

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA  
FACULTAD DE INGENIERÍA  
ESCUELA DE CIENCIAS Y SISTEMAS  
SISTEMAS OPERATIVOS 1 SECCIÓN A  
ING. SERGIO ARNALDO MENDEZ AGUILAR  
AUX. FERNANDO ALBERTO MAZARIEGOS TAJIBOY  
VACACIONES DICIEMBRE 2020



# PROYECTO 1

## MODULOS KERNEL, PLANIFICACIÓN DE PROCESOS Y MONITOREO DE RECURSOS

### OBJETIVOS

- Conocer el kernel de Linux y los módulos que actúan sobre el directorio /proc.
- Creación de módulos.
- Conocer la planificación de procesos de Linux.
- Investigar la función del directorio /proc para obtener la información del sistema.
- Generar gráficas dinámicas dentro de una solución web, para facilitar la lectura de información por parte del usuario final.
- Conocer los estados de los procesos en los sistemas basados en Unix.

### DESCRIPCIÓN

Se deberá implementar un servidor web en Golang, creando una aplicación simple para obtener información sobre la memoria y CPU del sistema haciendo uso de módulos del kernel escritos en C para obtener la información y escribirla en archivos dentro de la carpeta /proc para posteriormente leerlos con Golang.

Queda a discreción del estudiante que tecnología utilizar para la interfaz gráfica que deberá ser visible a través del navegador.

### MÓDULOS A IMPLEMENTAR

#### Módulo de Memoria(sysinfo)

El módulo deberá sobrescribir un archivo en el directorio /proc.

#### Características a implementar:

- Importar librería <sys/sysinfo.h>
- Debe imprimir el número de carnet del estudiante al cargar el módulo (insmod).
- Debe imprimir el nombre del curso al momento de descargar el módulo (rmmod).
- La información que se mostrará en el módulo debe ser obtenida por medio de los struct de información del sistema operativo y no de la lectura de otro archivo.
- El nombre del módulo será: memo\_<<carnet>>

## Módulo CPU(task\_struct)

El módulo deberá sobrescribir un archivo en el directorio /proc.

Características a implementar:

- Importar librerías: <linux/sched.h>, <linux/sched/signal.h>
- Debe imprimir el nombre del estudiante al cargar el módulo (insmod).
- Debe imprimir “Diciembre 2020” al momento de descargar el módulo (rmmod).
- La información a mostrar debe ser obtenida por medio de los struct de datos del sistema operativo y no de la lectura de archivos o comandos de consola.
- El nombre del módulo será: cpu\_<<carnet>>

## APLICACIÓN WEB

La aplicación web permite visualizar gráficas dinámicas que muestren el uso del CPU y de la memoria RAM del servidor.

La aplicación web permite mostrar la información básica de los procesos que se ejecutan y permite terminar los procesos (kill) que se encuentran en ejecución.

### Página Principal

Esta debe mostrar de manera tabulada todos los procesos que están siendo ejecutados en el servidor, así como un resumen general de los procesos. La información general que se debe mostrar es la siguiente:

Procesos en ejecución	Número de procesos en estado de ejecución (running)
Procesos suspendidos	Número de procesos en estado de suspendido (sleeping)
Procesos detenidos	Número de procesos en estado de detenido (stopped)
Procesos zombie	Número de procesos zombie
Total de procesos	Número total de procesos

Además, se debe mostrar de manera tabulada la siguiente información de cada proceso:

PID	Identificador del proceso
Nombre	Nombre del proceso
Usuario	Usuario que ejecutó el proceso
Estado	Estado en el que se encuentra el proceso
%RAM	Porcentaje de utilización de RAM por el proceso

En esta sección, debe estar un botón el cual desarrolla la función de terminar (kill) al proceso seleccionado en la tabla ya descrita.

Se deberá mostrar un árbol de procesos dinámico, es decir, se debe mostrar una lista de procesos que permita seleccionar uno y desplegar a todos sus hijos con PID y nombre.

## Monitor de CPU

El monitor de CPU debe mostrar la información del consumo de CPU del servidor, en el cual se podrá visualizar la siguiente información:

- Porcentaje de CPU utilizado
- Gráfica de utilización de CPU en tiempo real

La gráfica debe ser similar a un polígono de frecuencia, el cual debe mostrar el consumo del CPU del servidor en tiempo real sin que el usuario necesite estar refrescando la página para monitorear el comportamiento de la utilización del CPU.

## Monitor de RAM

El monitor de memoria RAM es similar al de CPU, debe mostrar la información del consumo de RAM del servidor, en el cual se podrá visualizar la siguiente información:

- Total de memoria RAM del servidor (en MB)
- Total de memoria RAM consumida (en MB)
- Porcentaje de consumo de RAM

Al igual que el monitor de CPU, la gráfica debe mostrarse como un polígono de frecuencias en tiempo real del consumo de memoria del servidor. En las gráficas toda información mostrada debe estar en MB.

## LOCUST

Para poder visualizar de una mejor manera el funcionamiento de todo lo anterior utilizaremos Locust, la cual es una herramienta de pruebas de tráfico.

## RESTRICCIONES

- El proyecto se realizará de manera individual.
- La aplicación debe tener un aspecto profesional.
- La obtención de la información se hará a través de directorio /proc, a excepción de las estadísticas de CPU, de lo contrario se tendrá una nota de cero puntos.
- Cualquier copia parcial o total será reportada a la Escuela de Ciencias y Sistemas para que proceda como indica el reglamento.
- El código fuente debe ser administrado por medio de un repositorio de github, al momento de la calificación se bajará la última versión y será ejecutada en el servidor.

## ENTREGABLES

- Código fuente de los módulos desarrollados.
- Código fuente de la solución.
- Manual del proceso de creación de los módulos.

## FORMA DE ENTREGA

Es indispensable que se realice la entrega de 2 formas:

- Mediante UEDI, subiendo el enlace del repositorio con el código fuente y el manual del proceso de creación de módulos, en una carpeta comprimida [SO1]Proyecto1\_<<carnet>>.rar.
- Por medio de un repositorio de GitHub, el cual debe ser privado con el nombre: proyecto1\_<<carnet>>\_so1.

**La entrega se debe realizar antes del jueves 17 de diciembre de 2020**