

PROGRAMA DE LABORATORIO

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

**Sistemas Operativos 1**

CÓDIGO:	281	PONDERACIÓN:	5
ESCUELA DE INGENIERÍA EN:	CIENCIAS Y SISTEMAS	ÁREA A LA QUE PERTENECE:	DESARROLLO DE SOFTWARE
PRE REQUISITO:	Org, Lenguajes y Compiladores 2 (781), Arq, Compu y Ensambladores 1 (788)	POST REQUISITO:	Sistemas Operativos 2 (285), Sistema de Bases 2 (775)
CATEGORÍA:	OBLIGATORIO	VIGENCIA:	SEGUNDO SEMESTRE 2025
HORAS POR SEMANA DEL CURSO:	4 horas	HORAS POR SEMANA DEL LABORATORIO:	2 horas
HORAS DE AUTOAPRENDIZAJE:	4 horas	TOTAL DE HORAS DE APRENDIZAJE:	6 horas
CATEDRÁTICO (A):	Sergio Arnaldo Mendez Aguilar	AUXILIAR:	Edgar Rolando Alvarez Rodriguez
EDIFICIO:	Salon Virtual	SECCIÓN:	P
SALÓN DEL CURSO:	MEET VIRTUAL	SALÓN DEL LABORATORIO:	MEET VIRTUAL
DIAS QUE SE IMPARTE EL CURSO:	Lunes y Miércoles	DIAS QUE SE IMPARTE EL LABORATORIO:	Jueves
HORARIO DEL CURSO:	17:20 - 19:00	HORARIO DEL LABORATORIO:	17:20 - 19:00

Breve 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.

Índice

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).....	5
Diseño Didáctico por Competencias.....	5
Sesión de Diagnóstico.....	6
Evaluación de conocimientos previos.....	6
Presentación del tutor.....	6
Presentación de los estudiantes.....	6
Presentación del programa del curso.....	6
Evaluación de conocimientos del laboratorio actual.....	6
Sesión No. 1, Unidad No. 1 - Introducción al sistema operativo Linux.....	7
Valor de la semana (Saber ser).....	7
Conocimiento (Saber).....	7
Habilidades (Saber Hacer).....	7
Sesión No. 2, Unidad No. 1 - Bash conceptos básicos.....	8
Valor de la semana (Saber ser).....	8
Conocimiento (Saber).....	8
Habilidades (Saber Hacer).....	8
Sesión No. 3, Unidad No. 1 -Scripts intermedios en Bash.....	9
Valor de la semana (Saber ser).....	9
Conocimiento (Saber).....	9
Habilidades (Saber Hacer).....	9
Sesión No. 4, Unidad No. 1 - Virtualización ligera con contenedores Docker.....	10
Valor de la semana (Saber ser).....	10
Conocimiento (Saber).....	10
Habilidades (Saber Hacer).....	10
Sesión No. 5, Unidad No. 1 - Fundamentos del lenguaje C para modulos de kernel y uso con contenedores de bajo y alto nivel.....	11
Valor de la semana (Saber ser).....	11
Conocimiento (Saber).....	11
Habilidades (Saber Hacer).....	11
Sesión No. 6, Unidad No. 1 - C avanzado y llamadas al sistema con rust.....	12
Valor de la semana (Saber ser).....	12
Conocimiento (Saber).....	12

Habilidades (Saber Hacer).....	12
Sesión No. 7, Unidad No. 2 - Kubernetes en la nube con Google Cloud Platform.....	13
Valor de la semana (Saber ser).....	13
Conocimiento (Saber).....	13
Habilidades (Saber Hacer).....	13
Sesión No. 8, Unidad No. 2 - Introducción a Kubernetes.....	14
Valor de la semana (Saber ser).....	14
Conocimiento (Saber).....	14
Habilidades (Saber Hacer).....	14
Sesión No. 9, Unidad No. 2 - Gestión de recursos y configuración en Kubernetes.....	15
Valor de la semana (Saber ser).....	15
Conocimiento (Saber).....	15
Habilidades (Saber Hacer).....	15
Sesión No. 10, Unidad No. 2 - Escalado automático y monitoreo de salud en Kubernetes.....	16
Valor de la semana (Saber ser).....	16
Conocimiento (Saber).....	16
Habilidades (Saber Hacer).....	16
Sesión No. 11, Unidad No. 2 - Locust para carga de prueba en Kubernetes.....	17
Valor de la semana (Saber ser).....	17
Conocimiento (Saber).....	17
Habilidades (Saber Hacer).....	17
Tiempo de Auto-aprendizaje.....	18
Rúbrica de Evaluación.....	18
Resumen de Ponderaciones.....	18
Normativa Académica y Ética del Curso.....	19
Equipo Académico.....	20
Coordinador del Área.....	20
Sección P.....	20
Bibliografía.....	21
E-Grafía.....	21

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 por Competencias

Esta sección organiza las sesiones del laboratorio en función de las competencias que el estudiante debe desarrollar. Cada clase incluye valores (saber ser), contenidos teóricos (saber) y habilidades prácticas (saber hacer), permitiendo un aprendizaje integral y aplicado. Las actividades están alineadas con los objetivos del curso y el perfil del egresado.

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 Kahoot	Identificar el nivel de conocimientos previos en programación en Linux conceptos básicos de sistemas operativos y contenedores, 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 comparten 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 Didáctico: Empleo de entornos aislados, plataformas de desarrollo y soluciones basadas en contenedores.	Identificar el nivel de conocimientos previos en programación y conceptos básicos de contenedores.

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

Valor de la semana (Saber ser)

Nombre: Introducción al sistema operativo Linux
Comprende la estructura general del sistema operativo Linux y sus componentes clave.

Conocimiento (Saber)

Competencia(s)	
Comprende la estructura general del sistema operativo Linux y sus componentes clave.	
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

Habilidades (Saber Hacer)

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

Sesión No. 2, Unidad No. 1 - Bash conceptos básicos

Valor de la semana (Saber ser)

Nombre: Creación de Scripts en Bash

Introducción al lenguaje de programacion Bash y su uso en Linux

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

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
Crea scripts Bash simples para automatizar tareas básicas en Linux.	Práctica en laboratorio	0
Desarrollo de Contenedores y Gestión de Imágenes en Entornos Virtualizados	Proyecto 1	15

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

Valor de la semana (Saber ser)

Nombre: Scripts intermedios en Bash
Uso de estructuras de control, condicionales y bucles para crear scripts más avanzados en Bash.

Conocimiento (Saber)

Competencia(s)	
Aplica estructuras de control de flujo para construir scripts más robustos y funcionales.	
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

Habilidades (Saber Hacer)

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

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

Valor de la semana (Saber ser)

Nombre: Virtualización ligera con contenedores Docker
Introducción al uso de Docker para crear, gestionar y desplegar contenedores de forma eficiente.

Conocimiento (Saber)

Competencia	
Comprende los conceptos fundamentales de virtualización ligera mediante contenedores, y el funcionamiento básico de Docker.	
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

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
Práctica Única	Práctica	2

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

Valor de la semana (Saber ser)

Nombre: Fundamentos del lenguaje C
Estudio de la sintaxis básica del lenguaje C, tipos de datos y estructuras de control fundamentales.

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

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
Crea programas básicos en C que utilicen entrada/salida y estructuras de control.	Práctica en laboratorio	0
Módulo de Kernel para listado y monitoreo básico de procesos	Proyecto 2	15

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

Valor de la semana (Saber ser)

Nombre: C avanzado y llamadas al sistema
Aplicación de conceptos avanzados del lenguaje C como punteros, manejo de memoria y uso de llamadas al sistema para interacción con el sistema operativo.

Conocimiento (Saber)

Competencia	
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

Habilidades (Saber Hacer)

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

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

Valor de la semana (Saber ser)

Nombre: Kubernetes en la nube con Google Cloud Platform
Configuración y despliegue de clústeres Kubernetes utilizando GKE (Google Kubernetes Engine) para entornos reales.

Conocimiento (Saber)

Competencia	
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

Habilidades (Saber Hacer)

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

Sesión No. 8, Unidad No. 2 - Introducción a Kubernetes

Valor de la semana (Saber ser)

Nombre: Introducción a Kubernetes
Introducción al sistema de orquestación Kubernetes, su arquitectura y el despliegue básico de aplicaciones en contenedores.

Conocimiento (Saber)

Competencia	
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

Habilidades (Saber Hacer)

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

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

Valor de la semana (Saber ser)

Nombre: Gestión de recursos y configuración en Kubernetes
Implementación de recursos avanzados en Kurnet

Conocimiento (Saber)

Competencia	
Comprende el uso de recursos esenciales en Kubernetes para la gestión de configuraciones, actualizaciones y almacenamiento persistente.	
Tema	Subtema
Deployments	Declaración, escalado manual, actualizaciones y rollback
Archivos YAML avanzados	Buenas prácticas en definición de múltiples objetos

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
Define archivos YAML	Práctica en Laboratorio	0
Proyecto Kubernetes	Proyecto 3	50

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

Valor de la semana (Saber ser)

Nombre: Escalado automático y monitoreo de salud en Kubernetes
Implementación de estrategias de escalado horizontal y configuración de prometheus para asegurar la disponibilidad y estabilidad de los pods en Kubernetes.

Conocimiento (Saber)

Competencia	
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

Habilidades (Saber Hacer)

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

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

Valor de la semana (Saber ser)

Nombre: Pruebas de carga con Locust en aplicaciones desplegadas
Uso de Locust para simular usuarios concurrentes y evaluar el rendimiento de servicios desplegados en Kubernetes.

Conocimiento (Saber)

Competencia	
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

Habilidades (Saber Hacer)

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

Tiempo de Auto-aprendizaje

Tipo	Horas de Auto-aprendizaje
Proyectos	55
Prácticas	2
Tareas	1
Total	58

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.

Resumen de Ponderaciones

Tipo	Valor
Actividades en Clase	1
Proyectos	80
Prácticas	2
Tareas	2
Examen Final	15
Total	100

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.

Equipo Académico

Coordinador del Área

Nombre: Luis Fernando Espino Barrios	Correo electrónico: ---
--	----------------------------

Sección P

Docente

Nombre del Docente Sergio Arnaldo Mendez Aguilar	Correo electrónico ---
--	---------------------------

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

Tutor(es)

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

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

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=Autocompletar%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=Autocompletar%20de%20kubectl%20en%20Bash%20(MacOS),sesiones%20de%20terminal%20de%20macOS.)