

# Proyecto Final.

## Centro Multimedia

Báez Cadena Diestefano Michel

16 Enero 2023

## 1 Objetivo

Elaborar un sistema embebido utilizando los conocimientos adquiridos a lo largo del semestre.

## 2 Introducción

El proyecto final consiste en un prototipo de centro multimedia, en el cual se podrá acceder a distintas plataformas de streaming por ejemplo, *Netflix*, *HBO Max*, *Spotify*, *Youtube*, *etc.* Así como leer un dispositivo USB para reproducir archivos (*.mp3*, *.mp4* *.jpg*).

## 3 Antecedentes

A lo largo del semestre realizamos diferentes tipos de prácticas para conocer los componentes, utilidades y usos que le podíamos dar a la Raspberry Pi, todos esos conocimientos sirven de antecedente para poder diseñar, crear e implementar este proyecto, ya que se utilizan diversos conocimientos adquiridos durante el curso.

## 4 Materiales

- Raspberry Pi 3B+ con su respectivo eliminador 5V 2A
- Teclado inalámbrico tipo "Gamepad" para poder manipular el cursor y el teclado cuando sea requerido.
- Monitor con entrada HDMI.
- Cable HDMI-HDMI para conectar la tarjeta al monitor.

## 5 Desarrollo

### 5.1 Información sobre el cuidado de la salud y advertencias de riesgos

- Se puede llegar a sufrir de alguna quemadura leve si es que la tarjeta Raspberry recibe una descarga eléctrica y llega a quemar alguno de sus componentes.

## 5.2 Configuración de la tarjeta controladora.

### 5.2.1 Configuración del sistema operativo.

- Cambiar la imagen de arranque de raspbian Quitar la imagen predefinida: Editar el archivo */boot/config.txt* y agregar al final *disable\_splash = 1* para eliminar la imagen de inicio predeterminado.

Cambiar la imagen: La imagen debe llamarse "splash.png", después copiar con la siguiente línea.

```
$ sudo cp (ubicacion_imagen)/splash.png /usr/share/plymouth/themes/pix
```

- Configurar la Raspberry en modo consola.

```
$ sudo raspi-config
```

Al ejecutar *raspi-config* abrirá la configuración de la tarjeta, se tiene que seleccionar la opción 3 *Boot options*, después la opción B1 *Desktop / CLI* y por último seleccionar la opción B2 *Console Autologin*; esta configuración hará que al reiniciar arranque en modo consola.

- Configurar script de arranque Abrir el archivo *rc.local*

```
$ sudo nano /etc/rc.local
```

Insertar el script justo antes de *exit 0*

```
if
...
fi
/home/pi/Desktop/multimedia.py
exit 0
```

- Librerías a instalar

1. Tkinter - Librería que funciona para la creación y el desarrollo de aplicaciones de escritorio.

```
$ pip install tk
```

2. Pyautogui - Librería de Python que permite controlar o automatizar tareas como controlar el mouse y teclado en el sistema operativo.

```
$ python3 -m pip install pyautogui
```

3. Keyboard - Librería que permite simular que una tecla es presionada.

```
$ pip install keyboard
```

4. Pygame - Librería para el desarrollo de videojuegos que en este caso la usaremos para crear el reproductor de música.

```
$ sudo apt-get install python-pygame
```

5. Opencv - Librería que se utilizará para la reproducción de fotos.

```
$ pip install opencv-python-headless
```

6. VLC - Librería que se utilizará para la reproducción de videos.

```
$ pip install python-vlc
```

## 5.3 Desarrollo de los componentes de software.

### 5.3.1 Creación de la ventana

Para crear la ventana del sistema, se utiliza la librería Tkinter, que tambien nos ayudará a crear botones,label, etc.

```
# Creación de ventana principal
window = tk.Tk()
window.config(bg=colorFondo, cursor="circle")
# Geometria de la ventana
window.geometry("%dx%d" % (width, height))
# Atributos de la ventana, en este caso tiene que ser en pantalla completa
window.attributes('-fullscreen', True)
# Nombre de la ventana
window.title("Multimedia Center")
#Iniciar la ventana
window.mainloop()
```

### 5.3.2 Creación de botones para aplicaciones.

```
# URL's de las paginas web de servicios de streaming
urlN = 'https://www.netflix.com/mx/'
# Creación de funciones para cada servicio de streaming
def funcion_Netflix():
    wb.open_new(urlN) #Abre el buscador con el URL asignado
    window.after(500,pantCompleta) #Abre la aplicacion en pantalla completa

img_netflix = tk.PhotoImage(file="./iconos/netflix.png")
# Creación de los botones
botonNetflix = tk.Button(window, image=img_netflix, #Se asigna la imagen al boton
                          borderwidth=0, bg=colorFondo,
                          cursor="X_cursor",
                          command = funcion_Netflix) #Se llama a la funcion
```

### 5.3.3 Configuración de red WiFi

Para la configuración de red es necesario modificar el archivo *wpa\_supplicant.conf*, para esto hay que otorgar permisos al usuario para poder editarlo, y después de editarlo es necesario reiniciar el sistema para que se ejecuten los cambios.

```
def conectarRed(ssid, key):
    arch = '/etc/wpa_supplicant/wpa_supplicant.conf'
    subprocess.call(['sudo', "chmod", "777", arch]) # Se da permiso de edición
    #Entra al archivo y edita la configuración de la nueva conexión
    with open(arch, 'w') as fp:
        fp.write('ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev')
    with open(arch, 'a') as fp:
        fp.write('\nupdate_config=1')
    with open(arch, 'a') as fp:
        fp.write('\ncountry=MX')
    with open(arch, 'a') as fp:
        fp.write('\nnetwork={ \n')
    with open(arch, 'a') as fp:
```

```

        fp.write('\tssid="{}" \n'.format(str(ssid)))
with open(arch, 'a') as fp:
    fp.write('\tpsks="{}" \n'.format(str(key)))
with open(arch, 'a') as fp:
    fp.write('\tkey_mgmt=WPA-PSK')
with open(arch, 'a') as fp:
    fp.write('\n}')

#Reinicio para guardar y ejecutar los cambios en la red.
def reinicio():
    subprocess.run(["reboot"])

```

### 5.3.4 Leer y reproducir Fotos

Utilizamos OpenCV para acceder al directorio, obtenemos todas las imágenes dentro del directorio y las reproducimos automáticamente.

```

def funcion_Fotos():
    input_images_path = "/media/pi/USB/"
    #Se obtienen las imágenes
    files_names = os.listdir(input_images_path)
    #Se recorren todos los archivos en el directorio
    for file_name in files_names:
        image_path = input_images_path + "/" + file_name
        #Se guardan solo las imágenes
        image = cv2.imread(image_path)
        if image is None:
            continue
        #Cambiamos de tamaño la foto a pantalla completa
        image = cv2.resize(image, (width, height), interpolation=cv2.INTER_CUBIC)
        cv2.namedWindow("WindowName", cv2.WINDOW_FULLSCREEN)
        Se muestra la imagen
        cv2.imshow("Image", image)
        #Retraso de segundo y medio para pasar de foto
        cv2.waitKey(1500)
    cv2.destroyAllWindows()

```

### 5.3.5 Leer y reproducir videos

Se utiliza la librería VLC para obtener los videos y reproducirlos, la función tiene la misma for a de las imágenes pero se utiliza VLC para una mejor reproducción.

```

def funcion_Video():
    input_videos_path = "/media/pi/USB/"
    files_names = os.listdir(input_videos_path)
    for file_name in files_names:
        video_path = input_videos_path + "/" + file_name
        print(video_path)
        video = vlc.MediaPlayer(video_path)
        if video is None:
            continue
        video.play()

```

### 5.3.6 Leer y reproducir música

La función añadir, muestra una ventana para añadir las canciones que se quieran escuchar al reproductor.

```
def añadir():
    canciones = filedialog.askopenfilenames(initialdir="/media/pi/USB/",
    title="Elige una canción",filetypes=(("mp3","*.mp3"),("allfiles","*..*")))
    #cambiar la extension del nombre de la cancion for cancion in canciones:
    for cancion in canciones:
        cancion=cancion.replace("/media/pi/USB/","")
        cancion=cancion.replace(".mp3","")
        #añadir cancion a la pantalla
        pantalla.insert (END, cancion)
    cancion= pantalla.get(ACTIVE)
    cancion=f'{cancion}.mp3'
    #reproduce la canción
    pygame.mixer.music.load(cancion)
    pygame.mixer.music.play(loops=0)
```

Función que pone pausa a la música.

```
def pause(is_paused):
    global paused
    paused = is_paused
    if paused:
        pygame.mixer.music.unpause()
        paused=False
    else:
        pygame.mixer.music.pause()
        paused=True
```

### 5.3.7 Función para apagar el sistema

Para apagar el sistema necesitamos ejecutar comandos desde python, para eso se llama *subprocess* que ejecuta comandos del sistema operativo.

```
def apagar():
    #Se ejecuta el comando shutdown
    subprocess.call(['shutdown', "-h", "now"])
def funcion_Salir():
    labelSalir = Label(window, text="Hasta pronto.",
        fg="#fff",      # Foreground
        bg=colorFondo,  # Background
        font=("Verdana Bold",60))
    #Muestra la despedida
    labelSalir.place(relx=0, rely=0, relheight=1, relwidth=1)
    #Retraso de tiempo antes de apagar
    window.after(5000, apagar)
```

## 6 Conclusiones

A lo largo del semestre se realizaron diferentes prácticas para poder llegar a esta instancia con las bases necesarias para poder llevar a cabo el diseño, análisis, planeación, implementación y pruebas de un sistema

embebido. Sin los conocimientos que se tomaron durante el curso no hubiera sido posible llevar a cabo este proyecto, aunque ya teníamos conocimientos previos, este semestre se fortalecieron y se dirigieron en la dirección de crear un proyecto como el que se presentó.

## 7 Bibliografía de consulta

- ¿Cómo configurar una red Wi-Fi en Raspberry Pi? (2019, septiembre 16). Descubrearduino.com. <https://descubrearduino.com/como-configurar-una-red-wi-fi-en-tu-raspberry-pi/>
- Carles, J. (2018, abril 1). Iniciar Raspberry Pi en modo consola o en modo gráfico. geekland; Joan Carles. <https://geekland.eu/iniciar-raspberry-pi-en-modo-consola/>
- Cómo ejecutar un comando del sistema. (s/f). Recursos Python. <https://micro.recursospython.com/recursos/como-ejecutar-un-comando-del-sistema.html>
- Fernandez, R. (2018, noviembre 16). Tkinter - Label, Entry, método place y grid. <https://unipython.com/tkinter-label-entry-metodos-place-y-grid/>
- Fromaget, P. (s/f). ¿Raspberry Pi: Cómo Iniciar Automáticamente Un Programa? Raspberrytips.es. <https://raspberrytips.es/iniciar-un-programa-raspberry-pi/>
- Greyrat, R. (s/f). ¿Cómo crear una ventana de pantalla completa en Tkinter? Barcelonageeks.com. <https://barcelonageeks.com/como-crear-una-ventana-de-pantalla-completa-en-tkinter/>
- Vaibhav. (2022, abril 14). Simular entradas de teclado en Python. Delft Stack. <https://www.delftstack.com/es/howto/python/python-simulate-keyboard-input/>