Scrum 的

* 三種角色
  + 產品負責人 (Product Owner)
  + SCM (Scrum Master)
  + 團隊 (Team)
* 四種會議
  + 衝刺計畫會議 (Planning)
  + 每日站立會議 (Daily Standup)
    - 從昨天到現在我做了什麼
    - 今天我準備做什麼
    - 是否有遭遇到什麼困難
  + 展示會議 (Review)
  + 回顧會議 (Retrospective)
* 三個產物
  + 待辦產品 (Product Backlog)
  + 衝刺待辦產品 (Sprint Backlog)
  + 增益(遞增成品) (Increment)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 產品待辦事項 | 衝刺待辦事項 | 預備 (分析及文件製作) | 開發(進行中) | 開發(完成) | 測試 | 發佈 | 完成 |
|  | 10 | 2 | 2 |  | 1 |  |  |

P.119 圖4-9

* 個人看板 (Personal Kanban)：提供給個人或小團隊的精實思想 (Lean Thinking for Individuals and Small Teams.)
* 類專案管理 (Similar Project Management)
* 需求-估算-文件製作與分析-開發-測試-展示與回饋
* TODO(ready)-Doing-Done

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 待進行工作 | 準備 | 今日工作 | 正在進行的工作 | 等待中 | 完成的工作 |
|  |  |  |  |  |  |

* 類文件Similar Document (ex: 活文件or測試案例)  
  驗收程式的測試計畫，即將開發的功能的測試案例  
  迅速喚起回憶的東西，系統架構說明書或是描述開發程式的規格文件
* 專案底下開Document資料夾，底下再分為：
  + 文件目錄: 提醒自己把程式的功能在這裡做簡單扼要的描述(寫一個系統說明文件)。
  + 測試目錄: 提醒自己這支程式能做到哪些功能了。
* 「交給自己的文件」沒有任何原始碼的程式應該獨自存在，而沒有文件的伴隨。
* 程式需要文件來替它做說明，更需要測試程式來保證它的好壞，也需要系統架構以便進行維護運作。
* 三種基本必備的文件
  + 需求說明文件：做說明用。
  + 系統概述文件：便利維護運作。
  + 活文件：提供自動化測試的基礎。
* 文件製作的價值觀：Good Enough及四個準則
* 文件包含「概述部分」與「明確描述」，兩者都必須遵守文件的敏捷規範：
  + 一定要維持輕量化(lightweight)。
  + 產出文件一定要維持高品質，也就是具有：準確、最新的、高可讀性、足夠簡潔跟嚴謹的結構。
  + 撰寫文件必須採用方便、易開發維護、並能夠產出高品質文件的工具。
  + 明確描述的文件部分，必須可以跟著程式碼做改變，也就是與程式同步。
* 提供概述部分的Big Picture的文件。
  + 「需求說明」文件：能讓人很快弄清楚程式目的的文件。
  + 「系統概述」文件：能讓人很快弄清楚系統架構的文件(User Story Mapping是最佳的文件之一)。
* 提供明確描述的活文件(Living Document)。
  + 解決「沒有銀子彈」的活文件，活文件提供自動化測試的基礎。能夠伴隨著程式開發同步更新的文件謂之「活文件」。