

# 專案題目：Synesthesia: Exploring the Universe of Words in Colors

小組名稱：S. A. M. E.

成員：陳舒毓(組長)，林奕汝，單子芸，阮郁善

## 概述(Introduction)

人類文字發展至今已有一段歷史，文字、語言經常是人類之間用來表達、接收訊息的工具，不同民族、地域有不同的語言，然而及文字之所以能夠用以溝通，歸因於它是建立在最基本、人類所能見的現實生活中。因此，不同民族、不同語系的人類可藉由學習來理解、運用其他文字、語言。

而色彩對於人類，相較於文字，是更加直覺的一種訊息形式；人類能夠接收的可見光範圍相當侷限，但色彩可以說掌控了人類的文明及生活，許多心理學上的實驗，甚至直接證明房間顏色對人類心理活動的影響。

對此，我們希望以遊戲的模式，蒐集大家對文字與顏色間的直覺聯想關係，可望作為這方面研究的前置實驗數據。

## 設計概念(Design concept)

### 遊戲化設計

此專案的主要目的為將任務遊戲化以吸引使用者遊玩，透過使用者的遊戲行為產生量化的資料。目前的設計係仿敲磚塊的小遊戲，遊戲時玩家必須用板子將色球反彈，較為不同的是，目標並非敲碎方塊，而是要將色球彈到畫面右上方的太空船中。

遊戲背景故事是迷失的太空人，需要補充太空船能量（色球），才能返回地球。遊戲開始時，Player1扮演外星人，負責出題（一句英文句子），並在 Player2 進行接球動作時進行干擾。Player2 負責選擇自己認為與單字最相關的顏色，使用方向鍵移動人物，讓人物手上的板子將該顏色的色球反彈到空中，且在數次反彈後送到太空船中。如果沒有接到任何球，或是在接到球後卻沒接到反彈後的球，皆算失敗，會扣生命值。反之，成功接到太空船內會進入下一關。生命值扣完太空人會墜入黑洞，完成所有題目則會讓太空船順利航返地球。

而我們會將題目與 Player2 接住的球的顏色輸出到檔案內，做為資料收集的數據庫。我們利用球受重力影響會墜落的特性使玩家在短時間內做出最直覺的決定。

## 技術細節

### 1. Swing(GUI):

- 登入畫面：我們的遊戲是一個雙人遊戲，因此登入遊戲的 UI 也是由 Swing 完成的。
- 開始畫面：遊戲最初 Player 1 輸入句子的畫面是由 JPanel 作為基底，加上 ActionListener 以及 KeyListener。此外，JButton 利用 ImageIcon 以符合我們想達到的美術效果，當滑鼠移至 button 上時圖案會有不同，使玩家能夠清楚知道現在自己滑鼠所在的位置。比較特別的是，在這裡也用到了 Swing 的 Timer 實作文字的動畫。
- TopBar:顯示玩家的生命值，以及關卡的題目。

### 2. Processing

整個遊戲的介面幾乎都是由 Processing 來實作，是我們 final project 的重點技術。

- 遊戲畫面：Player 1 主要的功能是移動雷射槍以及發射雷射光。Player 2 的主要任務則是移動接球的板子，好讓球能夠到達太空船（終點）完成補給。這部分全都是利用 Processing 完成。
- 色球掉落以及彈跳：球的掉落亦是利用 Processing 完成。雖然球體掉落並不是由真正的物理公式寫出來的，但是我們利用每個固定時間區段，球體在這個 interval 掉落的距離會越來越長這項物理現象，增加球的 y 值；至於反彈時的情況則是相反，在達到反彈最高點之前，球體在固定 interval 所移動的距離會越來越短，我們根據這樣的現象減少球的 y 值；而球體 x 方向則是維持等速。我們就是利用以上幾個事實，模擬現實生活中球體掉落及反彈的效果。
- Transmitter(發射器管): 顯示剩餘球數與顏色，當發射出色球後，會重新顯示剩餘球色。此外，一開始球色的順序是隨機產生的，所以每一關的順序都會有所不同。
- 太空船動畫：依據球的位置，當超過一定距離，就會出現太空船開啟的動畫。
- 背景移動：隨機從12種背景抓取其一，並透過更動x位置，讓背景有移動效果。

### 3. Sprite：Player 2 太空船的動畫以及背景的捲動效果皆是利用這個技巧。

### 4. Thread：透過 multithreading 控制 Player 1 及 Player 2 的遊戲流程。

### 5. Socket：

Player 1 與 Player 2 的 synchronization 以及 interaction 皆是透過 Socket 才得以實現，以下有幾個重點：

- 同步：主要由 Player 2 傳送球的顏色、位置，以及 Plank 的 x 位置。另外，Player 1 發出雷射光時，也會傳 y 座標給 Player 2 判斷球是否被射

中而改變位置。

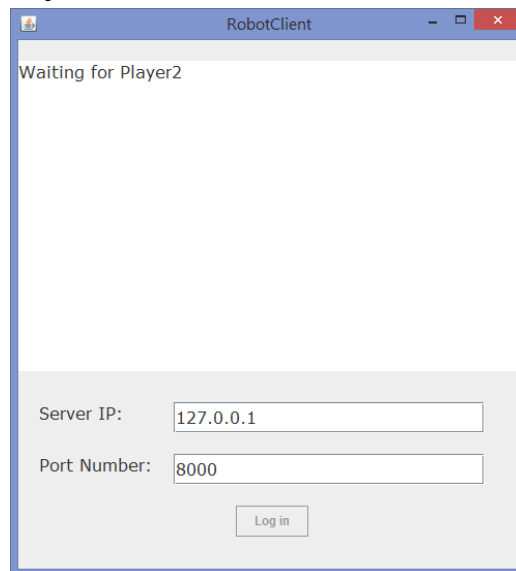
- 互動： 遊戲剛開始 Player 1 輸入的句子，會送到 Server 端以後進行切割，再傳給 Player 2 以分割成不同關卡，而球的顏色選擇也讓 Player 2 最後勝利畫面得以顯示。

6. 繪圖：

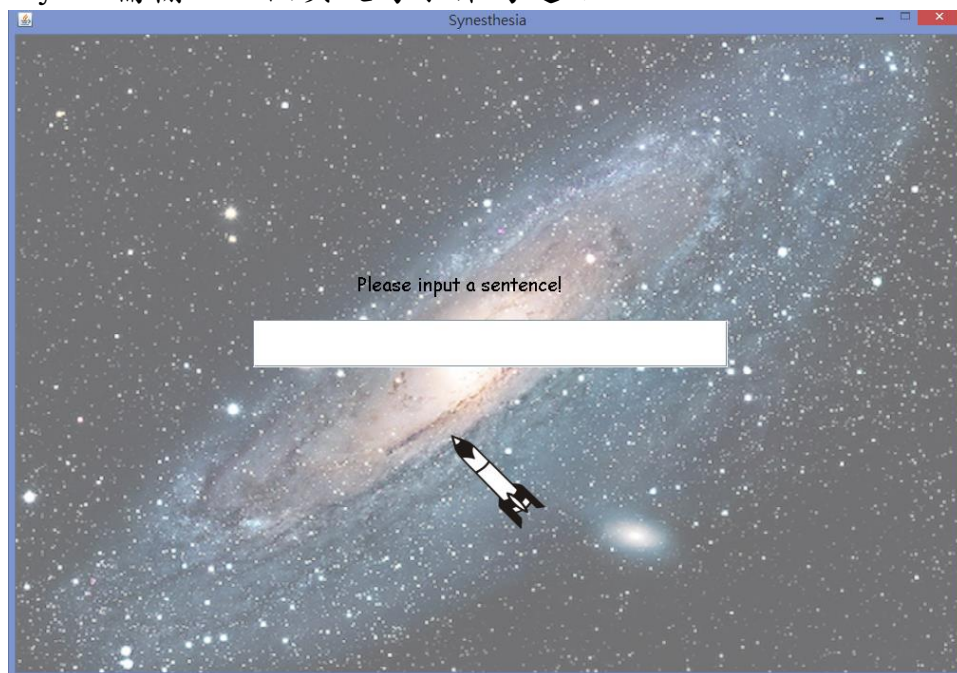
- Pixlr Editor
- 小畫家
- fiito

## 實作成果

1. 登入畫面，當Player2也登入時，會直接開啟遊戲初始畫面。



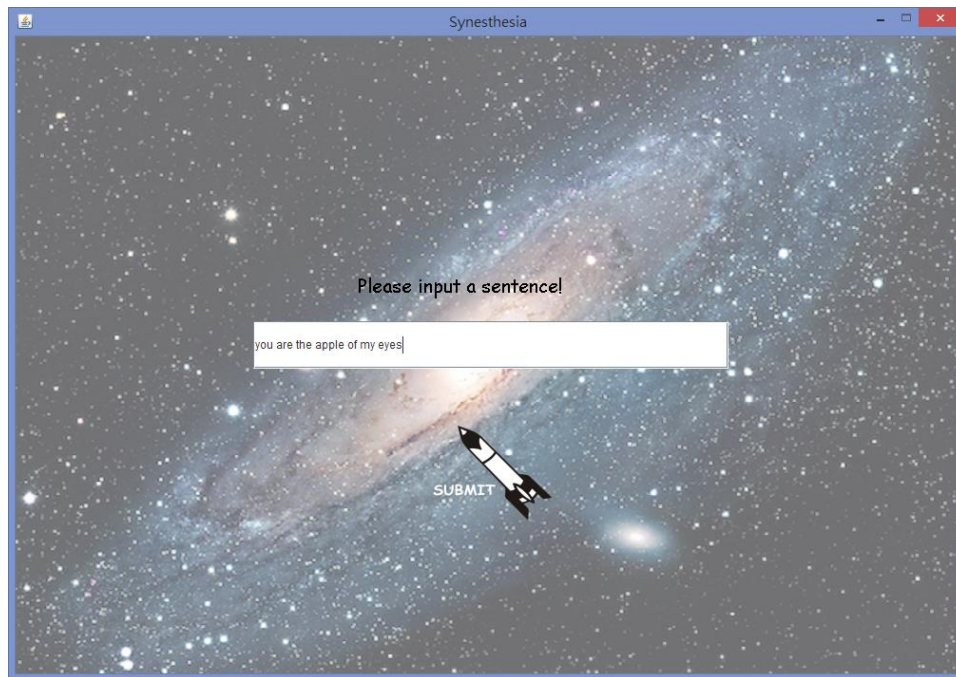
2. player1需輸入一個英文句子作為題目。



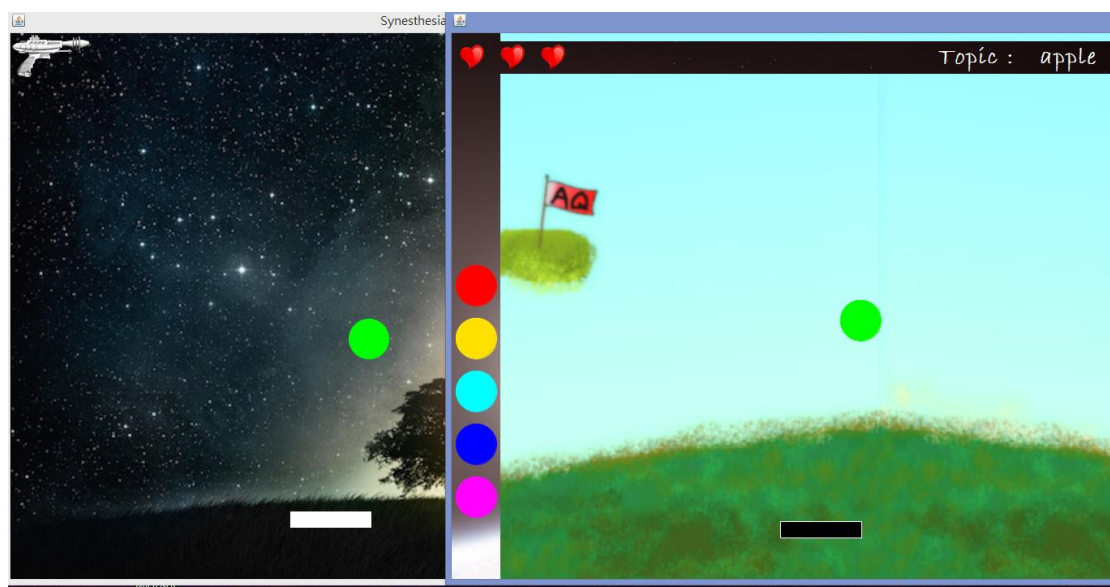
Player2則進行等待畫面。



Player1 輸入完句子後，可點火箭Submit，進入遊戲。

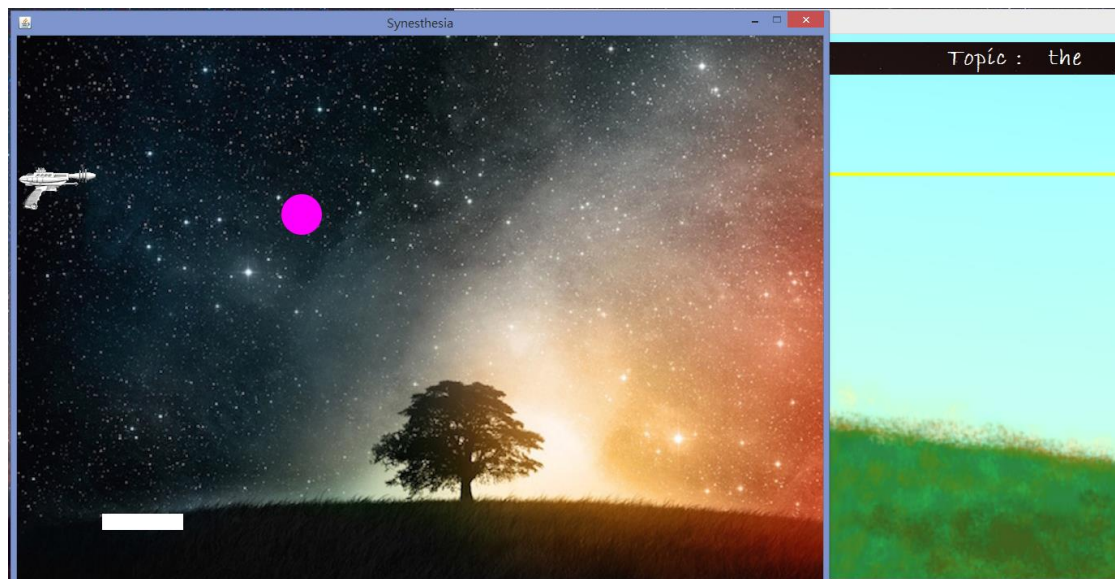


### 3. Player1 與 Player2 畫面為同步進行



Player1 可利用空白鍵發射干擾波，當球被擊中時，球的位置會被隨機放置。

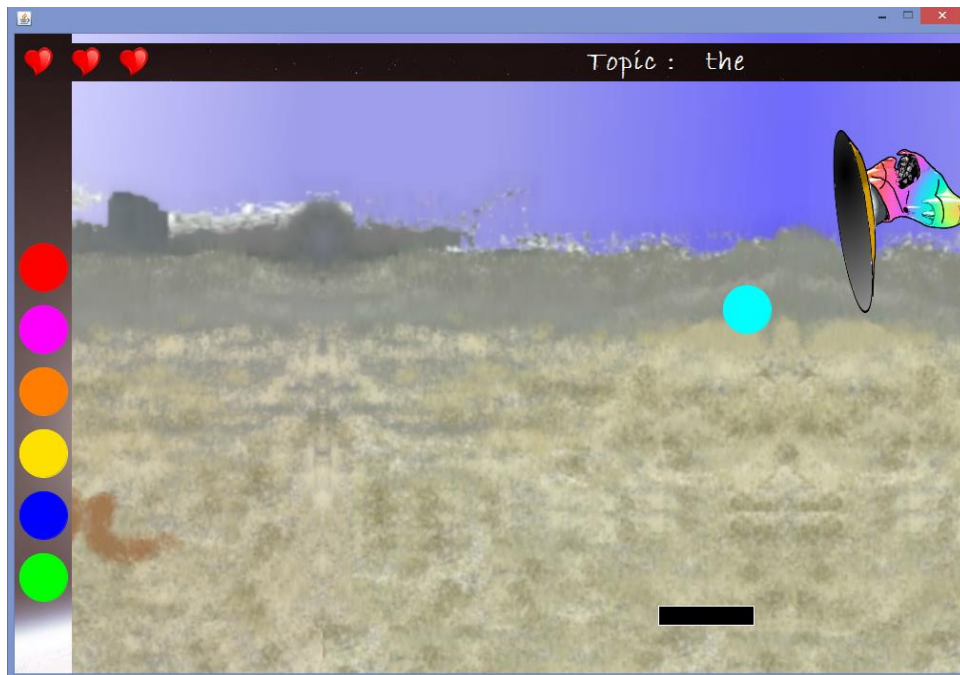




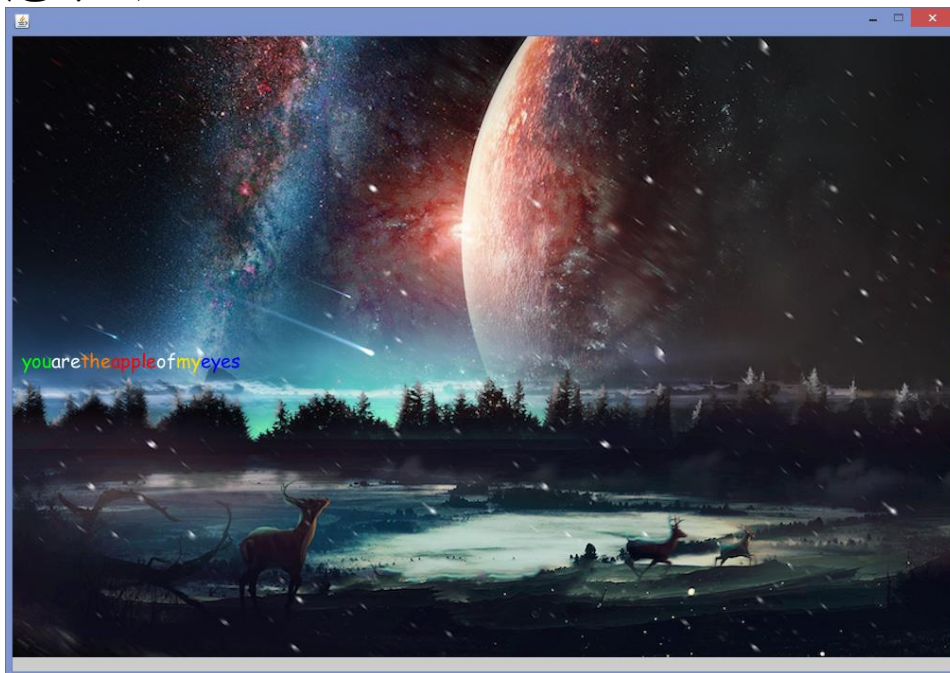
4. Player2 遊戲畫面，利用左右鍵控制底下的小板，將球運送到太空船。 遊戲背景有12種，會隨機出現。



當運送到後面時，太空船自動會出現，使Player2 可順利將球傳送到太空船中。

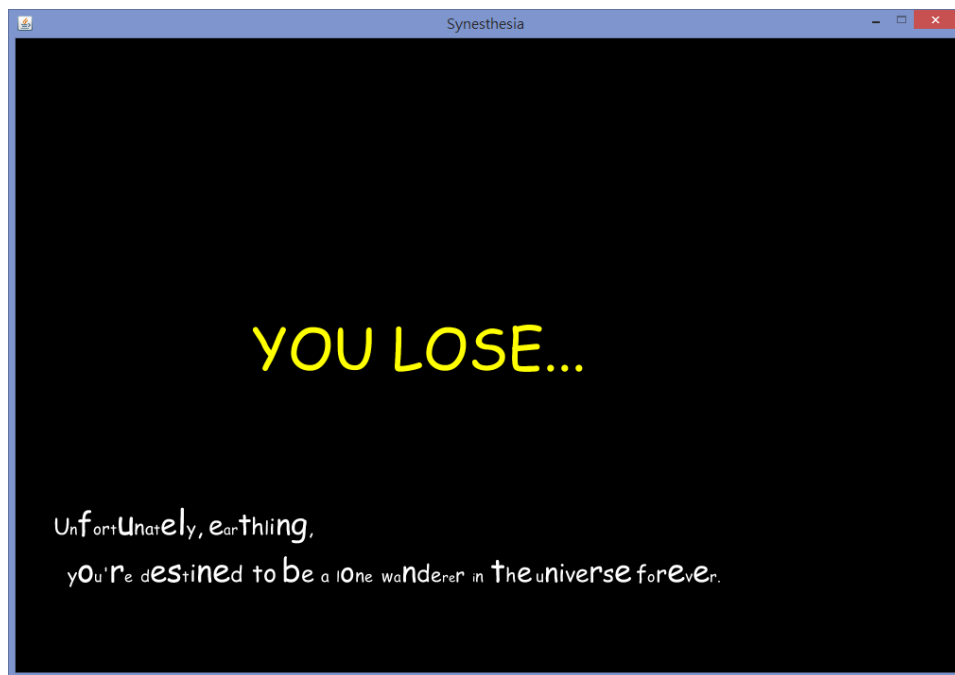


5. 當關卡破玩後，Player1 與 Player2 畫面會出現Player2所選字與顏色的組合。



當無生命值時，Player1 和 Player2 會出現輸的畫面。





## 遇到的困難：

在寫Player 1遊戲開始畫面所遇到的困難是，當使用者於TextField輸入句子時，會影響到文字動畫的進行，使得文字動畫的字母中出現了duplicates或是IndexOutOfBoundsException。在上網搜尋過許多資料還有嘗試用multithreading之後，還是無法解決這個問題。於是，我們最後的方法是每次使用者按下鍵盤的任意按鍵時，讓timer暫停；而當使用者release按鍵時，讓timer restart才能夠避免問題的發生。

## 分工

陳舒毓(組長)	25%	Proposal撰寫 美術設計 遊戲關卡核心撰寫 (Player 2：判斷干擾波、背景捲動、太空船動畫) Final Report撰寫
林奕汝	25%	Proposal撰寫 程式碼整合 玩家互動核心撰寫 (Socket) Final Report撰寫
單子芸	25%	Proposal撰寫 遊戲化設計 使用者介面設計

		遊戲關卡核心撰寫 (Player 1:輸入句子及文字動畫、發射干擾波。輸贏畫面) Final Report撰寫
阮郁善	25%	Proposal撰寫 系統架構設計 使用者介面設計 遊戲關卡核心撰寫 (Player 2: TopBar、Tansmitter、球體掉落、板子接球) Final Report撰寫

## 專業完成度及與原先計畫的差異

原先在proposal裡預定要做的功能，幾乎都有完成，有差異的部分在於，背景從原本預想的平面背景，轉變成移動式背景，並從單一畫面，便成為12種隨機畫面，富新鮮感。

	Week1	Week2	Week3	Week4	Week5
主題發想					
計劃書撰寫					
使用者介面設計					
系統架構設計					
技術工具先期研究					
遊戲關卡核心撰寫					
玩家互動核心撰寫					
程式碼整合					
內部測試					
專題展示					

## 參考資料

<https://processing.org> Processing

<http://docs.oracle.com/javase/tutorial/uiswing/components/index.html>

Java Tutorials

<http://stackoverflow.com> stackoverflow