1. **簡介**
   1. 動機
   2. 分工
2. **遊戲介紹**
   1. 遊戲說明
      1. 遊戲方式
      2. 遊戲規則
      3. 特殊功能
      4. 密技
   2. 遊戲圖形
   3. 遊戲音效
3. **程式設計**
   1. 程式架構
   2. 程式類別
   3. 程式技術
4. **結語**
   1. 問題及解決方法
   2. 時間表
   3. 貢獻比例
   4. 檢核表
   5. 收穫
   6. 心得感想
5. **簡介**
6. **動機**

在三個可選擇的程式語言當中，比較想熟悉的是javascript，所以選擇做HTML遊戲。前陣子滿流行橫向無限捲軸的遊戲，主角固定位置而地圖持續移動的遊戲方式，於是就選了這樣的遊戲作為這次物件導向程式設計實習課程的範本。



1. **分工**

自己一組：102820016，四電資三，謝孟樺

程式撰寫、Debug、遊戲測試、製作和修改圖片、音樂

1. **遊戲介紹**
2. **遊戲說明**
   * 1. **遊戲方式**

用空白鍵控制，keydown主角人物往上移動，keyup則往下掉。 

* + 1. **遊戲規則**

控制主角避開障礙物，並在遊戲中收集星星，以最長存活距離為遊戲目標。

* + 1. **遊戲功能**
       1. 人物控制
       2. 紀錄星星數
       3. 計算目前距離
       4. 障礙物偵測、扣血
       5. 遊戲暫停、重新開始、回到開始畫面
    2. **密技**

無

1. **遊戲圖形**

|  |  |  |
| --- | --- | --- |
| 圖形 | 說明 | |
|  | 主角y軸向上移動，循環播放動畫。 | |
|  | 主角y軸向下移動。 | |
|  | 當主角在地面上，播放跑步動畫。 | |
|  | 當主角碰到障礙物，播放閃爍動畫。 | |
|  | 暫停畫面的主角圖片。 | |
|  | 遊戲道具：生命值、星星。 | |
|  | 遊戲中的地板tile，形成主要遊戲地圖、移動邊界。 | |
|  | 障礙物，主角碰到會扣血。 | |
|  | 按鈕：暫停遊戲 | |
|  | 按鈕：回到開始畫面、重新開始遊戲、開始遊戲 | |
|  | 按鈕：開啟或關閉背景音樂 | |
|  | 按鈕：開啟或關閉音效 | |
|  | 遊戲背景 | |
| Game Level | | 說明 |
|  | | Game Level：menu  開始畫面，開啟遊戲的第一頁 |
|  | | Game Level：level1  進入正式遊戲前的載入頁面。 |
|  | | Game Level：level1  主要遊戲畫面 |
|  | | Game Level：over  遊戲結束，顯示計分資訊。 |

1. **遊戲音效**

|  |  |
| --- | --- |
| 觸發情境 | 說明及使用音效 |
|  | 當遊戲開始即播放。  音效：bensound-funkyelement.mp3 |
|  | 每碰到一顆星星，播放一次音效。  音效：electronic\_bubbles.mp3 |
|  | 當主角向上移動，開始循環播放，放開空白鍵往下移動，則同時停止播放音效。  音效：engine.mp3 |
|  | 當主角碰到障礙物時播放。  音效：rock\_crash.mp3 |

1. **程式設計**
   1. **程式架構**

mainGame

MyGame

Menu

GameOver

GameMap

Character

* 1. **程式類別**

|  |  |  |
| --- | --- | --- |
| 類別名稱 | .js檔行數 | 說明 |
| Menu | 25 | 遊戲主選單，play按鈕開始遊戲 |
| MyGame | 379 | 主要遊戲程式，播放音樂及各種觸發音效，GameMap和Character的互動 |
| GameOver | 29 | 結束畫面，紀錄並顯示距離、星星數 |
| GameMap | 157 | 用陣列寫出遊戲地圖、星星分布，並使它持續移動 |
| Character | 27 | 載入主角sprite，定義各個動畫的播放 |
| 總行數 | 617 |  |

* 1. **程式技術**
     1. GameMap中只有移動到接近canvas的tile才會被draw出來
     2. 判斷主角的螢幕位置及其上下左右分別是何種類型的tile，藉此做出相對應的觸發事件

1. 形成邊界：限制主角移動範圍不超過一定範圍
2. 地圖停止移動：當主角的右邊判斷為石磚，地圖停止移動
3. 障礙物偵測：當主角身體範圍與障礙物位置重疊，判斷為障礙物碰撞，並減少HP。
4. 吃星星：當主角身體範圍與星星位置重疊，星星消失，並記錄星星數。
   * 1. 因星星和遊戲地圖適用的tile大小不同，故分為兩個陣列來寫，也因此要分別換算兩者的地圖座標與螢幕座標。
     2. 換算地圖位置與玩家目前距離。
     3. 使用draw畫出血條、顯示計分資訊。
     4. 血條寬度依剩餘血量比例顯示。
     5. 偵測滑鼠點擊位置，觸發遊戲暫停、重新開始、回到開始畫面等功能。
     6. 當主角HP為零，跳到結束畫面。
5. **結語**
   1. **問題及解決方法**

|  |  |
| --- | --- |
| 問題 | 解決方法 |
| 第一次寫HTML Game，面對都是新的東西，不知道如何下手 | 反覆練習tutorial、研究框架內容，從範例程式中相同功能的程式碼開始慢慢改寫。 |
| 對框架不熟悉，不知道有很多方便的函式可以套用，導致進度不如預期 | 把src的程式碼一一看過，大概知道有哪些功能 |
| 為了讓地圖移動，在update裡修改position值，但是竟然不會動！？ | 請教老師與助教，學到了一些原本不知道的瀏覽器debugger用法，才知道問題在draw而不是原以為的update。助教修正框架後問題暫時解決。 |
| 加入player後，當player碰到邊界地圖停止draw | 請教老師與助教，重寫class。 |
| 用判斷式寫player移動範圍，結果碰到邊界就不能移動 | 用瀏覽器debugger觀察程式碼，發現是player在每個frame移動的單位會造成誤差，導致判斷式失效，把邊界值設定往內縮一個誤差就解決了。 |
| 看似不相關的兩個class居然互相影響，怎麼除錯都找不出問題在哪 | 請教老師助教，但找不出問題，暫時把問題物件移到畫面外。 |
| 為地圖寫邊界失敗，屢屢遭受挫折 | 不斷嘗試不同方法，最後把邊界判斷移出GameMap Class，改由主角位置判斷邊界 |
| 嘗試寫吃星星的功能一直失敗 | 試了很多方法，找出可行的寫法。 |

* 1. **時間表**

|  |  |  |
| --- | --- | --- |
| 週次 | 日期 | 花費時間 |
| 1 | 2016-03-04 ～ 2016-03-11 | 3小時 |
| 2 | 2016-03-11 ～ 2016-03-18 | 5小時 |
| 3 | 2016-03-18 ～ 2016-03-25 | 6小時 |
| 4 | 2016-03-25 ～ 2016-04-01 | 6小時 |
| 5 | 2016-04-01 ～ 2016-04-08 | 8小時 |
| 6 | 2016-04-08 ～ 2016-04-15 | 7小時 |
| 7 | 2016-04-15 ～ 2016-04-22 | 3小時 |
| 8 | 2016-04-22 ～ 2016-04-29 | 2小時 |
| 9 | 2016-04-29 ～ 2016-05-06 | 8小時 |
| 10 | 2016-05-06 ～ 2016-05-13 | 8小時 |
| 11 | 2016-05-13 ～ 2016-05-20 | 10小時 |
| 12 | 2016-05-20 ～ 2016-05-27 | 3小時 |
| 13 | 2016-05-27 ～ 2016-06-03 | 2小時 |
| 14 | 2016-06-03 ～ 2016-06-10 | 1小時 |
| 15 | 2016-06-10 ～ 2016-06-17 | 1小時 |
| 總時數 |  | 73小時 |

* 1. **貢獻比例**

自己一組

100 %

* 1. **檢核表**

|  |  |  |  |
| --- | --- | --- | --- |
|  | 檢查項目 | 完成否 | 無法完成的原因 |
| 1 | 解決Memory leak | □已完成　□未完成 |  |
| 2 | 自定遊戲Icon | □已完成　□未完成 |  |
| 3 | 全螢幕啟動 | □已完成　□未完成 |  |
| 4 | 修改Help->About | □已完成　□未完成 |  |
| 5 | 初始畫面說明按鍵及滑鼠之用法與密技 | □已完成　□未完成 |  |
| 6 | 上傳setup檔 | □已完成　□未完成 |  |
| 7 | 報告字型、點數、對齊、行距、頁碼等格式正確 | □已完成　□未完成 |  |
| 8 | 報告封面、側邊格式正確 | □已完成　□未完成 |  |

* 1. **收獲**

　　之前只用過Unity寫遊戲，第一次要像這樣沒有使用圖形介面的寫出一個遊戲，一開始覺得非常難下手。接觸javascript也算第一次，透過本課程讓我學到了一個新的語言，未來除了用來寫程式之外，也可以應用在網頁前端等。寫遊戲的過程中也會遇到不少問題，很謝謝老師跟助教認真的指導，除了幫我找出問題，也教我如何善用除錯工具來debug。我覺得在本課程獲得最大的收穫是學習如何解決問題，訓練自己的邏輯思考能力。當我想實作某一個功能，必須不斷思考我要如何用程式的邏輯達成。

* 1. **心得感想**

　　剛開始覺得非常困難，要用新接觸的語言寫出一個遊戲，而且一開始的進度不理想，遇到不知道怎麼解決的問題覺得很痛苦，在專案起頭就一直卡在同一個問題，但很感謝老師和助教一直都不吝指導。

　　這次課程用助教寫好的框架，很多功能都已經幫我們包好，讓我們寫程式難度降低，但如果要學好程式，還是要看懂框架內容，了解這個功能是如何產生的。

　　我覺得我在這堂課收穫很多，老師不會一步一步教，反而是讓大家自由發揮，所以大家會遇到的問題也都不一樣。這堂課比起其他一般課程，更需要鞭策自己，自我監督每周達成一定進度，更重要的是必須自己想辦法解決問題，對我而言我覺得這堂課最大的收穫是，訓練自己思考變得更靈活。

**附錄**

|  |
| --- |
| **mainGame.js** |
| //當有要加關卡時, 可以使用addNewLevel  //第一個被加進來的Level就是啟動點, 所以一開始遊戲就進入MyMenu  Framework.Game.addNewLevel({menu: new Menu()});  Framework.Game.addNewLevel({level1: new MyGame()});  Framework.Game.addNewLevel({over: new GameOver()});  //讓Game開始運行  Framework.Game.start(); |

|  |
| --- |
| **Menu.js** |
| var Menu = Framework.Class(Framework.Level , {    load: function(){  this.pic = new Framework.AnimationSprite({url:[define.imagePath+'game\_menu\_1.jpg', define.imagePath+'game\_menu\_2.jpg']});  this.pic.position = {x:Framework.Game.getCanvasWidth()/2, y:Framework.Game.getCanvasHeight()/2};  this.pic.start({from:0, to: 1});  },    update: function(){  this.rootScene.update();  },    draw: function(parentCtx){  this.rootScene.draw(parentCtx);  this.pic.draw();  },  click: function (e) {  console.log('x = '+e.x+', y = '+e.y);  if(e.x>1050 && e.x<1277 && e.y>485 && e.y<650){  Framework.Game.goToNextLevel();  }  },  }); |

|  |
| --- |
| **myGameLevel1.js** |
| var MyGame = Framework.Class(Framework.Level , {    initializeProgressResource: function() {  this.loading = new Framework.Sprite(define.imagePath + 'loading.jpg');  this.loading.position = {x: Framework.Game.getCanvasWidth() / 2 , y: Framework.Game.getCanvasHeight() / 2};  },  loadingProgress: function(ctx, requestInfo) {  this.loading.draw(ctx);  ctx.fillStyle = '#82aa16';  ctx.fillRect(470,425,400\*Math.round(requestInfo.percent)/100,49);  ctx.font ='30pt Arial';  ctx.textAlign = 'center';  ctx.fillStyle = 'white';  ctx.fillText(Math.round(requestInfo.percent) + '%' , 1000 , 462);  },    load: function(){    this.gameMap = new GameMap();  this.gameMap.load();  this.rootScene.attach(this.gameMap);    this.audio = new Framework.Audio({  bgMusic: {  mp3: define.musicPath + 'bensound-funkyelement.mp3'  }, star:{  mp3: define.musicPath + 'electronic\_bubbles.mp3'  }, engine:{  mp3: define.musicPath + 'engine.mp3'  }, crash:{  mp3: define.musicPath + 'rock\_crash.mp3'  }  });    this.audio.play({name: 'bgMusic', loop: true});    this.char = new Character(define.imagePath + 'player.png', {  position: {x:400,y:500},  run: {from: 0, to: 1},  moveUp: {from: 2, to: 5},  moveDown: {from: 6, to: 6},  hurt: {from: 6, to: 12}  });  this.rootScene.attach(this.char.sprite);    this.heart = new Framework.Sprite(define.imagePath + 'heart.png');  this.hp = 50;    this.pauseButton = new Framework.Sprite(define.imagePath + 'pause\_btn.png');  this.playerPause = new Framework.Sprite(define.imagePath + 'player\_front.png');  this.homeButton = new Framework.Sprite(define.imagePath + 'home\_btn.png');  this.replayButton = new Framework.Sprite(define.imagePath + 'replay\_btn.png');  this.playButton = new Framework.Sprite(define.imagePath + 'play\_btn.png');  this.bgDark = new Framework.Sprite(define.imagePath + 'bg\_dark.png');          var characterPosition;    this.isStop = false;  this.isPlayed = false;  this.clock = new Framework.Sprite(define.imagePath + 'clock.png');  this.clock.scale = 0.3;  this.clock.position = {  x: 0,  y: 0  };    //characterPosition = {x: 0, y: -1138 \* this.clock.scale};  characterPosition = {x: 0, y: -1138};  this.secondHand = new Framework.Sprite(define.imagePath + 'secondHand.jpg');  this.firen = new Character\_ex(define.imagePath + 'firen.png', {position: characterPosition, run: {from: 20, to: 22}, beHit: {from:30, to: 35}, hit: {from: 10, to: 13}});  //this.freeze = new Character(define.imagePath + 'freeze.png', {position: characterPosition, scale: 1, run: {from: 29, to: 27}, beHit: {from:39, to: 35}, hit: {from: 19, to: 16}});  this.freeze = new Character\_ex(define.imagePath + 'firen.png', {position: characterPosition, run: {from: 20, to: 22}, beHit: {from:30, to: 35}, hit: {from: 10, to: 13}});    this.clockCenter = new Framework.Scene();  this.clockCenter.position = {  x: -10.5 \* this.clock.scale,  y: 51 \* this.clock.scale  };  this.clockCenterNeg = new Framework.Scene();  this.clockCenterNeg.position = {  x: -10.5 \* this.clock.scale,  y: 51 \* this.clock.scale  };  this.secondHand.position = {  x: 0,  y: -100  };  this.wholeClock = new Framework.Scene();  this.wholeClock.position = {  x: Framework.Game.getCanvasWidth() / 2,  y: Framework.Game.getCanvasHeight() / 2  };    this.secondHandRotationRate = 0.3;  //this.wholeClock.attach(this.clock);  //this.clockCenter.attach(this.secondHand);  //this.clockCenter.attach(this.firen.sprite);  this.clockCenterNeg.attach(this.freeze.sprite);  this.wholeClock.attach(this.clockCenterNeg);  //this.wholeClock.attach(this.clockCenter);  this.rootScene.attach(this.wholeClock);    /\*  //繪製Sprite的boundry (Debug用)  this.firen.sprite.isDrawBoundry = true;  this.clock.isDrawBoundry = true;  //載入要被播放的音樂清單  //資料夾內只提供mp3檔案, 其餘的音樂檔案, 請自行轉檔測試  this.audio = new Framework.Audio({  kick: {  mp3: define.musicPath + 'kick2.mp3',  //ogg: define.musicPath + 'kick2.ogg',  //wav: define.musicPath + 'kick2.wav'  }, song1:{  mp3: define.musicPath + 'NTUT\_classic.mp3',  //ogg: define.musicPath + 'Hot\_Heat.ogg',  //wav: define.musicPath + 'Hot\_Heat.wav'  }, song2:{  mp3: define.musicPath + 'NTUT\_modern.mp3',  //ogg: define.musicPath + 'The\_Messenger.ogg',  //wav: define.musicPath + 'The\_Messenger.wav'  }  });  //播放時, 需要給name, 其餘參數可參考W3C  this.audio.play({name: 'song2', loop: true});  this.rectPosition = {  x: Framework.Game.getCanvasWidth() / 2 - 130,  y: Framework.Game.getCanvasHeight() / 2 - 90  };    this.position = {  x: 100,  y: 100  }  this.rotation = 0;\*/  },  initialize: function() {      },  update: function() {    this.rootScene.update();    // 人物右邊是空的地圖才會動，否則地圖靜止  if(this.gameMap.getTileType(this.char.sprite.position.x+this.char.sprite.width, this.char.sprite.position.y)!='tile'  && this.gameMap.getTileType(this.char.sprite.position.x+this.char.sprite.width, this.char.sprite.position.y+this.char.sprite.height/2)!='tile'){  this.gameMap.position={  x: this.gameMap.position.x - this.gameMap.moveSpeed,  y: this.gameMap.position.y  }  }    /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  \*\*\*\*\*\*\*人物移動\*\*\*\*\*\*  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/    if(this.isSpace){  if(this.gameMap.getTileType(this.char.sprite.position.x+120, this.char.sprite.position.y-10)!='tile' && this.gameMap.getTileType(this.char.sprite.position.x+40, this.char.sprite.position.y-10)!='tile'){  this.char.sprite.position.y -= 4;  }else{  this.char.sprite.position.y += 0;  }  }else{  if(this.gameMap.getTileType(this.char.sprite.position.x+120,this.char.sprite.position.y+130)!='tile' && this.gameMap.getTileType(this.char.sprite.position.x+40,this.char.sprite.position.y+130)!='tile'){  this.char.sprite.position.y += 4;  this.char.moveDown();  }else{  this.char.sprite.position.y += 0;  this.char.run();  }  }      /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  \*\*\*\*\*\*\*\*吃星星\*\*\*\*\*\*\*  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/    for(let i=this.gameMap.findStarY(this.char.sprite.position.y)-2; i<this.gameMap.findStarY(this.char.sprite.position.y); i++){  for(let j=this.gameMap.findStarX(this.char.sprite.position.x); j<this.gameMap.findStarX(this.char.sprite.position.x)+3; j++){  if(this.gameMap.star[i][j]==1){  this.gameMap.star[i][j]=0;  this.gameMap.starCount++;  this.audio.play({name: 'star'});  }  }  }      /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  \*\*\*\*\*\*\*碰到扣HP\*\*\*\*\*\*  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/    if(this.gameMap.getTileType(this.char.sprite.position.x+20, this.char.sprite.position.y+80)=='sting'){  this.hp -= 0.1;  this.char.hurt();  this.audio.play({name: 'crash'});  }    if(Math.floor(this.hp) == 0){  var distance = Math.floor(this.gameMap.position.x/-20);  this.level = Math.ceil(this.gameMap.distance/400);  Framework.Game.goToNextLevel();  }  console.log(distance);      /\*  //以下為當被攻擊時會停下來, 並且當被攻擊的動畫播放完時便繼續跑的Scenario  if(this.firen.collide(this.freeze) && !this.isStop && !this.isPlayed) {  this.isStop = true;  this.isPlayed = true;  //當碰攻擊時, 播放音效(可一次播放多首音樂)  this.audio.play({name: 'kick'});  this.firen.hit(function() {  game.freeze.beHit(function() {  game.isStop = false;  game.freeze.run();  });  game.firen.run();  });    }  else if(!this.firen.collide(this.freeze)){  this.isPlayed = false;  this.clockCenter.rotation += this.secondHandRotationRate;  this.clockCenterNeg.rotation = -this.clockCenter.rotation;  }  else if(this.firen.collide(this.freeze) && !this.isStop)  {  this.clockCenter.rotation += this.secondHandRotationRate;  this.clockCenterNeg.rotation = -this.clockCenter.rotation;  }  //以上為當被攻擊時會停下來, 並且當被撞到的動畫播放完時便繼續跑的Scenario  this.isPlayHit = this.firen.collide(this.freeze) \*/  },  draw:function(parentCtx){    this.rootScene.draw(parentCtx);    parentCtx.fillStyle = '#e03f32';  parentCtx.fillRect(70,95,this.hp\*4,30);  this.heart.scale = 0.8;  this.heart.position = {x:60,y:110};  this.heart.scale = 1.5  this.heart.draw();  this.gameMap.starPic.position = {x:60,y:180};  this.gameMap.starPic.draw();  parentCtx.font = '32pt Arial';  parentCtx.fillStyle = 'lightgrey';  parentCtx.textAlign = 'left';  parentCtx.fillText(Math.floor(this.hp).toString(),110,125,100);  parentCtx.fillStyle = '#f1d133';  parentCtx.fillText(this.gameMap.starCount.toString(),110,200,100);  parentCtx.fillStyle = 'white';  parentCtx.fillText(Math.floor(this.gameMap.position.x/-20).toString()+' m',30,270,150);    this.pauseButton.position = {x:Framework.Game.getCanvasWidth()-60,y:110};  this.pauseButton.scale = 1.5;  this.pauseButton.draw();  this.playerPause.position = {x:Framework.Game.getCanvasWidth()/2,y:250};  this.replayButton.position = {x:Framework.Game.getCanvasWidth()/2,y:500};  this.homeButton.position = {x:Framework.Game.getCanvasWidth()/2-140,y:500};  this.playButton.position = {x:Framework.Game.getCanvasWidth()/2+170,y:500};  this.bgDark.position = {x:750,y:500};  this.playerPause.scale = 1.5;  this.replayButton.scale = 1.5;  this.homeButton.scale = 1.5;  this.playButton.scale = 1.5;  if(this.isPause){  this.bgDark.draw();  this.playerPause.draw();  this.replayButton.draw();  this.homeButton.draw();  this.playButton.draw();  }          /\*  //可支援畫各種單純的圖形和字  parentCtx.fillStyle = (this.secondHandRotationRate > 0)?'green':'red';  parentCtx.fillRect(this.rectPosition.x , this.rectPosition.y, 260, 90);  parentCtx.font = '65pt bold';  parentCtx.fillStyle = 'white';  parentCtx.textBaseline = 'top';  parentCtx.textAlign = 'center';  parentCtx.fillText('Click Me', this.rectPosition.x + 130, this.rectPosition.y, 260);  \*/    },  keydown:function(e){  if(e.key === 'Space'){  this.isSpace = true;  this.char.moveUp();  this.audio.play({name: 'engine', loop: true});  }  },    keyup:function(e){  this.isSpace = false;  this.audio.stop('engine');  },  touchstart: function (e) {  //為了要讓Mouse和Touch都有一樣的事件  //又要減少Duplicated code, 故在Touch事件被觸發時, 去Trigger Mouse事件  this.click({ x: e.touches[0].clientX, y: e.touches[0].clientY });  },    click: function (e) {    if(e.x>this.pauseButton.position.x-25 && e.x<this.pauseButton.position.x+25 && e.y>this.pauseButton.position.y-25 && e.y<this.pauseButton.position.y+25){  this.isPause = true;  Framework.Replay.pause();  }    if(e.x>755 && e.x<920 && e.y>425 && e.y<570){  Framework.Replay.resume();  }    if(e.x>480 && e.x<580 && e.y>450 && e.y<540){  Framework.Game.goToLevel('menu');  this.isPause = false;  }    if(e.x>620 && e.x<720 && e.y>450 && e.y<550){  Framework.Game.goToLevel('level1');  this.isPause = false;  }  console.log(e.x, e.y);  if (!this.rectPosition) {  return;  }    if(e.x >= this.rectPosition.x && e.x <= this.rectPosition.x + 260 && e.y >= this.rectPosition.y && e.y <= this.rectPosition.y + 90) {  if(!this.isClockStop) {  this.secondHandRotationRate = 0;  this.isClockStop = true;  //Audio可以一次暫停所有的音樂  this.audio.pauseAll();  } else {  this.isClockStop = false;  this.secondHandRotationRate = 0.3;  //Audio也可以針對一首歌進行操作(繼續播放)  this.audio.resume('song2');  }  } else if(e.x >= this.clock.upperLeft.x && e.x <= this.clock.lowerRight.x && e.y >= this.clock.upperLeft.y && e.y <= this.clock.lowerRight.y) {  //由於Click Me在太小的螢幕的情況下會蓋到Clock, 導致點擊Click Me時, 會回到前一個Level,  //故使用else if, 並優先選擇Click Me會觸發的條件  this.audio.stopAll();  Framework.Game.goToPreviousLevel();  return;  }  },  }); |

|  |
| --- |
| **GameMap.js** |
| class GameMap{    constructor(){  this.tileW = 120;  this.tileH = 120;  this.starW = 60;  this.starH = 60;  this.position = {x: 0, y: -this.tileH/2};  this.moveSpeed = 5;  this.starCount = 0;  this.distance = 0;    this.map = [[1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1],  [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,4,1,1,1,1,1,1,1,5,0,0,2,2,2,2,2,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,4,1,5,0,4,1,5,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],  [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,2,2,2,2,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,7,0,0,0,0,0,0,0,2,0,0,0,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,3,3,3,3,3,3,3,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],  [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,4,1,1,1,1,1,1,1,5,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],  [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,3,0,0,0,3,0,0,0,3,0,0,0,0,0,8,0,0,0,8,0,0,0,8,0,0,0,0,0,0,2,2,2,2,2,2,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],  [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,3,3,3,3,3,3,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,6,0,0,0,0,0,0,0,0,0,4,1,5,0,4,1,5,0,4,1,5,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],  [1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1]];    this.star = [[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],  [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],  [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],  [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,0,0,0,0,0,0,0,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],  [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,1,0,0,0,1,0,0,0,0,0,1,0,0,0,1,0,0,0,0,0,0,0,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,1,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],  [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,1,0,0,0,0,0,1,0,0,0,1,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,0,1,0,1,0,1,0,1,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],  [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,0,0,0,0,0,0,0,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],  [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],  [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],  [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]];  }    load(){  this.bgPic = new Framework.Sprite(define.imagePath + 'bg.jpg');  this.tilePic = new Framework.Sprite(define.imagePath + 'tile.png');  this.stingBottomPic = new Framework.Sprite(define.imagePath + 'sting\_bottom.png');  this.stingTopPic = new Framework.Sprite(define.imagePath + 'sting\_top.png');  this.stingLeftPic = new Framework.Sprite(define.imagePath + 'sting\_left.png');  this.stingRightPic = new Framework.Sprite(define.imagePath + 'sting\_right.png'); // 5  this.ob1 = new Framework.Sprite(define.imagePath + 'obstacle\_1.png');  this.ob2 = new Framework.Sprite(define.imagePath + 'obstacle\_2.png');  this.ob3 = new Framework.Sprite(define.imagePath + 'obstacle\_3.png');  this.starPic = new Framework.Sprite(define.imagePath + 'star.png');  }    update(){    this.firstTilePos = {  x: this.position.x + this.tileW/2,  y: this.position.y + this.tileH/2  };  }    draw(ctx){    for(let x=0; x<this.map[0].length/6; x++){  var bgPosition = {  x: this.position.x + (this.bgPic.width\*x) + this.bgPic.width/2,  y: this.position.y + this.bgPic.height/2-10 + this.tileH  };    if(bgPosition.x <= Framework.Game.getCanvasWidth()+this.bgPic.width/2){ // 只畫出Canvas內的  this.bgPic.position = bgPosition;  this.bgPic.draw(ctx);  }  }    for(let i=0; i<this.map.length; i++){  for(let j=0; j<this.map[0].length; j++){    var picPosition = {  x: this.position.x + (this.tileW\*j) + this.tileW/2,  y: this.position.y + (this.tileH\*i) + this.tileH/2  };    if(picPosition.x <= Framework.Game.getCanvasWidth()+100){ // 只畫出Canvas內的  switch(this.map[i][j]){  case 0:  break;  case 1:  this.tilePic.position = picPosition;  this.tilePic.draw(ctx);  break;  case 2:  this.stingBottomPic.position = picPosition;  this.stingBottomPic.draw(ctx);  break;  case 3:  this.stingTopPic.position = picPosition;  this.stingTopPic.draw(ctx);  break;  case 4:  this.stingLeftPic.position = picPosition;  this.stingLeftPic.draw(ctx);  break;  case 5:  this.stingRightPic.position = picPosition;  this.stingRightPic.draw(ctx);  break;  case 6:  this.ob1.position = picPosition;  this.ob1.draw(ctx);  break;  case 7:  this.ob2.position = picPosition;  this.ob2.draw(ctx);  break;  case 8:  this.ob3.position = picPosition;  this.ob3.draw(ctx);  break;  }  }  }  }  for(let i=0; i<this.star.length; i++){  for(let j=0; j<this.star[0].length; j++){    var starPosition = {  x: this.position.x + (this.starW\*j) + this.starW/2,  y: this.position.y + (this.starH\*i) + this.starH/2 + 120  };    switch(this.star[i][j]){  case 0:  break;  case 1:  this.starPic.position = starPosition;  this.starPic.draw(ctx);  break;  }  }  }  }    getTileType(x, y){  var gy = Math.floor(y/this.tileH);  var gx = Math.floor((x -this.firstTilePos.x)/this.tileW);  if(this.map[gy][gx] == 1){  return 'tile';  }  else{  for(i=2; i<=5; i++){  if(this.map[gy][gx] == i)  return 'sting';  }  }  }    findStarX(x){  return Math.floor((x -this.firstTilePos.x)/this.starW);  }    findStarY(y){  return Math.floor(y/this.starH);  }  } |

|  |
| --- |
| **Character.js** |
| var Character = function(file, options) {  this.url = file;  this.sprite = new Framework.AnimationSprite({url:this.url, col:5 , row:3 , loop:true , speed:1});    this.sprite.position = options.position || {x: 0, y: 0};  this.sprite.scale = options.scale || 1;  this.run = function() {  this.sprite.start({ from: options.run.from, to: options.run.to, loop: true });  };    this.moveUp = function() {  this.sprite.start({ from: options.moveUp.from, to: options.moveUp.to, loop: true });  };    this.moveDown = function() {  this.sprite.start({ from: options.moveDown.from, to: options.moveDown.to, loop: true });  };    this.hurt = function() {  this.sprite.start({ from: options.hurt.from, to: options.hurt.to, loop: true });  };  //預設人物就是在跑步  this.run();  }; |

|  |
| --- |
| **GameOver––.js** |
| var GameOver = Framework.Class(Framework.Level , {  load: function(){  this.pic = new Framework.Sprite(define.imagePath + 'game\_over.jpg');  this.pic.position = {x:Framework.Game.getCanvasWidth()/2, y:Framework.Game.getCanvasHeight()/2};  },    update: function(){  this.rootScene.update();  },    draw: function(parentCtx){  this.rootScene.draw(parentCtx);  this.pic.draw();  parentCtx.fillStyle = 'white Arial';  parentCtx.fillText(this.distance.toString()+' m',270,270,150);  parentCtx.fillStyle = '#f1d133';  parentCtx.fillText(this.gameMap.starCount.toString(),400,500,100);  },    click: function (e) {  console.log('x = '+e.x+', y = '+e.y);  if(e.x>875 && e.x<1080 && e.y>500 && e.y<670){  Framework.Game.goToLevel('level1');  }else if(e.x>1155 && e.x<1285 && e.y>570 && e.y<670){  Framework.Game.goToLevel('menu');  }  },  }); |