

---

[草稿] (無主旨)

---

寄件者 ericyeh92094@hotmail.com

#PMBootcamp

#PM私塾

工時估計真的很重要?

拖了好一陣子，PM 私塾總算開出第一篇 (不過題目跟原來所打算的不一樣就是了)。先前社群上蠻熱鬧在討論“估計工時”這件事情，看樣子許多工程師都覺得這件事形式大過實質，做了也是白做。不過，也有 PM 覺得雖然估不準，但是協作的角度來說，還是要做：

“重要的不是時程有多準，而是「它為什麼不準，我忽略了什麼，下次如何讓它更準」。”

首先，我十分認同這位 PM 的正向思考，也同意文章中所提到的其他協作面向。不過，如果只把焦點放在“工時能不能越估越準”這件事上，根據我半輩子都耗在開發上經驗，日子會越過越鬱卒。就算是非常成熟的團隊，非常厲害的 **developers**，經驗十足的經驗 PM，我也從來沒看過估準過的工時，或是十足十根據 PM 的規劃進行的情形。預估 10 天完成的工作，最後拖了一個月或甚至整個 **milestone** 的事所在多有。話說回來，因為這樣團隊每天都處在“危機模式”？情況也不是每次都那麼嚴重。

所以說，工時估計真的不重要？當然不是，但是重點不是全在工時，真正的關鍵是在“工法”。

團隊在做規畫的時候，不會只看時間排得進排不進時程表，而是要 **review** 整個功能要怎樣做出來。有沒有要靠其他 **API** 的地方？有沒有算法不清楚的地方？有沒有人要休假？成熟的團隊，會在事前將問號盡量開出來。問號越多，工時越不準，大家心裡都有數。但是團隊知道有哪些代辦事情，計畫還是可控的。另一個成熟團隊的指標，是當計畫或規格有變化時，能不能很快的擬定相對應的工法以及相關的問號 (其實就是相依事項 **dependency**)。

真正時間的控制，就是必須完成的時程內，選擇砍規格之後的簡單工法 (相依度低) 或是增加功能的複雜工法 (相依度高)；而不是死守一個工法，來要求 PM 一定要給出多少時間。

所以嚴格來說，“工時”在這裡只是個綜合指標，代表團隊對開發執行的掌握度。斤斤計較對它的估計準不準，就是拿 **KPI** 當成果的謬誤了。