



# ¡De Pesca!

## Advanced Programing Final Project

Show your fishing skills in this minigame, where you will have to catch as many fish as possible and avoid obstacles and enemies!

### Scenes

Scene	Description	Image
<a href="#">Home Page</a>	Scene where the player will select the game mode and his equipment.	
<a href="#">Classic game</a>	Game Mode of normal difficulty where little by little more enemies will begin to appear, the game ends when the player runs out of hooks or after a certain number of rounds.	
<a href="#">Against time game</a>	Game Mode of extreme difficulty, from the beginning all types of enemies appear, against time, the objective is to catch the largest number of fish in 1 minute.	
<a href="#">Game Over</a>	Scene where the statistics of the game and the best scores are shown, it asks you if you want to play again or return to the home page.	

### Gadgets

Name	Description	Image
<a href="#">Fish</a>	They work as points the more you catch the better. They swim across the game scene.	
<a href="#">Hook</a>	They work as lives or opportunities, if the player runs out of Hooks the player loses the game.	
<a href="#">Bait can</a>	They work like extra lives, catching one adds a hook	
<a href="#">Coins</a>	They are obtained at the end of the game and are used to make improvements to the rod (not yet final) or buy objects.	
<a href="#">Premium rod</a>	Includes an extra fish in game that is more valuable than the common fish, it can be bought with coins (Not yet final)	

### Enemies

Name	Descripción	Imagen
<a href="#">Obstacles (Boot, Barrel)</a>	If the rod has a fish and hits the obstacle the fish escapes.	
<a href="#">Jellyfish</a>	If the jellyfish touches the rod, it is electrocuted and loses a hook and the fish escapes.	
<a href="#">Shark</a>	If the shark bites the rod, it loses a hook and the fish escapes.	
<a href="#">Crab</a>	If the crab cuts the rod, it loses a hook and the fish escapes.	

```

/*****
Copyright (c) 2017-2018 Xiamen Yaji Software Co., Ltd.

http://www.cocos2d-x.org

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
*****/

#include "HelloWorldScene.h"

USING_NS_CC;

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

// Print useful error message instead of segfaulting when files are not there.
static void problemLoading(const char* filename)
{
    printf("Error while loading: %s\n", filename);
    printf("Depending on how you compiled you might have to add 'Resources/' in front of filenames in HelloWorldScene.cpp\n");
}

// on "init" you need to initialize your instance
bool HelloWorld::init()
{
    ////////////////////////////////////
    // 1. super init first
    if ( !Scene::init() )
    {
        return false;
    }

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

    auto item_menu_1 = MenuItemFont::create("Play",CC_CALLBACK_1(HelloWorld::Play,this));
    item_menu_1->setPosition(Point(visibleSize.width / 2, (visibleSize.height / 5) * 4));
    auto *menu = Menu::create(item_menu_1,nullptr);
    menu->setPosition(Point(0, 0));
    this->addChild(menu);

    ////////////////////////////////////

    // Code

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

    ////////////////////////////////////
    // 2. add a menu item with "X" image, which is clicked to quit the program
    //    you may modify it.

    // add a "close" icon to exit the progress. it's an autorelease object
    auto closeItem = MenuItemImage::create(
        "CloseNormal.png",

```

```

        "CloseSelected.png",
        CC_CALLBACK_1(HelloWorld::menuCloseCallback, this));

if (closeItem == nullptr ||
    closeItem->getContentSize().width <= 0 ||
    closeItem->getContentSize().height <= 0)
{
    problemLoading("'CloseNormal.png' and 'CloseSelected.png'");
}
else
{
    float x = origin.x + visibleSize.width - closeItem->getContentSize().width/2;
    float y = origin.y + closeItem->getContentSize().height/2;
    closeItem->setPosition(Vec2(x,y));
}

// create menu, it's an autorelease object
auto menu = Menu::create(closeItem, NULL);
menu->setPosition(Vec2::ZERO);
this->addChild(menu, 1);

//////////
// 3. add your codes below...

// add a label shows "Hello World"
// create and initialize a label

auto label = Label::createWithTTF("¡De Pesca!", "fonts/Marker Felt.ttf", 24);
if (label == nullptr)
{
    problemLoading("'fonts/Marker Felt.ttf'");
}
else
{
    // position the label on the center of the screen
    label->setPosition(Vec2(origin.x + visibleSize.width/2,
        origin.y + visibleSize.height - label->getContentSize().height));

    // add the label as a child to this layer
    this->addChild(label, 1);
}

// add "HelloWorld" splash screen"
auto sprite = Sprite::create("Files/Escenas/GameWallpaper.png");
if (sprite == nullptr)
{
    problemLoading("Files/Escenas/GameWallpaper.png");
}
else
{
    // position the sprite on the center of the screen
    sprite->setPosition(Vec2(visibleSize.width/2 + origin.x, visibleSize.height/2 + origin.y));

    // add the sprite as a child to this layer
    this->addChild(sprite, 0);
}*/
return true;
}

void HelloWorld::menuCloseCallback(Ref* pSender)
{
    //Close the cocos2d-x game scene and quit the application
    Director::getInstance()->end();

    /*To navigate back to native iOS screen(if present) without quitting the application ,do not use Director::getInstance()->end()

    //EventCustom customEndEvent("game_scene_close_event");
    //_eventDispatcher->dispatchEvent(&customEndEvent);

}

void HelloWorld::Play(HelloWorld::Ref* pSender){
    auto director = Director::getInstance();
    auto tamano = director->getWinSize();
    auto visibleSize = Director::getInstance()->getVisibleSize();
    Vec2 origin = Director::getInstance()->getVisibleOrigin();

    // Wallpaper
    auto spriteFondo = Sprite::create("Files/Escenas/GameWallpaper.png");

```

```
spriteFondo->setScale(tamano.width/(1900+15), tamano.height/(1300+15));

spriteFondo->setAnchorPoint(Vec2::ZERO);
spriteFondo->setPosition(origin);

addChild(spriteFondo,0);

// Pescador
_pescador = Sprite::create("Files/Sprites/Pescador/pescando1blue.png");
_pescador->setScale(0.16, 0.16);
_pescador->setFlippedX(true);
_pescador->setAnchorPoint(Vec2::ZERO);
_pescador->setPosition(119,82);

addChild(_pescador, 1);
}
```