

BeApp 測試計畫 前置導讀 範本

目錄

| | |
|-----------------------|---|
| 一 · 概述..... | 2 |
| 1. 測試優序判定..... | 2 |
| 2. 往下個階段的決定..... | 2 |
| 二 · 測試階段目標..... | 2 |
| 1. 穩定版..... | 2 |
| 2. Alpha 版 | 3 |
| 3. 準備營運版..... | 4 |
| 三 · 測試目標對應技術..... | 5 |
| 四 · 目標與測試工作的行程對應..... | 6 |

一．概述

1. 測試優序判定

專案導入測試流程至上架營運會經歷多個版本，可總結為三個階段版本：穩定版、Alpha 版、準備營運版(High Usability)。

測試在對應的目標版本，使用的測試技巧與問題判定優序有所不同，例如：在「穩定版」階段著重在功能正確性，採取功能正向測試，驗證功能執行流程如規格書所設計，而其他的測試方式，例如：關連性測試、反向測試就不在「穩定版」階段，即便發現問題，其優序低於修復基礎功能的問題。

2. 往下個階段的決定

每個階段都有結束與開始的特徵，但不代表滿足特徵就可往下個階段進行，因此需要負責專案的主管人員根據專案進度、測試結果..等各項數據做整體評估後，再決定專案前往下個階段。

雖然每個階段的意義是虛擬的，沒有實際感受，但在專案成員的心理壓力與預期時間具有正向影響，例如：專案處於「準備營運版」階段，每個專案成員的心理預期，最多就一個月要上架營運，再努力一下。

二．測試階段目標

1. 穩定版

- a. 目標：功能、系統的正确性驗證，保證執行結果符合預期。
- b. 時程：1 個月為 1 個 Milestone，每週為 1 個 Check Point，共計 1 個月最少 3 個 Check Point 版本。
- c. 說明：

工程師開發過程中，雖然會進行單元測試，但功能是由多個單元組成，因此功能執行結果通常不如預期，仍需修復/修改、產出新版、測試，三個步驟的反覆循環，最後完成功能。

系統是由多個功能所組成，系統如預期運作仰賴各個功能之間的資料傳遞與邏輯關連正確性，即便是同一個系統，但不同的操作流程造就不同功能執行順序的排列組合，這龐大的排列組合超過工程師、系統分析師可承擔工作量，也超過本身的專業。

因此透過測試人員模擬使用者的操作習慣，將各種可能的問題定義優先修復順序，並且透過發現問題的重現步驟，協助工程師快速的定義問題、修復問題。

d. 開始與結束的特徵

開始

專案已經沒有規劃新功能製作，所有的設計均已實做成為功能。
版本有 80% ~ 90% 的功能是可以操作。

結束

現有功能經過版本推進，所發現新問題的數量逐步減少，修復問題的數量逐步上升。

2. Alpha 版

a. 目標：導入使用者行為，聘請 Target User 評估系統。

b. 時程：

目標導向沒有固定週期與時間，根據功能完成度與目標契合的程度。
確認時間週期略長於穩定版，如：每個月 2 次 Check Point 版本。

c. 說明：

使用者不一定根據的系統設計操作功能，因此測試員模擬使用者，或是聘請 Target User，專案透過每個人的不同操作習慣，發現原本沒有想到，或是非預期但為正常的操作問題。

此階段的測試員常用的測試手法包括：關連性測試、反向測試、邊界值測試..等。

導入使用者測試為了追求成效，通常會多個使用者同時操作，因此 Server 效能調校也在本階段進行，透過壓力測試工具的開發與反覆調校與測試驗證，統計測試數據，協助工程師定位效能問題，逐次調校至預期目標。

本階段發現的問題需仰賴上個階段的測試結果，讓發現問題的影響變數趨近單純。

d. 開始與結束的特徵

開始

模擬使用者的測試，導入 Target User 評估。還有效能調校。

結束

版本測試結果貼近目標。例如，下頁所列目標：

「User Friendly」

共定義 10 個項目，均測試通過，沒有發現新問題。

「Server 效能每小時服務 500 人」

模擬 500 人連續 1 小時內持續與 Server 取得資料，沒有發現 Server 記憶體異常上升或 Crash 狀況。

3. 準備營運版

a. 目標：進行商業邏輯、架設正式台與資料備份、模糊測試的執行，在專案上架前的最後確認。

b. 時程：距離專案計畫上架營運前的時間點

c. 說明：

功能正確性與各項測試均已完成，就功能而言，專案功能處於高可用度 (High Usability) 的狀態，專案可進行商業邏輯的調整，例如：付費下載的價格、功能推撥的頻率，另為了穩定提供服務，將可考量系統故障的備援機制。

測試人員均已完成原計畫的測試項目，而採以模糊測試、猴子測試..甚至漏洞測試等測試手法，嘗試在發現版本中的潛藏問題，並且計畫上架營運開放初期的「開服測試」。

d. 開始與結束的特徵

開始

完成專案目標達 90% 以上。

結束

專案上架營運。

三 · 測試目標對應技術

以表格歸納第二章描述的版本各階段的週期與對應測試技術。

本專案未含蓋項目以灰色字體表示。

| 階段目標 | 測試週期 | 測試技術 |
|---------|------------------------------------------|-----------------------------------------------------------|
| 穩定版 | 每月 1 個 Milestone 每週 1 個 CheckPoint | 基礎功能、系統整合測試。 |
| Alpha 版 | 每月 1 個 Milestone 每 2 週 1 個 CheckPoint | 模擬使用者操作，包括：關連性測試、反向測試、邊界值測試。 Server 測試項目，包括：效能測試、穩定測試。 |
| 準備營運版 | 目標導向 每週 1 個 Milestone | 商業邏輯驗證、壓力測試、回歸測試、模糊測試、開服測試。 |

四．目標與測試工作的行程對應

以表格方式將本專案的測試項目與週次時間做對應

| 週次 | 週目標 | 分類 | 星期一 | 星期二 | 星期三 | 星期四 | 星期五 | |
|-------|-------------------------------|------|-----------------------------|-------------------------------|------------------------------------|---------|------------------|---------|
| 1 | 基礎功能 (穩定版) | App | 會員 帳號管理 | My EMotion | 更新版本 > 驗證修復 Bug | AvaChat | | |
| | | | 新手教學 | Ultimate EMotion | > Bug 關連功能測試 | | | |
| | | | 裝置管理 | | IF TTT | | | |
| | | 後台 | | | | | Dashboard | Emotion |
| | | 效能測試 | | | | | 帳號管理 會員管理 | 推撥通知 |
| | | | | | | | 測試準備：確認 API 使用方式 | |
| 2 | 關連性 測試 (Alpha 版) | App | 跨平台資料傳遞 聊天室文字、表情、影片 等 | | 更新版本 > 驗證修復 Bug > Bug 關連功能測試 | 相容性測試 | | |
| | | | | | | 回歸測試 | | |
| | | 後台 | Homepage | EMotion 在編輯/刪除 資料後 的關連功能影響 | | | 回歸測試 | |
| | | | Language | | | | | |
| IFTTT | | | | | | | | |
| | | 效能測試 | 測試準備：編寫 Robot Script | | | | | |
| 3 | 正式環 境驗證 (準備 營運 版) | App | 使用者體驗 模糊測試 | 配合因效能調校產出新版本而做功能正確性複驗 | | | | |
| | | 後台 | 無 | | | | | |
| | | 效能測試 | 執行效能測試與效能調校驗證 | | | | | |