

A dark blue vertical bar on the left side of the page. A blue arrow points to the right from the bar, containing the date.

05/01/2023

Compte rendu

Game avec cpp cocos2d (pico park)

Several thin, curved lines in dark blue and light grey that sweep upwards from the bottom left corner.

KADDOUR FERDAOUS
[NOM DE LA SOCIETE]

MENU PRINCIPAL:

Tous avec cpp est facile grâce à l'orienté objet

*« **helloworldscene.h** » c'est la classe qui englobe tous.*

au sein de la dernière on définit tous les attributs et les méthodes

on a fait inclure cocos2d.h la bibliothèque logicielle de cet jeu 2d

ainsi que notre classe globale helloworld

```
#ifndef __HELLOWORLD_SCENE_H__
#define __HELLOWORLD_SCENE_H__

#include "cocos2d.h"

class HelloWorld : public cocos2d::Scene
{
public:
    static cocos2d::Scene* createScene();

    virtual bool init();

    // a selector callback
    void menuCloseCallback(cocos2d::Ref* pSender);

    // implement the "static create()" method manually
    CREATE_FUNC(HelloWorld);
    void GoToPlayScene(Ref* pSender);
};

#endif // __HELLOWORLD_SCENE_H__
```

Nous avons déclaré une fonction membre helloworld::gotoplayscene pour commencer le jeu

[helloworld.cpp](#)

dans cette etape nous allons créer un menu,et un button « start.png » pour demarer le jeu

ains qu'un background comme un debut du jeu



```

#include "PlayScene.h"
#include "HelloWorldScene.h"

USING_NS_CC;

Scene* HelloWorld::createScene()
{
    return HelloWorld::create();
}

//initilaser l'instance
bool HelloWorld::init()
{
    //////////////////////////////////////
    // 1. super init first
    if ( !Scene::init() )
    {
        return false;
    }

    auto visibleSize = Director::getInstance()->getVisibleSize();
    Vec2 origin = Director::getInstance()->getVisibleOrigin();

    // ----- -menu-----
    auto menuItem = MenuItemImage::create("pico park.jpg");
    auto playItem = MenuItemImage::create("arreter.png", "start.png", CC_CALLBACK_1(HelloWorld::GoToPlayScene, this));

```

```

background->setPosition(Vec2(visibleSize.width/2 +
origin.x, visibleSize.height/2 + origin.y));

```

Pausemenu

Pausemenu.h



C'est la classe qui nous donne le choix soit de démarrer le jeu, soit pour afficher le menu, ou bien quitter le jeu

```
#define __GAMEOVER_SCENE_H__

#include "cocos2d.h"

class GameOver : public cocos2d::Layer
{
public:

    static cocos2d::Scene* createScene();

    /
    virtual bool init();

    void update(float dt);

    void GoToMenuScene(Ref* pSender);
    void GoToHelloWorldScene(Ref* pSender);
};

#endif // __PLAY_SCENE_H__
```

Comme on a déclaré des méthodes dans « pausemenu.h » on vient de les initialiser ici à « pausemenu.cpp »

Créer un objet de layer et un objet de la scène

GoToMainMenuScene

La fonction crée d'abord une instance de scène locale de la scène du menu principal, puis elle apparaît

la scène actuelle hors de la pile et la scène de jeu est remplacée par le menu principal

La fonction Retry crée d'abord une instance de scène locale de la scène Game, puis

il fait sortir la scène actuelle de la pile et la scène de jeu actuelle est remplacée par

la nouvelle scène de jeu (redémarre le jeu)

scène

```
Scene* PauseScene::createScene()
{
    auto scene = Scene::create();
    auto layer = PlayScene::create();
    scene->addChild(layer);
    return scene;
}

bool PauseScene::init()
{
    if (!Layer::init())
    {
        return false;
    }

    Size visibleSize = Director::getInstance()->getVisibleSize();
    Point origin = Director::getInstance()->getVisibleOrigin();

    void PauseScene::Resume(cocos2d::Ref* pSender)
    {
        Director::getInstance()->popScene();
    }

    void PauseScene::GoToHelloWorldScene(cocos2d::Ref* pSender)
    {
        auto scene = HelloWorld::createScene();
```

GoToMenuScene va s'allumer lorsque le joueur veut reprendre le jeu

GoToHelloWorldScene est bien lorsque le joueur clique sur menu

Gameoverscene

```
#define __GAMEOVER_SCENE_H__

#include "cocos2d.h"

class GameOver : public cocos2d::Layer
{
public:

    static cocos2d::Scene* createScene();

    /
    virtual bool init();

    void update(float dt);

    void GoToMenuScene(Ref* pSender);
    void GoToHelloWorldScene(Ref* pSender);
};

#endif // __PLAY_SCENE_H__
```

```

USING_NS_CC;
Scene* gameover::createScene()
{
    auto scene = Scene::create();
    auto layer = gameover::create();
    scene->addChild(layer);
    return scene;
}

bool gameover::init()
{
    if (!Layer::init())
    {
        return false;
    }
    Director::getInstance()->replacescene(scene);
    Director::getInstance()->replacescene(scene);
    Size visibleSize = Director::getInstance()->getVisibleSoize();
    Point origin = Director::
    return true;
}

void gameover::GoToGameScene* cocos2d::Ref* pSender){
    auto scene = gamescreen::createscene();
    auto layer = gameover::create();

    Director::getInstance()->replacescene(scene);
}

void gameover::GoToHelloWorldScene* cocos2d::Ref* pSpender){
    auto scene = HelloWorldscene::createscene();

```


Gamescene.h

```

#ifndef __PLAY_SCENE_H__
#define __PLAY_SCENE_H__

#include "cocos2d.h"

class PlayScene : public cocos2d::Layer
{
public:

    static cocos2d::Scene* createScene();

    virtual bool init();

    void menuCloseCallback(cocos2d::Ref* pSender);

    CREATE_FUNC(PlayScene);

    cocos2d::Sprite* mySprite;

    void GoToPauseScene(Ref* pSender);
    void GoToGameOverScene(Ref* pSender);

};

#endif
```

Gamescene.cpp

ng.cpp HelloWorldScene.cpp pausemenu.cpp [] gamescene.cpp gameoverscene.cpp

```
7
3 Scene* PlayScene::createScene()
3 {
3     return PlayScene::create();
1 }
2
3
4
5
6
7 bool PlayScene::init()
7 {
3     if (!Scene::init())
3     {
3         return false;
1     }
2     Size visibleSize = Director::getInstance()->getVisibleSize();
3     Point origin = Director::getInstance()->getVisibleOrigin();
4
5
6
7     auto background = Sprite::create("bggame.jpg");
3     background->setPosition(Vec2(visibleSize.width / 2 + origin.x, visibleSize.height / 2 + origin.y));
3     background->setScaleY(9.0);
3     background->setScaleX(9.0);
1     this->addChild(background, -4);
2
3
4     auto ground1 = Sprite::create("bg.jpg");
5     ground1->setScaleY(4.0);
6
```