



S.I.G.D.

Sistemas Operativos III

ProgWare

ROL	APELLIDO	NOMBRE	C.I	E-MAIL	TEL/CEL
Coordinador	Gallas	Lucas	5.363.476-5	lucasgallas2003@gmail.com	093766017
Sub-coordinador	Gonzalo	Martinez	5.230.446-8	Gonzalom747@gmail.com	094663018
Integrante	Vidir	Kevin	5.646.391-3	kevinvidir@gmail.com	094230963
Integrante 2	Alvez	Mauricio	5.450.509-8	alvez.mauricio04@gmail.com	091240243
Integrante 3	Almeyra	Valentín	5.348.527-1	vaalca2017@gmail.com	092954187

Docente: Rodríguez, Carlos

**Fecha de
culminación**

5/9/2022

SEGUNDA ENTREGA

I.S.B.O.

3°BC



Índice

Índice	1
Usuarios necesarios para el Sistema Operativo	2
Menús Scripts para los Usuarios del Sistema	3
Rutinas de Backup con Crontab	16

S.I.G.D.

I.S.B.O.

3°BC ¹



Usuarios necesarios para el Sistema Operativo

- Administrador de Backups
- Administrador de Base de Datos
- Administrador de Logs
- Root

```
## Alias de Comando

# Alias de Comando - Backups
Cmnd_Alias Backups = (ALL) /var/spool/cron/crontabs
Cmnd_Alias BScript = (ALL) /Scripts/Backup.sh
Adm_Backups ALL=(ALL) ALL

# Alias de Comando - Base de Datos
Cmnd_Alias BD = (ALL) /var/lib/mysql
Adm_BD ALL=(ALL) ALL

# Alias de Comando - Logs
Cmnd_Alias Logs = (ALL) /var/log
Adm_Logs ALL=(ALL) ALL
```



Menús Scripts para los Usuarios del Sistema

S.I.G.D.

I.S.B.O.

3°BC ₃



```
#!/bin/bash
function _menuPrincipal()
{
    echo "1)Operador"
    echo "2)Administrador"
    echo "9)Salir"
    echo
    echo n- "Indica una opcion:"
}

function _menuOperador()
{
    echo "1)Hay Usuarios Conectados?"
    echo "2)Ip del Servidor"
    echo "3)Base de datos funcionando?"
    echo "4)Cambiar Password"
    echo "5)RespalDOS"
    echo "6)Recuperacion"
    echo "9)Salir"
    echo
    echo n- "Indica una opcion:"
}

function _menuAdministrador()
{
    echo "1)Hay Usuarios Conectados?"
    echo "2)Ip del Servidor"
    echo "3)Base de datos funcionando?"
    echo "4)Cambiar Password"
    echo "5)RespalDOS"
    echo "6)Recuperacion"
    echo "7)Usuarios"
    echo "8)Grupos"
    echo "9)Salir"
    echo
    echo n- "Indica una opcion:"
}

```



```
function _submenuUsuarios()
{
    echo "1)Creacion"
    echo "2)Modificacion - Nombre "
    echo "3)Modificacion - Contraseña "
    echo "4)Borrado"
```

```
    echo "9)Salir"
    echo
    echo n- "Indica una opcion:"
}
```

```
function _submenuGrupos()
{
    echo "1)Creacion"
    echo "2)Modificacion"
    echo "3)Borrado"
    echo "9)Salir"
    echo
    echo n- "Indica una opcion:"
}
```

```
opc=0
until [ $opc -eq 9 ]
do
    case $opc in
        1)
            opc1=0
            until [ $opc1 -eq 9 ]
            do
                case $opc1 in
                    1)
                        echo "Operador - Hay Usuarios Conectados"
```



```
W
  _menuOperador
..
11
  2)
  echo "Operador - IP Servidor"
  ifconfig
  _menuOperador
..
11
  3)
  echo "Operador - Base de Datos Funcionando?"
  systemctl status mariadb
  _menuOperador
..
11
  4)
  echo "Operador - Cambiar Password"
  sudo passwd
  _menuOperador
```

```
..
11
  5)
  echo "Operador - Respaldos"
  _menuOperador
..
11
  6)
  echo "Operador - Recuperacion"
  _menuOperador
..
11
  *)
  _menuOperador
..
11
```

```
esac
```



```
    esac
    read opc1

done
_menuPrincipal
..
2)
opc2=0
until [ $opc2 -eq 9 ]
do
    case $opc2 in
        1)
            echo "Administrador - Hay usuarios Conectados?"
            w
            _menuAdministrador
            ..
        2)
            echo "Administrador - IP Servidor"
            ifconfig
            _menuAdministrador
            ..
        3)
            echo "Operador - Base de Datos Funcionando?"
            systemctl status mariadb
            _menuAdministrador
            ..
        4)
            echo "Operador - Cambiar Password"
```

```
sudo passwd
_menuAdministrador
```




```
_menuAdministrador
..
11
5)
echo "Operador - RespalDOS"
..
11
6)
echo "Operador - Recuperacion"
..
11
7)
echo "Administrador - Usuarios"
opc3=0
until [ $opc3 == 9 ]
do
case $opc3 in
1) echo "Administrador - Usuarios - Creacion"
echo "Insertar nombre"
read user
echo "Agregar Contraseña"
read contra
adduser $user -p $contra
echo "usuario actualizado"
echo
_submenuUsuarios
..
11

2) echo "Administrador - Usuarios - Modificacion Nombre"
echo "Ingresar usuario a modificar"
read user
echo "Ingresar nuevo nombre"
read nuser
usermod -l $nuser $user
echo "usuario actualizado"
echo
_submenuUsuarios
..
11

3) echo "Administrador - Usuarios - Modificacion Contraseña"
echo "Ingrese usuario a modificar"
read user
passwd $user
```



```
echo "usuario actualizado"
echo
```

```

_submenuUsuarios
..
11

4) echo "Administrador - Usuarios - Borrar"
echo "usuario a eliminar"
read user
userdel -r $user
echo "usuario actualizado"
echo
_submenuUsuarios

..
11
*)
_submenuUsuarios
..
11
esac
read opc3
done
_menuAdministrador

..
11
8)
echo "Administrador - Grupos"
opc4=0
until [ $opc4 -eq 9 ]
do
case $opc4 in
1) echo "Administrador - Grupos - Creacion"
echo "Ingrese Nombre del Grupo"
read grupo

```



```
groupadd $grupo
echo "Grupo actualizado"
echo
_submenuGrupos
..
11
```

```
2) echo "Administrador - Grupos - Modificacion Nombre"
echo "Grupo a modificar"
read grupo
echo "nuevo Nombre del grupo"
read ngrupo
groupmod -n $ngrupo $grupo
```

```
echo "Grupo actualizado"
echo
_submenuGrupos
..
11
```

```
2) echo "Administrador - Grupos - Modificacion ID"
echo "Grupo a modificar"
read grupo
echo "Nuevo GID"
read gid
groupmod -g $gid $grupo
```

```
echo "Grupo actualizado"
echo
_submenuGrupos
..
11
```

```
3) echo "Administrador - Grupos - Borrar"
echo "Grupo a Modificar"
```



```

echo "Grupo a modificar"
read grupo
echo "Nuevo GID"
read gid
groupmod -g $gid $grupo

echo "Grupo actualizado"
echo
_submenuGrupos
..
..

3) echo "Administrador - Grupos - Borrar"
echo "Grupo a Modificar"
read grupo
groupdel $grupo
echo "Grupo actualizado"
echo
_submenuGrupos
..
..
*)
_submenuGrupos
..
..
esac
read opc4
done
_menuAdministrador

..
..

*)
_menuAdministrador
..
..
esac
read opc2
done
_menuPrincipal

```



Codigo Del Menu:

```
#!/bin/bash

function _menuAdministrador()
{
    echo "1)Hay Usuarios Conectados?"
    echo "2)Ip del Servidor"
    echo "3)Base de datos funcionando?"
    echo "4)Cambiar Password"
    echo "5)Respalidos"
    echo "6)Recuperacion"
    echo "7)Usuarios"
    echo "8)Grupos"
    echo "9)Salir"
    echo
    echo n- "Indica una opcion:"
}

function _submenuUsuarios()
{
    echo "1)Creacion"
    echo "2)Modificacion - Nombre "
    echo "3)Modificacion - Contraseña "
    echo "4)Borrado"
    echo "9)Salir"
    echo
    echo n- "Indica una opcion:"
}

function _submenuGrupos()
```



```
{  
echo "1)Creacion"  
echo "2)Modificacion"
```

```
echo "3)Borrado"  
echo "9)Salir"  
echo  
echo n- "Indica una opcion:"  
}
```

```
opc=0  
until [ $opc -eq 9 ]  
do  
case $opc in
```

```
2)  
Opc1=0  
until [ $opc1 -eq 9 ]  
do  
case $opc2 in
```

```
1)
```

```
echo "Administrador - Hay usuarios Conectados?"
```



```
w
_menuAdministrador
;;
2)
echo "Administrador - IP Servidor"
ifconfig
_menuAdministrador
;;
3)
echo "Operador - Base de Datos Funcionando?"
systemctl status mariadb
_menuAdministrador
;;
```

```
4)
echo "Operador - Cambiar Password"
sudo passwd
_menuAdministrador
;;
5)
echo "Operador - RespalDOS"
;;
```



6)

```
echo "Operador - Recuperacion"  
;;
```

7)

```
echo "Administrador - Usuarios"  
opc3=0  
until [ $opc3 == 9 ]  
do  
case $opc3 in  
    1)      echo "Administrador - Usuarios - Creacion"  
            echo "Insertar nombre"  
            read user  
            echo "Agregar Contraseña"  
            read contra  
            adduser $user -p $contra  
            echo "usuario actualizado"  
            echo  
            _submenuUsuarios  
            ;;  
  
    2)      echo "Administrador - Usuarios - Modificacion Nombre"  
            echo "Ingresar usuario a modificar"  
            read user|
```




Rutinas de Backup con Crontab

Script de Backup: Este script crea el respaldo comprimiendo la carpeta del programa, mediante este código:

```
cd /backups && tar -cpzf Proyecto_Backup.tar.gz /home/ProgWare_Server/Proyecto
echo Hecho!
```

Para que este script se ejecute de forma rutinaria se lo ejecutó con crontab de forma total e incremental, de esta forma:

```
#Total
30 18 * * * 6 /Scripts/Backup.sh

#Incremental
00 19 * * * /Scripts/Backup.sh
```

En el respaldo total la fecha y hora de este serán a las 18:30 de cada domingo de la semana y el mes.

En el respaldo incremental la fecha y hora de este serán a las 19:00 todos los días de la semana y el mes.



Hoja Testigo:

S.I.G.D.

I.S.B.O.

3°BC 17