

**UNIVERSIDAD SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA
ESCUELA DE CIENCIAS Y SISTEMAS
SISTEMAS OPERATIVOS 2
LABORATORIO SISTEMAS OPERATIVOS 2**

**PROYECTO
FASE 1**

MANUAL TÉCNICO

**Diego Andrés Obín Rosales
201903865**

Guatemala, 27 de febrero de 2023

VERSIONES UTILIZADAS.

Para la elaboración de este proyecto fueron utilizadas las diferentes herramientas con sus respectivas versiones:

Editor de Código: **Visual Studio Code V. 1.75**

Lenguajes de Programación:

- **Golang V. 1.20**
- **NodeJs V. 18.12**
- **Wails V. 2.3**

INSTALACIONES.

Golang.

Para la instalación de golang fue necesaria la descarga del archivo de la página oficial de Golang: <https://go.dev/dl/>

NodeJS.

Instalación de NodeJS de la página oficial de descarga: <https://nodejs.org/en/>

Wails.

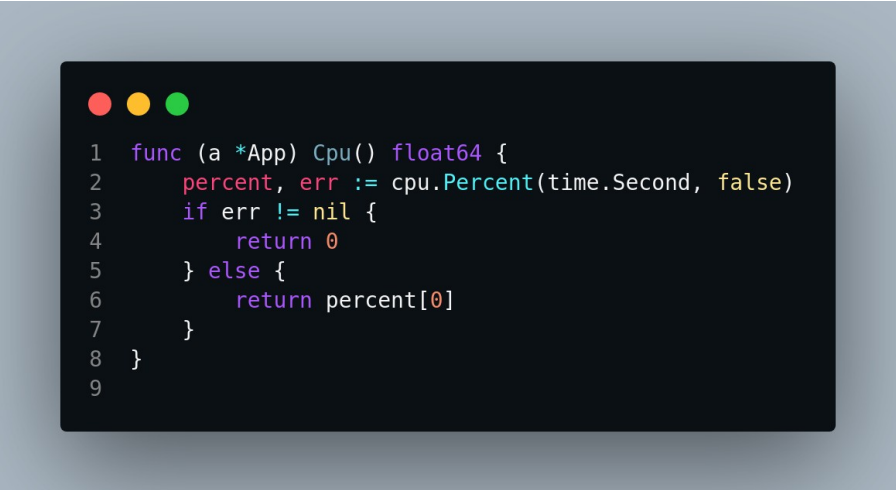
Pasos para la instalación de Wails: <https://wails.io/docs/gettingstarted/installation/>

CREACIÓN DEL PROYECTO.

Para la creación del proyecto utilizando la herramienta **Wails** es necesario ejecutar el siguiente comando: `wails init -n APP -t react`

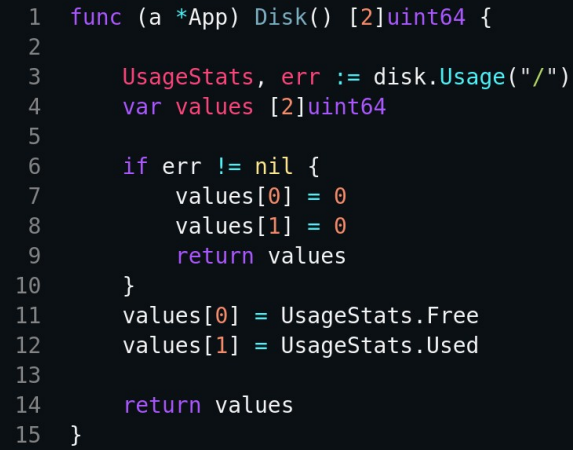
FUNCIONES.

Obtención del módulo de CPU:



```
1 func (a *App) Cpu() float64 {
2     percent, err := cpu.Percent(time.Second, false)
3     if err != nil {
4         return 0
5     } else {
6         return percent[0]
7     }
8 }
9
```

Obtención del módulo de Memoria:



```
1 func (a *App) Disk() [2]uint64 {
2
3     UsageStats, err := disk.Usage("/")
4     var values [2]uint64
5
6     if err != nil {
7         values[0] = 0
8         values[1] = 0
9         return values
10    }
11    values[0] = UsageStats.Free
12    values[1] = UsageStats.Used
13
14    return values
15 }
```

COMPILACIÓN.

Para llevar a cabo la compilación del proyecto se debe de ejecutar el comando: “wails build” que generará el código listo para producción en el directorio “build/bin”, ahora si se desea correr el código en forma desarrollador se debe de ejecutar el comando: “wails dev” así podrá realizar cambios conforme se actualiza el código fuente del proyecto.