

# 程式人《十分鐘系列》



用十分鐘

讓你的專案一開始就搞砸

陳鍾誠

2016 年 10 月 19 日

# 大部分的程式人

- 都曾經做過專案！

而且

- 大家都知道 ...

專案要搞砸很容易

要搞好卻很困難！

有那些困難呢？

# 第一個困難是

- 你做的專案沒有人要用！

為甚麼會沒有人要用呢？



原因很多

# 像是對於創新型專案

- 創新就是做沒有人做過的東西
- 既然沒有人做過當然就沒有人用過
- 那做出來的東西當然就不知道有沒有人會想用
- 如果你自己都不想用，那結果常常就是沒有人要用囉！

# 所以創新型專案

- 失敗是很常見的  
成功反而是很少見的！

# 而對於那些委託型專案

- 客戶通常不知道自己想要甚麼
- 開發者通常不是使用者，也沒辦法把系統納入日常生活運作
- 於是開發出來的東西，客戶沒辦法納入日常運作的情況就很常見！

# 更不用說那些學生專題

- 他們很多只是想要老師給歐趴，  
壓根就沒想要拿來用！

# 所以

- 專案做完十之八九沒有人用
- 有人用的專案真的不太多...

# 對了

- 我還沒提到像國科會那種專案

那只是要錢用的

- 錢拿到專案就沒用了...



# 抱歉

- 再講下去就要離題了...

讓我們拉回主題

主題是甚麼？

# 歐、對了

- 主題是專案搞砸的原因...

除了沒人用之外

專案搞砸的另一個原因是

團隊沒有起來

# 換句話說



# 就是

- 團隊離心離德，根本就沒有用心做專案，所以專案自然就搞砸了！

# 大家知道

- 軟體專案是出了名的難搞...

但是在搞定專案之前

一定要先搞定人的問題

如果人都搞不定

那專案又怎麼做得出來呢？

# 在專案執行中

- 人的問題有很多 ...

# 像是

- 不想做、不願意做
- 想做、但不想為某人做
- 還有一堆關於政治上的問題...



# 關於這點

- 國父孫中山先生早就說過：  
政是眾人之事，治是治理，  
治理眾人之事就是政治...

# 問題是

- 程式人比較擅長搞機器的事  
但是不太擅長搞人的事情...

# 所以

- 程式專案會那麼常失敗，  
也是有原因的啦...

# 既然

- 程式專案這麼常失敗，那麼到底要怎樣才能減少失敗的機率呢？

# 關於這點

- 我其實也沒有甚麼高明的想法了...

# 或許各位可以請教

- 那些《成功學》和《心靈雞湯》的老師
- 只是如果沒效千萬不要說是我推薦的...

# 我只知道

- 如果要失敗，那就快點發現
- 別等到最後才失敗，最好第一天就知道會失敗，然後趕快把專案停掉，這樣才不會損失太大！

# 就像我們寫程式的時候

- 那些可能會發生錯誤的地方，就乾脆先用 `assert` 或 `throw` 把錯誤丟出來，讓整個系統直接停掉，而不是假裝錯誤不存在繼續執行
- 愈早發現錯誤代價愈小，愈晚發生的愈致命...



# 舉例而言

- 像最近發生的三星 Note7 手機爆炸案，如果在設計或試產時就發現問題，那損失會很小，但是當貨物都已經出了幾千萬支才發現，那可就是致命一擊了.....

# 所以

- 對於軟體工程有認識的人都知道
- 從《瀑布模式》到《螺旋模式》
- 從後來的《整合建置》，《每日建置》，到現在的《continuous integration》的那些故事！

# 這種盡早失敗的法則

- 其實也適用在人的上面...

# 如果你知道

- 某人並不想做這個專案
- 或某人不想為某人做專案
- 或你自己根本不想做這個專案
- 那還是及早處理，代價會比較小一點...

# 如果你接客戶的專案

- 那麼盡早讓客戶瞭解你的專案，使用你的專案，那麼就比較容易知道專案是否走偏了方向！

# 當然

- 如果你只想拿客戶的錢，卻不想完成專案...
- 或者那個客戶是個奧客，讓他知道之後會來亂，那麼是可以考慮不要太早讓客戶知道...
- 不過在後一種情況下，其實你或許不應該接那個客戶的專案.....

# 如果你是個專題學生

- 知道某些人根本不想做專案，只是想混過專題，那最好一開始就不要找他們進來，就算找進來之後也趕快請他們出去，否則你的專題就會....

# 變成這樣





# 相信每個做過專題的同學

- 都應該能夠理解上面那張圖片的意義才對！

# 所以快速失敗

- 是我所知道的《專案管理方法》  
當中，最重要的一條原則！

# 這個原則

- 不只可以用在程式上面
- 也可以用在人、事、物、以及公司管理上

# 所以

- 失敗不是壞事

但是太晚才發生的失敗

那就是天大的壞事了！

而且

- 愈晚發生的失敗就愈致命！

最後



祝你早日失敗！

早日解脫

# 這就是我們今天的

- 十分鐘系列！

我們下回見！

Bye Bye!