**國立台灣大學  
106學年度第一學期軟體工程課程**

專案執行計畫書  
Shake and Chat 專案計畫

指導教授：李允中 教授

王瀚磊、陳立誠、李旭恩、綦家志、王本奕、高睦修

# 目錄

目錄 2

版次變更紀錄 3

1.專案內容 4

1.1專案起源 4

1.2開發工具與資料管理 4

2. 專案生命週期與工作分配 5

3. 里程碑 6

4.專案相關參與成員 6

4.1相關參與成員名單 6

4.2開發工具與資源 7

4.3知識技能需求 7

4.4計畫訓練表 7

5.專案風險管理 8

6. 時程與進度審查監控機制說明 9

6.1落後百分比 9

6.2矯正基準 9

6.3矯正措施 9

# 版次變更紀錄

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 版次 | 負責人 | 日期 | 變更項目敘述 | 審查者 | 審查日期 |
| 0.1 | 陳立誠 | 10/15/17 | 建立PEP初稿 | 全體組員 | 10/17/17 |
| 0.2 | 王本奕 | 11/01/17 | 補完開發工具與effort預估 | 全體組員 | 11/02/17 |
| 0.3 | 王瀚磊 | 11/02/17 | 調整專案起源、開發工具敘述，以及知識需求、訓練計畫表。最後新增風險管理之補充敘述。 | 全體組員 | 11/03/17 |

# 1.專案內容

**1.1專案起源**

隨著時代演進，電話通訊逐漸被網路通訊取代。藉由網路的便利，通訊的模式變得更多元、更有趣。通訊不再只是文字訊息與聲音訊息的傳遞，更可以是照片、影片或是文件的傳遞，充滿著無限的可能。

網路通訊中，又以即時通訊系統最為常見，舉凡現今流行的Line, Whatsapp, Facebook Messenger … 等，這些應用皆環繞在mobile裝置上，讓使用者可以無時無刻即時傳遞訊息給對方。觀望現今使用者透過即時通訊軟體之交流，雖然縮短了朋友之間的距離，卻讓使用者侷限在自身既定的交友圈，反而失去了與周遭其他人的互動。

本專案著重於加強現有即時通訊系統之功能，並利用此功能促進使用者與身旁周圍的陌生人有進一步的互動，擴增交友圈。本專案藉由即時通訊在手機上應用之特色，我們可以透過網路或是GPS取得使用者的定位。透過這些即時的定位資訊，將周圍其他陌生的使用者顯示，並期望使用者能在與其他陌生人的交流之中拓展自己的多方交友。

本專案使用Zulip，開源之即時通訊系統，並新增其定位功能、搖晃手機偵測功能，透過以上功能實作出搖一搖(Shake and Chat)。

**1.2開發工具與資料管理**

本專案使用的即時通訊系統為Zulip。會議記錄使用Microsoft Office 2013。 資料庫為Postgre SQL 9.1，程式語言為python3，環境為ubuntu 16.04。手機client端使用Android Studio2 開發，使用套件包含android SDK API23、Google APIs、 Google Play Services。版本控制使用gitlab。

# 2. 專案生命週期與工作分配

­

圖1:專案執行甘特圖

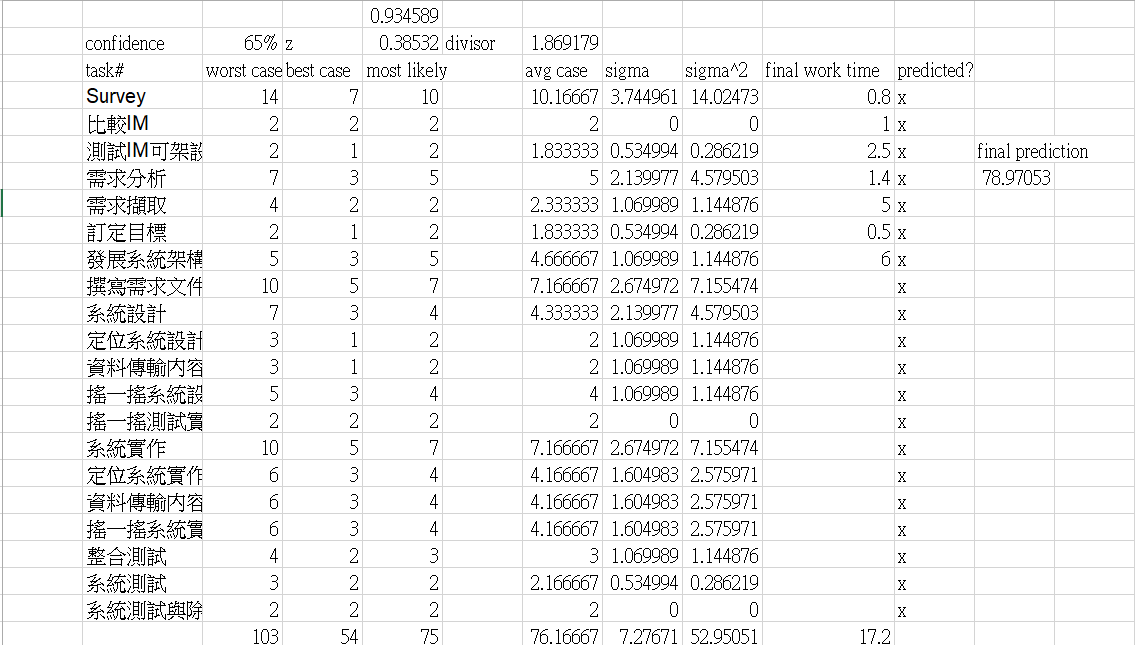


表1:WBS以及總Effort預估表

# 3. 里程碑

|  |  |  |
| --- | --- | --- |
| 里程碑 | 日期 | 內容 |
| M1 | 2017/10/15 | 決定IM與應用 |
| M2 | 2017/10/24 | 需求文件產出 |
| M3 | 2017/11/7 | 設計文件產出 |
| M4 | 2017/11/21 | 模組驗收 |
| M5 | 2017/12/12 | 系統原始碼產出 |
| M6 | 2017/12/19 | 測試文件產出 |
| M7 | 2017/12/26 | 驗收測試 |

表1: 里程碑

**4.專案相關參與成員**

**4.1相關參與成員名單**

|  |  |  |
| --- | --- | --- |
| 姓名 | 職稱 | 資源角色 |
| 李允中 | 國立台灣大學資訊工程系教授 | Senior Manager |
| 王瀚磊 | 資工所碩二 | Project Manager |
| 陳立誠 | 電子所碩二 | Team Member |
| 李旭恩 | 資工系大四 | Team Member |
| 綦家志 | 電子所碩一 | Team Member |
| 王本奕 | 資工所碩一 | Team Member |
| 高睦修 | 海洋工程所碩二 | Team Member |

表2:相關參與成員名單

## 4.2開發工具與資源

本專案計畫預計將會使用下列開發資源工具：

* 硬體設備：個人電腦\*6、學術網路、android手機\*6
* 軟體設備：Gantt Project2.8.5、Microsoft Office 2013、Android Studio 2、ubuntu 16.04、spyder IDE、vim、google hangout、Postgre SQL 9.1。

## 4.3知識技能需求

|  |  |  |  |
| --- | --- | --- | --- |
| 專業知識技能 | 人數 | 預計受訓人員 | 說明 |
| Zulip Server/Client | 6 | 全體 | 本專案所套用之Instant Messenger，其server/client端原始API之應用與熟悉 |
| Google Map API | 6 | 全體 | 在client端(mobile)上之定位系統係透過google map之API取得其定位資訊，讓 API 從全球定位系統 (GPS) 及 WiFi 與手機基地台訊號資料等可用的位置提供者，儘可能測定最精確的位置 |
| Android Studio 2 | 6 | 全體 | Client端之應用皆採用Android Studio 2作為開發平台，因此全員需在專案進行前熟悉此開發工具 |
| Python 3程式設計 | 6 | 無 | 本專案的成員已有相關經驗 |
| PostgreSQL 9.1 | 6 | 全體 | Zulip Server端所使用之database，由於新增之定位功能需仰賴資料庫之擴充欄位及搜尋等功能，因此全員需在專案進行前熟悉此工具 |

表3:知識技能需求

## 4.4計畫訓練表

|  |  |  |  |
| --- | --- | --- | --- |
| 訓練項目 | 參與人員 | 訓練方式 | 訓練時間 |
| Zulip | 全體 | 將參與人員分配成兩個群組，分別研究及熟悉server以及client之功能 | 2017/10/15~  2017/10/29 |
| Google Map API | 全體 | 由參與人員自行學習，並互相交流。對於研究client之群組成員需要求高掌握度 | 2017/10/15~  2017/10/29 |
| PostgreSQL 9.1 | 全體 | 由參與人員自行學習，並互相交流。對於研究server之群組成員需要求高掌握度 | 2017/10/15~  2017/10/29 |
| Android Studio 2 | 全體 | 由參與人員自行學習，並互相交流。對於研究client之群組成員需要求高掌握度 | 2017/10/15~  2017/10/29 |

表4:計畫訓練表

**5.專案風險管理**

* 低影響程度：風險項目對整體專案影響不超過30%
* 中影響程度：風險項目對整體專案影響大於30%，小於60%
* 高影響程度：風險項目對整體專案影響超過60%

|  |  |  |  |
| --- | --- | --- | --- |
| 風險項目 | 影響程度 | 發生機率 | 解決方式 |
| 實際進度預期不同 | 高 | 高 | 加開會議，討論是否變更進度 |
| 需求擷取模糊、不完善 | 高 | 高 | 學習相關領域知識以及参考現有系統 |
| 開發技術不熟悉 | 高 | 高 | 規劃教育訓練 |
| Coding能力不一致 | 中 | 高 | 加強學習以及分配匹配能力的工作 |
| Coding標準不一致，難以維護 | 中 | 高 | 訂定並遵從Coding Standard |
| 開發文件管理不易 | 中 | 中 | 使用Gitlab集中管理 |
| Android Studio所開發之API版本不相容 | 高 | 低 | 經由所有開發者協同後統一API進行開發 |
| 測試伺服器(public IP)取得不易 | 中 | 高 | 借用組員之實驗室伺服器，平時關閉，若要測試再通知負責組員。 |
| 美工不足 | 高 | 高 | 參考各家知名instant messenger之美工設計 |
| 期中考導致進度落後 | 高 | 高 | 之後加快進度 |
| 開會時間難以統一配合 | 高 | 高 | 每次開會結束後先規劃好下次會議時間，並盡量挑選較多人能參與會議的時間，同時盡量避免線上會議所導致之效率低迷 |

表5:風險列表

**6. 時程與進度審查監控機制說明**

本專案對於進行中的工作每週監控一次，由專案負責人針對所完成的工作比例進行進度審視的方式。

**6.1落後百分比**

(現在時間-截止時間)

D= X 100%

(截止時間-開始時間)

**6.2矯正基準**

* 106/9/13~106/12/26矯正基準定為20%，當時程落後超過20%必須實施矯正措施。

**6.3矯正措施**

矯正措施由專案負責人邀請相關人員召開會議，討論如何修改時程規劃，並實行之。本專案的監控項目如下所列：(下面為預設監控項目，但計畫執行中可依此監控項目對計畫之重要性及計畫資源情況，設定是否為監控項目)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 編號 | 里程碑項目 | 預期完成 | 實際完成 | 矯正基準 | 矯正機制 |
| M1 | 決定IM與應用 | 2017/10/15 | 2017/10/15 |  | 明確列出未完成的部分，之後每隔2日確認完成的進度 |
| M2 | 需求文件產出 | 2017/10/24 |  |  | 明確列出未完成的部分，之後每隔2日確認完成的進度 |
| M3 | 設計文件產出 | 2017/11/7 |  |  | 明確列出未完成的部分，之後每隔2日確認完成的進度 |
| M4 | 模組驗收 | 2017/11/21 |  |  | 列出落後之百分比數，之後加派人力並於期限完成 |
| M5 | 系統原始碼產出 | 2017/12/12 |  |  | 列出落後之百分比數，之後加派人力並於期限完成 |
| M6 | 測試文件產出 | 2017/12/19 |  |  | 列出落後之百分比數，之後加派人力並於期限完成 |
| M7 | 驗收測試 | 2017/12/26 |  |  | 列出落後之百分比數，之後加派人力並於期限完成 |