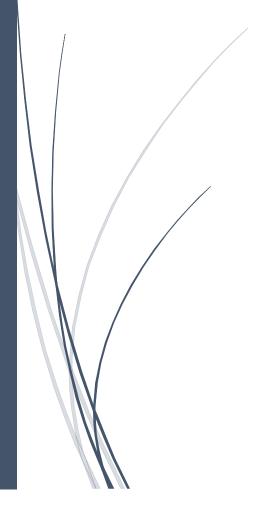
05/01/2023

Compte rendu

Game avec cpp cocos2d (pico park)



KADDOUR FERDAOUS [NOM DE LA SOCIETE]

MENU PRINCIPAL:

Tous avec cpp est facile grâce a l'oriente objet « helloworldscene.h »c'est la classe qui englobe tous.

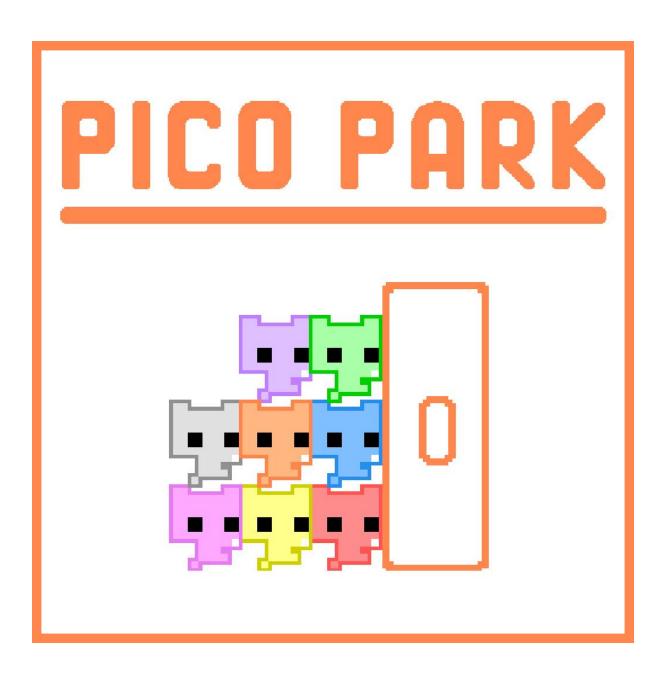
au seins de la dernière on définit tous les attributs et les méthodes on a fait inclure cocos2d.h la bibliothèque logicielle de cet jeu 2d ainsi que notre classe globale helloworld

Nous avons declarer une fonction membre helloword ::gotoplayscene pour commencer le jeu

helloword.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



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, o 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 déclarer des méthodes dans « pausemenu.h » on vient de les initialiser ici a « pausemenu.cpp »

Créer 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::GoToHelloWodldScene(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();
```

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