

氣球遊戲設計

大學部人工智慧應用
數位系三年級
吳智鴻



教學網站：[HTTP://120.108.221.55/PROFCHWU/DCTEC](http://120.108.221.55/PROFCHWU/DCTEC)

FB社團： 106 數位系人工智慧

APP INVENTOR PROJECT: BALLOON

水平儀遊戲設計

目標

- 設計一個水平儀遊戲

資料結構

變數
資料庫 (Firebase)*

演算法

- 依照水平儀參數，移動圖片精靈。
- 檢測圖片精靈與球型精靈的碰撞。



所需素材（音樂與圖片）



AI2 專案檔



ball.png



boom.png



boom.wav



dnb.wav



light_plank.png



stick1.png



stick2.png



stick3.png



stick4.png



dark_plank.png



需要瞭解的程式技巧

按鈕控制

圖片精靈與球型精靈控制

圖片控制

If else判斷

變數

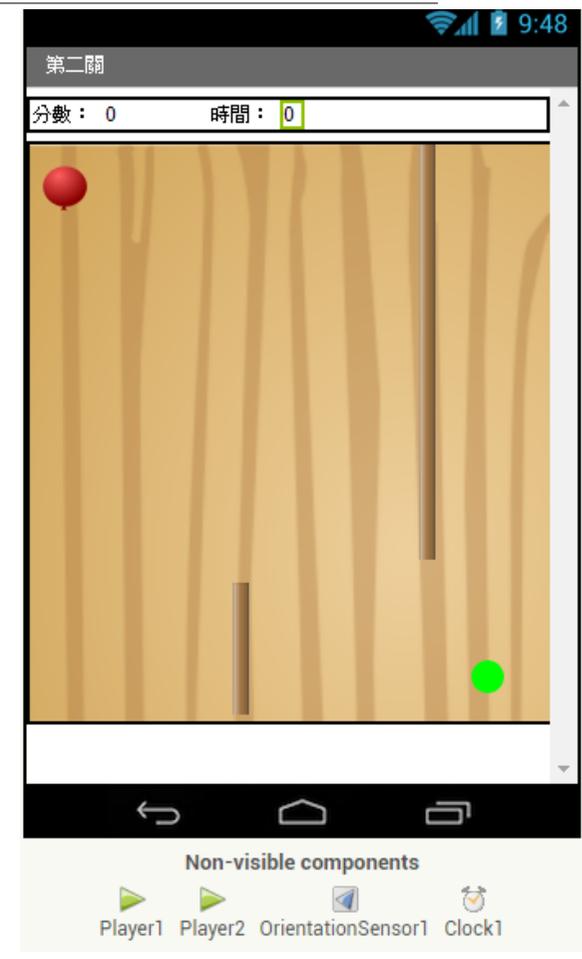
音樂播放

多螢幕處理

定時器

資料庫存取

螢幕設計



第一關 螢幕設計



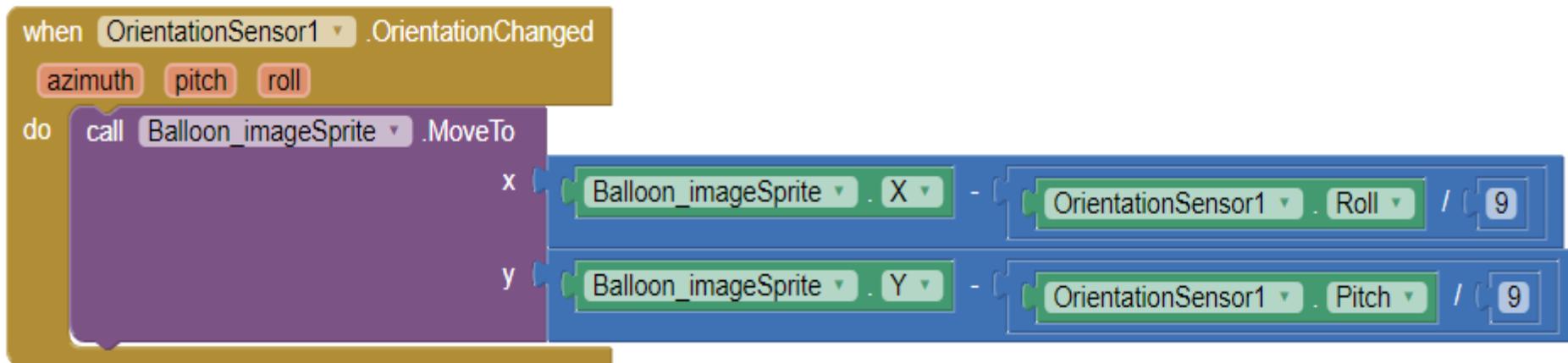
The screenshot shows the development environment interface. The Components panel on the left lists the following components: Game1, TableArrangement1 (containing Label1, Score1_Label, Label3, and Time1_Label), Canvas1 (containing Goal_Ball, Balloon_imageSprite, ImageSprite1, Player1, Player2, OrientationSensor1, and Clock1). The Properties panel on the right shows the properties for the selected Goal_Ball component: Enabled (checked), Heading (0), Interval (100), PaintColor (Green), Radius (10), Speed (0.0), Visible (checked), X (7), Y (320), and Z (1.0). The Media panel at the bottom lists various assets, including ballon.png, ball.png, boom.png, boom.wav, dark_plank.png, dnb.wav, light_plank.png, stick1.png, and stick2.png.

程式碼（控制移動）

以水平儀控制移動

變更+或-，或者變更數值，可以改變移動方式與速度

```
when OrientationSensor1 .OrientationChanged
  azimuth pitch roll
do
  call Balloon_imageSprite .MoveTo
    x Balloon_imageSprite . X - OrientationSensor1 . Roll / 9
    y Balloon_imageSprite . Y - OrientationSensor1 . Pitch / 9
```



程式碼（判斷目標碰撞）

當目標被碰撞時，播放過關音樂。

當結束後跳至第二關，並將分數傳到第二關

```
when Goal_Ball .CollidedWith
  other
  do
    set OrientationSensor1 . Enabled to false
    call Balloon_imageSprite . MoveTo
      x 0
      y 320
    call Player2 . Start

when Player2 . Completed
  do
    open another screen with start value
      screenName "Game2"
      startValue Score1_Label . Text
```

The image shows two Scratch code blocks. The first block is a 'when' block triggered by 'Goal_Ball . CollidedWith' with an 'other' block. Inside the 'do' block, it contains: 'set OrientationSensor1 . Enabled to false', 'call Balloon_imageSprite . MoveTo' with 'x' set to 0 and 'y' set to 320, and 'call Player2 . Start'. The second block is a 'when' block triggered by 'Player2 . Completed' with a 'do' block containing: 'open another screen with start value' with 'screenName' set to 'Game2' and 'startValue' set to 'Score1_Label . Text'.

程式碼（判斷障礙物碰撞）

當障礙物被碰撞時，呼叫重新開始。

```
when ImageSprite1 .CollidedWith  
  other  
do call re_start
```

重新開始的設定。

更改氣球為爆炸圖案。

```
to re_start  
do set OrientationSensor1 .Enabled to false  
   set Balloon_imageSprite .Picture to "boom.png"  
   call Player1 .Start
```

設定圖片，並將氣球回覆到初始位置。

```
when Player1 .Completed  
do set Balloon_imageSprite .Picture to "ball.png"  
   call Balloon_imageSprite .MoveTo  
     x 5  
     y 10  
   set OrientationSensor1 .Enabled to true
```

程式碼 (第一關) 全

```
when OrientationSensor1 .OrientationChanged
  azimuth pitch roll
do
  call Balloon_imageSprite .MoveTo
    x Balloon_imageSprite . X - OrientationSensor1 . Roll / 9
    y Balloon_imageSprite . Y - OrientationSensor1 . Pitch / 9
```

```
when Goal_Ball .CollidedWith
  other
do
  set OrientationSensor1 .Enabled to false
  call Balloon_imageSprite .MoveTo
    x 0
    y 390
  call Player2 .Start
```

```
when Player2 .Completed
do
  open another screen with start value screenName "Game2"
  startValue Score1_Label .Text
```

```
when ImageSprite1 .CollidedWith
  other
do
  call re_start
```

```
to re_start
do
  set OrientationSensor1 .Enabled to false
  set Balloon_imageSprite .Picture to "boom.png"
  call Player1 .Start
```

```
when Player1 .Completed
do
  set Balloon_imageSprite .Picture to "ball.png"
  call Balloon_imageSprite .MoveTo
    x 5
    y 10
  set OrientationSensor1 .Enabled to true
```

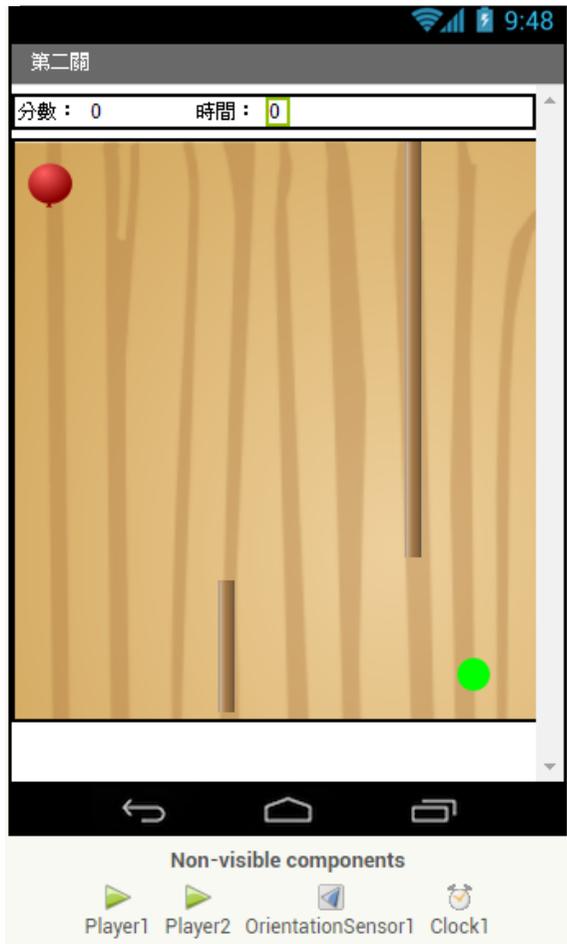
第二關



第二關跟第一關一樣，你是
否要自己嘗試看看。



第二關 螢幕設計



Components

- Game2
 - TableArrangement1
 - Label1
 - Score2_Label
 - Label3
 - Time2_Label**
 - Canvas1
 - Ballon_ImageSprite
 - Goal_Ball
 - ImageSprite2
 - ImageSprite1
 - Player1
 - Player2
 - OrientationSensor1
 - Clock1

Rename Delete

Media

- ballon.png
- ball.png
- boom.png
- boom.wav
- dark_plank.png
- dnb.wav**
- light_plank.png
- stick1.png
- stick2.png

Properties

Time2_Label

BackgroundColor
 None

FontBold

FontItalic

FontSize
14.0

FontTypeface
default ▾

HTMLFormat

HasMargins

Height
Automatic...

Width
Automatic...

Text
0

TextAlignment
left : 0 ▾

TextColor
 Black

Visible

第二關程式碼 (全)

```
when OrientationSensor1 .OrientationChanged
  azimuth pitch roll
do call Ballon_ImageSprite .MoveTo
  x Ballon_ImageSprite . X - OrientationSensor1 . Roll / 9
  y Ballon_ImageSprite . Y - OrientationSensor1 . Pitch / 9
```

```
when ImageSprite1 .CollidedWith
  other
do call re_start2
```

```
when ImageSprite2 .CollidedWith
  other
do call re_start2
```

```
when Goal_Ball .CollidedWith
  other
do set OrientationSensor1 . Enabled to false
  call Player2 .Start
```

```
when Player2 .Completed
do close screen with value result Score2_Label . Text
```

```
to re_start2
do set OrientationSensor1 . Enabled to false
  set Ballon_ImageSprite . Picture to "boom.png"
  call Player1 .Start
```

```
when Player1 .Completed
do set Ballon_ImageSprite . Picture to "ball.png"
  call Ballon_ImageSprite .MoveTo
    x 5
    y 10
  set OrientationSensor1 . Enabled to true
```



後續延伸

加入分數計算。

加入倒數計時，限制每關30秒內。

最高分數連結資料庫。

第一關 加入倒數計時

計時器每1000毫秒(1秒) 執行一次

Properties

Clock1

TimerAlwaysFires



TimerEnabled



TimerInterval

1000

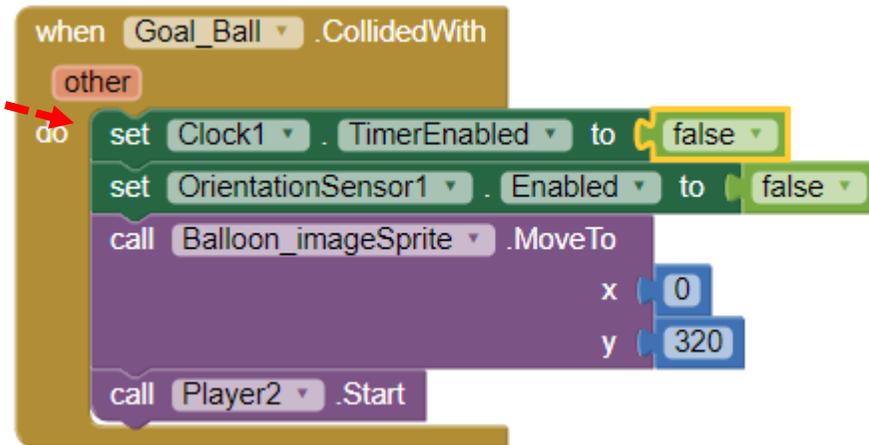
倒數計時每一秒減一

initialize global time1 to 30

```
when Clock1.Timer
do
  set global time1 to get global time1 - 1
  set Time1_Label.Text to get global time1
```

第一關 新增 過關時停止計數器

設定停止計數器1



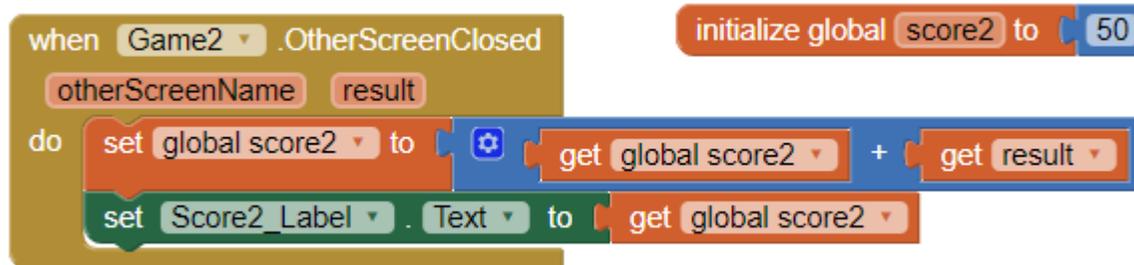
```
when Goal_Ball .CollidedWith  
  other  
  do  
    set Clock1 .TimerEnabled to false  
    set OrientationSensor1 .Enabled to false  
    call Balloon_imageSprite .MoveTo  
      x 0  
      y 320  
    call Player2 .Start
```

The image shows a Scratch code block with a yellow background. It starts with a 'when' block: 'when Goal_Ball .CollidedWith other'. Below this is a 'do' block containing four actions: 'set Clock1 .TimerEnabled to false', 'set OrientationSensor1 .Enabled to false', 'call Balloon_imageSprite .MoveTo' with 'x' set to 0 and 'y' set to 320, and 'call Player2 .Start'. A red dashed arrow points from the text '設定停止計數器1' to the 'set Clock1 .TimerEnabled to false' block. The 'false' value in the first 'set' block is highlighted with a yellow box.

第二關 分數傳遞與取得

讀取上一關傳來的數值

累加至這一關的分數score2



```
when Game2 .OtherScreenClosed
  initialize global score2 to 50
  do
    otherScreenName result
    set global score2 to get global score2 + get result
    set Score2_Label . Text to get global score2
```

The image shows a Scratch script with the following blocks:

- when Game2 .OtherScreenClosed** (yellow block)
- initialize global score2 to 50** (orange block)
- do** (blue block) containing:
 - otherScreenName result** (pink block)
 - set global score2 to get global score2 + get result** (orange block)
 - set Score2_Label . Text to get global score2** (green block)

先增第二關 計時與分數計算

計時與分數計算

initialize global time1 to 30

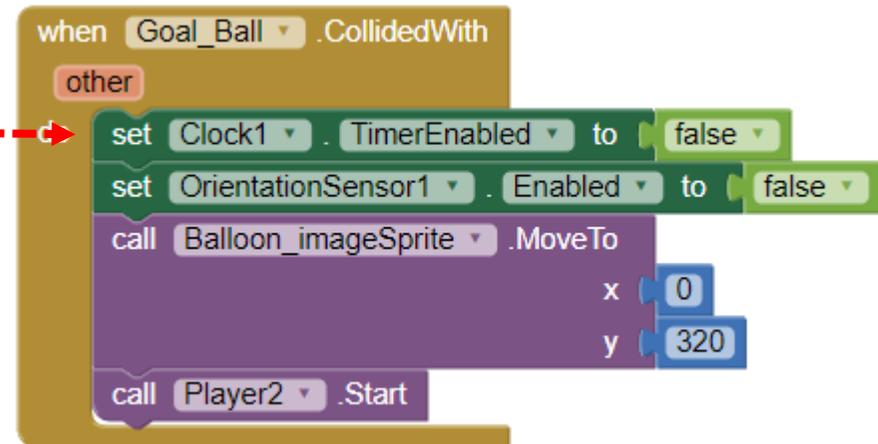
initialize global score1 to 30

```
when Clock1.Timer
do
  set global time1 to get global time1 - 1
  set Time1_Label.Text to get global time1
  set global score1 to get global score1 - 1
  set Score1_Label.Text to get global score1
```

第一關與第二關

第一關與第二關都要加入

碰到目的地時停止計時



```
when Goal_Ball .CollidedWith  
  other  
    set Clock1 .TimerEnabled to false  
    set OrientationSensor1 .Enabled to false  
    call Balloon_imageSprite .MoveTo  
      x 0  
      y 320  
    call Player2 .Start
```

The image shows a Scratch code block for a collision event. The event is triggered when 'Goal_Ball' collides with 'other'. The code block contains the following actions: 'set Clock1 .TimerEnabled to false', 'set OrientationSensor1 .Enabled to false', 'call Balloon_imageSprite .MoveTo' with x-coordinate 0 and y-coordinate 320, and 'call Player2 .Start'. A red dashed arrow points from the text '碰到目的地時停止計時' to the first 'set' block.

第一關程式碼全

initialize global time1 to 30

initialize global score1 to 30

```
when Clock1.Timer
do
  set global time1 to get global time1 - 1
  set Time1_Label.Text to get global time1
  set global score1 to get global score1 - 1
  set Score1_Label.Text to get global score1
```

```
when OrientationSensor1.OrientationChanged
  azimuth pitch roll
do
  call Balloon_imageSprite.MoveTo
    x Balloon_imageSprite.X - OrientationSensor1.Roll / 5
    y Balloon_imageSprite.Y - OrientationSensor1.Pitch / 5
```

```
when Goal_Ball.CollidedWith
  other
do
  set Clock1.TimerEnabled to false
  set OrientationSensor1.Enabled to false
  call Balloon_imageSprite.MoveTo
    x 0
    y 320
  call Player2.Start
```

```
when Player2.Completed
do
  close screen with value result get global score1
```

```
when ImageSprite2.CollidedWith
  other
do
  call re_start
```

```
when ImageSprite1.CollidedWith
  other
do
  call re_start
```

```
to re_start
do
  set OrientationSensor1.Enabled to false
  set Balloon_imageSprite.Picture to "boom.png"
  call Player1.Start
```

```
when Player1.Completed
do
  set Balloon_imageSprite.Picture to "ball.png"
  call Balloon_imageSprite.MoveTo
    x 5
    y 10
  set OrientationSensor1.Enabled to true
```

第二關程式碼全

```
when Clock1.Timer
do
  set global time2 to get global time2 - 1
  set Time2_Label.Text to get global time2
  set global score2 to get global score2 - 1
  set Score2_Label.Text to get global score2
```

```
initialize global time2 to 30
```

```
initialize global score2 to 30
```

```
when OrientationSensor1.OrientationChanged
  azimuth pitch roll
do
  call Ballon_ImageSprite.MoveTo
    x Ballon_ImageSprite.X - OrientationSensor1.Roll / 5
    y Ballon_ImageSprite.Y - OrientationSensor1.Pitch / 5
```

```
when ImageSprite1.CollidedWith
  other
do
  call re_start2
```

```
when ImageSprite2.CollidedWith
  other
do
  call re_start2
```

```
when Goal_Ball.CollidedWith
  other
do
  set Clock1.TimerEnabled to false
  set OrientationSensor1.Enabled to false
  call Player2.Start
```

```
to re_start2
do
  set OrientationSensor1.Enabled to false
  set Ballon_ImageSprite.Picture to "boom.png"
  call Player1.Start
```

```
when Player2.Completed
do
  close screen with value result get global score2
```

```
when Player1.Completed
do
  set Ballon_ImageSprite.Picture to "ball.png"
  call Ballon_ImageSprite.MoveTo
    x 5
    y 10
  set OrientationSensor1.Enabled to true
```

加入時間限制



加入時間停止的判斷

第一關

```
when Clock1.Timer
do
  set global time1 to get global time1 - 1
  set Time1_Label.Text to get global time1
  set global score1 to get global score1 - 1
  set Score1_Label.Text to get global score1
  if get global time1 = 0
  then call re_start
```

第二關

```
when Clock1.Timer
do
  set global time2 to get global time2 - 1
  set Time2_Label.Text to get global time2
  set global score2 to get global score2 - 1
  set Score2_Label.Text to get global score2
  if get global time2 = 0
  then call re_start2
```

主畫面程式(全)

分數累加與 關卡切換都在主畫面

```
when Start_Button .Click  
do  
  open another screen screenName "Game1"  
  
initialize global total_score to 0
```

```
when Screen1 .OtherScreenClosed  
  otherScreenName result  
do  
  set global total_score to  
    get global total_score + get result  
  if  
    get otherScreenName = "Game2"  
  then  
    call Notifier1 .ShowMessageDialog  
      message join "分數：" get global total_score  
      title "過關訊息"  
      buttonText "OK"  
  if  
    get otherScreenName = "Game1"  
  then  
    open another screen screenName "Game2"
```