



UNIVERSIDAD DE OVIEDO

ESCUELA POLITÉCNICA DE INGENIERÍA DE GIJÓN

INGENIERO DE TELECOMUNICACIÓN

LENGUAJES Y SISTEMAS INFORMÁTICOS

PROYECTO FIN DE CARRERA N° 3133481

**GRABACION Y REPRODUCCION DE AUDIO-VIDEO EN
UN PUESTO MULTIPANTALLA**

DOCUMENTO N° 6

CÓDIGO

**YÚSEF HABIB FERNÁNDEZ
NOVIEMBRE 2014**

**TUTOR: CLAUDIO DE LA RIVA
COTUTOR: MARCELINO AGUINAGA**

TABLA DE CONTENIDO

1.	INTRODUCCIÓN.....	4
2.	INTERFAZ DE USUARIO	5
2.1	Menu.sh	5
2.2	menu2.sh.....	6
2.3	menu3.sh.....	7
2.4	menu4.sh.....	8
3.	GRABACIÓN DE LA ESTACIÓN	11
3.1	saveSession.sh	11
3.2	record.sh.....	13
3.3	videoRecord.sh	15
3.4	audioRecord.sh	16
3.5	concat.sh	17
3.6	resoluciones.sh	18
4.	GENERACIÓN ARCHIVOS FINALES	19
4.1	fileGenerator.sh	19
5.	REPRODUCCIÓN DE LA ESTACIÓN	21
5.1	playStation.sh	21
6.	INSTALACIÓN	22
6.1	downloadSoft.sh	22
6.2	fromPath.sh	23
6.3	fromInternet.sh.....	25
7.	INFORMACIÓN	28
7.1	userGuide.sh	28
7.2	installationGuide.sh	30
7.3	credits.sh	31

IDENTIFICACIÓN DEL DOCUMENTO

Proyecto	Grabación y reproducción de audio-video en un puesto multipantalla
Nº proyecto	3133481
Autor	Yúsef Habib Fernández
Tutor	Claudio de la Riva Álvarez
Cotutor	Marcelino Aguinaga Izquierdo
Documento	Código
Fecha	Noviembre 2014

1. INTRODUCCIÓN

El presente documento contiene todo el código desarrollado para la realización de este prototipo, es por ello que no se puede encontrar el código asociado a librerías de terceros que hayan sido empleadas. Esto se debe a que no resultan de interés en términos del proyecto presentado pues no fueron modificadas ni alteradas en ningún grado para su correcto uso.

Se dividirá la presentación de todo el código en función de la finalidad de este para que resulte más sencillo de ordenar. Por tanto cada una de las entradas del índice corresponderá con una funcionalidad, y cada uno de los subíndices con cada script.

2. INTERFAZ DE USUARIO

2.1 Menu.sh

```
#!/bin/bash

#*****
#Menu.sh
#*****

#-----
#Script maestro para el manejo del software. Este es lanzado a partir del comando "bash Menu.sh"
#-----

#Code
while true; do
    # Limpieamos las X
    tput clear

    # Movemos el cursor a la localización X,Y (la esquina izquierda es 0,0)
    tput cup 3 15

    # Configuramos el color del fondo a traves del condigo de escape ANSI
    tput setaf 3
    echo "Indra SoftLabs"
    tput sgr0

    tput cup 5 17
    # Seleccionamos el modo de video "revertido"
    tput rev
    echo "M E N U - P R I N C I P A L"
    tput sgr0

    tput cup 7 15
    echo "1. Grabación de la estación"

    tput cup 8 15
    echo "2. Generación de archivos reproducibles"

    tput cup 9 15
    echo "3. Reproducción de la estación"

    tput cup 10 15
    echo "4. Instalación software"

    tput cup 11 15
    echo "5. Información"

    tput cup 12 15
    echo "6. Salir"

    # Seleccionamos modo "negrita"
    tput bold
    tput cup 14 15
    read -p "Enter your choice [1-6] " opcion

case $opcion in
```

```

        1) bash menu2.sh;;
        2) bash fileGenerator.sh;;
        3) bash playStation.sh;;
        4) bash menu4.sh;;
        5) bash menu3.sh;;
        6) echo "Adiós";
            exit 1;;
*)
    tput clear
    tput sgr0
esac
done

```

2.2 menu2.sh

```

#!/bin/bash

#*****
#menu2.sh
#*****

#-----
#      Extiende las opciones del Menú maestro
#-----

#Code
while true; do
    # Limpieamos las X
    tput clear

    # Movemos el cursor a la localización X,Y (la esquina izquierda es 0,0)
    tput cup 3 15

    # Configuramos el color del fondo a traves del condigo de escape ANSI
    tput setaf 3
    echo "Indra SoftLabs"
    tput sgr0

    tput cup 5 17
    # Seleccionamos el modo de video "revertido"
    tput rev
    echo "M E N U - P R I N C I P A L"
    tput sgr0

    tput cup 7 15
    echo "1. Grabación de la estación"

    tput bold
    tput cup 8 15
    echo "  1 Crear nueva sesión de grabación"
    tput cup 9 15
    echo "  2 Comenzar sesión de grabación"
    tput cup 10 15
    echo "  3 Volver a la anterior ventana"

    tput sgr0
    tput cup 11 15

```

```

echo "2. Generación de archivos reproducibles"

tput cup 12 15
echo "3. Reproducción de la estación"

tput cup 13 15
echo "4. Instalación software"

tput cup 14 15
echo "5. Información"

tput cup 15 15
echo "6. Salir"

# Seleccionamos modo "negrita"
tput bold
tput cup 17 15
read -p "Enter your choice [1-3] " opcion

case $opcion in
    1) bash saveSession.sh;;
    2) bash record.sh;;
    3) exit -1;;
    *)
        tput clear
        tput sgr0
esac
done

```

2.3 menu3.sh

```

#!/bin/bash

#####
#menu3.sh
#####

#-----
#      Extiende las opciones del Menu maestro
#-----

#Code
while true; do
    # Limpieamos las X
    tput clear

    # Movemos el cursor a la localización X,Y (la esquina izquierda es 0,0)
    tput cup 3 15

    # Configuramos el color del fondo a traves del condigo de escape ANSI
    tput setaf 3
    echo "Indra SoftLabs"
    tput sgr0

    tput cup 5 17
    # Seleccionamos el modo de video "revertido"
    tput rev
    echo "M E N U - P R I N C I P A L"

```

```

tput sgr0

tput cup 7 15
echo "1. Grabación de la estación"

tput cup 8 15
echo "2. Generación de archivos reproducibles"

tput cup 9 15
echo "3. Reproducción de la estación"

tput cup 10 15
echo "4. Instalación software"

tput cup 11 15
echo "5. Información"

tput bold
tput cup 12 15
echo "  1 Información de uso"
tput cup 13 15
echo "  2 Información de instalación"
tput cup 14 15
echo "  3 Información de licencias"
tput cup 15 15
echo "  4 Volver a la anterior ventana"

tput sgr0
tput cup 16 15
echo "6. Salir"

# Seleccionamos modo "negrita"
tput bold
tput cup 18 15
read -p "Enter your choice [1-4] " opcion

case $opcion in
    1) bash userGuide.sh;;
    2) bash installationGuide.sh;;
    3) bash credits.sh;;
    4) exit -1;;
    *)
        tput clear
        tput sgr0
esac
done

```

2.4 menu4.sh

```

#!/bin/bash

#*****
#menu4.sh
#*****

#-----
#Extiende las opciones del Menu maestro
#-----

```



```

#Code
while true; do
    # Limpiamos las X
    tput clear

    # Movemos el cursor a la localización X,Y (la esquina izquierda es 0,0)
    tput cup 3 15

    # Configuramos el color del fondo a traves del condigo de escape ANSI
    tput setaf 3
    echo "Indra SoftLabs"
    tput sgr0

    tput cup 5 17
    # Seleccionamos el modo de video "revertido"
    tput rev
    echo "M E N U - P R I N C I P A L"
    tput sgr0

    tput cup 7 15
    echo "1. Grabación de la estación"

    tput cup 8 15
    echo "2. Generación de archivos reproducibles"

    tput cup 9 15
    echo "3. Reproducción de la estación"

    tput cup 10 15
    echo "4. Instalación software"

    tput bold
    tput cup 11 15
    echo "  1 Descargar Software"
    tput cup 12 15
    echo "  2 Instalación sin internet"
    tput cup 13 15
    echo "  2 Instalación con internet"
    tput cup 14 15
    echo "  4 Volver a la anterior ventana"

    tput sgr0
    tput cup 15 15
    echo "5. Información"

    tput cup 16 15
    echo "6. Salir"

    # Seleccionamos modo "negrita"
    tput bold
    tput cup 18 15
    read -p "Enter your choice [1-4] " opcion

case $opcion in
    1) bash downloadSoft;;
    2) bash instal.sh;;
    3) asdad;;
    4) exit -1;;
*)

```

```
    tput clear
    tput sgr0
esac
done
```

3. GRABACIÓN DE LA ESTACIÓN

3.1 saveSession.sh

```
#!/bin/bash

#####
#saveSession.sh
#####

#-----
#Script encargado de recopilar toda la información necesaria para la creación de una nueva configuración.
#Se realiza a través de un asistente que pide cada uno de los parámetros a introducir.
#-----

clear

#Variables
declare SESSION
declare RUTAFFMPEG
declare RUTAVIDEOS
declare RES1
declare RES2
declare POS1
declare POS2
declare CRF
declare PRESET
declare FPS
declare RECORDTIME
declare SESSIONTIME
declare CURRENTDIR=`pwd`
#-----

#Funciones

#En caso de que exista la carpeta directamente entra en ella, en otro caso la crea y luego entra
gotDirectorio()
{
    if [ ! -d $1 ]; then
        mkdir $1
    fi
    cd $1
}

#Método para la confirmación de que todos los valores son correctos
confirmacion()
{
    tput sgr0
    tput bold
    while true; do
        echo; read -p "¿Está seguro de los valores introducidos o desea reiniciar el proceso? (s/n) " yn
        case $yn in
            s ) break;;
            n ) cd $CURRENTDIR; bash saveSession.sh;;
            * ) echo "por favor responda sí(s) o no(n)";;
        esac
    done
}
```

```

}

#Muestra la configuración actual para la sesión de grabación
showConfig()
{
    tput sgr0
    tput setaf 5
    echo "-----"
    echo " La configuración escogida es:"
    echo "   Nombre de la sesión: $SESSION"
    echo "   Ruta a los archivos FFmpeg: $RUTAFFMPEG"
    echo "   Ruta al directorio de grabación deseado: $RUTAVIDEOS"
    echo "   Resolución pantalla principal: $RES1"
    echo "   Resolución pantalla secundaria: $RES2"
    echo "   Posición pantalla principal: $POS1"
    echo "   Posición pantalla secundaria: $POS2"
    echo "   Tasa de refresco: $FPS fps"
    echo "   Valor de CRF: $CRF"
    echo "   Valor de preset: $PRESET"
    echo "   Duracion de la grabacion: $RECORDTIME "
    echo "   Duración de las sesiones de video: $SESSIONTIME minutos"
    echo "   Posiciones de los monitores: Principal: $POS1 Secundaria: $POS2"
    echo "-----"
}

#Se encarga de grabar en un archivo .txt las opciones de configuración, cada una en una línea nueva.
grabacion()
{
    echo $RUTAFFMPEG > $SESSION
    echo $RUTAVIDEOS >> $SESSION
    echo $RES1 >> $SESSION
    echo $RES2 >> $SESSION
    echo $POS1 >> $SESSION
    echo $POS2 >> $SESSION
    echo $FPS >> $SESSION
    echo $CRF >> $SESSION
    echo $PRESET >> $SESSION
    echo $RECORDTIME >> $SESSION
    echo $SESSIONTIME>> $SESSION
}
#-----

#Code
tput cup 1 1
echo "Asistente para la creación de una sesión de grabación"

tput cup 3 2
tput sgr0
echo "Introduzca los siguientes parámetros:"
tput cup 5 3
read -p "Nombre de la sesión: " SESSION
tput cup 6 3
read -p "Ruta a los archivos FFmpeg: " RUTAFFMPEG
tput cup 7 3
read -p "Ruta al directorio de grabación deseado: " RUTAVIDEOS
tput cup 8 3
echo "Selección de las resoluciones y posiciones de las pantallas: "
gnome-terminal --tab -e "bash resoluciones.sh"
tput cup 9 3
read -p "-Resolución pantalla principal: " RES1

```

```

tput cup 10 3
read -p "-Resolución pantalla secundaria: " RES2
tput cup 11 3
read -p "-Posición pantalla principal: " POS1
tput cup 12 3
read -p "-Posición pantalla secundaria: " POS2
tput cup 13 3
read -p "-Frecuencia de refresco(10fps es lo óptimo): " FPS
tput cup 14 3
read -p "-Indique valor de CRF: " CRF
tput cup 15 3
read -p "-Indique valor de preset: " PRESET
tput cup 16 3
read -p "Duración de la sesión de grabación(hh:mm:ss): " RECORDTIME
tput cup 17 3
read -p "Duración de los videos generados (minutos): " SESSIONTIME
echo

POS1=0
OIFS=$IFS && IFS='x' read -a array <<< "$RES1" && POS2=${array[0]} && IFS=$OIFS
SESSION=$SESSION.txt

showConfig
confirmacion
grabacion

```

3.2 record.sh

```

#!/bin/bash

#####
#record.SH
#####

#-----
#Script encargado de lanzar cada una de las instancias de grabación, para ello a través de un diálogo con
#el usuario le pide que seleccione una configuración de grabación.
#Se creará una carpeta para la sesión total, luego otra para cada día de grabación y un más pequeña en
#función del tamaño final de los archivos.
#-----

clear

#Variables
declare -a array
declare RUTAFFMPEG
declare RUTAVIDEOS
declare RES1
declare RES2
declare FPS
declare CRF
declare PRESET
declare RECORDTIME
declare SESSIONTIME
declare POS1
declare POS2
declare waitingTime
#-----

```

#Funciones

#Lanza las 3 instancias necesarias de FFmpeg

lanzarFFmpeg()

```
{
    gnome-terminal --tab -e "bash videoRecord.sh $RUTAFFMPEG $RUTAVIDEOS/$dateDMY/$dateHM
$RES1 $FPS $CRF $PRESET $RECORDTIME $POS1 1" && gnome-terminal --tab -e "bash videoRecord.sh
$RUTAFFMPEG $RUTAVIDEOS/$dateDMY/$dateHM $RES2 $FPS $CRF $PRESET $RECORDTIME
$POS2 2"
    gnome-terminal --tab -e "bash audioRecord.sh $RUTAFFMPEG $RUTAVIDEOS/$dateDMY/$dateHM
$RECORDTIME"
}
```

#Carga los parámetros de configuración de una sesión almacenados en un .txt

#Lee este recursivamente y otorga su valor a la variable

loadFile()

```
{
    i=0
    while read line; do
        array[i]=$line
        let "i+=1"
    done < $FILE
    RUTAFFMPEG=${array[0]} && RUTAVIDEOS=${array[1]}
    RES1=${array[2]} && RES2=${array[3]}
    POS1=${array[4]} && POS2=${array[5]}
    FPS=${array[6]} && CRF=${array[7]}
    PRESET=${array[8]} && RECORDTIME=${array[9]}
    SESSIONTIME=${array[10]}
    cd ..
}
```

#Muestra la configuración actual para la sesión de grabación

showConfig()

```
{
    echo
    tput sgr0
    tput setaf 5
    echo "-----"
    echo " La configuración escogida es:"
    echo " Ruta a los archivos FFmpeg: $RUTAFFMPEG"
    echo " Ruta al directorio de grabación deseado: $RUTAVIDEOS"
    echo " Resolución pantalla principal: $RES1"
    echo " Resolución pantalla secundaria: $RES2"
    echo " Tasa de refresco: $FPS fps"
    echo " Selecccion el valor CRF: $CRF "
    echo " Selecccion el preset: $PRESET"
    echo " Duracion de las grabaciones: $RECORDTIME "
    echo " Duración de las sesiones de video: $SESSIONTIME minutos"
    echo " Posiciones de los monitores: Principal: $POS1 Secundaria: $POS2"
    echo "-----"
}
```

#Método para la confirmación de que todos los valores son correctos

confirmacion()

```
{
    tput sgr0
    tput bold
    while true; do
        echo;read -p "Está seguro de los valores introducidos o desea reiniciar el proceso? (s/n)" yn
    done
}
```

```

        case $syn in
            s ) break;;
            n ) bash record.sh;;
            * ) echo "por favor responda sí(s) o no(n)";;
        esac
    done
}

#En caso de que exista la carpeta directamente entra en ella, en otro caso la crea y luego entra
gotDirectorio()
{
    if [ ! -d $1 ]; then
        mkdir $1
    fi
}
#-----

#Code
tput sgr0
tput cup 1 1
echo "Script basado en FFMPEG para la grabación de video y audio"

tput cup 2 2
echo "A continuación se muestran las configuraciones disponibles:" && cd Configuraciones
tput cup 3 4
ls
tput cup 5 1
read -p "¿Qué configuración desea utilizar? " FILE

loadFile $FILE
showConfig
confirmacion
gotDirectorio $RUTAVIDEOS
waitingTime=600
contador=$((SESSIONTIME/10))

while [ $contador>0 ]; do
    dateDMY=$(date +%d%b%Y)
    dateHM=`date +%H:%M`
    gotDirectorio $RUTAVIDEOS/$dateDMY
    gotDirectorio $RUTAVIDEOS/$dateDMY/$dateHM
    lanzarFFmpeg && contador=$((contador-1))
    echo "Press 'Ctrl+C' to exit from the record, next record will start in $waitingTime" && sleep
$waitingTime
done

```

3.3 videoRecord.sh

```

#!/bin/bash

#*****
#VideoRecord.sh
#*****

#-----
#Script encargado de la grabación del video generado por una pantalla.
#Se le pasan los siguientes parámetros necesarios:

```

```

#la ruta a las librerías y dependencias de FFMpeg,
#la ruta a donde se va a almacenar las grabaciones,
#resolución del monitor, CRF, preset, tiempo de grabación,
#posición del monitor (respecto al principal), le numero de pantalla.
#-----

clear

#Varibales
declare RUTAFFMPEG=$1
declare RUTAVIDEOS=$2
declare RES=$3
declare FPS=$4
declare CRF=$5
declare PRESET=$6
declare RECORDTIME=$7
declare POS=$8
declare NP=$9
declare NAME="Pantalla"
declare CURRENTDIR=`pwd`
#-----

#Funciones

#En caso de que exista la carpeta directamente entra en ella, en otro caso la crea y luego entra
gotDirectorio()
{
    if [ ! -d $1 ]; then
        mkdir $1
    fi
}
#-----

#Code
folder=$RUTAVIDEOS/$NAME$NP
gotDirectorio $folder
list=$folder/mylist-"`date +%H:%M`.txt

echo "Comenzando grabación ..."
echo -e "\aPuedes minimizar esta ventana, pues la grabación comenzará en 4 segundos..."
sleep 4

cd $RUTAFFMPEG
for (( i=0; i<10; i+=1)); do
    file=$folder/^date +%H:%M:%S:%4N`.mp4
    ./ffmpeg -t $RECORDTIME -f x11grab -video_size $RES -framerate $FPS -i :0.0+$POS -vcodec
    libx264 -preset $PRESET -crf $CRF -y $file && printf "file \"$file\"\n" >> $list
done
echo "Se va a proceder a generar el archivo concatenado de las 15 instancias de audio" && sleep 5
bash $CURRENTDIR/concat.sh $list $folder.mp4

```

3.4 audioRecord.sh

```

#!/bin/bash

#*****
#AudioRecord.ah
#*****

```



```
#-----
#Script encargado de la grabación del audio generado por el micro.
#Se le pasan los parámetros necesarios:
#la ruta a las librerías y dependencias de FFMpeg,
#la ruta a donde se va a almacenar las grabaciones, y la duración de estas.
#-----

clear

#Variables
declare RUTAFFMPEG=$1
declare RUTAVIDEOS=$2
declare RECORDTIME=$3
declare NAME="Audio"
declare CURRENTDIR=`pwd`
#-----

#Funciones

#En caso de que exista la carpeta directamente entra en ella, en otro caso la crea y luego entra
gotDirectorio()
{
    if [ ! -d $1 ]; then
        mkdir $1
    fi
}
#-----

#Code
folder=$RUTAVIDEOS/$NAME
gotDirectorio $folder
list=$folder/mylist-"`date +%H:%M`.txt

echo "Comenzando grabación ..."
echo -e "\aPuedes minimizar esta ventana, pues la grabación comenzará en 4 segundos..."
sleep 4

cd $RUTAFFMPEG
for (( i=0; i<10; i+=1)); do
    file=$folder/^date +%H:%M:%S:%4N`.mp3
    ./ffmpeg -f alsa -i hw:0 -codec:a libmp3lame -qscale:a 4 -t $RECORDTIME $file && printf "file
"$file"\n" >> $list
done
echo "Se va a proceder a generar el archivo concatenado de las 15 instancias de audio" && sleep 5
bash $CURRENTDIR/concat.sh $list $folder.mp3
```

3.5 concat.sh

```
#!/bin/bash

#*****
#concat.sh
#*****

#-----
#Este script se encarga de concatenar todos los archivos que vienen reflejados en el "directorio".
#Este es un .txt, en el que previamente se han grabado todos los archivos generados. Por lo que el script
```

```
#accede a este documento y va concatenando todas las entradas que va encontrando.
```

```
#-----
```

```
clear
```

```
#Variables
```

```
declare directorio=$1
```

```
declare folder=$2
```

```
#-----
```

```
#Code
```

```
cd directorio
```

```
ffmpeg -f concat -i $directorio -c copy $folder
```

3.6 resoluciones.sh

```
#!/bin/bash
```

```
#####
```

```
#resoluciones.sh
```

```
#####
```

```
#-----
```

```
#Script que muestra las resoluciones soportadas por todos los monitores conectados. También
```

```
#informa de la resolución recomendada mediante un asterisco
```

```
#-----
```

```
echo " ----- "
```

```
echo "A continuación se mostrarán las resoluciones de las pantallas conectadas(la que lleva asterisco funciona actualmente):"
```

```
xrandr
```

```
echo " ----- "
```

```
read -p "Presione una tecla para salir.... " enter
```

4. GENERACIÓN ARCHIVOS FINALES

4.1 fileGenerator.sh

```
#!/bin/bash

#####
#fileGenerator.sh
#####

#-----
#Su función es la de generar los archivos de reproducción final. Para ello hace uso de las funciones
#de FFmpeg concat para realizar la concatenación del video del monitor principal y del audio, y overlay,
#para construir un archivo de salida a partir de las 2 instancias de video y la de audio.
#-----

clear

#Variables
declare video1
declare video2
declare audio
declare output
declare folder
#-----

#Code
path=~/.Proyecto/Grabaciones
softwarePath=`pwd`

tput sgr0
tput bold
tput cup 1 1
echo "Script basado en FFmpeg para la generación de los archivos finales"
cd $path
tput sgr0
tput cup 3 2
echo "Dias disponibles"
ls
tput cup 6 2
read -p "Día? " dia
cd $dia
tput cup 8 2
echo "Sesiones disponibles"
ls
tput cup 10 2
read -p "Sesión? " sesion
cd $sesion

folder=$path/$dia/$sesion
video1=$folder/Pantalla1.mp4
video2=$folder/Pantalla2.mp4
video=$folder/Pantalla1Audio.mp4
audio=$folder/Audio.mp3
output=$folder/Overlay.mp4
```

```
#Concateno audio y pantalla1
cd ~/bin && ./ffmpeg -i $video1 -i $audio -c:v copy -c:a aac -strict experimental $video

#Genero archivo overlay con pantalla1, pantalla2, y audio
cd $softwarePath && gnome-terminal --tab -e "bash overlay.sh $video1 $video2 $audio $output"

#cd ~/bin && ./ffmpeg -i $video1 -i $video2 -i $audio -filter_complex "[0:v]setpts=PTS-STARTPTS,
pad=iw*2:ih[bg]; \
#[1:v]setpts=PTS-STARTPTS[fg]; [bg][fg]overlay=w" $output
```

5. REPRODUCCIÓN DE LA ESTACIÓN

5.1 playStation.sh

```
#!/bin/bash

#####
#playStation.sh
#####

#-----
#DESCRIPCION
#-----

clear

#Variables
declare video1
declare video2
declare audio
declare output
declare folder
#-----

#Code
path=~/.Proyecto/Grabaciones

tput sgr0
tput bold
tput cup 1 1
echo "Script basado en FFmpeg para la de una sesión en modo multipantalla"
cd $path
tput sgr0
tput cup 3 2
echo "Dias disponibles"
ls
tput cup 6 2
read -p "Día? " dia
cd $dia
tput cup 8 2
echo "Sesiones disponibles"
ls
tput cup 10 2
read -p "Sesión? " sesion
cd $sesion

folder=$path/$dia/$sesion
video1=$folder/Pantalla1Audio.mp4
video2=$folder/Pantalla2.mp4

vlc $video1 --input-slave=$video2
```

6. INSTALACIÓN

6.1 downloadSoft.sh

```
#!/bin/bash

#####
#downloadSoft.sh
#####

#-----
#Este script se encarga de conectarse a los repositorios necesarios para la descarga de todos
#los paquetes necesarios para le funcionamiento del programa.
#Crea una carpeta para las descargas donde se le indique en su llamada.
#-----

clear

#Variables
declare directorio=$1
#-----

#Funciones

#En caso de que exista la carpeta directamente entra en ella, en otro caso la crea y luego entra
gotDirectorio()
{
    if [ ! -d $1 ]; then
        mkdir $1
        cd $1
    elif then
        cd $1
    }
#-----

#Code -> PROBAR ESO

tput clear
tput cup 3 9
echo "Script encargado de la descarga de todo el software necesario"
tput cup 4 9
read -p "Introduzca por favor la ruta hasta el directorio en el que quiere almacenar todo el software descargado."
directorio
gotDirectorio $directorio

wget http://www.tortall.net/projects/yasm/releases/yasm-1.3.0.tar.gz
wget http://downloads.xvid.org/downloads/xvidcore-1.3.2.tar.gz
wget
http://downloads.sourceforge.net/project/lame/lame/3.99/lame-3.99.5.tar.gz?r=http%3A%2F%2Fsourceforge.net
%2Fprojects%2F%2Flame%2Ffiles%2F&ts=1346843156&use_mirror=dfn
wget http://downloads.sourceforge.net/faac/faac-1.28.tar.gz
wget
http://downloads.sourceforge.net/project/opencore-amr/opencore-amr/opencore-amr-0.1.3.tar.gz?r=http%3A%2
F%2Fsourceforge.net%2Fprojects%2Fopencore-amr%2Ffiles%2Fopencore-amr%2F&ts=1346843453&use_mir
ror=dfn
wget http://downloads.xiph.org/releases/ogg/libogg-1.3.0.tar.gz
```

```
wget http://downloads.xiph.org/releases/vorbis/libvorbis-1.3.3.tar.gz
wget http://downloads.xiph.org/releases/theora/libtheora-1.1.1.tar.bz2
git clone git://git.videolan.org/x264.git
wget http://downloads.sourceforge.net/opencore-amr/vo-aacenc-0.1.2.tar.gz
git clone http://git.chromium.org/webm/libvpx.git
wget http://zlib.net/zlib-1.2.8.tar.gz
git clone git://git.videolan.org/ffmpeg.git ffmpeg
```

6.2 fromPath.sh

```
#!/bin/bash
```

```
#####
#fromPath.sh
#####
```

```
#-----#
#Su función es la de intalar todas las librerías, dependencias, programas auxiliares, ...
#necesarios. Para ello inicialmente instala a través del repositorio. Está script logra la instalación
#en una máquina que corra una distribución linux del tipo Ubuntu.
#Descarga, descomprime, configura, compila e instala cada uno de los paquetes.
#-----
```

```
clear
```

```
#Variables
```

```
declare directorioDownload
```

```
declare directorioInstall
```

```
#-----
```

```
#Funciones
```

```
#En caso de que exista la carpeta directamente entra en ella, en otro caso la crea y luego entra
```

```
gotDirectorio()
```

```
{
  if [ ! -d $1 ]; then
    mkdir $1
    cd $1
  elif then
    cd $1
  }
}
```

```
#Extrae el archivo comprimido que se le pasa como argumento. Para ello compara su extensión con las
```

```
#listadas y en función de eso aplica una determinada descompresión.
```

```
extract () {
  if [ -f $1 ] ; then
    case $1 in
      *.tar.bz2) tar xjvf $1 ;;
      *.tar.gz) tar xzvf $1 ;;
      *.bz2)    bunzip2 $1 ;;
      *.rar)    unrar e $1 ;;
      *.gz)     gunzip $1 ;;
      *.tar)    tar xf $1 ;;
      *.tbz2)   tar xjf $1 ;;
      *.tgz)    tar xzf $1 ;;
      *.zip)    unzip $1 ;;
      *.Z)      uncompress $1 ;;
      *.7z)     7z x $1 ;;
    esac
  fi
}
```

```

*)    echo "$1' cannot be extracted via extract()" ;;
    esac
    else
        echo "$1' is not a valid file"
    fi
}
#-----

#Instala de los repositorios todos los paquetes y librerías necesarias
sudo apt-get update
sudo apt-get -y install autoconf automake build-essential libass-dev libfreetype6-dev libgpg-dev \
    libsdl1.2-dev libtheora-dev libtool libva-dev libvdpau-dev libvorbis-dev libx11-dev \
    libxext-dev libxfixes-dev pkg-config texi2html zlib1g-dev
#-----

tput clear
tput cup 3 9
echo "Script encargado de la instalación desde una carpeta de todo el software necesario"
tput cup 4 9
read -p "Introduzca por favor la ruta hasta el directorio en el que están almacenados todos los paquetes."
directorioDownload

tput cup 5 9
read -p "Introduzca por favor la ruta hasta el directorio en el que desea realizar la instalación"
directorioDownload
gotDirectorio directorioInstall

#YASM
extract directorioDownload/yasm-1.3.0.tar.gz
cd yasm-1.3.0
./configure --prefix="$HOME/Escritorio/PFC/Ubuntu/ffmpeg_build"
--bindir="$HOME/Escritorio/PFC/Ubuntu/ffmpeg_bin"
make
make install
make distclean
cd ..

#libx264
extract directorioDownload/last_x264.tar.bz2
cd x264-snapshot*
PATH="$PATH:$HOME/Escritorio/PFC/Ubuntu/ffmpeg_bin" ./configure
--prefix="$HOME/Escritorio/PFC/Ubuntu/ffmpeg_build"
--bindir="$HOME/Escritorio/PFC/Ubuntu/ffmpeg_bin" --enable-static --disable-opencl
PATH="$PATH:$HOME/Escritorio/PFC/Ubuntu/ffmpeg_bin" make
make install
make distclean
cd ..

#libopus
extract directorioDownload/xzvf_opus-1.1.tar.gz
cd opus-1.1
./configure --prefix="$HOME/Escritorio/PFC/Ubuntu/ffmpeg_build" --disable-shared
make
make install
make distclean
cd ..

#libpx

```



```

extract directorioDownload/xjvf libvpx-v1.3.0.tar.bz2
cd libvpx-v1.3.0
./configure --prefix="$HOME/Escritorio/PFC/Ubuntu/ffmpeg_build" --disable-examples
make
make install
make clean
cd ..

#FFmpeg
extract directorioDownload/xjvf ffmpeg-snapshot.tar.bz2
cd ffmpeg
PATH="$PATH:$HOME/Escritorio/PFC/Ubuntu/ffmpeg_build/bin"
PKG_CONFIG_PATH="$HOME/Escritorio/PFC/Ubuntu/ffmpeg_build/lib/pkgconfig" ./configure \
--prefix="$HOME/Escritorio/PFC/Ubuntu/ffmpeg_build" \
--extra-cflags="-I$HOME/Escritorio/PFC/Ubuntu/ffmpeg_build/include" \
--extra-ldflags="-L$HOME/Escritorio/PFC/Ubuntu/ffmpeg_build/lib" \
--bindir="$HOME/Escritorio/PFC/Ubuntu/ffmpeg_build/bin" \
--enable-gpl \
--enable-libass \
--enable-libfdk-aac \
--enable-libfreetype \
--enable-libmp3lame \
--enable-libopus \
--enable-libtheora \
--enable-libvorbis \
--enable-libvpx \
--enable-libx264 \
--enable-nonfree \
--enable-x11grab
PATH="$PATH:$HOME/Escritorio/PFC/Ubuntu/ffmpeg_build/bin" make
make install
make distclean
hash -r
cd ..

```

6.3 fromInternet.sh

```

#!/bin/bash

#*****
# fromInternet.sh
#*****

#-----#
#Su función es la de intalar todas las librerías, dependencias, programas auxiliares, ...
#necesarios. Para ello inicialmente instala a través del repositorio. Está script logra la instalación
#en una máquina que corra una distribución linux del tipo Ubuntu.
#Descarga, descomprime, configura, compila e instala cada uno de los paquetes.
#-----

clear

#Variables
declare directorioDownload
declare directorioInstall
#-----

#Funciones

```

```

#En caso de que exista la carpeta directamente entra en ella, en otro caso la crea y luego entra
gotDirectorio()
{
    if [ ! -d $1 ]; then
        mkdir $1
        cd $1
    elif then
        cd $1
    }

#Extrae el archivo comprimido que se le pasa como argumento. Para ello compara su extensión con las
#listadas y en función de eso aplica una determinada descompresión.
extract () {
    if [ -f $1 ] ; then
        case $1 in
            *.tar.bz2) tar xjvf $1 ;;
            *.tar.gz) tar xzvf $1 ;;
            *.bz2) bunzip2 $1 ;;
            *.rar) unrar e $1 ;;
            *.gz) gunzip $1 ;;
            *.tar) tar xf $1 ;;
            *.tbz2) tar xjf $1 ;;
            *.tgz) tar xzf $1 ;;
            *.zip) unzip $1 ;;
            *.Z) uncompress $1 ;;
            *.7z) 7z x $1 ;;
            *) echo "$1' cannot be extracted via extract()" ;;
        esac
    else
        echo "$1' is not a valid file"
    fi
}

#-----

#Instala de los repositorios todos los paquetes y librerías necesarias
sudo apt-get update
sudo apt-get -y install autoconf automake build-essential libass-dev libfreetype6-dev libgpg-dev \
    libssl1.2-dev libtheora-dev libtool libva-dev libvdpau-dev libvorbis-dev libx11-dev \
    libxext-dev libxfixes-dev pkg-config texi2html zlib1g-dev
#-----

tput clear
tput cup 3 9
read -p "Instalador del Software desde Internet"

bash downloadSoft.sh

#YASM
extract directorioDownload/yasm-1.3.0.tar.gz
cd yasm-1.3.0
./configure --prefix="$HOME/Escritorio/PFC/Ubuntu/ffmpeg_build"
--bindir="$HOME/Escritorio/PFC/Ubuntu/ffmpeg_bin"
make
make install
make distclean
cd ..

```

```

#libx264
extract directorioDownload/last_x264.tar.bz2
cd x264-snapshot*
PATH="$PATH:$HOME/Escritorio/PFC/Ubuntu/ffmpeg_bin" ./configure
--prefix="$HOME/Escritorio/PFC/Ubuntu/ffmpeg_build"
--bindir="$HOME/Escritorio/PFC/Ubuntu/ffmpeg_bin" --enable-static --disable-openc
PATH="$PATH:$HOME/Escritorio/PFC/Ubuntu/ffmpeg_bin" make
make install
make distclean
cd ..

#libopus
extract directorioDownload/xzvf opus-1.1.tar.gz
cd opus-1.1
./configure --prefix="$HOME/Escritorio/PFC/Ubuntu/ffmpeg_build" --disable-shared
make
make install
make distclean
cd ..

#libpx
extract directorioDownload/xjvf libvpx-v1.3.0.tar.bz2
cd libvpx-v1.3.0
./configure --prefix="$HOME/Escritorio/PFC/Ubuntu/ffmpeg_build" --disable-examples
make
make install
make clean
cd ..

#FFmpeg
extract directorioDownload/xjvf ffmpeg-snapshot.tar.bz2
cd ffmpeg
PATH="$PATH:$HOME/Escritorio/PFC/Ubuntu/ffmpeg_bin"
PKG_CONFIG_PATH="$HOME/Escritorio/PFC/Ubuntu/ffmpeg_build/lib/pkgconfig" ./configure \
--prefix="$HOME/Escritorio/PFC/Ubuntu/ffmpeg_build" \
--extra-cflags="-I$HOME/Escritorio/PFC/Ubuntu/ffmpeg_build/include" \
--extra-ldflags="-L$HOME/Escritorio/PFC/Ubuntu/ffmpeg_build/lib" \
--bindir="$HOME/Escritorio/PFC/Ubuntu/ffmpeg_bin" \
--enable-gpl \
--enable-libass \
--enable-libfdk-aac \
--enable-libfreetype \
--enable-libmp3lame \
--enable-libopus \
--enable-libtheora \
--enable-libvorbis \
--enable-libvpx \
--enable-libx264 \
--enable-nonfree \
--enable-x11grab
PATH="$PATH:$HOME/Escritorio/PFC/Ubuntu/ffmpeg_bin" make
make install
make distclean
hash -r
cd ..

```

7. INFORMACIÓN

7.1 userGuide.sh

```
#!/bin/bash

#*****
#                               userGuide.sh
#*****

#-----
#Información de uso
#-----

#Code

tput clear

# Movemos el cursor a la localización X,Y (la esquina izquierda es 0,0)
tput cup 3 55

# Configuramos el color del fondo a traves del condigo de escape ANSI
tput setaf 3
echo "Indra SoftLabs"
tput sgr0

tput cup 5 50
# Seleccionamos el modo de video "revertido"
tput rev
echo "G U I A   D E   U S O"
tput sgr0

tput cup 7 7
echo "Este manual tiene como intención el de ofrecer algunas pautas de uso para el prototipo desarrollado."

tput cup 8 7
echo "A través de las diferentes entradas del menu principal se puede acceder a las diferentes funcionalidades del programa."

tput cup 10 7
echo "La primera entrada del menú podemos realizar las siguientes tareas:"

tput cup 11 9
echo "1. Se lanza el asistente para la creación de una nueva configuración de grabación. Entre los parámetros a introducir,"
tput cup 12 10
echo "destacar los relativos a las posiciones de los monitores, recordar el principal viene asociado con el valor 0, mientras"
tput cup 13 10
echo "que el secundario tomará el valor de la resolución horizontal del primario, si está conectado a la derecha de este, si"
tput cup 14 10
echo "si estuviese a la izquierda sería el mismo valor pero negativo. Por otro lado, los valores de codificación se recomienda"
tput cup 15 10
```

```
echo "emplear 18 para el CRF y ultrafast para el preset. Más posibilidades se encuentran en el documento 4
Manual de Usuario "
```

```
tput cup 16 9
echo "2. Se lanza el reproductor en el cual solo habrá que seleccionar la configuración que deseemos emplear en
la grabación"
tput cup 17 10
echo "de la sesión que vamos a iniciar. El proceso se realiza las veces necesarias para poder grabar el tiempo
establecido en"
tput cup 18 10
echo "el archivo. Si se desea detenerlo, solo hace falta pulsar Cntrl+C en la ventana del menu."
```

```
tput cup 20 7
echo "La segunda entrada es la encargada de lanzar el script que generará los archivos finales codificados,
concatenados y"
tput cup 21 7
echo "superpuestos. A traves de un diálogo con el usuario se accede a la sesión con la que se va a trabajar."
```

```
tput cup 22 7
echo "La tercera entrada permite reproducir una sesión grabada mediante dos instancias, por lo que se pueden ver
y escuchar"
tput cup 23 7
echo "todos los flujos audiovisuales grabados."
```

```
tput cup 25 7
echo "La cuarta entrada permite la descarga e instalación de todo el software necesario, puede encontrar más
información de"
tput cup 26 7
echo "esta entrada seleccionando la opción 4 del menu."
```

```
tput cup 28 7
echo "Por último las dos últimas entradas permiten mostrar esta información, al igual que la relacionada con la
instalación"
tput cup 29 7
echo "del software o las licencias. También la opción para abandonar el presente programa."
```

```
# Seleccionamos modo "negrita"
```

```
tput bold
```

```
tput cup 32 15
```

```
read -p "Presion 'q' para salir o 'x' para mostrar el manual de instalación: " opcion
```

```
case $opcion in
```

```
  x) bash installationGuide.sh;;
```

```
  q) exit 1;;
```

```
*)
```

```
  tput clear
```

```
  tput sgr0
```

```
esac
```

```
done
```

7.2 installationGuide.sh

```
#!/bin/bash

#*****
#          installationGuide.sh
#*****

#-----
#Información para la instalación
#-----

#Code

tput clear

# Movemos el cursor a la localización X,Y (la esquina izquierda es 0,0)
tput cup 3 55

# Configuramos el color del fondo a traves del condigo de escape ANSI
tput setaf 3
echo "Indra SoftLabs"
tput sgr0

tput cup 5 45
# Seleccionamos el modo de video "revertido"
tput rev
echo "G U I A   D E   I N S T A L A C I Ó N"
tput sgr0

tput cup 7 7
echo "Este manual tiene como intención el de ofrecer algunas pautas para la instalación del software necesario."

tput cup 10 7
echo "A partir de tres entradas del submenu, se pueden realizar las siguientes acciones:"

tput cup 11 8
echo "1.Nos permite descargar todo el software necesario en el directorio que se le indique a través de un diálogo con"
tput cup 12 9
echo "el script."

tput cup 14 8
echo "2.Esta segunda entrada permite instalar el software necesario descargado previamente en otro equipo con conexión a"
tput cup 15 9
echo "internet. Para ello le especificamos el directorio en el que están los paquetes descargados y el directorio sobre el"
tput cup 16 9
echo "quequeremos realizar la instalación, y este procederá con todos los pasos: configuración, compilación e instalación."

tput cup 19 7
echo "Por último esta estrad9a permite realizar todo el proceso, desde la descarga hasta la instalación automaticamente."
tput cup 20 7
echo "Es la indicada cuando tenemos un equipo con acceso a internet."
```

```
# Seleccionamos modo "negrita"
tput bold
tput cup 23 15
read -p "Presion 'q' para salir o 'x' para mostrar el manual de uso: " opcion

case $opcion in
  x) installationGuide.sh;;
  q) exit 1;;
*)
  tput clear
  tput sgr0
esac
done
```

7.3 credits.sh

```
#!/bin/bash

#*****
#                               credits.sh
#*****

#-----
#Créditos y licencias de uso
#-----

#Code

tput clear

# Movemos el cursor a la localización X,Y (la esquina izquierda es 0,0)
tput cup 3 55

# Configuramos el color del fondo a traves del condigo de escape ANSI
tput setaf 3
echo "Indra SoftLabs"
tput sgr0

tput cup 5 50
# Seleccionamos el modo de video "revertido"
tput rev
echo "L I C E N C I A S   D E   U S O"
tput sgr0

tput cup 7 7
echo "Este proyecto fue desarrollado por Yúsef Habib Fernández para la empresa Indra Software Labs,"
tput cup 8 7
echo "como Proyecto Final de Carrera para la obtención del título de Ingeniero Superior de Telecomunicación"
tput cup 9 7
echo "el cual fue ofertado a través de la Cátedra Indra de la Universidad de Ovideo."

tput cup 11 7
echo "Todos los programas que son necesarios han sido liberados bajo licencia GNU GPL por lo que se garantiza la libertad"
```

```
tput cup 12 7
echo "para compartir y modificar el software cubierto por ella, asegurando la liberta de este software para todos
los usuarios"
tput cup 13 7
echo "ya sean inviduos o empresas."

# Seleccionamos modo "negrita"
tput bold
tput cup 16 15
read -p "Presion 'q' para salir o 'x' para mostrar instalacion: " opcion

case $opcion in
q) exit 1;;

*)
tput clear
tput sgr0
esac
done
```