# PLANTILLA GENERAL para aplicaciones con ARDUINO

Este documento detalla los elementos, estructura y orden que deben tener el código, archivos, recursos y herramientas dentro de la unidad que detalla el desarrollo de una aplicación en la plataforma Arduino.

Se desarrollan elementos de diversos aspectos de las aplicaciones. Se prueban y ejercitan las opciones que pueden encontrarse en las aplicaciones.

### Esquemas de introducción de datos

* Interruptores simples (terminales digitales de la tarjeta)
* Interruptores en entradas analógicas (terminales analógicas de la tarjeta)
* Interruptores con registros de corrimiento (CD4011)
* Encodificadores rotativos
* Pantalla táctil (de pantallas TFT)
* Módulos de I/O (TM1638, PCF8574, otros)

### Esquemas de desplegado de información

* Terminal Serial
* LEDs
* LEDs inteligentes (WS2812)
* Matrices de LEDs (Gráficas, Texto, Mixtas)
* Pantallas LCD
* Pantallas TFT
* Módulos de I/O (TM1638, PCF8574, Otras)
* Aplicaciones en PC (Processing, C#, Visual Basic, Otras)
* Aplicaciones en Android

### Esquemas de comunicación

* Bluetooth
* Ethernet
* RF
* Paralela
* Serial (RS232, RS485, Otras)
* USB
* WiFi
* IrDa

### Esquemas de sensado

* Aceleración y Magnéticos (Aceleración lineal, Posición, Giroscopía, Magnetómetro, Orientación (Brújula), Otros)
* Biométrico (Temperatura, Presión arterial, Pulsaciones cardíacas, Oximetría, Huella digital, Peso, Altura, Otros)
* Distancia (Ultrasónico, Laser, Microondas, Radar, Otros)
* Entradas y Salidas (Encodificador rotativo, Flexión, Inclinación, Interruptores, Potenciómetros, Sensores táctiles, Teclados, Otros)
* Luz y Color (Intensidad luminosa, Luz ambiental, Color, Otros)
* Medioambiente (Calidad del aire y agua, Peso, Presencia, Presión barométrica, Temperatura, Humedad, Otros)
* Posición y Movimiento (Seguidores de línea, GPS, Otros)

### Esquemas de actuación

* LCD (Pantallas de caracteres y gráficas, con interface paralela e I2C, Pantallas Oled, Otras)
* LEDs (Arreglos de LEDs, Tiras de LEDs con CI, Matrices de LEDs, Arreglos para gráficas, Letreros, Otras)
* Motores (Motores de corriente directa, Servomotores, Motores de Pasos, Otros)
* Pantallas TFT (Menús, Carrusel, Otras)
* Sonido (Módulos MP3 (grabado, reproducción, secuencia de sonidos, narración orientada, Etc.), Procesadores de audio, Sintetizadores, Amplificadores, Otros)

## Organización del Código

La aplicación tendrá el nombre: App***Xyzw***, en donde ***Xyzw*** establecen algún tipo de identificación.

Los archivos de la aplicación deberán estar en la carpeta AppXyzw

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |
| Carpeta  AppXyzw | Carpeta  Info |  |  |  |  |  |  |  |  |  |
|  | AppXyzw.ino |  |  |  |  |  |  |  |  |  |
|  | AppXyzw.h |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |

--| AppXyzw |----¬

[ info ] 🡪 Carpeta que contiene información relevante

AppXyzw.h 🡪 Archivo de cabecera de la aplicación.

AppXyzw.ino 🡪 Archivo principal de la aplicación

Estructura.h 🡪 Estructura y organización del Menú

Estructura.ino 🡪 Código “callback” de las funciones que responden al Menú [Interacción con Usuario – Menú]

Menu\_Disp.cpp 🡪 Código para rutinas de apoyo a mecanismo de Despliegue de información en Menú

Menu\_Nav.cpp 🡪 Código para rutinas de apoyo a mecanismo de Navegación del Menú

Archivo: AppXyzw.h

## Uso General del MENÚ.

Teclas (opciones) que se emplean:

ESCape Permite salir de la actual opción o del Menú.

🡨 Selecciona la opción anterior (cíclica)

🡪 Selecciona la siguiente opción (cíclica)

SELección Ejecuta la selección actual (pasa a ese nivel o selecciona el valor mostrado)

Para entrar al MENÚ es necesario oprimir SEL. Se mostrará el letrero “M E N U”

Cada menú está compuesto de las principales opciones de selección (Menú Raíz)

Cuando se SELecciona una opción de la raíz del menú se entra a un sub-menú, en el cual se puede acceder a otras opciones.

Lo que a continuación se muestra como: **🡨 🡪** es el uso de la tecla **🡨** o la tecla **🡪**

Cuando se especifica **[SEL]** se oprime la tecla SELección

Si se muestra **[ESC]** se oprime la tecla ESCape

Funcionamiento Básico del MENÚ del ejemplo:

**[SEL]**

Se imprime: “M E N U”

“Pruebas Entrada” **🡨 🡪** "Comunicaciones" **🡨 🡪** "LED" **🡨 🡪** "Mas Menú" **🡨 🡪** "Flip-Flop" **🡨 🡪** "Otro Flip-Flop"

Si estando en alguna de las opciones se oprime **[SEL]** se pasa a esa opción de sub-sub-menú, por ejemplo:

**[SEL]** “Pruebas Entrada”

“Entrada Datos”

"Lista Frutas” **🡨 🡪** "Booleano” **🡨 🡪** "Integer 8" **🡨 🡪** "Integer 16" **🡨 🡪** "Integer 32" **🡨 🡪** "Hex 16" **🡨 🡪** "Float" **🡨 🡪** "Unidades Ing" **🡨 🡪** "Reset del Menu"

Si se seleciona alguna de las opciones anteriores se continúa con esa secuencia en particular, por ejemplo:

**[SEL]** “Lista Frutas”

“Lista Frutas”

"Fruta:[Manzana]” **🡨 🡪** "Fruta:[Pera]” **🡨 🡪** "Fruta:[Naranja]” **🡨 🡪** "Fruta:[Platano]” **🡨 🡪** "Fruta:[Pi#a]” **🡨 🡪** "Fruta:[Durazno]”

Nota: La longitud del letrero depende del valor propuesto en su declaración.

Otras opciones.

Actualización de elementos del reloj o calendario:

//-----[ HORAS - MINUTOS - SEGUNDOS - DÍA de SEMANA - MES - DIA de MES - AÑO ]-----

void \*mnuIValueRqst(MD\_Menu::mnuId\_t id, bool bGet) { // Devolución ("callback") para variables Enteras.

🡪 LECTURA del TIEMPO ACTUAL

🡪 Probablemente RTC.readTime(); // Deja en variables RTC.\*\*\* (RTC.h, RTC.m, RTC.s, RTC.dow…) los valores actuales

switch (id) {

case 12:

if (bGet)

return ((void \*)&Horas);

🡪 Cambiar la variable adecuada, por ejemplo: RTC.h = Horas;

🡪 Hacer el cambio adecuado para cada caso.

else {}

break;

case 13:

if (bGet)

return ((void \*)&Minutos);

else {}

break;

case 14:

if (bGet)

return ((void \*)&Segundos);

else {}

break;

case 15:

if (bGet)

return ((void \*)&DiaSem);

else {}

break;

case 16:

if (bGet)

return ((void \*)&Mes);

else {}

break;

case 17:

if (bGet)

return ((void \*)&DiaMes);

else {}

case 18:

if (bGet)

return ((void \*)&Ayo);

else {}

break;

}

🡪 Hacer el cambio global para todo el TIEMPO: RTC.writeTime(); // Actualiza TODO el TIEMPO.

return (nullptr);

}

MODIFICACIONES en las LIBRERÍAS de SOPORTE:

🡪 Verificar en la de TECLADOS, en particular lo referente al “Debounce Delay”, originalmente propuesto para 100 mS.

Ahora se tiene que sensar LONGPRESS en lugar de PRESS.

Verificar que los interruptores estén asociados en arreglos para hacer más simple el análisis, sin importar el tipo de sistema de entrada de información.

Otras cosas...