
 - 9.1 Introducción
 - 9.2 Applets
 - 9.3 Introducción a los Servlets
 - 9.4 Servlets HTTP y su funcionamiento
 - 9.5 Mantenimiento de la información de estado en Servlets
 - 9.5.1 Campos ocultos de formularios
 - 9.5.2 Cookies
 - 9.5.3 Variables de Servlet
 - 9.5.4 Objeto session
 - 9.6 Servicios web
 - 9.7 Protocolo SOAP
 - 9.8 Breve reseña de la arquitectura REST
 - 9.9 Conclusiones
- 10 Paradigmas avanzados
 - 10.1 Introducción
 - 10.2 Paradigma MOM
 - 10.3 Paradigma de agentes software móviles
 - 10.4 Paradigma de espacio de objetos
 - 10.5 Computación colaborativa
 - 10.6 Tendencias futuras en computación distribuida
 - 10.7 Conclusiones

Contenidos Prácticos

Durante el desarrollo de la asignatura se realizarán las siguientes actividades prácticas:

- Resolución de cuestiones y ejercicios prácticos sobre paradigmas de computación distribuida.
- Resolución de un caso práctico: uso de sockets en Java.
- Debate/coloquio sobre la comunicación en grupo
- Resolución de un caso práctico: invocación de métodos remotos Java.
- Resolución de cuestiones y ejercicios prácticos y supuestos sobre CORBA
- Resolución de un caso práctico: uso de Servlets, HTML, XML y JSP en servicios Web
- Trabajo de documentación sobre SOA y REST/POX-RPC
- Debate/coloquio sobre los paradigmas avanzados de computación

Evaluación

El sistema de evaluación del aprendizaje de la UDIMA contempla la realización de diferentes tipos de actividades de evaluación y aprendizaje. El criterio de valoración establecido se detalla a continuación:

| | |
|--|-------------|
| Actividades de aprendizaje | 10% |
| Controles | 10% |
| Actividades de Evaluación Continua (AEC) | 20% |
| Examen final presencial | 60% |
| TOTAL | 100% |

Bibliografía

- David Lizcano (2015). *Sistemas Distribuidos*. Madrid: Ed. CEF
- M. L. Liu (2004). *Computación Distribuida, Fundamentos y Aplicaciones*. Pearson Addison Wesley.