

**Programación Distribuida y Tiempo
Real****Carrera/ Plan:*****Licenciatura en Informática Plan 2015*****Año 2020****Año: 4º****Régimen de Cursada: Semestral****Carácter (Obligatoria/Optativa): Obligatoria****Correlativas:****Profesor/es: Fernando G. Tinetti****Hs. semanales: 6 hs****FUNDAMENTACIÓN**

Se presenta el contexto y los conceptos básicos de las herramientas de programación en los entornos distribuidos. Agrega una visión de mayor nivel de abstracción de sistemas operativos y de redes combinados con conceptos propios de sistemas distribuidos. Se realiza una presentación de los temas más importantes de los sistemas de tiempo real, el impacto en el desarrollo y evaluación de los sistemas y la relación o el impacto de los requerimientos de tiempo real en los programas distribuidos

OBJETIVOS GENERALES

Analizar las arquitecturas de procesamiento distribuido y los mecanismos de comunicación y sincronización entre procesos. Integrar los conocimientos anteriores con el manejo de datos distribuidos. Desarrollar prácticas experimentales sobre redes LAN y WAN con ambientes de desarrollo orientados a la programación distribuida. Caracterizar los sistemas de tiempo real (STRs) y los sistemas distribuidos de tiempo real (SDTRs), en particular en relación con el desarrollo de software para los mismos. Estudiar aspectos propios de la arquitectura y comunicaciones de los SDTRs.

COMPETENCIAS

- CGS2- Comunicarse con efectividad en forma oral y escrita.
- CGT1- Identificar, formular y resolver problemas de Informática.
- CGT5- Utilizar de manera efectiva las técnicas y herramientas de aplicación de la Informática.
- CGT8 Capacidad de interpretación y resolución de problemas multidisciplinares, desde los conocimientos de la disciplina informática.
- LI- CE1 – Planificar, dirigir, realizar y/o evaluar proyectos de especificación, diseño, implementación, verificación, validación, puesta a punto, mantenimiento y actualización para arquitecturas de sistemas de procesamiento de datos, con capacidad de incorporar aspectos emergentes del cambio tecnológico.
- LI- CE4 – Planificar, dirigir, realizar y/o evaluar proyectos de relevamiento de problemas del mundo real, especificación formal de los mismos, diseño, implementación, prueba, verificación, validación, mantenimiento y control de calidad de sistemas de software/sistemas de información que se ejecuten sobre equipos de procesamiento de datos, con capacidad de incorporación de tecnologías emergentes del cambio tecnológico. Capacidad de análisis, diseño y evaluación de interfases humano computador y computador-computador.

CONTENIDOS MINIMOS (de acuerdo al Plan de Estudios)

- Procesamiento distribuido. Modelos y paradigmas.
- Distribución de datos y procesos.
- Bases de datos distribuidas.
- Migración de procesos en ambientes distribuidos.
- Programación de aplicaciones en ambientes distribuidos.
- Características de los Sistemas de Tiempo real y su software.
- Procesamiento distribuido y sistemas de Tiempo Real.
- Aplicaciones de Sistemas Distribuidos de Tiempo Real

PROGRAMA ANALÍTICO

Unidad 1.- Sistemas Distribuidos. Conceptos introductorios. Motivación. Definiciones. Caracterización y problemas a resolver. Arquitecturas de procesamiento paralelo y su evolución a entornos/arquitecturas distribuidos. Redes de interconexión. Modelos de procesamiento distribuido. Distribución de datos y procesos. Ejemplos de utilización actual.

Unidad 2.- Patrones Básicos de Interacción de Procesos. Productores/Consumidores: relaciones con pipelines de procesamiento y filtros de los sistemas operativos. Cliente/ servidor: conceptos, ejemplos, sistemas operativos distribuidos, relación con Internet, relación con threads. Interacción entre pares (peer-to-peer): relación con cómputo paralelo clásico, identificación de áreas de aplicación.

Unidad 3.- Pasaje de Mensajes. Pasaje de mensajes asincrónicos. Clientes y servidores. Interacción entre pares. Pasaje de mensajes sincrónicos, implementación de mensajes asincrónicos con mensajes sincrónicos. Redes y sockets. Ejemplos con lenguajes y bibliotecas como C, MPI y Java.

Unidad 4.- RPC (Remote Procedure Call, Llamada a Procedimiento Remoto) y Rendezvous. Concepto de RPC, extensión del modelo de ejecución procedural tradicional, relación con módulos de programas. Definición de rendezvous, ejemplos cliente/servidor e interacción entre pares. Ejemplos con lenguajes de programación como Ada y Java.

Unidad 5.- Problemas/Conceptos Relacionados. Bases de datos distribuidas: motivación, antecedentes, relación con programación distribuida, consultas y transacciones distribuidas, migración y replicación de datos. Modelo de cómputo/ aplicaciones N-tier. Migración de procesos en entornos distribuidos, algunas ideas implementadas en Condor. DSM (Distributed Shared Memory, memoria compartida físicamente distribuida): motivación, implementaciones. Otros modelos y lenguajes: Bag-of-Tasks (bolsa de tareas), Manager/worker (manejador/trabajador), Master/worker (maestro/trabajador), Algoritmos de heartbeat y/o sistólicos, Algoritmos de broadcast, Servidores múltiples

Unidad 6.- Sistemas de Tiempo Real (STR) y Sistemas Distribuidos de Tiempo Real (SDTR). Definiciones de sistemas de tiempo real. Tipos de restricciones de tiempo. Características de los Sistemas de Tiempo real y su software. Procesamiento distribuido y sistemas de Tiempo Real. Aplicaciones de Sistemas Distribuidos de Tiempo Real. Impacto sobre los sistemas operativos, sobre el software en general y sobre el software distribuido en particular.

BIBLIOGRAFÍA

- Sistemas Distribuidos: Principios y Paradigmas, 2da. Ed., A. S. Tanenbaum, M. van Steen, Pearson Educación, México, 2008.
- Distributed Systems: Principles and Paradigms, 2nd Ed., A. S. Tanenbaum, M. van Steen, Prentice Hall, 2007.
- Distributed Systems: Principles and Paradigms, 2nd Ed., A. S. Tanenbaum, M. van Steen, 2017. Se puede descargar una copia personal, generada por los propios autores del libro desde el sitio <https://www.distributed-systems.net/index.php/books/distributed-systems-3rd-edition-2017/>
- Distributed Systems: Concepts and Design, 4th Ed., G. Coulouris, J. Dollimore, T. Kindberg, Addison Wesley, 2005.
- Distributed Computing: Principles and Applications, M. L. Liu, Addison-Wesley, 2004.

METODOLOGÍA DE ENSEÑANZA

La modalidad de enseñanza incluye:

- 1) Clases teóricas, normalmente guiadas a partir de diapositivas proyectadas y explicaciones de algunos detalles en pizarrón. Se presentan respuestas a los alumnos utilizando el pizarrón y en algunos casos ejemplos funcionando sobre un sistema distribuido para aclaración de conceptos.
- 2) Clases prácticas, mayormente para: a) presentación de los temas de los trabajos prácticos, y b) consultas de los trabajos prácticos.
- 3) La actividad de los trabajos prácticos se presenta a través de informes que se elaboran de manera grupal o individual y se evalúan de manera individual y oral.

EVALUACIÓN

La evaluación se lleva a cabo mediante exámenes parciales y final.

Los exámenes parciales consisten en la defensa de los informes correspondientes a los trabajos prácticos. Esta defensa es oral e individual a partir de los informes y de los programas que los alumnos desarrollan para la resolución de los trabajos prácticos.

El examen final es oral e incluye todos los temas presentados en la asignatura. Cuando los alumnos deciden llevar a cabo un trabajo final integrador, el examen se suele concentrar en la defensa del trabajo y la justificación de cada una de las decisiones tomadas en el mismo a partir de los conceptos vistos de la asignatura.

CGS2- Comunicarse con efectividad en forma oral y escrita.

Los alumnos deben presentar un trabajo escrito (individual/en comisión) que es evaluado teniendo en cuenta el contenido técnico, pero también la estructura, organización, sintaxis, claridad conceptual y la bibliografía consultada que debe ser citada rigurosamente. Esto se refleja en planillas escritas que conforman documentación de evaluación del trabajo.

CGT1- Identificar, formular y resolver problemas de Informática.

En la cátedra se pone énfasis en el proceso de identificación de problemas del mundo real, especificación de los mismos como problemas resolubles desde la informática y en el desarrollo de soluciones verificables para los mismos.

La evaluación de esta competencia forma parte de las evaluaciones de trabajos prácticos y examen final de la asignatura y se refleja en la corrección de las pruebas escritas del alumno.

CGT5- Utilizar de manera efectiva las técnicas y herramientas de aplicación de la Informática.

En la cátedra se pone énfasis en la capacidad del alumno para conocer técnicas y herramientas de aplicación en Informática (analizando y eligiendo las mejores características de las tendencias marcadas por el cambio tecnológico) y en la aplicación efectiva de las mismas.

CGT8 Capacidad de interpretación y resolución de problemas multidisciplinares, desde los conocimientos de la disciplina informática.

En la cátedra se tratan proyectos multidisciplinares, en muchas oportunidades interactuando con profesionales/especialistas de otras disciplinas y/o analizando el trabajo colaborativo con otras disciplinas. Se trata de acompañar al alumno en la interpretación del rol del Informático como articulador de soluciones en áreas de conocimiento muy diferentes que requieren participación de expertos extra-disciplinares.

La cátedra acompaña el aprendizaje del alumno (individual o según su rol en un equipo de proyecto) sobre esta competencia con materiales bibliográficos de casos de interés que resulten informativos y motivadores para el alumno.

La evaluación de esta competencia forma parte de las evaluaciones de trabajos prácticos y examen final de la asignatura (esta redacción puede tener matices según la asignatura) y se refleja en la corrección de las pruebas escritas (o coloquios) del alumno.

CRONOGRAMA DE CLASES Y EVALUACIONES

Clase/s	Fecha/s	Contenidos/Actividades
1 y 2	10/8/20 19/8/20	Conceptos introductorios. Motivación. Definiciones. Caracterización y problemas a resolver. Arquitecturas de procesamiento paralelo y su evolución a entornos/arquitecturas distribuidos. Redes de interconexión. Modelos de procesamiento distribuido. Distribución de datos y procesos. Ejemplos de utilización actual.
3	26/8/20	Consulta de TP
4, 5 y 6	2/9/20 9/9/20	Productores/Consumidores: relaciones con pipelines de procesamiento y filtros de los sistemas operativos. Cliente/servidor: conceptos, ejemplos, sistemas operativos distribuidos, relación con Internet, relación con threads. Interacción entre pares (peer-to-peer): relación con cómputo paralelo clásico, identificación de áreas de aplicación. Consulta de TP
7	16/9/20	Entrega/Evaluación de TP
8	23/9/20	Pasaje de mensajes asíncronos. Clientes y servidores. Interacción entre pares. Pasaje de mensajes sincrónicos, implementación de mensajes asíncronos con mensajes sincrónicos. Redes y sockets. Ejemplos con lenguajes y bibliotecas como C, MPI y Java. Consulta de TP
9	30/09/20	Concepto de RPC, extensión del modelo de ejecución procedural tradicional, relación con módulos de programas. Definición de rendezvous, ejemplos cliente/servidor e interacción entre pares. Ejemplos con lenguajes de programación como Ada y Java. Consulta de TP
10 y 11	7/10/20 14/10/20	Entrega/Evaluación de TP

12, 13 y 14	21/10/20 28/10/20 4/11/20	Bases de datos distribuidas: motivación, antecedentes, relación con programación distribuida, consultas y transacciones distribuidas, migración y replicación de datos. Modelo de cómputo/ aplicaciones N-tier. Migración de procesos en entornos distribuidos, algunas ideas implementadas en Condor. DSM (Distributed Shared Memory, memoria compartida físicamente distribuida): motivación, implementaciones. Otros modelos y lenguajes: Bag-of-Tasks (bolsa de tareas), Manager/worker (manejador/trabajador), Master/worker (maestro/trabajador), Algoritmos de heartbeat y/o sistólicos, Algoritmos de broadcast, Servidores múltiples.. Consulta de TP
15	11/11/20	Consulta de TP
16	18/10/20	Definiciones de sistemas de tiempo real. Tipos de restricciones de tiempo. Características de los Sistemas de Tiempo real y su software. Procesamiento distribuido y sistemas de Tiempo Real. Aplicaciones de Sistemas Distribuidos de Tiempo Real. Impacto sobre los sistemas operativos, sobre el software en general y sobre el software distribuido en particular.
17	25/11/20	Entrega/Evaluación de TP
18	2/12/20	Consultas para Reentregas

Evaluaciones previstas	Fecha
Definición de Reentregas	9/12/20
1er. Recuperatorio de Entregas	16/12/20
2do. Recuperatorio de Entregas	3/2/21

Contacto de la cátedra (mail, sitio WEB, plataforma virtual de gestión de cursos):

Se utilizará la plataforma ideas (que reemplazó a webunlp): <https://ideas.info.unlp.edu.ar/>. En esta plataforma se puede llevar a cabo toda la comunicación y manejo de material de la asignatura, desde la comunicación por correo electrónico, almacenamiento de material como apuntes de clases, enunciados de trabajos prácticos, entregas de material por parte de los estudiantes, etc. En caso de ser necesario, se acuerdan clases de consulta específicas utilizando el correo electrónico, con el cual se acuerdan días/horarios a tal fin.

Firma del/los profesor/es