

PROGRAMA DE LABORATORIO

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA
ESCUELA DE CIENCIAS Y SISTEMAS

**LABORATORIO SISTEMAS OPERATIVOS 1 - SECCIÓN P**

CÓDIGO:	281	PUNTEO NETO LABORATORIO:	P
ESCUELA DE INGENIERÍA EN:	CIENCIAS Y SISTEMAS	ÁREA A LA QUE PERTENECE:	CIENCIAS DE LA COMPUTACIÓN
PRE REQUISITO:	Org, Lenguajes y Compiladores 2 (781), Arq, Compu y Ensambladores 1 (778)	POST REQUISITO:	Sistemas Operativos 2 (285), Sistema de Bases de Datos 2 (775), Seminario de Sistemas 1 (797)
CATEGORÍA:	OBLIGATORIO	VIGENCIA:	PRIMER SEMESTRE 2026

Descripción del Laboratorio

El laboratorio del curso 'Sistemas Operativos 1' está diseñado para complementar la teoría impartida en clase, mediante la aplicación práctica de los conceptos fundamentales del kernel, gestión de procesos y recursos del sistema. Su propósito es fomentar el aprendizaje activo mediante la experimentación directa con componentes del sistema operativo, desarrollar habilidades para la resolución de problemas de bajo nivel usando llamadas al sistema, y comprender la interacción entre hardware y software mediante el uso de herramientas como compiladores GCC, depuradores GDB y comandos esenciales de Bash.

Resumen de Ponderaciones y Tiempo de Auto-aprendizaje

TIPO	PONDERACIÓN	HORAS DE AUTO-APRENDIZAJE
Proyectos	80	79
Cortos	5	1
Examen Final	15	0
TOTAL	100	80

Equipo Académico

Coordinador del Área


Nombre: M.Sc. Luis Fernando Espino Barrios	Correo electrónico: usac.sistemas@gmail.com
---	--

Docente

Sergio Arnalado Méndez Aguilar	Correo electrónico del Docente
--------------------------------	--------------------------------

	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Día	X		X			
Horario	17:20 - 19:00		17:20 - 19:00			
Lugar	Virtual		Virtual			

Tutor(es)

Nombre del Tutor	Edgar Rolando Alvarez Rodriguez	
Correo electrónico institucional	3014174660101@ingenieria.usac.edu.gt	

Nombre del Tutor	José Daniel Lorenzana Medina	
Correo electrónico institucional	3718164320101@ingenieria.usac.edu.gt	

Tipo		Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Clase	Día				X		
	Horario				17:20 - 19:00		
	Lugar				Virtual		
Atención al Estudiante	Día				X		
	Horario				19:00 - 19:30		
	Lugar				Virtual		

Índice

Descripción del Laboratorio.....	1
Resumen de Ponderaciones y Tiempo de Auto-aprendizaje.....	1
Equipo Académico.....	2
Coordinador del Área.....	2
Docente.....	2
Tutor(es).....	2
Competencias Vinculadas al Perfil del Egresado.....	4
Competencias Específicas.....	4
Competencias Generales.....	4
Competencias del Laboratorio.....	4
Competencia(s) Específica(s).....	4
Competencia(s) General(es).....	4
Diseño Didáctico.....	5
Sesión de Diagnóstico.....	5
Sesión No. 1, Unidad No. 1 - Introducción al sistema operativo Linux.....	6
Sesión No. 2, Unidad No. 1 - Bash Conceptos Básicos.....	7
Sesión No. 3, Unidad No. 1 - Scripts intermedios en Bash e Inicios en Docker.....	8
Sesión No. 4, Unidad No. 1 - Virtualización ligera con contenedores Docker.....	9
Sesión No. 5, Unidad No. 1 - Fundamentos del lenguaje C para módulos de kernel y uso con contenedores de bajo y alto nivel.....	10
Sesión No. 6, Unidad No. 1- C avanzado y llamadas al sistema con rust.....	11
Sesión No. 7, Unidad No. 2 - Kubernetes en la nube con Google Cloud Platform.....	11
Sesión No. 8, Unidad No. - Introducción a Kubernetes.....	12
Sesión No. 9, Unidad No. 2 - Gestión de recursos y configuración en Kubernetes.....	13
Sesión No. 10, Unidad No. 2 - Escalado automático y monitoreo de salud en Kubernetes.....	14
Sesión No. 11, Unidad No. 2 - Locust para carga de prueba en Kubernetes.....	15
Rúbrica de Evaluación.....	16
Normativa Académica y Ética del Curso.....	16
Bibliografía.....	17
E-Grafía.....	17

Competencias Vinculadas al Perfil del Egresado

Competencias Específicas

No.	Competencia
1	Diseño e Implementación de Soluciones a Nivel de Sistema
2	Resolución de Problemas de Bajo Nivel
3	Administración de diversos tipos de Softwares

Competencias Generales

No.	Competencia
1	Pensamiento Computacional y Resolución de Problemas
2	Programación y Desarrollo de Software
3	Administración de Sistemas

Competencias del Laboratorio

Competencia(s) Específica(s)

No.	Competencia	Nivel de Aprendizaje
1	Desarrollar módulos básicos del kernel en Linux	Aplicación
2	Gestionar procesos, memoria y cpu	Análisis
3	Implementar scripts de automatización	Aplicación
4	Configurar clusters básicos con Kubernetes	Análisis
5	Crear y gestionar contenedores Docker	Aplicación

Competencia(s) General(es)

No.	Competencia	Nivel de Aprendizaje
1	Gestionar tecnologías de virtualización y contenedores	Aplicación
2	Desarrollar software de sistema	Aplicación

Diseño Didáctico

Sesión de Diagnóstico

Evaluación de conocimientos previos

Se aplicará una actividad diagnóstica con el objetivo de identificar el nivel de conocimientos y habilidades que los estudiantes poseen al inicio del curso. No influye en la nota final, pero es obligatoria para todos los estudiantes.

Tipo de Actividad	Descripción
Cuestionario de Conocimientos por medio de Wayground	Identificar el nivel de conocimientos previos en programación en Linux conceptos básicos de sistemas operativos, contenedores y temas de cursos prerequisites, para ajustar el ritmo de las sesiones prácticas.

Presentación del tutor

El tutor se presenta formalmente al grupo, compartiendo su formación académica, experiencia profesional y educativa, así como sus expectativas sobre el curso. También se abordan aspectos como normas de convivencia, canales de comunicación, disponibilidad para consultas y métodos de acompañamiento.

Presentación de los estudiantes

Se escogen un grupo de estudiantes al azar. En su presentación, se les pedirá que compartan información básica como su nombre, intereses personales o profesionales, experiencias previas relacionadas con el curso y sus expectativas. Esta actividad busca promover la interacción, el reconocimiento entre pares y la construcción de un entorno participativo y respetuoso.

Presentación del programa del curso

Se presenta el contenido del programa del curso, se aclaran dudas y se fomenta el compromiso del estudiante con su aprendizaje.

Evaluación de conocimientos del laboratorio actual

Se realiza una evaluación o práctica que permite conocer el grado de familiaridad de los estudiantes con las herramientas, entornos o competencias técnicas necesarias para el laboratorio actual.

Tipo de Actividad	Descripción
Cuestionario de Conocimientos actuales por medio de Wayground	Identificar el nivel de conocimientos que los estudiantes poseen sobre los temas que se impartirán en el laboratorio para saber en cuáles hacer más énfasis.

Sesión No. 1, Unidad No. 1 - Introducción al sistema operativo Linux

Área Actitudinal (Saber ser)

Nombre del valor: Perseverancia
Aprender Linux puede ser desafiante; promover la práctica constante y no rendirse ante errores transmite este valor fundamental.

Área de Conocimiento (Saber)

Competencia(s)	
El estudiante diseña e implementa redes de computadoras aplicando principios de enrutamiento, segmentación de redes, redundancia y gestión de servicios de red para garantizar la disponibilidad, eficiencia y seguridad de los servicios de red en entornos empresariales	
El estudiante administra interfaces y configuraciones de seguridad en dispositivos de red mediante el uso de comandos, técnicas y buenas prácticas para asegurar el correcto funcionamiento y la seguridad de la red	
Tema	Subtema
Sistema de archivos en Linux	Jerarquía de directorios, rutas absolutas y relativas
Permisos de archivos	Lectura, escritura, ejecución, chmod, chown
Procesos y gestión	ps, top, kill, prioridades
Usuarios y grupos	adduser, passwd, groups, sudo

Área de Habilidades (Saber Hacer)

Competencia	
Navega y gestiona el sistema de archivos utilizando comandos de terminal.	
Tipo de Actividad	Ponderación
Práctica en laboratorio	0

Sesión No. 2, Unidad No. 1 - Bash Conceptos Básicos

Área Actitudinal (Saber ser)

Nombre del valor: Perseverancia
Aprender Linux puede ser desafiante; promover la práctica constante y no rendirse ante errores transmite este valor fundamental.

Área de Conocimiento (Saber)

Competencia(s)	
Comprende el entorno de línea de comandos en Linux y la estructura de scripts Bash.	
Tema	Subtema
Introducción a Bash	Historia, uso y ventajas del shell Bash
Comandos básicos de Linux	Navegación, manipulación de archivos
Estructura de scripts en Bash	Sintaxis, comentarios, permisos de ejecución
Variables y operadores en Bash	Tipos de variables, operadores aritméticos

Área de Habilidades (Saber Hacer)

Competencia	
Crea scripts Bash simples para automatizar tareas básicas en Linux.	
Tipo de Actividad	Ponderación
Práctica en laboratorio, Proyecto 1	0, 15

Sesión No. 3, Unidad No. 1 - Scripts intermedios en Bash e Inicios en Docker

Área Actitudinal (Saber ser)

Nombre del valor: Perseverancia
Aprender Linux puede ser desafiante; promover la práctica constante y no rendirse ante errores transmite este valor fundamental.

Área de Conocimiento (Saber)

Competencia(s)	
Uso de estructuras de control, condicionales y bucles para crear scripts más avanzados en Bash.	
Comprender el funcionamiento de contenedores y Docker	
Tema	Subtema
Condicionales en Bash	if, else, elif, case
Bucles en Bash	for, while, until
Funciones en Bash	Declaración y uso de funciones en scripts
Entrada y salida de datos	read, argumentos (\$1, \$2...), redirección y pipes
Docker	instalación de docker y extras

Área de Habilidades (Saber Hacer)

Competencia	
Desarrolla scripts con estructuras de control de flujo para automatizar tareas.	
Tipo de Actividad	Ponderación
Práctica en laboratorio	0

Sesión No. 4, Unidad No. 1 - Virtualización ligera con contenedores Docker

Área Actitudinal (Saber ser)

Nombre del valor: Perseverancia
Aprender Linux puede ser desafiante; promover la práctica constante y no rendirse ante errores transmite este valor fundamental.

Área de Conocimiento (Saber)

Competencia(s)	
Introducción al uso de Docker para crear, gestionar y desplegar contenedores de forma eficiente.	
Tema	Subtema
Contenedores vs máquinas virtuales	Diferencias, ventajas y desventajas
Arquitectura de Docker	Cliente, demonio, imágenes, contenedores, DockerHub
Imágenes y contenedores	docker pull, docker run, docker ps, docker stop
Dockerfile y construcción	Sintaxis del Dockerfile, docker build, docker exec

Área de Habilidades (Saber Hacer)

Competencia	
Comprender la virtualización y funcionamiento de Docker	
Tipo de Actividad	Ponderación
Práctica Única	2

Sesión No. 5, Unidad No. 1 - Fundamentos del lenguaje C para módulos de kernel y uso con contenedores de bajo y alto nivel

Área Actitudinal (Saber ser)

Nombre del valor: Perseverancia
Aprender C puede ser desafiante; promover la práctica constante y no rendirse ante errores transmite este valor fundamental.

Área de Conocimiento (Saber)

Competencia(s)	
Comprende los fundamentos del lenguaje C para desarrollar programas simples.	
Tema	Subtema
Sintaxis básica de C	Declaraciones, funciones, compilación con gcc
Tipos de datos y operadores	int, float, char, operadores aritméticos y lógicos
Entrada/Salida estándar	printf(), scanf()
Estructuras de control	if, else, switch, for, while, do while

Área de Habilidades (Saber Hacer)

Competencia	
Crea programas básicos en C que utilicen entrada/salida y estructuras de control.	
Tipo de Actividad	Ponderación
Proyecto 2	15

Sesión No. 6, Unidad No. 1- C avanzado y llamadas al sistema con rust

Área Actitudinal (Saber ser)

Nombre del valor: Perseverancia
Aprender C puede ser desafiante; promover la práctica constante y no rendirse ante errores transmite este valor fundamental.

Área de Conocimiento (Saber)

Competencia(s)	
Comprende el manejo de memoria, punteros y la interacción directa con el sistema operativo mediante llamadas al sistema.	
Tema	Subtema
Punteros	Declaración, punteros a variables, punteros a funciones
Manejo dinámico de memoria	malloc, calloc, free
Archivos en C y Rust	fopen, fread, fwrite, fclose
Llamadas al sistema	fork, exec, wait, exit, getpid, kill

Área de Habilidades (Saber Hacer)

Competencia	
Utiliza llamadas al sistema para crear procesos y gestionar su ejecución.	
Tipo de Actividad	Ponderación
Práctica en el laboratorio	0

Sesión No. 7, Unidad No. 2 - Kubernetes en la nube con Google Cloud Platform

Área Actitudinal (Saber ser)

Nombre del valor: Perseverancia
Aprender Kubernetes puede ser desafiante; promover la práctica constante y no rendirse ante errores transmite este valor fundamental.

Área de Conocimiento (Saber)

Competencia(s)	
Comprende el proceso de creación, configuración y gestión de un clúster Kubernetes en GCP.	
Tema	Subtema
Introducción a GCP	Activación de proyecto, consola, GCP CLI
GKE: Google Kubernetes Engine	Creación de clúster, zonas y regiones, tamaños
Conexión desde terminal	gcloud auth, gcloud container clusters get-credentials
Despliegue desde GCP	Uso de kubectl en GKE, dashboard de GCP

Área de Habilidades (Saber Hacer)

Competencia	
Despliega una aplicación Ingress previamente desarrollada en un entorno en la nube.	
Tipo de Actividad	Ponderación
Tarea del tema	2

Sesión No. 8, Unidad No. - Introducción a Kubernetes

Área Actitudinal (Saber ser)

Nombre del valor: Perseverancia
Aprender Kubernetes puede ser desafiante; promover la práctica constante y no rendirse ante errores transmite este valor fundamental.

Área de Conocimiento (Saber)

Competencia(s)	
Comprende la arquitectura básica de Kubernetes y su modelo de despliegue basado en Pods.	
Tema	Subtema
¿Qué es Kubernetes?	Historia, ventajas, diferencias con Docker Compose
Arquitectura de Kubernetes	Master, Node, API server, kubelet, etcd, scheduler, controller
Objetos básicos: Pod y Service	Creación de pods, servicios, kubectl básico
Despliegue simple	kubectl apply, kubectl get, kubectl describe, YAML básico

Área de Habilidades (Saber Hacer)

Competencia	
Despliega una aplicación simple en Kubernetes usando pods y servicios.	
Tipo de Actividad	Ponderación
Práctica en laboratorio	0

Sesión No. 9, Unidad No. 2 - Gestión de recursos y configuración en Kubernetes

Área Actitudinal (Saber ser)

Nombre del valor: Perseverancia
Aprender Kubernetes puede ser desafiante; promover la práctica constante y no rendirse ante errores transmite este valor fundamental.

Área de Conocimiento (Saber)

Competencia(s)	
Implementación o correcto gestionamiento de recursos avanzados en Kuberne	
Tema	Subtema
Deployments	Declaración, escalado manual, actualizaciones y rollback
Pods	Correcto entendimiento de la estructura de los Pods
Archivos YAML avanzados	Buenas prácticas en definición de múltiples objetos

Área de Habilidades (Saber Hacer)

Competencia	
Correcto entendimiento de la estructura de los manifest, así como los distintos Objetos de Kubernetes	
Tipo de Actividad	Ponderación
Define archivos YAML , Proyecto Kuberne	0, 50

Sesión No. 10, Unidad No. 2 - Escalado automático y monitoreo de salud en Kubernetes

Área Actitudinal (Saber ser)

Nombre del valor: Responsabilidad
Aprender Kubernetes puede ser desafiante; promover la práctica constante y para cumplir con todas las metas de la unidad.

Área de Conocimiento (Saber)

Competencia(s)	
Comprende cómo Kubernetes gestiona la disponibilidad y escalabilidad automática de aplicaciones mediante prometheus y recursos de control.	
Tema	Subtema
Namespaces	Organización de recursos, separación de ambientes
Escalado en Kubernetes	Horizontal Pod Autoscaler (HPA), recursos CPU/memoria, límites

Área de Habilidades (Saber Hacer)

Competencia	
Configura un HPA para escalar una aplicación según el uso de CPU.	
Tipo de Actividad	Ponderación
Práctica en laboratorio	0

Sesión No. 11, Unidad No. 2 - Locust para carga de prueba en Kubernetes

Área Actitudinal (Saber ser)

Nombre del valor: Responsabilidad
Aprender Kubernetes puede ser desafiante; promover la práctica constante y para cumplir con todas las metas de la unidad.

Área de Conocimiento (Saber)

Competencia(s)	
Comprende cómo generar pruebas de carga controladas para evaluar la capacidad de respuesta y escalabilidad de servicios.	
Tema	Subtema
Introducción a Locust	Instalación, propósito, comparación con otras herramientas

Área de Habilidades (Saber Hacer)

Competencia	
Crea un script de carga con Locust y su uso para probar un cluster de kubernetes	
Tipo de Actividad	Ponderación
Actividad en clase	1

Rúbrica de Evaluación

Cada una de las actividades del laboratorio (proyectos, prácticas, tareas y otras) cuenta con una rúbrica de evaluación específica, la cual está detallada en el documento que se entrega al estudiante al momento de asignar la actividad. Estas rúbricas describen los criterios de evaluación, niveles de desempeño esperados y la ponderación correspondiente de cada aspecto evaluado.

Es **responsabilidad del estudiante** leer detenidamente la rúbrica asignada antes de iniciar el desarrollo de la actividad. Comprender los criterios de evaluación no solo permite orientar adecuadamente el trabajo, sino también mejorar el desempeño académico y fomentar la autorregulación del aprendizaje.

En caso de no recibir la rúbrica al momento de la asignación, el estudiante **debe solicitarla directamente al tutor académico**, ya que constituye una herramienta esencial para el cumplimiento de los objetivos de aprendizaje y la evaluación transparente.

Normativa Académica y Ética del Curso

En concordancia con el perfil del estudiante de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, se espera un alto nivel de compromiso con la excelencia académica y la ética profesional. Por ello, que se establece los siguientes lineamientos de carácter obligatorio que regulan el comportamiento académico del estudiante:

Plagio y copias

- Todo proyecto será sometido a verificación para confirmar su autoría y originalidad, con la finalidad de evitar cualquier plagio, copia o que la actividad no haya sido realizada por el estudiante.
- Cualquier evidencia de lo antes descrito en las distintas actividades será sancionada con una calificación de 0 (cero) y el caso será reportado al Docente quien a su vez informará a la Escuela de Ciencias y Sistemas para su seguimiento institucional.

Prórrogas y reposiciones

- No se otorgarán prórrogas para entregas de actividades.
- No se permitirá la reposición de proyectos bajo ninguna circunstancia.

Requisitos para evaluación final del curso

- Es obligatorio aprobar el laboratorio para tener derecho a la evaluación final del curso.
- La calificación de prácticas, proyectos y otras actividades que se indique será asignada de forma presencial, en la fecha y hora establecidas por el tutor académico.

Asistencia

- Para obtener la nota del laboratorio, se requiere un mínimo del 80% de asistencia a las sesiones de laboratorio.
- En caso de inasistencia, sólo se aceptarán justificaciones válidas respaldadas por constancia oficial.

Entregas

- No se aceptarán entregas tardías de tareas, prácticas, exámenes cortos, exámenes finales o proyectos sin justificación.

Medio oficial de entrega

- La plataforma UEDI de la Facultad será el único medio oficial para la entrega de actividades del curso.

Bibliografía

- Gigi, S. (2021).
- Learn Kubernetes in a month of lunches (1st ed.). Manning Publications.
- ISBN-13: 978-1617297061

E-Grafía

- Komodor. (2024, 29 julio). kubectl autocomplete: Enabling and Using in Bash, Zsh & PS.
- [https://komodor.com/learn/kubectl-autocomplete-enabling-and-using-in-bash-zsh-and-powershell/#:~:text=Autocomplete%20de%20Kubectl%20en%20Bash%20\(MacOS\),sesiones%20de%20terminal%20de%20macOS.](https://komodor.com/learn/kubectl-autocomplete-enabling-and-using-in-bash-zsh-and-powershell/#:~:text=Autocomplete%20de%20Kubectl%20en%20Bash%20(MacOS),sesiones%20de%20terminal%20de%20macOS.)