

PROYECTO FINAL: REPRODUCTOR MULTIMEDIA(PHOENIX)

RUIZ AGUILAR EDUARDO

UNIVERSIDAD NACIONAL AUTÓNOMA DE MEXICO

CIUDAD DE MÉXICO
16 DE AGOSTO 2021

OBJETIVO

Creación de centro multimedia donde se pueden reproducir películas, vídeos, música, y fotografías haciendo uso de la Raspberry Pi 3 model b y a través de sus puertos usb leer un medio extraíble (USB) para observar contenido mixto.

LISTA DE MATERIALES DE COMPONENTES ELECTRÓNICOS, INCLUYENDO POSIBLES EQUIVALENTES SI ASÍ FUERE NECESARIO.

Raspberry pi 3 model b.

Cable de alimentación
para la Raspberry que
debe suministrar 5V a
3A.

Cable de red ethernet

Monitor con HDMI

Teclado alámbrico USB.

Mouse USB

DESCRIPCIÓN DEL FUNCIONAMIENTO DE LOS COMPONENTES ELECTRÓNICOS RELEVANTES (CIRCUITOS INTEGRADOS, ENCAPSULADOS, CONTROLADORES, ETCÉTERA)

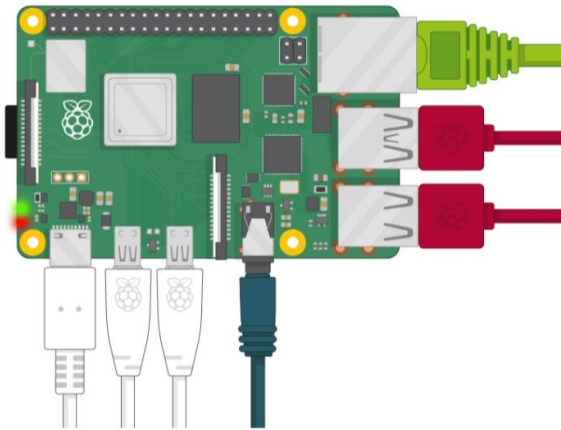


Imagen 1: Componentes de la Raspberry

Imagen obtenida de: <https://projects.raspberrypi.org/es-LA/projects/raspberry-pi-setting-up/>

En la imagen se pueden observar los componentes de la raspberry y su manera de conectarlos; al ser una computadora de bajo costo su poder de cómputo es menor, pero esto no quiere decir que deje de ser funcional ya que se puede trabajar en ella y hacer las tareas que realiza con computador.

Elementos que contiene:

Puertos USB	Entrada para microSD	HDMI
Fuente de alimentación de hasta 2.5 A	Base o entrada para Ethernet	GPIO(Son los pines de entrada y salida)

Cada uno de estos elementos ayudan a que se pueda visualizar la información de manera correcta y trabajar en casos prácticos para la implementación de diferentes herramientas que puedan ser una solución

INFORMACIÓN SOBRE EL CUIDADO DE LA SALUD Y ADVERTENCIAS DE RIESGOS (EJ. QUEMADURAS, ELECTROCUCIÓN, ETCÉTERA)

Evitar quemaduras: Si no se quiere correr riesgo de una posible quemadura se debe tener cautela para que la fuente de alimentación no sea mayor a 5V.

Se debe guardar la tarjeta en su bolsa antiestática.

Todos los modelos de Raspberry Pi realizan un cierto grado de gestión térmica para evitar el sobrecalentamiento con cargas pesadas. Los SoC tienen un sensor de temperatura interno, cuyo software en la GPU sondea para garantizar que las temperaturas no superen un límite predefinido; esto es 85° Celsius en todos los modelos. Es posible establecer esto en un valor más bajo, pero no en uno más alto. A medida que el dispositivo se acerca al límite, se reducen varias frecuencias y, a veces, los voltajes utilizados en el chip (ARM, GPU). Esto reduce la cantidad de calor generado, manteniendo la temperatura bajo control. (Raspberry Pi, s.f.)¹

INFORMACIÓN SOBRE EL CUIDADO DE LOS COMPONENTES ELECTRÓNICOS DELICADOS (EJ. MEMORIAS) QUE ASÍ LO REQUIERAN, TALES COMO USO DE PULSERA ANTI-ESTÁTICA, ACUMULACIÓN DE GRASA EN TERMINALES, LÍMITES DE TEMPERATURA, ETCÉTERA

Evitar alguna pérdida o romper el dispositivo de memoria SSD.

Evitar tocar la tarjeta Raspberry durante su funcionamiento.

Evitar que los cables de ethernet se rompan o se enrollen durante la conexión.

Aunque la Raspberry cuenta con una bolsa antiestática se recomienda el uso de una pulsera antiestática.

Otro error más o menos frecuente a la hora de usar una Raspberry Pi es no tener en cuenta que se trata de un ordenador. En miniatura, sí, pero un ordenador, al fin y al cabo. Eso significa que cuanto más tiempo la uses y más le exijas, mayor será la necesidad de refrigerar su CPU. Déjale el espacio necesario para que disipe el calor o te llevarás un susto. Más allá del peligro de fusión por calor -o algo similar- el problema es que, a mayor temperatura, menor rendimiento de su procesador. (Álvarez, 2019)²

CONFIGURACIÓN DE LA TARJETA CONTROLADORA (EJ. RASPBERRY PI) O CIRCUITO MICROCONTROLADOR (EJ. MICROCHIP PIC A UTILIZAR, INCLUYENDO PUERTOS, PERIFÉRICOS, CONFIGURACIONES DEL SISTEMA OPERATIVO, PAQUETES Y LIBRERÍAS A INSTALAR, ETCÉTERA.

Con el uso de la Raspberry Pi 3 model b de forma física necesitaremos contar con un cable ethernet, una microSD se conectará a la computadora para guardar la distribución del sistema operativo a utilizar; el seleccionado fue Raspian 10.

Los pasos para configurar son los siguientes:

-Instalación del sistema operativo Raspian10.

-Para la memoria SD se tuvo que bootear para que pudiera funcionar mediante la conexión de puerto USB en la tarjeta Raspberry.

-Para la configuración no se contaba con un teclado y un mouse por lo que se tuvo que realizar una conexión vía remota con una dirección IP.

-Conectamos la fuente de alimentación a la Raspberry.

-Conectamos el cable ethernet a un modem.

-Realizamos la descarga del programa NMAP, su función nos servirá obtener los puertos de nuestra dirección IP.

-Ya instalado NMAP podemos observar el segmento de red para darnos cuenta cual está teniendo comunicación con la tarjeta Raspberry. Realizamos la descarga del programa Putty, su función es conectarse a través de un canal seguro que permitirá conectarnos de forma remota con la tarjeta Raspberry Pi.

-Procedemos a instalar el programa VNC, su función será proporcionarnos una interfaz gráfica en nuestra tarjeta Raspberry. Hay que recordar que la comunicación se realiza mediante SSH.

-Se despliega una terminal dentro de las RaspberryPi e instalamos la paquetería apt-get install screenfetch, su función será observar el sistema operativo.

-Ahora procedemos a instalar la paquetería de VNC viewer con la que observaremos nuestra interfaz gráfica.

```
pi@raspberrypi:~$ sudo fdisk -l
Command (m for help): Partition type
  0 = primary (1 primary, 0 extended, 3 free)
  1 = extended (contains for logical partitions)
Select (default p): Partition number (2-4, default 2): First sector (2048-606207
99, default 2048): Last sector, +/-sectors or +/-size(K,M,G,T,P) (532480-6062079
9, default 60620799):
Created a new partition 2 of type 'Linux' and of size 28.7 GiB.
Partition 2 contains a data signature.

Command (m for help):
Disk /dev/mmcblk0: 28.9 GiB, 31037849600 bytes, 60620800 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x97709164

Device      Boot  Start      End  Sectors  Size Id Type
/dev/mmcblk0p1      8192    532479    524288    256M  c W95 FAT32 (LBA)
/dev/mmcblk0p2    532480 60620799 60088320 28.7G 83 Linux

Command (m for help): The partition table has been altered.
Syncing disks.

pi@raspberrypi:~$ sudo apt-get install -y realvnc-vnc-server realvnc-vnc-viewer
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
E: No se ha podido localizar el paquete realvnc-vnc-server
pi@raspberrypi:~$ sudo apt-get install -y realvnc-vnc-server realvnc-vnc-viewer
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
El paquete indicado a continuación se instaló de forma automática y ya no es nec
esario.
  libmicrodnss0
Utilice «sudo apt autoremove» para eliminarlo.
Paquetes recomendados:
  cups
Se actualizarán los siguientes paquetes:
  realvnc-vnc-server realvnc-vnc-viewer
2 actualizados, 0 nuevos se instalarán, 0 para eliminar y 189 no actualizados.
Se necesita descargar 90355 kB de archivos.
Se utilizarán 12.3 kB de espacio de disco adicional después de esta operación.
```

Imagen 2: Se muestran las configuraciones para la raspberry

-Una vez finalizado lo siguiente es abrir el programa de VNC para establecer la conexión, proporcionándole la dirección IP y el nombre.

-Ahora se puede observar la interfaz gráfica de la Raspberry Pi.

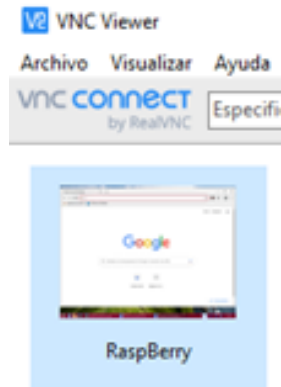


Imagen 3: Interfaz gráfica a utilizar

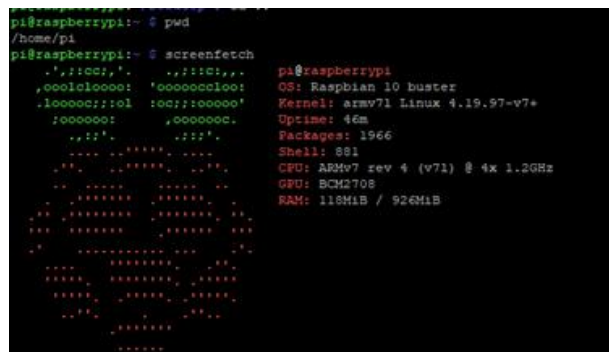


Imagen 4: Especificaciones de la tarjeta

DESARROLLO DE LOS COMPONENTES DE SOFTWARE

Los componentes de software que se utilizaron se listan a continuación:

Librería tkinter

Facilito el manejo de la interfaz con el uso de clases para la experiencia del usuario.

Librería para el navegador

Web browser: Con ella se estableció una conexión al navegador de una página en internet.

Librería vlc

Reproduce archivos como imágenes, video o música

Librería Pygame

Modulo que reproduce el audio

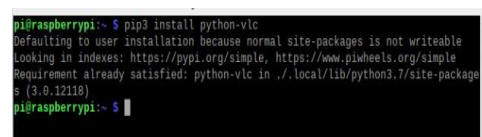


Imagen 5: Instalación de librería

```

1  """
2  #####
3  Proyecto Final: Media Center (PHOENIX)
4  Fundamentos de Sistemas Embebidos
5  Semestre 2021-2
6  Creación: 16/08/2021
7  Última modificación: 17/08/2021
8
9  Integrantes:González Pacheco Leonardo Alonso
10             Hernández Hernández Edgar
11             Ruiz Aguilar Eduardo
12  #####
13  """
14
15  import pygame
16  import threading
17  import vlc
18  import os
19  from tkinter import*
20  from PIL import ImageTk, Image

```

Imagen 6: Código donde se observan librerías implementadas.

```

102  memoria USB"""
103  class Ventana2:
104      #Constructor de la segunda ventana
105      def __init__(self,ventana,opcion):
106          self.opcion=opcion
107          self.ventana=ventana
108          ventana.title("Interfaz Memoria")
109          ventana['background']='#000000'
110
111          ext_audio=[".mp3"]
112          musica=encontrar(ext_audio)
113
114          ext_video=[".mp4",".mkv"]
115          videos=encontrar(ext_video)
116
117          ext_imagenes=[".jpg",".png",".jfif",".jpeg"]
118          fotos=encontrar(ext_imagenes)
119
120          if(len(musica)==0 and len(videos)==0 and len(fotos)==0):
121              self.lab=Label(ventana, bg="black", fg="white", font=('Arial',15),text="No hay contenido").place(x=570,y=170)

```

Imagen 7: Código para la lectura de archivos multimedia en la memoria.

```

52  class Ventana:
53      #Constructor de la ventana
54      def __init__(self,ventana):
55          self.ventana=ventana#El atributo ventana se inicializa con el
56                               #valor mandado por el constructor, que es un
57                               #objeto Tk, el cual permite crear ventanas con
58                               #diferentes objetos dentro de ella
59
60          self.opcion = 0
61          ventana['background']='#000000'
62          ventana.title("Reproductor Phoenix")#Ponemos el título a la ventana
63          self.imagen = ImageTk.PhotoImage(Image.open(r'/home/pi/Documents/CentroMultimedia/Imagenes/netflix.png').resize((100,100)))
64          self.boton = Button(ventana, image = self.imagen, width= 100, height=100,relief="flat", borderwidth=0, bg='#000000')
65
66          self.imagen1 = ImageTk.PhotoImage(Image.open(r'/home/pi/Documents/CentroMultimedia/Imagenes/Spotify.png').resize((100,100)))
67          self.boton1 = Button(ventana, image = self.imagen1, width= 100, height=100,relief="flat", borderwidth=0, bg='#000000')
68
69          self.imagen2 = ImageTk.PhotoImage(Image.open(r'/home/pi/Documents/CentroMultimedia/Imagenes/Amazon.png').resize((100,100)))
70          self.boton2 = Button(ventana, image = self.imagen2, width= 100, height=100,relief="flat", borderwidth=0, bg='#000000')
71
72          self.imagen3 = ImageTk.PhotoImage(Image.open(r'/home/pi/Documents/CentroMultimedia/Imagenes/Disney.png').resize((100,100)))
73          self.boton3 = Button(ventana, image = self.imagen3, width= 100, height=100,relief="flat", borderwidth=0, bg='#000000')

```

Imagen 8: Botones utilizados.

```

422 if len(arr)!=0 and len(arr2)==0 and len(arr3)==0:
423     a=1
424 elif len(arr)==0 and len(arr2)!=0 and len(arr3)==0:
425     a=3
426 elif len(arr)==0 and len(arr2)==0 and len(arr3)!=0:
427     a=2
428     '''En dado que no sea el caso, se debe de lanzar la segunda
429     ventana para elegir que archivo reproducirá'''
430 else:
431     root=Tk()
432     a=0
433     root.attributes("-zoomed",True)
434     vent=Ventana2(root,a)
435
436     root.mainloop()
437     a=vent.regresa()#Este valor determina que archivo se
438                     #reproducirá
439 '''a==1 es audio'''
440 if a==1:
441     ent=pygame.mixer.init()

```

Imagen 9: Método utilizado para la reproducción de audio.

```

475 elif(valor==1):
476     comando="/usr/bin/chromium-browser %s"
477     nav=webbrowser.get(comando)#Se utiliza el método get para obtener el navegador
478     webbrowser.register("chrome",None,nav)#Con este método se registra el navegador Chrome
479                                         #con el nombre de chrome
480     nav=webbrowser.get("chrome")#Se obtiene el navegador para utilizarlo
481     nav.open("https://www.netflix.com/mx/login")
482
483 elif(valor==2):
484     comando="/usr/bin/chromium-browser %s"
485     nav=webbrowser.get(comando)
486     webbrowser.register("chrome",None,nav)
487     nav=webbrowser.get("chrome")
488     nav.open("https://accounts.spotify.com/es/login/?continue=https:%2F%2Fwww.spotify.com%2Fapi%2Fgrowth%2F12l-redi")
489
490 elif(valor==3):
491     comando="/usr/bin/chromium-browser %s"
492     nav=webbrowser.get(comando)
493     webbrowser.register("chrome",None,nav)
494     nav=webbrowser.get("chrome")

```

Imagen 10: Direcccionamiento a plataformas de streaming.

```
503
504     else:
505         bandera=False#Para terminar con el ciclo while
```

Imagen 11: Línea que termina la ejecución del programa.



Imagen 12: Raspberry conectada a monitor y a modem con cable ethernet.

INTEGRACIÓN DE LOS COMPONENTES DE SOFTWARE O MÓDULOS EN UNA SOLUCIÓN DE SOFTWARE

El uso de librerías que logramos integrar en el proyecto dio respuesta a las necesidades del usuario. Se utilizaron los puertos USB para leer la memoria dentro de la Raspberry Pi, el uso de la microSD para almacenar nuestro sistema operativo y la función del cable ethernet para conectarnos a internet, logrando así poder leer archivos como imágenes, videos y fotos.

Uso de botones para:

Dirigir a servicios de streaming

Observar contenido de la memoria

Salir del contenido de memoria

Siguiente

Atrás

Pausa

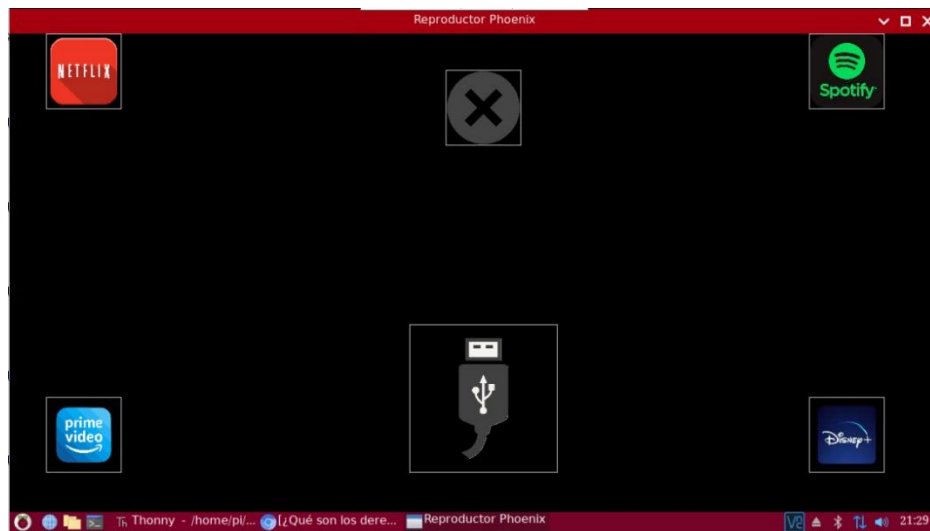


Imagen 13: Botones implementados en la interfaz con las librerías descritas anteriormente.

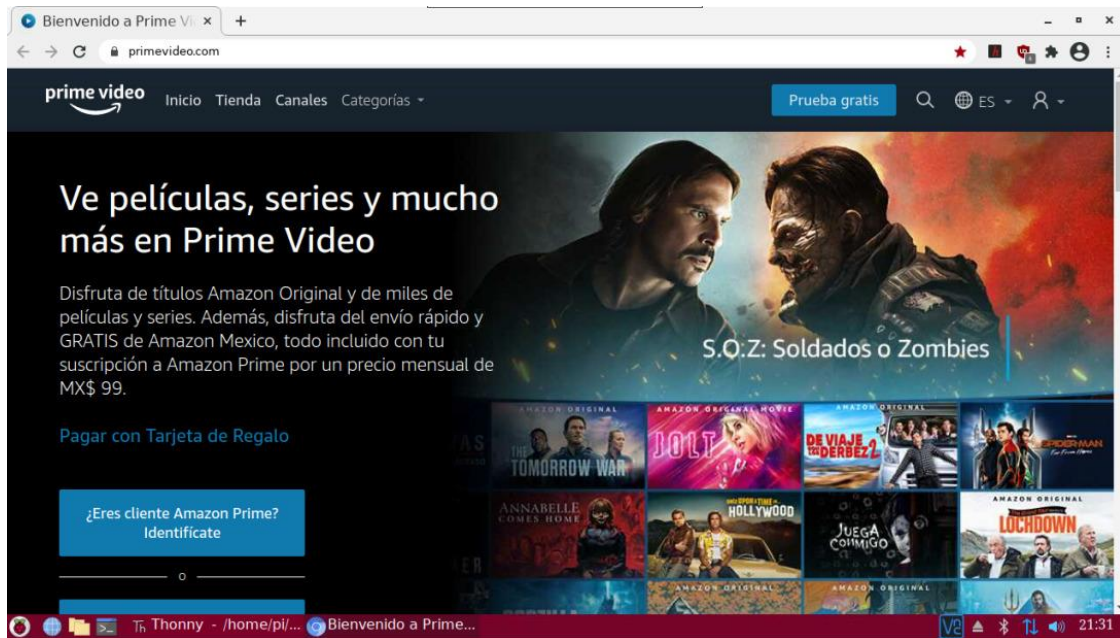


Imagen 14: Botón que abre el navegador para ir a Amazon.

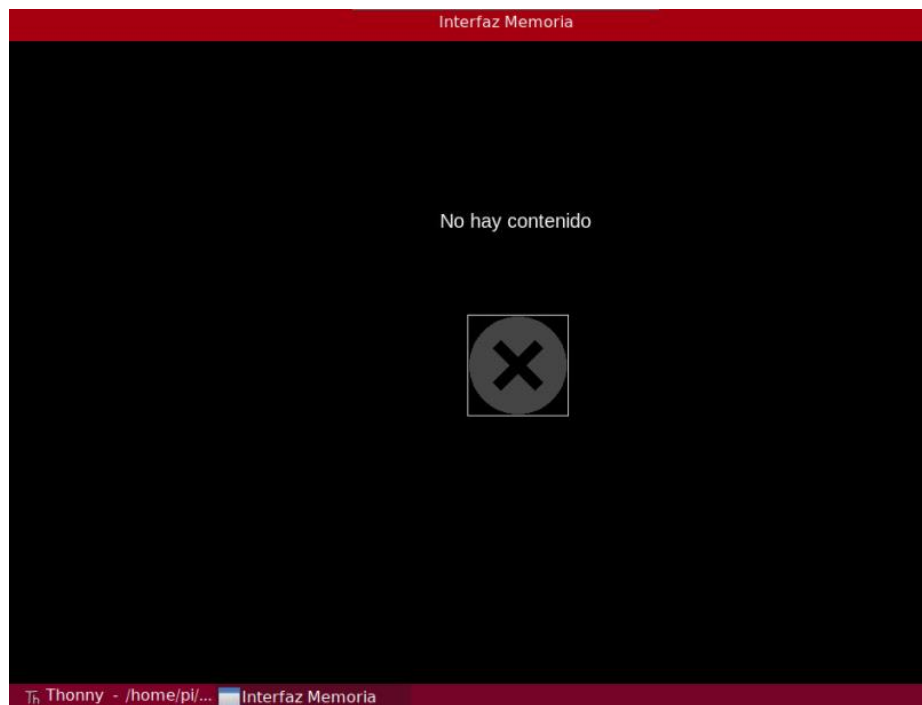


Imagen 15: Memoria insertada sin archivos multimedia.

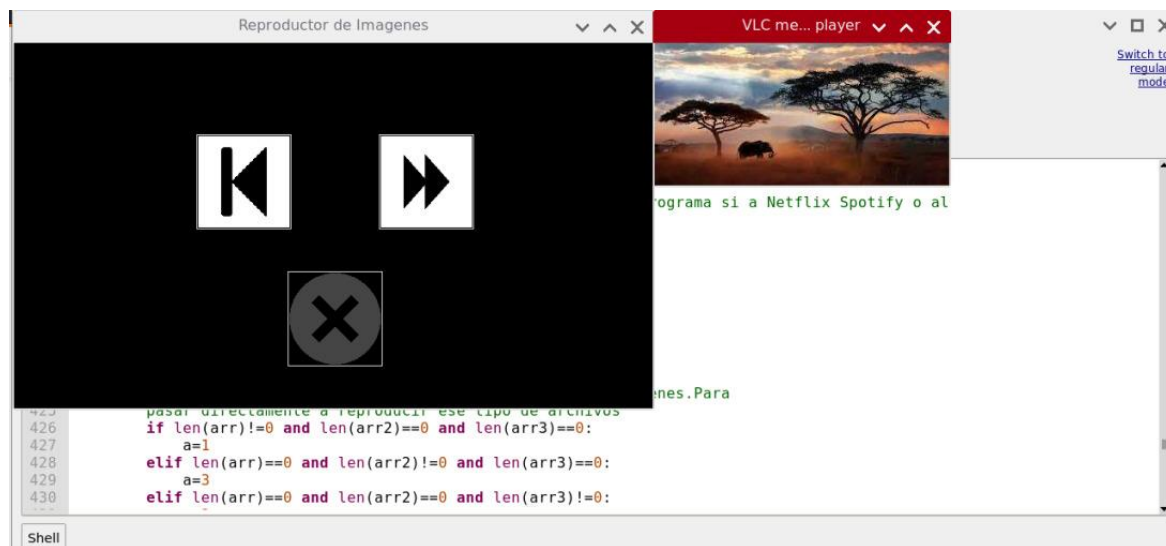


Imagen 16: Memoria con contenido y botones para visualizar los archivos.

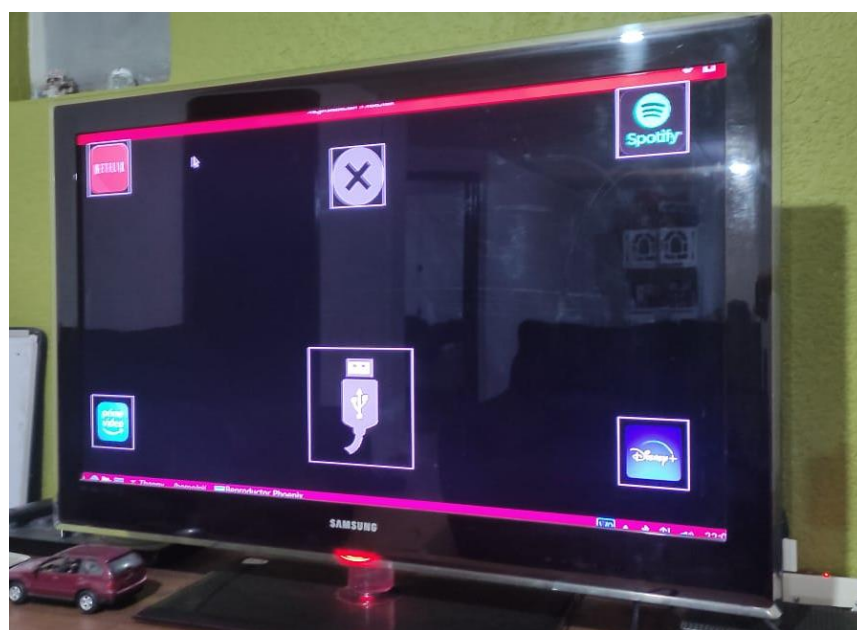


Imagen 17: Centro multimedia en monitor conectado.

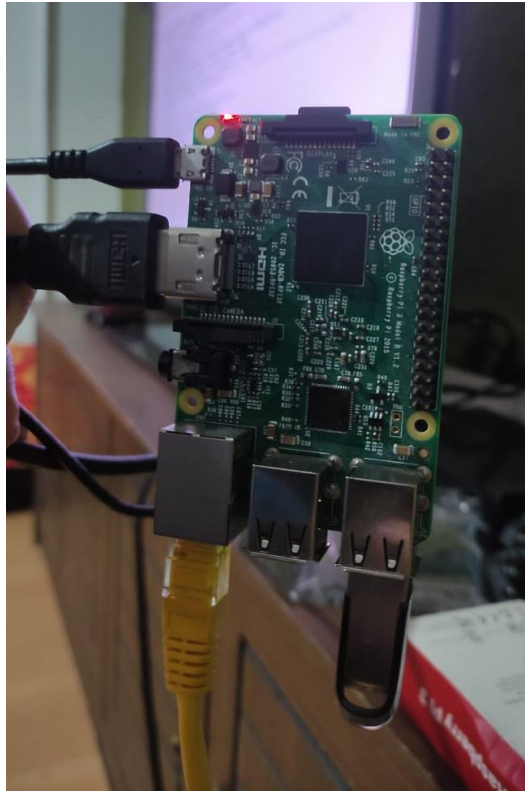


Imagen 18: Memoria conectada a Raspberry.

CUESTIONARIO

¿Por qué el uso de una Raspberry Pi física a una simulada?

R: Al realizar el análisis respecto a lo que necesitaríamos del proyecto pudimos observar que en cuestión de realizarlo simulado tendríamos que cuidar lo que nos comentó el profesor acerca de que sea compatible no solo para Intel si no igual para toda arquitectura de ARM. Además, teníamos la oportunidad de que el proyecto se mostrara mejor implementado si lo hacíamos con una tarjeta física.

¿Hubo complicaciones durante el desarrollo?

R: Si hubo una serie de complicaciones respecto al desarrollo porque algunas librerías que se utilizó cuando implementamos los botones no eran compatibles, además a la hora que quisimos mejorar la interfaz y le cambiábamos al código dejaba de ser funcional, se pudo resolver gracias a que íbamos creando versiones durante el desarrollo de nuestro código.

¿El objetivo fue cumplido?

R: Si totalmente fue cumplido nuestro objetivo porque pudimos implementar el centro de multimedia con un uso correcto, además pudimos crear una interfaz agradable y funcional.

CONCLUSIONES

Fue difícil lograr hacer que la memoria física de las Raspberry Pi se leyera mediante el puerto USB, debido a algunos errores que se nos presentaron a la hora de generar las librerías usadas en Python, además nunca habíamos manejado una Raspberry física sino más bien simulada. Optamos por usar la Raspberry física por las funcionalidades que puede realizar la tarjeta, con respecto a la solución pudimos usar una serie de botones para mostrar una solución mejor desarrollada.

En cuanto al aprendizaje del proyecto me agrado la forma en que pude observar como funciona una Raspberry Pi, donde demostramos lo aprendido en la clase de sistemas embebidos. Se reforzó la teoría con la práctica porque los conocimientos adquiridos complementaron de forma adecuada y simplificaron la solución.

REPOSITORIO PARA DESCARGAR EL CÓDIGO

<https://github.com/LeoAlonsoGonzalez/CentroMultimediaPhoenix>

BIBLIOGRAFÍA

¹Raspberry Pi. (s.f.). Documentación de Raspberry Pi. Recuperado el 16 de agosto de 2021, de <https://www.raspberrypi.org/documentation/computers/raspberry-pi.html>

²Álvarez, E. (25 de mayo de 2019). Computerhoy. Recuperado el 16 de agosto de 2021, de <https://computerhoy.com/listas/tecnologia/5-precauciones-debes-tomar-utilizar-raspberry-pi-primera-vez-427805>

³ Botón de Python - tkinter - programador clic. (s. f.). Botón de Python. Recuperado 16 de agosto de 2021, de <https://programmerclick.com/article/87091959012/>

⁴ The Raspberry Pi Foundation. (s. f.). Buy a 3 Model B –. Raspberry Pi. Recuperado 16 de agosto de 2021, de <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>