



Sistemas Operativos 2

Practica 1

Manual Técnico

Juan Antonio Solares Samayoa
Carnet 201800496
Primer Semestre 2023

Descripción de la aplicación	3
Configuración de la aplicación	4
Lenguajes de programación utilizados	4
Frameworks utilizados	4
Wails	4
React.js	4
Estructura del proyecto	5
Estructura de carpetas y archivos del proyecto	5
Estructura de carpetas y archivos de la aplicación frontend de React.js	6
Interacción con el sistema operativo	7
Obtención uso de CPU	7
Comando mpstat	7
Obtención uso de Disco	7
Comando df	7

Descripción de la aplicación

La aplicación de escritorio monitoreo consiste en una solución de software que ayuda a usuarios que nos están familiarizados a utilizar Sistemas Operativos basado en linux a poder monitorear de forma sencilla el uso de disco y de memoria sin necesidad de utilizar comandos del sistema, lo cual para usuarios principiantes puede ser un tema difícil de comprender. La aplicación tiene como objetivo brindar una manera fácil y visual para monitorear el uso de disco y CPU en sistemas Linux. Los usuarios podrán ver los recursos de sus sistemas en tiempo real y hacer un seguimiento de los cambios en el uso de recursos en un período determinado.

La interfaz de la aplicación será simple y fácil de usar, presentando gráficas de líneas intuitivas que muestran la utilización de disco y CPU en un período de tiempo seleccionado. Los usuarios podrán elegir el intervalo de tiempo para el monitoreo y seleccionar las unidades de medida (por ejemplo, porcentaje de uso de CPU, megabytes de espacio en disco). Además, la aplicación también contará con alarmas configurables que notificarán a los usuarios cuando los recursos del sistema alcancen un umbral determinado. Estas alarmas se presentarán visualmente en las gráficas de líneas para alertar a los usuarios de cualquier problema en el uso de recursos del sistema.

Configuración de la aplicación

Lenguajes de programación utilizados

- Golang
- Javascript
- Typescript

Frameworks utilizados

Wails

Wails es un framework de desarrollo de aplicaciones de escritorio que utiliza Golang para el desarrollo del backend y tecnologías web como Vue.js o React.js para el desarrollo del frontend. Wails permite crear aplicaciones de escritorio multiplataforma utilizando las tecnologías web más modernas y proporciona una interfaz para interactuar con la funcionalidad del sistema operativo, como el acceso al sistema de archivos y la comunicación con otros programas. Además, Wails es compatible con las principales plataformas de escritorio como Windows, macOS y Linux

React.js

React.js es una librería de JavaScript de código abierto para la construcción de interfaces de usuario (UI) desarrollada por Facebook. Permite crear componentes reutilizables que representan diferentes partes de la interfaz de usuario de una aplicación web y manejar los cambios en dicha interfaz de manera eficiente, mejorando así la experiencia del usuario.

Estructura del proyecto

Estructura de carpetas y archivos del proyecto

Carpeta	Descripción
assets	Ubicación de recursos como imágenes, iconos y fuentes.
build	En este directorio se almacenan los archivos compilados de la aplicación
frontend	En este directorio se ubicaran el código relacionado con el que la aplicación será construida (Varía de acuerdo al framework a utilizar).
main.go	Archivo principal de la aplicación, en el cual se define el punto de entrada
app.go	Archivo en el que se define el comportamiento de la aplicación, se definen métodos y funciones que serán ejecutadas por el frontend de la aplicación. En este archivo se ejecutan las funciones con las cuales se interactúa con el sistema operativo para obtener las métricas del sistema, tales como el uso de CPU y disco.
wails.json	<p>Es un archivo de configuración que es necesario para la construcción y empaquetamiento de la aplicación.</p> <p>En este archivo puede especificar diferentes configuraciones, como por ejemplo el nombre de la aplicación, versión, descripción, autor y ruta de salida de la aplicación que será empaquetada. También se pueden especificar los comandos necesarios para construir y ejecutar la aplicación</p>

Estructura de carpetas y archivos de la aplicación frontend de React.js

Carpeta	Descripción
node_modules	En esta carpeta se ubican todas las dependencias de node.js y Typescript necesarias para que la aplicación pueda ser ejecutada
public	Contiene los archivos públicos de la aplicación, como por ejemplo el archivo "index.html"
src	Contiene los archivos fuente de la aplicación, como son los componentes, servicios, páginas, utilidades y el archivo de entrada "index.tsx".
wailsjs	Es una directorio que se crea de manera automática durante la construcción de la aplicación, contiene los archivos y configuraciones necesarios para la integración entre la aplicación de React y la aplicación de Wails.
package.json	Es un archivo de configuración que contiene información sobre la aplicación y también de las dependencias utilizadas.
index.tsx	Es el punto de entrada de la aplicación y renderizar el componente App, que es el componente principal.
App.tsx	Archivo que contiene la lógica para poder enrutar a diferentes páginas en la aplicación.

Interacción con el sistema operativo

La recolección de datos para generar las gráficas en la aplicación serán por medio de comandos de linux, los cuales retornan métricas de uso del CPU y disco en tiempo real.

Obtención uso de CPU

Comando mpstat

El comando mpstat (Multi-Processor Statistics) en Linux es una herramienta que permite mostrar las estadísticas de uso del procesador y de los núcleos de procesamiento. Es especialmente útil para monitorear el uso de la CPU y la carga de trabajo del sistema.

La salida de mpstat incluye información sobre el uso promedio del procesador y de cada núcleo de procesamiento, así como la cantidad de tiempo que se dedica a procesamiento de usuario, procesamiento de sistema y espera de E/S (entrada/salida). Además, también muestra la cantidad de interrupciones por segundo y el porcentaje de tiempo de espera en la cola de E/S.

Ejemplo del uso del comando mpstat

```
(base) juanantonio@juanantonio:~$ mpstat
Linux 5.10.102.1-microsoft-standard-WSL2 (juanantonio) 02/26/23 _x86_64_ (20 CPU)
12:30:20 CPU %usr %nice %sys %iowait %irq %soft %steal %guest %gnice %idle
12:30:20 all 0.03 0.00 0.05 0.00 0.00 0.00 0.00 0.00 0.00 99.92
```

Obtención uso de Disco

Comando df

El comando df -h es utilizado en sistemas Linux para mostrar el espacio disponible en los diferentes sistemas de archivos montados en el sistema. Al utilizar la opción -h, el comando muestra los tamaños de los archivos en una forma fácil de leer para los usuarios.

La salida del comando muestra información para cada sistema de archivos montado, incluyendo el dispositivo de almacenamiento, la capacidad total, el espacio utilizado, el espacio libre y el porcentaje de espacio utilizado.

Ejemplo de uso del comando df

```
(base) juanantonio@juanantonio:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/sdc        251G   13G  226G   6% /
none            3.9G   32K   3.9G   1% /mnt/wslg
none            3.9G   4.0K   3.9G   1% /mnt/wsl
tools           456G  271G  186G  60% /init
none            3.9G   4.0K   3.9G   1% /run
none            3.9G     0   3.9G   0% /run/lock
none            3.9G     0   3.9G   0% /run/shm
none            3.9G     0   3.9G   0% /run/user
tmpfs           3.9G     0   3.9G   0% /sys/fs/cgroup
drivers          456G  271G  186G  60% /usr/lib/wsl/drivers
lib             456G  271G  186G  60% /usr/lib/wsl/lib
none            3.9G   76K   3.9G   1% /mnt/wslg/versions.txt
none            3.9G   76K   3.9G   1% /mnt/wslg/doc
drvfs           456G  271G  186G  60% /mnt/c
```

Figura #. Uso del comando “df”