```c
/* *********************************************************************** */
/** Descriptive File Name

  @Company
   Santiago Valiante

  @File Name
    Jeux.c

  @Summary
    Brief description of the file.

  @Description
    Describe the purpose of this file.
 */
/* *********************************************************************** */


/* *********************************************************************** */
/* *********************************************************************** */
/* Section: Included Files                                                 */
/* *********************************************************************** */
/* *********************************************************************** */

/* This section lists the other files that are included in this file.
 */

/* TODO:  Include other files here if needed. */
#include "Jeux.h"
#include "app.h"
#include "Buzzer.h"
#include "MenuJeu.h"
#include "Mc32Delays.h"


/* *********************************************************************** */
/* *********************************************************************** */
/* Section: File Scope or Global Data                                      */
/* *********************************************************************** */
/* *********************************************************************** */

/*  A brief description of a section can be given directly below the section
    banner.
 */

#define WAITSTART 0

#define START 1
#define WAITSTOP 2



MENU_JEUX time;
int nbrTuch = 0;


/* *********************************************************************** */
/** Descriptive Data Item Name

  @Summary
```

```c
    Brief one-line summary of the data item.

  @Description
    Full description, explaining the purpose and usage of data item.
    <p>
    Additional description in consecutive paragraphs separated by HTML
    paragraph breaks, as necessary.
    <p>
    Type "JavaDoc" in the "How Do I?" IDE toolbar for more information on tags.

  @Remarks
    Any additional remarks
 */


/* ************************************************************************** */
/* ************************************************************************** */
// Section: Local Functions                                                  */
/* ************************************************************************** */
/* ************************************************************************** */

/*  A brief description of a section can be given directly below the section
    banner.
 */


/* ************************************************************************** */

/**
  @Function
    int ExampleLocalFunctionName ( int param1, int param2 )

  @Summary
    Brief one-line description of the function.


  @Description
    Full description, explaining the purpose and usage of the function.
    <p>
    Additional description in consecutive paragraphs separated by HTML
    paragraph breaks, as necessary.
    <p>
    Type "JavaDoc" in the "How Do I?" IDE toolbar for more information on tags.

  @Precondition
    List and describe any required preconditions. If there are no preconditions,
    enter "None."

  @Parameters
    @param param1 Describe the first parameter to the function.


    @param param2 Describe the second parameter to the function.

  @Returns
    List (if feasible) and describe the return values of the function.
    <ul>
      <li>1   Indicates an error occurred
```

```
        <li>0   indicates an error did not occur
    </ul>

  @Remarks
    Describe any special behavior not described above.
    <p>
    Any additional remarks.

  @Example
    @code
    if(ExampleFunctionName(1, 2) == 0)
    {
        return 3;
    }
 */
int ModeNormal()
{
    bool static flag = true;
    int static flagEnd = 0;
    int static state = 0;

    //Si première fois

    if (flag == true)
    {
        //Remet tous à 0
        SetTime(0,0);
        state = 0;
        flag = false;
        flagEnd = false;
    }

    switch (state)
    {
        case WAITSTART:
            //Attente que le joueur retourne au départ
            if (PIN_STARTStateGet() == 0)
            {
                //Aller dans l'etat START
                state = START;
            }
            break;

        case START:
            //Lance un compte a rebours
            CompteARebours();

            //Si le joueur est plus sur le start
            if (PIN_STARTStateGet() == 1)
            {
                //Active le compteur en s
                DRV_TMR2_Start();

                //Aller dans l'etat WAITSTOP
                state = WAITSTOP;
            }
            break;
```

```c
        case WAITSTOP:
            //Si le joueur est arrivé
            if (PIN_STOPStateGet() == 0)
            {
                //Stop  le timer
                DRV_TMR2_Stop();
                //met la valeur de retour à 1
                flagEnd = 1;
                flag = true;

            }
            break;
    }

    //Met à jour l'affichage
    lcd_gotoxy(1,2);
    printf_lcd("Time: %2d:%2d", time.timemin ,time.timesec);

    //Détection de touch
    TouchDetectWire();

    //Met à jour l'affichage du nbr de touche
    lcd_gotoxy(1,3);
    printf_lcd("Tuch: %2d", nbrTuch);

    return flagEnd;
}


/* ************************************************************************** */
/* ************************************************************************** */
// Section: Interface Functions                                              */
/* ************************************************************************** */
/* ************************************************************************** */

/*  A brief description of a section can be given directly below the section
    banner.
 */

// **************************************************************************

/**
  @Function
    int ExampleInterfaceFunctionName ( int param1, int param2 )

  @Summary
    Brief one-line description of the function.

  @Remarks
    Refer to the example_file.h interface header for function usage details.
 */
void TouchDetectWire()
{
    bool static flagtuch = true; //incrément une fois si
```

```c
    //si on ne touhe pas le start et le stop
    if ((PIN_STARTStateGet == 1) && (PIN_STOPStateGet ==1)
    {
        //On regard si on touch
        if (PIN_WIREStateGet()== 0)
        {
            //Active le buzzer
            DRV_TMR1_Start();
            //Incrémente
            if (flagtuch == true)
            {
                nbrTuch++;
                flagtuch = false;
            }
        }
        else
        {
            //Stop le buzzer
            DRV_TMR1_Stop();
            flagtuch = true;
        }
    }
}

void CompteARebours ()
{
    lcd_ClearLine(1);
    lcd_ClearLine(2);
    lcd_ClearLine(3);

    lcd_gotoxy(11,2);
    printf_lcd("3");

    delay_ms(1000);

    lcd_gotoxy(11,2);
    printf_lcd("2");

    delay_ms(1000);

    lcd_gotoxy(11,2);
    printf_lcd("1");


    delay_ms(1000);

    lcd_gotoxy(9,2);
    printf_lcd("Go!!!");

    lcd_ClearLine(2);
}

void SetTime (int timeSecValue,int timeMinValue)
{
    time.timemin = timeMinValue;
```

```
    time.timesec = timeSecValue;
}


/* *************************************************************************
 End of File
 */
```