

REPORTE PRACTICA:
“RELOJ EN GLADE”

Omar I. Vizcaino Alarcon 24000207

Axel Arriola Fonseca 5802023

Karime García Xalteno 24100349

Microprocesadores

Puebla, Pue. A 21 de mayo de 2019

Introducción:

En esta práctica se realizó un programa en Glade que contenía 3 botones y una etiqueta. La finalidad fue hacer un reloj capaz de marcar la hora en la etiqueta de Glade, además de tener los botones de “stop”, “restart” y “start”.

Material:

- Raspberry Pi 3 (cables)
- Computadora o algún visualizador para la Raspberry

Procedimiento:

- Primero se creó el Glade, con una etiqueta, representando al reloj con un texto de contador “00:00:00”, y 3 botones, “start” “restart” “stop”.
- Posteriormente se creó el código en Geany para asimilar los botones a sus respectivas funciones, así como creando el loop para imprimir cada segundo el contador en la etiqueta del reloj.
- Finalmente se agregó un string para poder guardar los datos del contador así como poder hacer un restart sin perder el contador anterior.

Resultados:

Se presentaron problemas al crear el programa de Glade, ya que no tuvimos cuidado al asimilar bien las funciones, pero al final todo salió en orden y el reloj funcionó sin problema. Se agrega el código e imágenes.

Conclusión:

Se puede decir que esta práctica nos ayudó como repaso para poder aclarar nuestras dudas respecto a las funciones, asimilaciones y sistema gráfico en Glade, utilizando botones, entradas y etiquetas.

Anexos:

Código:

```

#include <gtk/gtk.h>

#include <glib/gi18n.h>

static int nSeg=0;
static gint timer=0;
static int bTimerCorriendo= FALSE;

void ActualizarSegundos(int valor);
void ActualizarSegundos(int valor){
printf("valor: %i",valor);
}

gint TimerCallback(gpointer data)
{
    nSeg++;
    ActualizarSegundos(nSeg);
    return 1;
}

void IniciarTimer()
{
    if (!bTimerCorriendo)
    {
        nSeg=0;
        timer=g_timeout_add(1000,TimerCallback,NULL);
        bTimerCorriendo=TRUE;
    }
}

void DetenerTimer(){
if(bTimerCorriendo){
g_source_remove(timer);
bTimerCorriendo=FALSE;}
}

```

```

int main (int argc,char *argv[])

{ GtkBuilder      *builder;

GtkWidget        *window;

gtk_init(&argc,&argv);

builder=gtk_builder_new();

gtk_builder_add_from_file(builder,"window.glade",NULL);

window=GTK_WIDGET(gtk_builder_get_object(builder,"window"));

gtk_builder_connect_signals(builder,NULL);


g_object_unref(builder);

gtk_widget_show(window);

gtk_main();

return 0;

}

void on_window_destroy () {

gtk_main_quit();

}

void on_blnicio_clicked(){

printf("Iniciando...");

IniciarTimer();}

```