

**UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA  
COORDINACIÓN DE FORMACIÓN BÁSICA  
COORDINACIÓN DE FORMACIÓN PROFESIONAL Y VINCULACIÓN UNIVERSITARIA  
PROGRAMA DE UNIDAD DE APRENDIZAJE**

**I. DATOS DE IDENTIFICACIÓN**

1. Unidad Académica: Facultad de Ciencias
2. Programa (s) de estudio: (Técnico, Licenciatura) Licenciatura en Ciencias Computacionales      3. Vigencia del plan:
4. Nombre de la Unidad de Aprendizaje: Sistemas Distribuidos      5. Clave: \_\_\_\_\_
6. HC: 2    HL 3    HT 0    HPC 0    HCL 0    HE 2    CR 7
7. Etapa de formación a la que pertenece: Terminal
8. Carácter de la Unidad de Aprendizaje:      Obligatoria x      Optativa \_\_\_\_\_
9. Requisitos para cursar la Unidad de Aprendizaje: Ninguna

Formuló: Dr. Ariel Arturo Quezada Pina

Vo. Bo. Dr. Alberto Leopoldo Morán y Solares

Fecha: Junio de 2016

Cargo: Subdirector de la Facultad de Ciencias

## **II. PROPÓSITO GENERAL DE LA UNIDAD DE APRENDIZAJE**

Esta asignatura le dará al estudiante las competencias que le permitan manipular de forma adecuada sistemas distribuidos. En esta asignatura se imparten las técnicas de diseño e implementación que permiten la construcción de servicios eficientes, escalables y seguros. Los tópicos incluidos son programación con consistencia de datos, tolerante a fallos, escalables y varios casos de estudio de sistemas distribuidos.

La asignatura se encuentra en la etapa terminal y es de carácter obligatorio.

## **III. COMPETENCIA DE DE LA UNIDAD DE APRENDIZAJE**

Formular soluciones eficientes a través de componentes y herramientas de software para la construcción de servicios y aplicaciones distribuidas con actitud crítica, propositiva y responsable.

## **IV. EVIDENCIAS DE DESEMPEÑO**

Desarrolla un proyecto final que incluya: el desarrollo de un sistema distribuido y el reporte. El reporte está constituido por las partes de un artículo científico que apliquen según el alcance del proyecto, determinado por el docente.

## V. DESARROLLO POR UNIDADES

### COMPETENCIA

Distinguir diferentes tipos de sistemas distribuidos para determinar el área de aplicación por medio del estudio de conceptos fundamentales de este tipo de sistemas con actitud crítica y responsable.

### CONTENIDO

**DURACIÓN 4 hr**

1. Introducción a los sistemas distribuidos.
  - 1.1 Definición de un sistema distribuido
  - 1.2 Objetivos
  - 1.3 Tipos de sistemas distribuidos

### COMPETENCIA

Discriminar las arquitecturas de sistemas distribuidos para identificar sus características principales por medio del estudio de conceptos fundamentales de este tipo de sistemas con actitud crítica y responsable.

### CONTENIDO

**DURACIÓN 4 hr**

2. Arquitecturas
  - 2.1 Modelos arquitectónicos
  - 2.2 Arquitecturas de sistemas
  - 2.3 Arquitecturas versus middleware
  - 2.4 Autoadministración en sistemas distribuidos

**COMPETENCIA**

Discriminar distintos tipos de procesos a través del análisis de distintos patrones para organizar eficientemente sistemas cliente-servidor con objetividad.

**CONTENIDO****DURACIÓN 4 hr**

## 3. Procesos

3.1 Hilos

3.2 Virtualización

3.3 Clientes

3.4 Servidores

3.5 Migración de código

**COMPETENCIA**

Identificar conceptos fundamentales de comunicación en un sistema distribuido para reconocer los mecanismos de intercambio de información llevado a cabo por distintos tipos de procesos con actitud crítica.

**CONTENIDO****DURACIÓN 4 hr**

## 4. Comunicación

4.1 Fundamentos

4.2 Llamadas a procedimientos remotos

4.3 Comunicación orientada a mensajes

4.4 Comunicación orientada a flujos

4.5 Comunicación por multitransmisión

**COMPETENCIA**

Analizar los métodos de nombrado a través de las técnicas y métodos para el descubrimiento de recursos en un sistema distribuido con objetividad.

**CONTENIDO****DURACIÓN 4 hr**

## 5. Nombres

5.1 Nombres, identificadores y direcciones

5.2 Nombres planos

5.3 Nombres estructurados

5.4 Nombres basados en atributos

**COMPETENCIA**

Organizar procesos para su coordinación en el tiempo a través del estudio de mecanismos de sincronización con actitud propositiva.

**CONTENIDO****DURACIÓN 4 hr**

## 6. Sincronización

6.1 Sincronización del reloj

6.2 Relojes lógicos

6.3 Exclusión mutua

6.4 Posicionamiento global de los nodos

6.5 Algoritmos de elección

**COMPETENCIA**

Determinar los mecanismos de manipulación de datos para mejorar su confiabilidad y disposición por medio de técnicas de replicación y consistencia con objetividad y actitud crítica.

**CONTENIDO****DURACIÓN 2 hr**

## 7. Consistencia y Replicación

7.1 Introducción

7.2 Modelos de consistencia centrada en los datos

7.3 Modelos de consistencia centrada en el cliente

7.4 Administración de Réplicas

7.5 Protocolos de consistencia

**COMPETENCIA**

Identificar diferentes tipos de fallas para la creación de sistemas robustos utilizando técnicas de estabilización en sistemas distribuidos con objetividad.

**CONTENIDO****DURACIÓN 2 hr**

## 8. Tolerancia a Fallas

8.1 Introducción

8.2 Atenuación de un proceso

8.3 Comunicación confiable entre cliente y servidor

8.4 Comunicación de grupo confiable

8.5 Realización distribuida

8.6 Recuperación

**COMPETENCIA**

Examinar los componentes principales de un paradigma específico de sistemas distribuidos para la implementación de un sistema de objetos distribuidos utilizando un software basado en objetos con actitud objetiva y propositiva.

**CONTENIDO****DURACIÓN 4 hr**

## 9. Sistemas Basados en Objetos Distribuidos

9.1 Arquitectura

9.2 Procesos

9.3 Comunicación

9.4 Asignación de nombres

9.5 Sincronización

9.6 Consistencia y replicación

9.7 Tolerancia a fallas

## VI. ESTRUCTURA DE LAS PRÁCTICAS

No. de Práctica	Competencia(s)	Descripción	Material de Apoyo	Duración
1	Experimentar un esquema de comunicación entre procesos para utilizar tecnologías de sockets empleando POSIX y Java con actitud objetiva y propositiva.	Probar distintas implementaciones para crear puertos de comunicación entre dos computadoras en red.	2 computadoras personales conectadas en red. Software de desarrollo.	4
2	Experimentar un esquema de comunicación entre procesos para utilizar el patrón de serialización empleando Java con actitud propositiva y responsable.	Probar distintas implementaciones para crear puertos de comunicación entre dos computadoras en red y utilizar el patrón de serialización para la transmisión de información en un sistema distribuido.	2 computadoras personales conectadas en red. Software de desarrollo.	6
3	Experimentar con un esquema de comunicación con hilos para atender distintas llamadas simultáneas a objetos en un ambiente distribuido utilizando Java con actitud objetiva y propositiva.	Probar distintas implementaciones para crear hilos de de procesos para atender peticiones de comunicación entre diferentes clientes y un servidor.	2 computadoras personales conectadas en red. Software de desarrollo.	6
4	Implementar un sistema	Utilizar distintas técnicas para lograr	2 computadoras	8

	distribuido para garantizar el uso eficiente de recursos compartidos por medio de componentes de sincronización con actitud propositiva y objetiva.	la sincronización de elementos en un sistema distribuido sencillo y comprobará cual es el mejor de ellos según el caso o problema que se le presente.	personales conectadas en red. Software de desarrollo.	
5	Implementar un sistema distribuido para verificar técnicas de consistencia de datos a través de componentes de replicación con actitud propositiva.	Desarrollar un sistema distribuido sencillo y comprobará la consistencia de datos al utilizar distintas formas de llevar a cabo replicación de información.	2 computadoras personales conectadas en red. Software de desarrollo.	8
6	Implementar un sistema distribuido para experimentar con distintos esquemas de estabilización empleando fallos con actitud crítica y responsable.	Desarrollar un sistema distribuido sencillo el cual debe contar con un conjunto de componentes de estabilización, le inducirá fallas con el objetivo de comprobar que el sistema sigue respondiendo adecuadamente.	2 computadoras personales conectadas en red. Software de desarrollo.	8
7	Implementar un sistema para utilizar objetos distribuidos e invocación remota empleando Java RMI con actitud científica y responsable.	Desarrollar un sistema distribuido completo utilizando exclusivamente la tecnología de Java RMI.	2 computadoras personales conectadas en red. Software de desarrollo.	8

## VII. METODOLOGÍA DE TRABAJO

### **El docente:**

- Expondrá en clase los temas del contenido de la unidad de aprendizaje.
- Facilitará material para revisión en horas extra clase.
- Solicitará la solución de problemas y ejercicios.
- Asignará temas de investigación y desarrollo de proyectos.

### **El alumno:**

- Atenderá a la clases para recibir instrucción en los temas del contenido de la unidad de aprendizaje.
- Revisará material asignado por el docente de forma extraclase los cuales pueden incluir sitios web, artículos de revistas y congresos.
- Resolverá problemas para desarrollar la capacidad analítica, el pensamiento crítico y actitud propositiva.
- Llevará a cabo ejercicios y prácticas de taller y laboratorio para reforzar los temas vistos en clase de forma teórica.
- Elaborará reportes con formato de artículo científico para formarse en la parte de divulgación académica.
- Elaborará una presentación por cada proyecto elaborado. Los proyectos serán llevados a cabo de forma individual o en equipo, según el alcance de los mismos.

## VIII. CRITERIOS DE EVALUACIÓN

### Porcentajes que componen la calificación

- 10% Reportes de soluciones de problemas y ejercicios asignados por el docente.
- 10% Reportes de las prácticas de laboratorio.
- 60% Exámenes
- 10% Reporte de proyecto final
- 10% Presentación de proyecto final

### Requisitos de acreditación de acuerdo con el estatuto escolar vigente

Para la acreditación de la unidad de aprendizaje se atenderá al Estatuto Escolar Vigente, artículos 70-71, por lo que el estudiante deberá contar un mínimo de 80% de asistencias en el periodo. Tener un mínimo aprobatorio de 60 en su calificación final.

### Procesos, instrumentos y productos

Los exámenes serán elaborados en base al material teórico impartido en clase. Su número y extensión será determinado por el docente.

Los reportes de problemas y ejercicios serán entregados en tiempo y forma, determinados por el docente.

El proyecto final será elaborado en equipo, del cual llevarán a cabo una presentación final y un reporte con formato de artículo científico.

## IX. BIBLIOGRAFÍA

### Básica

- George Coulouris and Jean Dollimore. Distributed Systems: Concepts and Design. 2011
- Tanenbaum, Van Steen. Distributed Systems, Principles and Paradigms. 2006 [CLÁSICO]

### Complementaria

- IEEE Transactions on Parallel and Distributed Systems.  
<http://ieeexplore.ieee.org/servlet/opac?punumber=71>
- Orfali, Robert. Cliente/servidor y Objetos. 2002IEEE Transactions on Parallel and Distributed Systems.  
<http://ieeexplore.ieee.org/servlet/opac?punumber=71>
- Saltzer, Jerome H.; M. Frans Kaashoek. Principles of Computer System Design: An Introduction. 2009 [CLÁSICO]
- Woungang, Isaac; Misra, Subhas Chandra; Misra, Sudip. Selected Topics in Communication Networks and Distributed Systems. 2010

## X. PERFIL DEL DOCENTE

Profesionista con licenciatura en el área de ciencias computacionales, experiencia docente y con conocimientos en el desarrollo de sistemas distribuidos de cómputo.