

SILABO

FACULTAD
ESCUELA PROFESIONAL
PROGRAMA DE ESTUDIOS

INGENIERIA ESTADISTICA E INFORMATICA
INGENIERIA ESTADISTICA E INFORMATICA
CARRERA PURA

I. INFORMACIÓN GENERAL

1.1 Identificación Académica

a) Curso	SISTEMAS DISTRIBUIDOS
b) Código	EST318
c) Prerequisito	EST316 - ARQUITECTURA DE REDES Y PROTOCOLOS
d) Número de Horas	03h teóricas, 02h prácticas, Total 05 horas
e) Créditos	04
f) Número de Horas virtuales	00
g) Año y Semestre Académico	2025-II
h) Ciclo de Estudios	VII
i) Duración	Del 25 de Agosto al 19 de Diciembre del 2025 (17 semanas)
j) Área Curricular	Estudios de especialidad
k) Características del Curso	Investigación, desarrollo e innovación (I+D+i)

1.2 Docente

a) Apellidos y Nombres	CAUNA HUANCA GERARDINO JUVENAL
b) Condición y Categoría	Contratado B3
c) Especialidad	Ing. Estadístico e informático, Maestría en Informática.
a) Apellidos y Nombres	HERRERA URTIAGA ALAIN PAUL
b) Condición y Categoría	Contratado B2
c) Especialidad	Ing. Estadístico e informático, Maestría en Informática.

1.3 Ambiente donde se realizó el aprendizaje

- a) 102

II. SUMILLA

El curso de sistemas distribuidos corresponde al área de estudios de especialidad y es de característica teórico práctica, su propósito es integrar los diversos aspectos de los sistemas operativos en su expresión más amplia de interconexión y gestión de redes de computadores y que conforman el sistema distribuido. El curso está dividido en dos unidades:

Unidad I : Definición de sistemas distribuidos y Modelos de sistemas distribuidos.

Unidad II: Sistemas distribuidos en nube y microservicios

III. PERFIL DEL EGRESADO EN RELACIÓN AL CURSO

CE4. Reconoce la lógica de los procesos estadísticos en la investigación científica multidisciplinaria para proponer soluciones innovadoras en diferentes contextos.

IV. LOGRO DE APRENDIZAJE DEL CURSO

Explica las técnicas usadas para la fragmentación, replicación y asignación de datos, durante el proceso de diseño de una base de datos distribuida.

V. TRATAMIENTO DE UNIDADES DIDÁCTICAS

UNIDAD 1		UNIDAD 1
LOGROS DE APRENDIZAJE DE LA UNIDAD Explica las arquitecturas paralelas		
TIEMPO DE DESARROLLO		Del 25 de Agosto al 20 de Octubre del 2025 (Total 40 horas)
HORAS DE ENSEÑANZA VIRTUAL/UNIDAD		00
SEMANAS	CRITERIOS DE DESEMPEÑO	CONOCIMIENTOS
Semana 1	Define correctamente los conceptos básicos, características y desafíos de los sistemas distribuidos y presenta un análisis de casos reales de implementación, demostrando comprensión de la evolución histórica de los sistemas distribuidos.	Introducción a los Sistemas Distribuidos - Definición, características y desafíos de los sistemas distribuidos. - Historia y evolución.

Semana 2	Describe los diferentes modelos arquitectónicos de sistemas distribuidos y diseña una arquitectura cliente-servidor básica utilizando sockets, evidenciando la aplicación de conceptos teóricos en la práctica.	Modelos de Sistemas Distribuidos - Arquitecturas cliente-servidor y peer-to-peer. - Sistemas distribuidos a gran escala: grids y clouds.
Semana 3	Explica los protocolos de comunicación y el rol del middleware en sistemas distribuidos, implementando una aplicación cliente-servidor con RPC o RMI que funcione correctamente.	Comunicación en Sistemas Distribuidos - Protocolos de comunicación. - Middleware y servicios de comunicación.
Semana 4	Analiza la consistencia de datos y describe los algoritmos de coordinación, implementando un algoritmo de consenso básico que gestione transacciones distribuidas de manera efectiva.	Coordinación y Consistencia - Consistencia de datos. - Algoritmos de coordinación: 2PC, Paxos, Raft.
Semana 5	Identifica amenazas y vulnerabilidades comunes en sistemas distribuidos y configura mecanismos de autenticación SSL/TLS en un sistema distribuido, demostrando medidas de control de acceso efectivas.	Seguridad en Sistemas Distribuidos - Amenazas y vulnerabilidades. - Mecanismos de autenticación y control de acceso.
Semana 6	Explica las estrategias de recuperación, replicación y particionamiento en sistemas distribuidos, y simula escenarios de fallo y recuperación en un sistema replicado, evaluando su eficacia.	Tolerancia a Fallos - Estrategias de recuperación. - Replicación y particionamiento.
Semana 7	Identifica técnicas de diseño escalable y evalúa el rendimiento de un sistema distribuido bajo diferentes cargas, aplicando pruebas de carga y análisis de optimización.	Escalabilidad y Rendimiento - Diseño escalable y técnicas de optimización. - Evaluación de rendimiento y prueba de carga.
Semana 8	Demuestra dominio de los conceptos teóricos y prácticos mediante la resolución de problemas y una evaluación que incluye aspectos clave de los contenidos vistos en la unidad.	Revisión y Evaluación Parcial - Repaso de los contenidos vistos. - Evaluación teórica y práctica de los conocimientos adquiridos.
PORCENTAJE DE AVANCE ACADÉMICO DE LA UNIDAD: 50%		

* El docente deberá programar “actividades de retroalimentación” los cuales se desarrollarán a lo largo de cada unidad didáctica con el fin de asegurar los adecuados aprendizajes de los estudiantes.

UNIDAD 2		UNIDAD 2
LOGROS DE APRENDIZAJE DE LA UNIDAD		
Explica los fundamentos de la computación paralela y distribuida		
TIEMPO DE DESARROLLO		Del 20 de Octubre al 19 de Diciembre del 2025 (Total 45 horas)
HORAS DE ENSEÑANZA VIRTUAL/UNIDAD		00
SEMANAS	CRITERIOS DE DESEMPEÑO	CONOCIMIENTOS
Semana 9	El estudiante instala y configura su entorno de trabajo, verificando que pueda ejecutar un servidor básico y realizar pruebas locales.	Preparación del entorno de desarrollo - Instalación y Configuración (IDE, dependencias, Postman/Insomnia). - Prueba de ejecución de un “Hello World” en un servidor local.
Semana 10	Diseña y expone una API REST básica con endpoints simples (ejemplo: CRUD de usuarios).	Comunicación entre procesos: APIs REST (Parte I) - Principios de REST. - Endpoints y verbos HTTP (GET, POST, PUT, DELETE). - Consumo de API desde un cliente simple (Postman o fetch/Axios).

Semana 11	Implementa autenticación simple (token/JWT) en una API REST y gestiona peticiones concurrentes.	Comunicación entre procesos: APIs REST (Parte II) • Autenticación y seguridad básica en APIs. • Manejo de concurrencia en servicios distribuidos. • Ejemplo práctico: mini servicio distribuido con login y operaciones protegidas.
Semana 12	El estudiante implementa un canal de comunicación en tiempo real utilizando WebSockets en un sistema cliente-servidor básico.	Comunicación en tiempo real: WebSockets • Concepto de comunicación full-duplex. • Diferencias entre HTTP polling, SSE y WebSockets. • Ejemplo práctico: chat o notificaciones en tiempo real.
Semana 13	El estudiante simula un sistema distribuido con replicación de datos y analiza cómo manejar fallos simples.	Tolerancia a fallos y replicación básica • Conceptos de replicación y redundancia. • Estrategias básicas de tolerancia a fallos. • Práctica: replicar datos en dos nodos simulados.
Semana 14	El estudiante configura un balanceador de carga simple y evalúa cómo afecta al rendimiento de una aplicación distribuida.	Escalabilidad y balanceo de carga • Conceptos de escalabilidad horizontal vs vertical. • Balanceo de carga (round-robin). • Práctica: balanceo en aplicaciones distribuidas pequeñas.
Semana 15	Desarrolla dos servicios independientes que se comunican vía API REST y los integra en una pequeña arquitectura distribuida.	Integración de servicios (Microservicios básicos) • Concepto de microservicios. • Comunicación entre servicios vía REST. • Ejemplo práctico: servicio de usuarios y servicio de pedidos.
Semana 16	Desarrolla en equipo un proyecto simple de sistema distribuido (ejemplo: chat, sistema de reservas, mini e-commerce) aplicando APIs REST y comunicación en tiempo real.	Proyecto Práctico de Implementación • Integración de API REST + WebSockets. • Uso de servicios en la nube. • Buenas prácticas de documentación y pruebas.
Semana 17	El estudiante presenta y defiende el proyecto final, demostrando competencias en los temas del curso a través de una evaluación práctica, articulando conocimientos teóricos y habilidades prácticas en sistemas distribuidos.	Evaluación y Examen Final • Presentación del proyecto final. • Examen práctico de los temas del curso.
PORCENTAJE DE AVANCE ACADÉMICO DE LA UNIDAD: 50%		

* El docente deberá programar “actividades de retroalimentación” los cuales se desarrollarán a lo largo de cada unidad didáctica con el fin de asegurar los adecuados aprendizajes de los estudiantes.

VI. ESTRATEGIAS METODOLÓGICAS

6.1 De Enseñanza

Clases Magistrales y Expositivas: Presentación de los conceptos teóricos fundamentales mediante clases interactivas, apoyadas en recursos audiovisuales, ejemplos prácticos y discusiones dirigidas.

Estudio de Casos: Análisis y discusión de casos reales de implementación de sistemas distribuidos en empresas y organizaciones para contextualizar los conceptos teóricos en situaciones reales.

Aprendizaje Basado en Proyectos (ABP): Los estudiantes desarrollan proyectos prácticos que integran conocimientos de distintas áreas del curso, favoreciendo la aplicación directa de los conceptos en problemas reales.

Demostraciones Prácticas y Laboratorios: Ejercicios guiados donde el docente muestra la implementación de aplicaciones distribuidas, fomentando el aprendizaje práctico y la experimentación.

6.2 De Aprendizaje

Aprendizaje Colaborativo: Fomento del trabajo en equipo para resolver problemas complejos, donde los estudiantes intercambian ideas y soluciones, potenciando la colaboración y el aprendizaje entre pares.

Resolución de Problemas: Actividades que desafían a los estudiantes a aplicar sus conocimientos para solucionar problemas técnicos específicos en sistemas distribuidos.

Uso de Plataformas y Simuladores: Aplicación de herramientas y simuladores de sistemas distribuidos para que los estudiantes puedan experimentar con configuraciones y escenarios reales.

Autoevaluación y Retroalimentación: Uso de cuestionarios y ejercicios de autoevaluación para que los estudiantes identifiquen áreas de mejora, reforzados con retroalimentación personalizada del docente.

6.3 De Investigación Formativa

Investigación Aplicada: Desarrollo de pequeñas investigaciones que permitan a los estudiantes explorar nuevas tecnologías y tendencias en sistemas distribuidos, como blockchain, edge computing o IA distribuida.

Revisión Bibliográfica: Análisis de artículos científicos y publicaciones recientes para mantener a los estudiantes actualizados en avances tecnológicos y metodologías emergentes en el campo de los sistemas distribuidos.

Proyectos de Innovación: Fomento de proyectos que incluyan la creación de soluciones innovadoras, donde los estudiantes integren conocimientos adquiridos con la exploración de nuevas ideas y tecnologías.

6.4 De Responsabilidad Social Universitaria

Proyectos con Impacto Social: Desarrollo de aplicaciones distribuidas que puedan tener un impacto positivo en la comunidad, como sistemas de monitoreo ambiental, gestión de recursos comunitarios o plataformas educativas distribuidas.

Ética y Sostenibilidad: Integración de discusiones sobre la ética en la administración de datos y la sostenibilidad de los sistemas distribuidos, promoviendo la conciencia social en el desarrollo tecnológico.

Actividades de Vinculación: Colaboración con organizaciones externas o proyectos comunitarios para aplicar los conocimientos del curso en contextos reales, beneficiando a la sociedad.

6.5 De Enseñanza Virtual

Aulas Virtuales y Plataformas de Aprendizaje: Uso de plataformas como aulas virtuales de la universidad, para organizar y distribuir materiales de estudio, realizar actividades interactivas y mantener la comunicación continua.

Clases Sincrónicas y Asincrónicas: Combinación de clases en vivo y sesiones grabadas para asegurar que los estudiantes tengan acceso al contenido en su propio ritmo y tiempo.

Recursos Multimedia: Uso de videos, tutoriales, simulaciones y demostraciones virtuales para complementar la enseñanza teórica y práctica.

VII. MEDIOS Y MATERIALES DIDÁCTICOS

1. Medios Didácticos:

- **Presentaciones Multimedia:** Uso de diapositivas interactivas con gráficos, diagramas y videos explicativos para facilitar la comprensión de conceptos complejos.
- **Videos Educativos:** Recursos audiovisuales que incluyen tutoriales, demostraciones y casos de estudio relacionados con sistemas distribuidos, accesibles a través de plataformas como YouTube, Coursera o canales especializados.
- **Simuladores y Entornos Virtuales:** Herramientas como Docker, Kubernetes, y plataformas de simulación que permiten a los estudiantes experimentar con configuraciones de sistemas distribuidos en un entorno seguro.
- **Aulas Virtuales y Plataformas de Aprendizaje:** Utilización de plataformas como aulas virtuales de la universidad para la gestión de contenido, asignaciones, exámenes y comunicación constante con los estudiantes.

2. Materiales Didácticos:

- **Guías de Laboratorio y Ejercicios Prácticos:** Documentos que detallan paso a paso las prácticas a realizar en cada sesión, incluyendo ejemplos de código y configuraciones específicas para experimentar con diferentes tecnologías distribuidas.
- **Manuales y Documentación Técnica:** Materiales de referencia que incluyen documentación de software, manuales de uso de herramientas como Hadoop, Spark, MQTT, y guías de configuración de sistemas distribuidos.
- **Libros y Artículos Especializados:** Textos recomendados que cubren tanto los fundamentos teóricos como las aplicaciones prácticas de los sistemas distribuidos, incluyendo títulos clásicos y publicaciones recientes.
- **Foros de Discusión y Comunidades en Línea:** Acceso a foros especializados y comunidades online como Stack Overflow, GitHub, y foros de desarrolladores que ofrecen soporte adicional y espacios para la discusión de problemas técnicos.

3. Recursos Digitales:

- **Repositorios de Código:** Acceso a repositorios en GitHub o GitLab con ejemplos de proyectos y código fuente para que los estudiantes puedan explorar y modificar según sus necesidades.
- **Software de Desarrollo y Pruebas:** Herramientas como Python, Java, IDEs (Eclipse, IntelliJ), plataformas de integración continua y software de virtualización como VirtualBox o VMware para realizar pruebas y desarrollos prácticos.
- **Bases de Datos en la Nube:** Acceso a servicios de bases de datos distribuidas como Amazon RDS, Google Cloud SQL o MongoDB Atlas para prácticas de gestión y administración en entornos distribuidos.

4. Materiales Interactivos:

- **Cuestionarios y Autoevaluaciones en Línea:** Herramientas interactivas para que los estudiantes evalúen su comprensión de los temas tratados y reciban retroalimentación inmediata sobre su desempeño.
- **Mapas Conceptuales y Diagramas de Flujo:** Diagramas visuales que ayudan a organizar y entender los conceptos clave, los flujos de datos y las interacciones entre los componentes de sistemas distribuidos.

VIII. EVALUACIÓN DEL APRENDIZAJE

8.1 Logro de aprendizaje, evidencias de desempeño, ponderación, técnicas e instrumentos de evaluación.

UNIDAD	LOGROS DE APRENDIZAJE	EVIDENCIAS DESEMPEÑO: De acción, objeto o producto (%)	PONDERACIÓN (Obligatorio en base a 100%)	TÉCNICAS	INSTRUMENTOS
1	Explica las arquitecturas paralelas	Realiza un proyecto de despliegue en una arquitectura distribuida.	50%	Resolución de caso real, exposición, desarrollo de proyecto	Proyecto, informe, práctica calificada

2	Explica los fundamentos de la computación paralela y distribuida	Desarrollo de Aplicaciones distribuidas	50%	Resolución de casos reales, desarrollo de soluciones	Aplicaciones distribuidas, informe, practica calificada, proyecto final
---	--	---	-----	--	---

8.2 Evidencias de aprendizaje del semestre académico.

LOGRO DE APRENDIZAJE DEL CURSO	EVIDENCIAS: ACCIÓN, OBJETO o PRODUCTO	FECHA DE PRESENTACIÓN
Explica las técnicas usadas para la fragmentación, replicación y asignación de datos, durante el proceso de diseño de una base de datos distribuida.	Diseña y Construye aplicaciones distribuidas	Última semana del semestre académico

• Evidencias que deben ser socializados con la comunidad como parte de la responsabilidad social universitaria. Por la emergencia sanitaria podría optarse por la difusión vía página web y redes sociales.

8.3 Calificación

La fórmula para la obtención del promedio final del curso es la siguiente:

$$\text{Promedio Final} = (50\%)IUPP + (50\%)IIUPP$$

Donde:

IUPP : Primero unidad promedio parcial

IIUPP : Segundo unidad promedio parcial

La fórmula para la obtención del promedio final del curso es la siguiente:

$$\text{Promedio Final} = (50\%)IUPP + (50\%)IIUPP$$

Donde:

IUPP : Primero unidad promedio parcial IIUPP : Segundo unidad promedio parcial

IX. FUENTES DE INFORMACIÓN

9.1 Bibliográficas

Básica

Coulouris, G., Dollimore, J., Kindberg, T., & Blair, G. (2011). Distributed Systems: Concepts and Design (5th ed.). Pearson Education.
Tanenbaum, A. S., & Van Steen, M. (2016). Distributed Systems: Principles and Paradigms (3rd ed.). Pearson Prentice Hall.
Kleppmann, M. (2017). Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems. O'Reilly Media.
Comer, D. E. (2015). Internetworking with TCP/IP: Principles, Protocols, and Architecture (6th ed.). Pearson.
Coulouris, G., & Blair, G. (2011). Distributed Systems Architecture: A Middleware Approach. Pearson.

Complementarias

Chandra, R., Gupta, R., & Singh, S. (2013). Cloud Computing: A Practical Approach for Learning and Implementation. Tata McGraw-Hill.
Erl, T., Puttini, R., & Mahmood, Z. (2013). Cloud Computing: Concepts, Technology & Architecture. Prentice Hall.
Haddad, M. (2021). Hands-On Microservices with Kubernetes: Build, Deploy, and Scale Applications Using Kubernetes (2nd ed.). Packt Publishing.
Shvachko, K., Kuang, H., Radia, S., & Chansler, R. (2010). The Hadoop Distributed File System. IEEE.
Andrews, G. R. (2000). Foundations of Multithreaded, Parallel, and Distributed Programming. Addison-Wesley.

Electrónicas

IEEE Xplore Digital Library. (<https://ieeexplore.ieee.org>)
ACM Digital Library. (<https://dl.acm.org>)
GitHub. (<https://github.com>)
Coursera y edX. (<https://www.coursera.org>) y (<https://www.edx.org>)
Stack Overflow. (<https://stackoverflow.com>)

Producción intelectual del docente relacionado con el curso

Materiales y guías elaborados por el docente

Puno, Diciembre del 2025