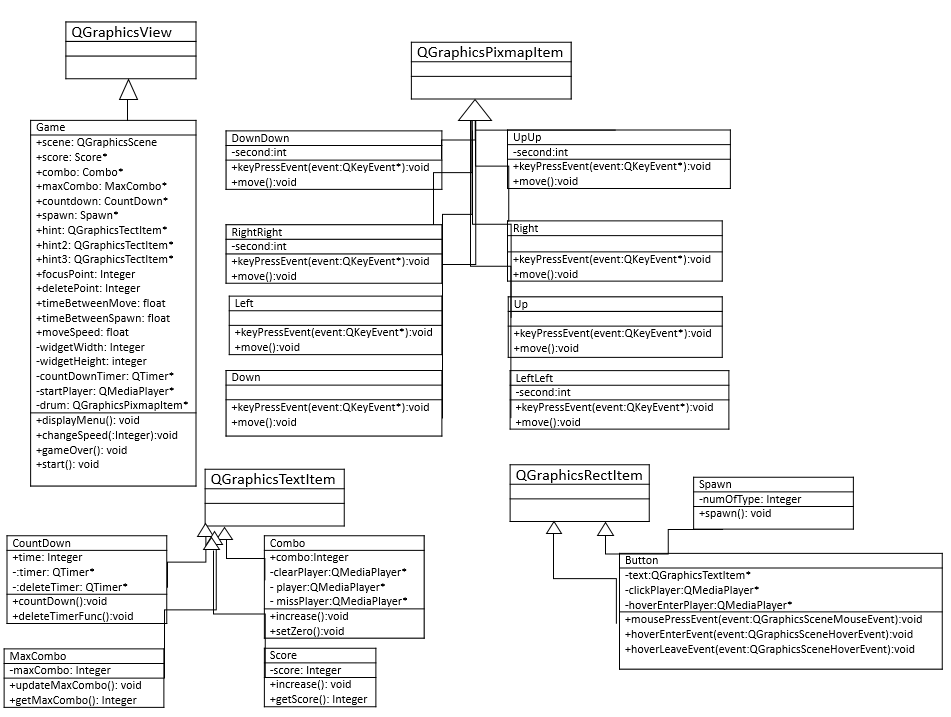
**Report**

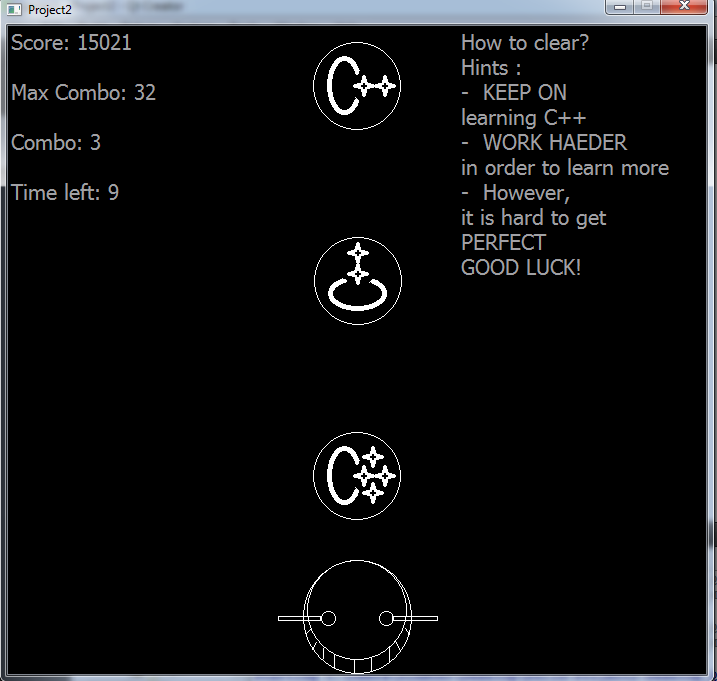
UML Class Diagram



Screen Shot



*//畫面什麼的就算了吧，大概是第一次用電腦繪圖=.=*



*//錯字已改*



// *這是滿分啊！！玩了好多次！！*

How to play

玩法在運行界面有提示：

Hint1: KEEP ON learning C++

學C++，當然就是往加的方向發展啦，所以若生成的『鼓』為『C++』時，單擊右鍵就可以消除咯，以此類推……

Hint2：WORK HARDER in other to learn more

遇到瓶頸時，當然要學習學習再學習！所以若生成的『鼓』為『C++++』時，雙擊右鍵就可以消除咯，以此類推……

Hint3：However，it is hard to be perfect

這只是想說不能perfect不要灰心，就像能把C++學完的人應該也不多……吧？不過其實多玩幾次要perfect也不太難啦嗯。

Program Architecture

基本簡介

1. Main function先宣告game物件，並呼叫Game::displayMainMenu(),生成Play和Quit按鍵。
2. 單擊Play呼叫Game::start();單擊Quit呼叫Game::close()。
3. Game::start()被呼叫後遊戲便開始了……
4. 結束時生成Play Again和Quit按鍵，單擊Play Again則再次呼叫Game::start()。

細節簡介

1. 首十秒只會生成需單擊消除的鼓，這是因為spawn物件在十秒後將生成範圍變廣（從rand()%4變成rand()%8）。
2. 鼓每十秒會變速，這是通過每十秒宣告兩個QTimer，一個定時生成鼓，另一個在9秒後刪除這兩個QTimer來達成（嗯……很像很爛不過我只想到這樣）。1秒空檔正好用來讀提示和適應改變的速度。
3. 倒數計時是通過宣告一個QTimer，然後每一秒將time減去1並更新畫面來實現。
4. 需要單擊消除的鼓在半進入可消除範圍時便可消除，而需滑動速度要雙擊消除的鼓則在接近消除範圍時便可開始雙擊（考慮延時）。
5. 計分與滑動速度、生成速度及combo次數相關。

（不）值得一提的是，鼓被消除時的音調會隨著combo增加而升高，由於無法持續升高，第40次combo及以上都是第七個八度的#F，那是我名字的諧音……（升Fa）

完