軟體工程─HW1 鄭凱文 104062223

(選定論文: Practices and Technologies in Computer Game Software Engineering)

1. 該論文的主要內容：

此論文主要探討在遊戲開發過程中套用軟體工程，需要注意的幾個重點。首先因為遊戲種類的多元化，試用該種類的開發技術，不一定能套用在另一種類上，即便是紙牌、棋類等模式差不多的遊戲，欲使用同樣的遊戲引擎、達到Reuse、Extension的目標，也需要相當可觀的開發費用和技術。

第二是遊戲開發上需要留意的核心，像Functional & Nonfunctional requirement，因為使用者導向的結果，後者往往掌控開發的主導權而使前者邊緣化。另外是Game Architecture Design，從運行硬體、軟體平台、到套用的Library以及network system，完成度將決定遊戲的成敗，是屬於開發者最珍貴的資產，需要被妥善保護(Keep Secret)，相對的這些Architecture能帶來很高的教育價值，因此強調多多接觸Free source的重要性。

再者提到現今適合的遊戲開發方法，講求Incremental development、Rapid release來符合使用者需求導向，加上需壓低製作成本的情況下，筆者比較推薦利用Agile method來開發遊戲。

最後，本文章為教育取向，裡頭提到的矛盾與問題，作者認為都是學生應該留意、學習的地方，也建議將遊戲開發當作軟體工程的課程入門。

1. 與課程內容的異同點：

<1>遊戲型態很多種，不可能一種思維、創作模式適用每一種遊戲，這個觀念就如同課文中提到的：在軟體工程諸多的開發方法中，沒有一種是絕對適用的。

<2>開發大型遊戲時，大多的使用者很難具體說明「他們想要什麼」，卻可以明白的說「哪些功能比較合他們胃口」，如同我們在評估軟體的Quality時，Usability(易用度)等評比項目，不易具體陳述好用的產品，多半採取比較方式呈現，是相對主觀的，無法單靠文字述說一個標準。

<3>文章中將Agile method，視為一個融合Incremental、Prototyping等特色的method，與課本分開討論的方式有些出入。

1. 從實務面上舉例說明如何應用該文章所提之方法：

舉例來說，如果今天要開發一個多人連線、以創新與社群聯繫為主打、橫跨Android和iOS平台的大型RPG遊戲，首先確認其為RPG劇情種類，聘請具有此種類經驗的開發者；強調創新，不適合用Unreal、Unity等現成Game Engine，需考量成本後額外訂做客製版；關係多人連線的穩定度，採用P2P或distributed server的network設計，架構上也必須支援兩種手機的軟硬體平台；因應玩家需求推出更新，在測試玩家回饋的真實程度初期，必須短而頻繁的推出新版本，加上壓低成本的考量，採取拋棄式、不著重於Reuse的修改方式；最後進入穩定經營，準備推出續作之期，再依據累積的玩家喜好保留部分內容，與完工的Game Engine一起reuse在新一代上面。

1. 心得與看法：

遊戲在程式開發中是近期最蓬勃發展的一塊，同時比起其他程式，遊戲平衡度、更新速度、平台的支援程度，這些nonfunctional requirement，重要性遠遠大過實際的內容，因此在開發上頭必然得著重於這些地方，畢竟大家最終目的仍是為了賺錢，而不是跟作者一樣注重於教育上，所以我認為現今遊戲的軟體工程上，缺少某些管理者認為應補齊的疏失，是非常合理的。就像課本中提到的，沒有一種開發方法能適用所有程式，去瞭解各自的優缺點，日後做出最好的決定，我覺得這是本文帶給我最大的教育意義。