計算機實習期末project

資工一A 111502522劉俊吾111502528 詹宸安

1. 專題簡介

遊戲名稱 : Cosmic Quest

遊戲類型 : 音樂遊戲

遊戲內容 : 玩家要根據音樂的節奏按下特定按鍵來得分

1. 遊戲規則與說明

根據點擊時機，我們分為 :

Perfect

(誤差30ms以內，得分100)

Great

(誤差30ms~60ms，得分60)

Bad

(誤差60ms~100ms，得分20)

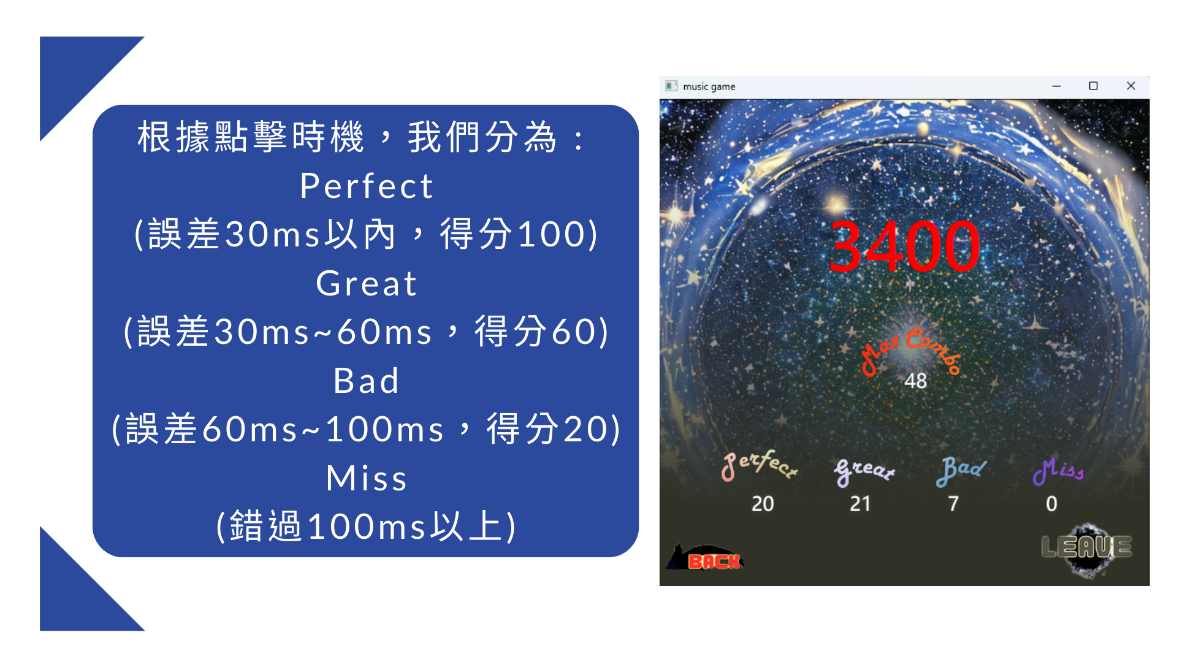
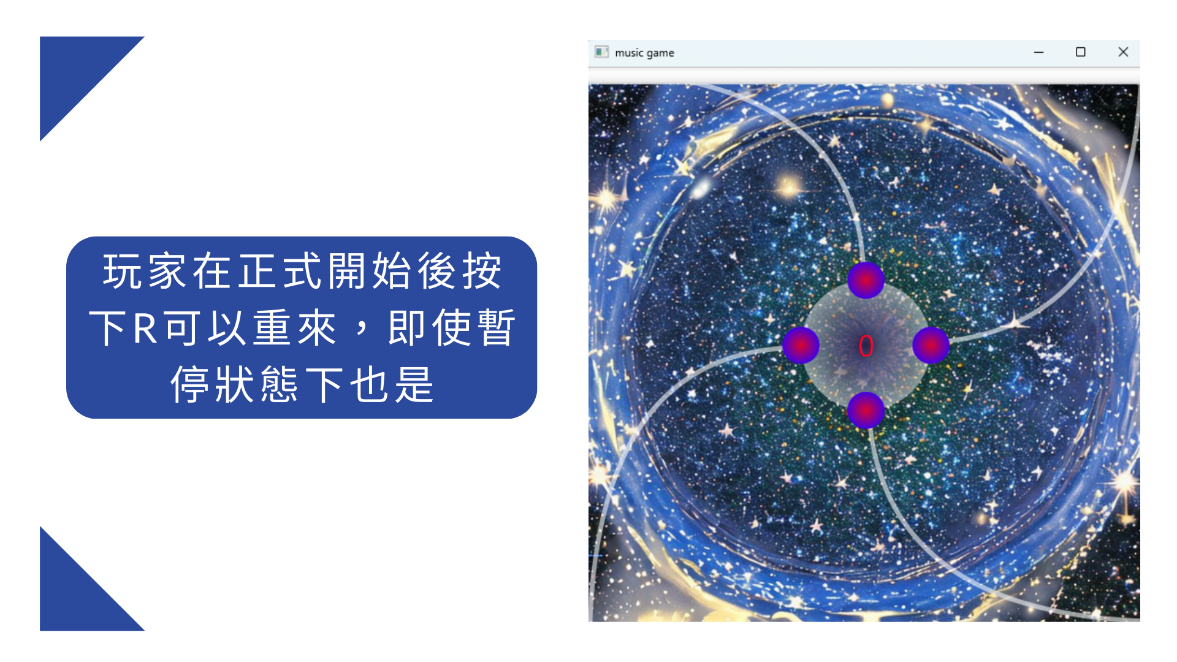
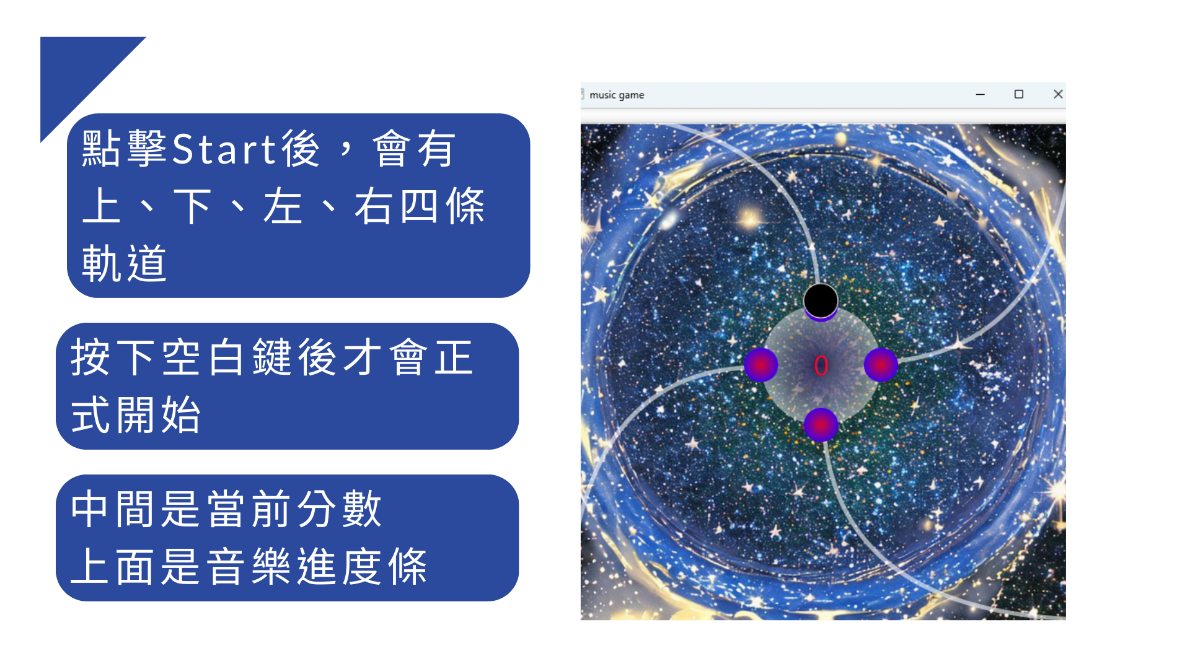
Miss

(錯過100ms以上)

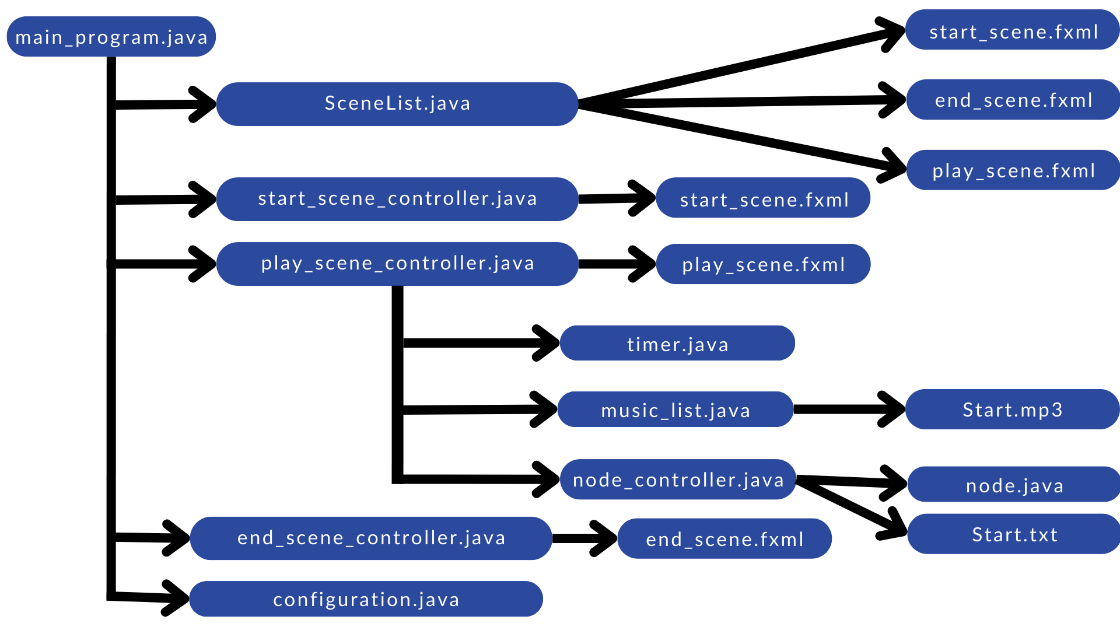
MaxCombo

連續非Miss的最大數量

1. 視窗介紹與功能(需附上截圖)



1. 程式架構和譜面設計



1. 技術困難點

* javafx,fxml,scenebuilder的相關操作:

我們用javafx 載入 scene builder 建出的 fxml 檔，還有後續的其他控制，例如在不同的fxml之間切換等等，一開始不熟悉相關語法和操作時在找資料上花費不少時間，而且有時候是環境沒有建好，像是javafx 要播音樂的話要在run configuration 的 vm argument那邊還要再加上--add-modules javafx.controls,javafx.media，類似我們要載入fxml時需要加上的--add-modules javafx.controls,javafx.fxml

* 設計架構:

設計架構時遇到了不少問題，像是要有幾個class，某個功能要不要切一個class出去，切出一個class後呼叫起來是會更簡單還是更麻煩。變數和method要設public還是private，有些東西runtime時才執行，不能設成static，所以要新增一個物件來使用他的method。

* 設計防呆機制和例外情形考慮:

我們也花不少時間模擬各種情況，讓該有作用的鍵在該作用的時機發揮作用，不該作用的鍵在不該作用的時機不要作用，因為我們的控制鍵都是有作用的，如果按鍵在不該作用時被觸發了，程式就會出包，包括但並不限於，遊戲開始前就按重製鍵，如果有人一直按著WASD鍵時要怎麼處理，如果有人在跳轉結算畫面時按下空白鍵，如果有人一直按著空白鍵不放要怎麼樣才能保證程式正常運行等等。

* 架構和執行優化:

目前程式再開始就把所有音符渲染好，並用PathTransion的setDelay()控制時間，還另外寫一個Timer來進行每個毫秒的判定。之後設計架構時，只要用Timeline在所有要生成音符的時間點執行生成音符並播放的函式，並利用Queue先進先出的特性儲存已生成但尚未消失的音符，這樣在暫停時，只要暫停Timeline和Queue中的音符就好。

* 音符的判定機制:

一開始我們是用音符和判定點的距離來決定音符的狀態，後來參考很多網路上的資料後決定用時間來判斷，也就是按下的時機點，這麼做的好處是更加精確的時機點安排，但壞處就是畫面可能會有細微落差。

* 兩段軌道:

我們預計讓生成的音符從畫面外移動到中心點，但又要控制到判定點的時間方便控制，所以寫了兩段動畫控制音符移動，其中，第一段就是Path加PathTransition的組合，第二段是Timeline，同時在第二段動畫中，加入淡出的效果。

* 畫面與音樂有落差:

考慮到種種可能會造成音符落下的時間點與音樂開始的時間點不同步，我們設計offset這個參數來控制音樂播放的延遲與提前。

* 一些小問題:

1. Text物件的錨點和ImageView物件不太一樣，導致設計時圖像與文字可能歪掉，透過SceneBuilder調整後，還是會有一點點偏移。
2. 原本希望ImageView物件可以直接使用onMouseClicked()來觸發事件，但結果無法運作，所以決定在圖案上方放一個Opacity為0.0的按鈕來控制。
3. 創意
   1. 自行設計架構

因為網路上查不到可以參考的音樂遊戲架構，所以就自己摸索，至少這個架構可以正常的運作。

* 1. 音樂創作和音符設計

音樂來自於詹宸安寫出來的一首樂曲，包含三連音的設計等等，在音符設計方面，也是以圓點表示，如此便不用處理音符轉向的問題

* 1. 設計圖標和介面

1. 圖標和畫面主要用Canva製作，部分內容有使用到圖庫的設計，而遊玩畫面中的背景為AI生成。
2. 介面方面採用正方形為基礎，使得音符在遊玩畫面中，移動到中心點的動畫更和諧。
3. 在設置介面使用菱形襯底，使方形的畫面更加生動。
4. 離開遊戲的圖標有下載免費圖片，以黑洞的概念表示離開的意象。
5. 設定的圖標以齒輪和行星為設計元素。
6. 回到開始畫面以箭頭為設計元素。
7. 音符的移動軌跡用曲線來增加中心的圓之主體
8. 未來展望
9. 更多神奇的音效和迷因
10. 新增教學頁面，選歌頁面，玩家排行榜。
11. 架構和執行優化。

目前程式再開始就把所有音符渲染好，並用PathTransion的setDelay()控制時間，還另外寫一個Timer來進行每個毫秒的判定。之後設計架構時，只要用TimeLine在所有要生成音符的時間點執行生成音符並播放的函式，並利用Queue先進先出的特性儲存已生成但尚未消失的音符，這樣在暫停時，只要暫停Timeline和Queue中的音符就好。

1. 更多首歌。
2. 做出長條音符以及更多的效果。