

# Socket programming project

資工三 A 106502002 魏廷芸

題目：英語猜字遊戲：Hangman

使用語言&連線方式：以 java 實作 multithread socket

遊玩方式：

1. server 一開始開啟 socket 時，要先輸入讓 client 端猜的單字  
(answer = hangman)

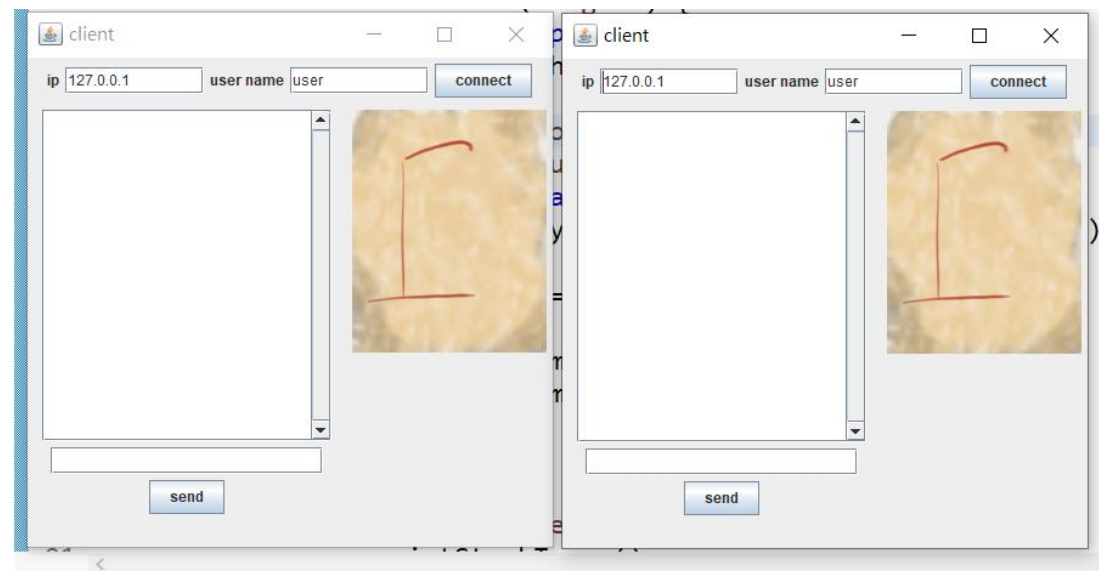
```
95         maxc = correct;
96     }
97     else {
98         broadcast(last);
99     }
100
101
102     broadcast(msg);
103     if(tmp.equals(answer)&&times<7) {
104         broadcast("-----WIN-----\n");
105         broadcast("congradulation, "+user+" !");
106     }
107     else if(times==7) {
```

Problems Javadoc Declaration Console Coverage

server (1) [Java Application] C:\Program Files\Java\jdk1.8.0\_231\bin\javaw.exe (2019年12月6日 下午10:31:53)

Set the answer :  
hangman

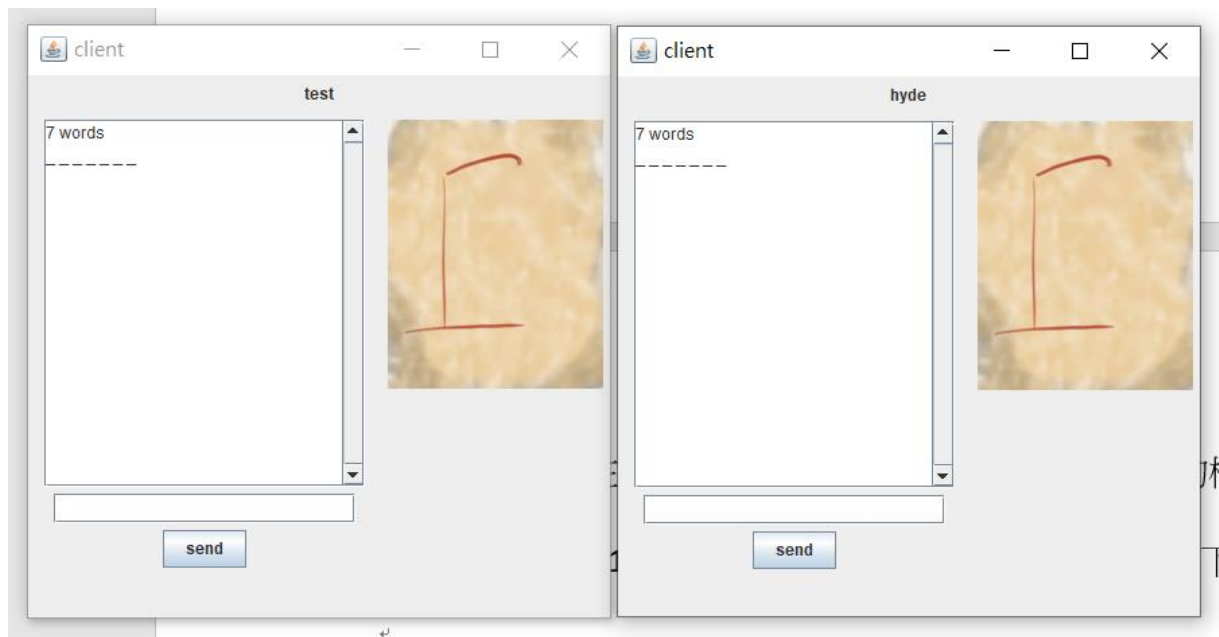
2. 接著開啟 client 端  
(以兩個 client 端為例)



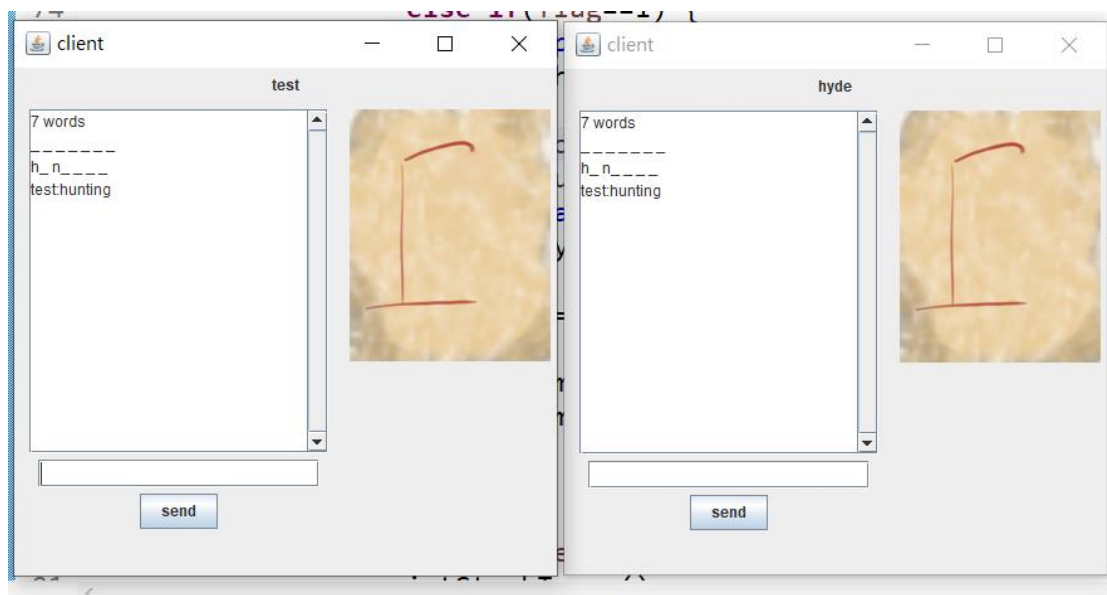
上圖可以看見在視窗最上方有輸入 IP 以及 user name 的欄位，(default 輸入為 127.0.0.1)，設好 ip 和 user name 後，按下 connect。

3. 輸入的 user name 會顯示在原本輸入 ip 和 user name 的位置，

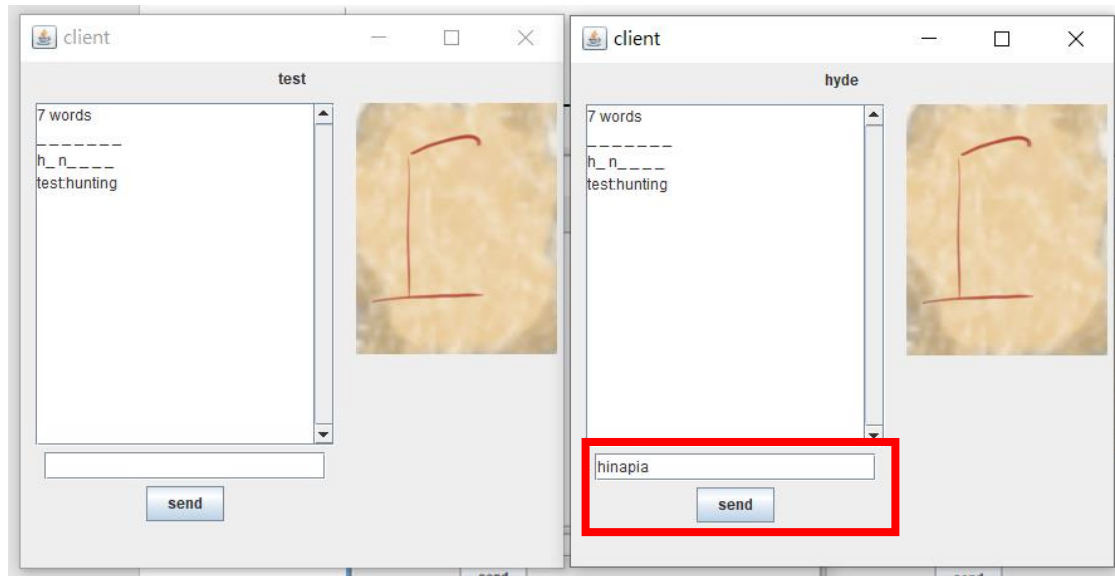
並且在對話視窗中顯示單字的字數。



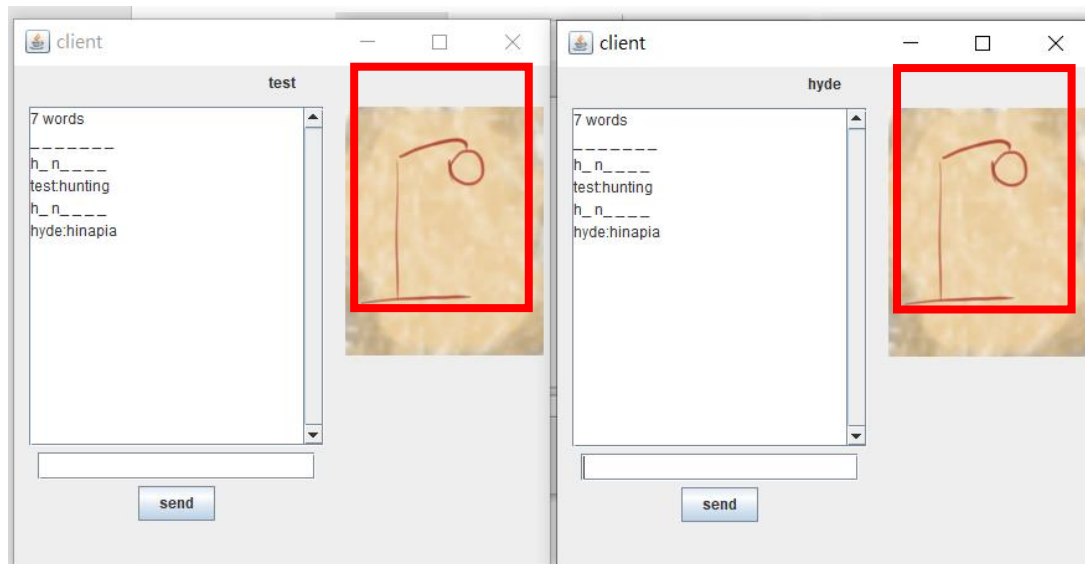
4. 玩家 **test** 在對話視窗下方的輸入欄位輸入猜測的單字：**hunting** 後，按下 **enter** 鍵(按 **send** 按鈕亦可)後，對話視窗顯示(1)更新的提示(猜中兩個字母)：**h\_n\_ \_ \_**，以及(2)玩家名稱：剛才輸入的單字



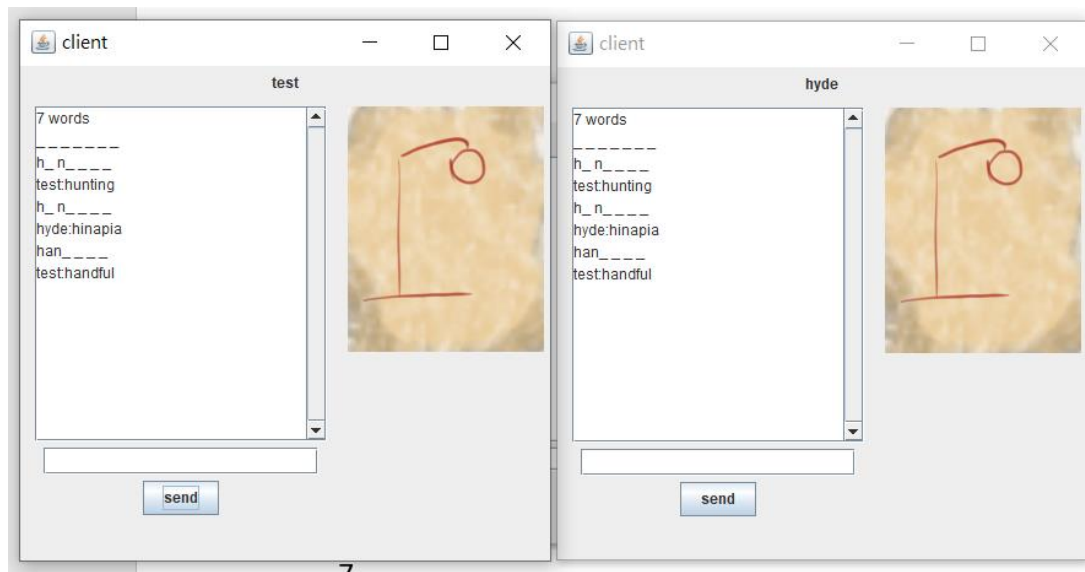
5. 玩家 **hyde** 在輸入欄打上猜測的單字後按下 **enter** 鍵(按 **send** 按鈕亦可)



6. 猜測無效，並沒有新猜到的字母。因此，絞刑台上被添了一筆。  
(按 send 按鈕亦可)

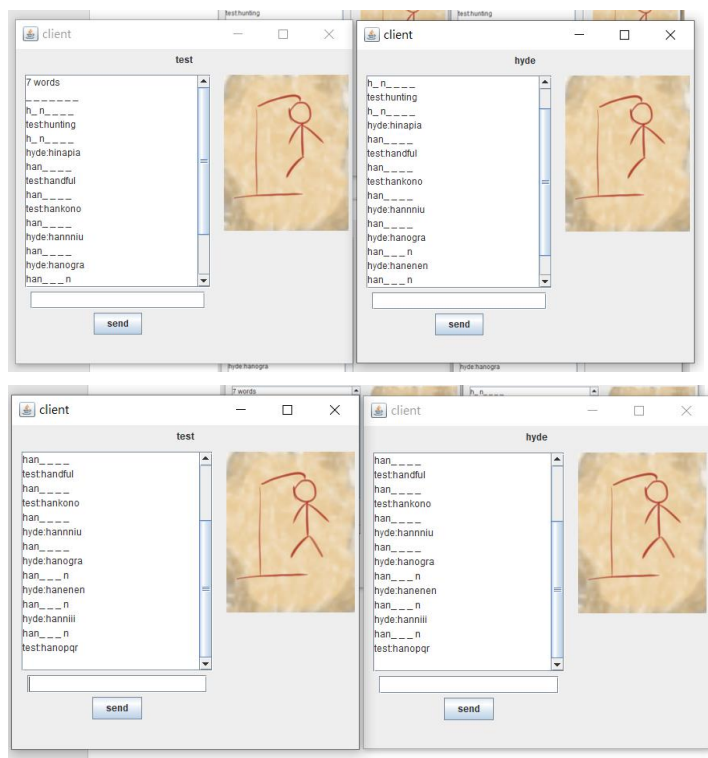


7. 玩家 test 輸入單字 handful，成功猜到第二個字母。  
提示更新為 han \_ \_ \_ \_，絞刑台維持不變

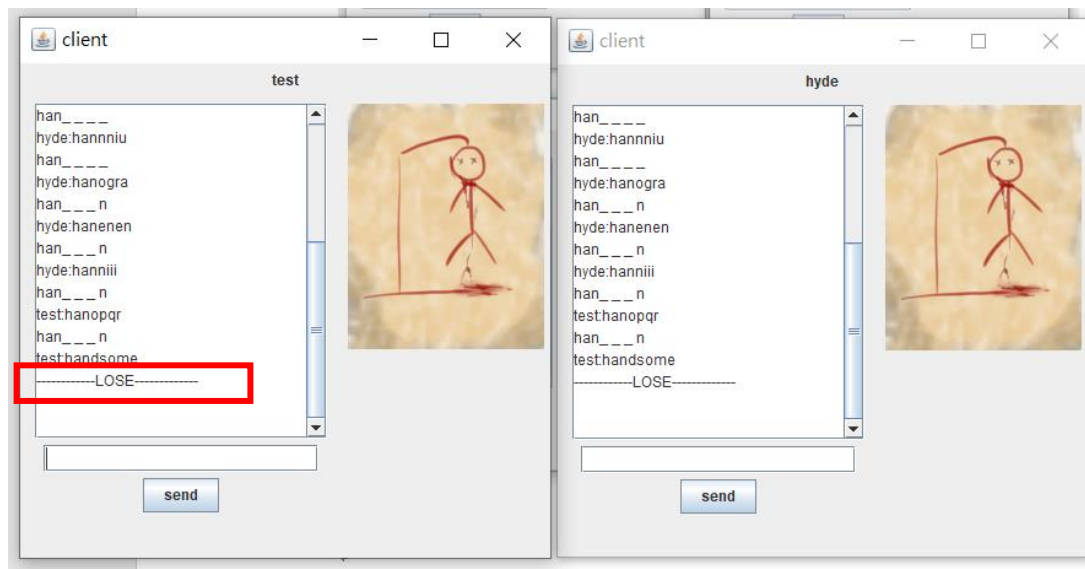


## 8. 持續猜測失敗

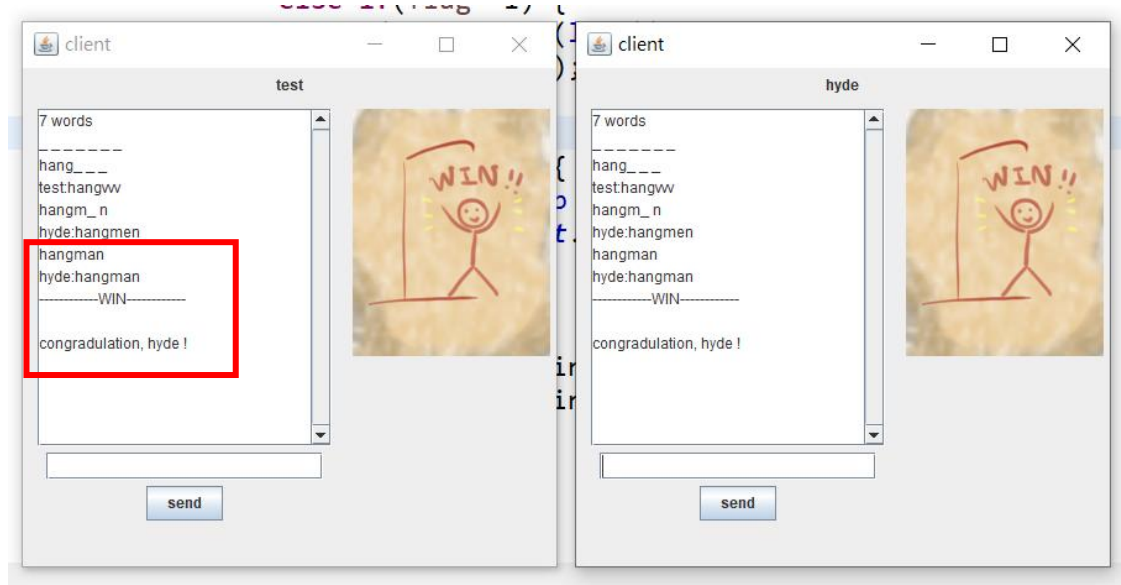




9. 在連續錯誤輸入後，絞刑台的人被完全畫出。  
遊戲失敗。



10. 反之，若成功猜到答案，則贏得勝利。



程式操作說明：

在程式中，使用的圖片路徑為絕對路徑，因此若要在自己的主機跑，須把路徑改為自己主機的絕對路徑。

```

JScrollPane roll = new JScrollPane(input);
ImageIcon i1 = new ImageIcon("C:\\Users\\hydrish\\eclipse-workspace\\networkc\\src\\networkc\\1.png");
ImageIcon i2 = new ImageIcon("C:\\Users\\hydrish\\eclipse-workspace\\networkc\\src\\networkc\\2.png");
ImageIcon i3 = new ImageIcon("C:\\Users\\hydrish\\eclipse-workspace\\networkc\\src\\networkc\\3.png");
ImageIcon i4 = new ImageIcon("C:\\Users\\hydrish\\eclipse-workspace\\networkc\\src\\networkc\\4.png");
ImageIcon i5 = new ImageIcon("C:\\Users\\hydrish\\eclipse-workspace\\networkc\\src\\networkc\\5.png");
ImageIcon i6 = new ImageIcon("C:\\Users\\hydrish\\eclipse-workspace\\networkc\\src\\networkc\\6.png");
ImageIcon i7 = new ImageIcon("C:\\Users\\hydrish\\eclipse-workspace\\networkc\\src\\networkc\\7.png");
ImageIcon win = new ImageIcon("C:\\Users\\hydrish\\eclipse-workspace\\networkc\\src\\networkc\\win.png");
ImageIcon lose = new ImageIcon("C:\\Users\\hydrish\\eclipse-workspace\\networkc\\src\\networkc\\8.png");
JLabel LA = new JLabel();

```

Server 和 client 分為兩個專案，client 端使用 swing 實作視窗化界面。

心得：

在思考連線 project 時，第一個想到的實作遊戲是你畫我猜，但因為要寫出一個繪圖板太複雜了，就改為小時候英語課常玩的吊死鬼遊戲。

連線程式比較難的點應該是設計者要去思考 server 端或 client 端發送訊息後，接收到的位置以及先後順序，我也為此苦惱了很久。

剛開始因為不熟悉 socket 功能，上網找了很多資料，也參考了網友做出連線聊天室的做法，最終才漸漸完成了這個簡單的小遊戲。

而這次的經驗也讓我學到了很多，我本身就是很喜歡遊戲的人，之前做的小遊戲卻很可惜都是單機遊戲，沒有連線功能。在這次學習後，希望未來有時間時，能夠建出一款有趣的連線遊戲。