

# Task Board (T. B.)

## Project Execution Plan (PEP)

---

Version: 1.0

Team6



Name	ID	E-mail
劉柏宏	103590001	lucy933626@gmail.com
鄭鴻仁	107598013	hong012078@gmail.com
陳巧宜	107598025	jammy1129@gmail.com
洪子軒	107598042	gay88358@yahoo.com.tw

Department of Computer Science & Information Engineering  
National Taipei University of Technology

09/26/2018

# 目錄

目錄 .....	1
版次變更記錄 .....	1
1. 專案規劃及查核點 .....	2
1.1 專案工作內容 .....	2
1.1.1 技術方法 .....	2
1.1.2 分工結構圖 .....	2
1.1.3 工作分包與工作項目估算模型與方法 .....	2
1.1.4 工作分包與工作項目總表 .....	3
1.1.5 工作分包與工作項目內容說明 .....	4
1.1.6 專案生命週期定義 .....	10
1.2 預定時程及查核點 .....	10
1.2.1 預訂查核點說明 .....	10
1.2.3 時程與進度審查監控機制說明 .....	11
2. 專案成員工作指派 .....	12
2.1 工作項目或工作分包預估需求與估算之假設條件 .....	12
2.2 計畫成員指派 .....	12
2.3 調整專案成員 .....	13
2.4 專案專業知識與技能需求 .....	14
2.5 訓練計畫表 .....	14
2.6 成員參與情況監控機制說明 .....	14
3. 資源需求 .....	15
3.1 計畫經費預算說明 .....	15
3.2 人事費用估算 .....	16
3.3 計畫經費預估表 .....	16
3.4 預算監控機制說明 .....	16
4. 資料管理規劃 .....	17
4.1 資料管理計畫 .....	17
4.2 列管資料總表 .....	17
4.3 列管資料監控機制說明 .....	17
5. 風險評估 .....	18
5.1 風險項目評估 .....	18
5.2 風險監控機制說明 (此項目為必要監控項目) .....	18
6. 建構管理計畫 .....	19
6.1 目的 .....	19
6.2 建立基準 .....	19
6.2.1 標示建構管理項目 .....	19

6.2.2運用建立建構管理系統.....	19
6.2.3建立基準.....	19
6.2.4版本控制程式.....	19
6.3異動追蹤與控制.....	19
6.3.1異動追蹤.....	19
6.3.2建構控制小組.....	19
6.3.3異動控制.....	20
6.4達成完整性.....	20
6.4.1建構管理記錄.....	20
6.4.2建構審核.....	20
7. 度量與分析計畫.....	21
7.1目的.....	21
7.2蒐集資訊的目的與資訊需求.....	21
7.3基礎度量.....	21
7.4度量與分析工具.....	21
8. 流程與產品品質保證計畫.....	22
8.1目的.....	22
8.2客觀檢視流程與產品.....	22
8.3 專案目標洞察.....	22
8.4管理架構.....	22

# 版次變更記錄

修訂			
----	--	--	--

版本	編輯作者	修改內容	日期
1.0	劉柏宏、陳巧宜 洪子軒、鄭鴻仁	初期規劃，規劃系統、系統開發時間、工作分配、	2018/10/07

# 1. 專案規劃及查核點

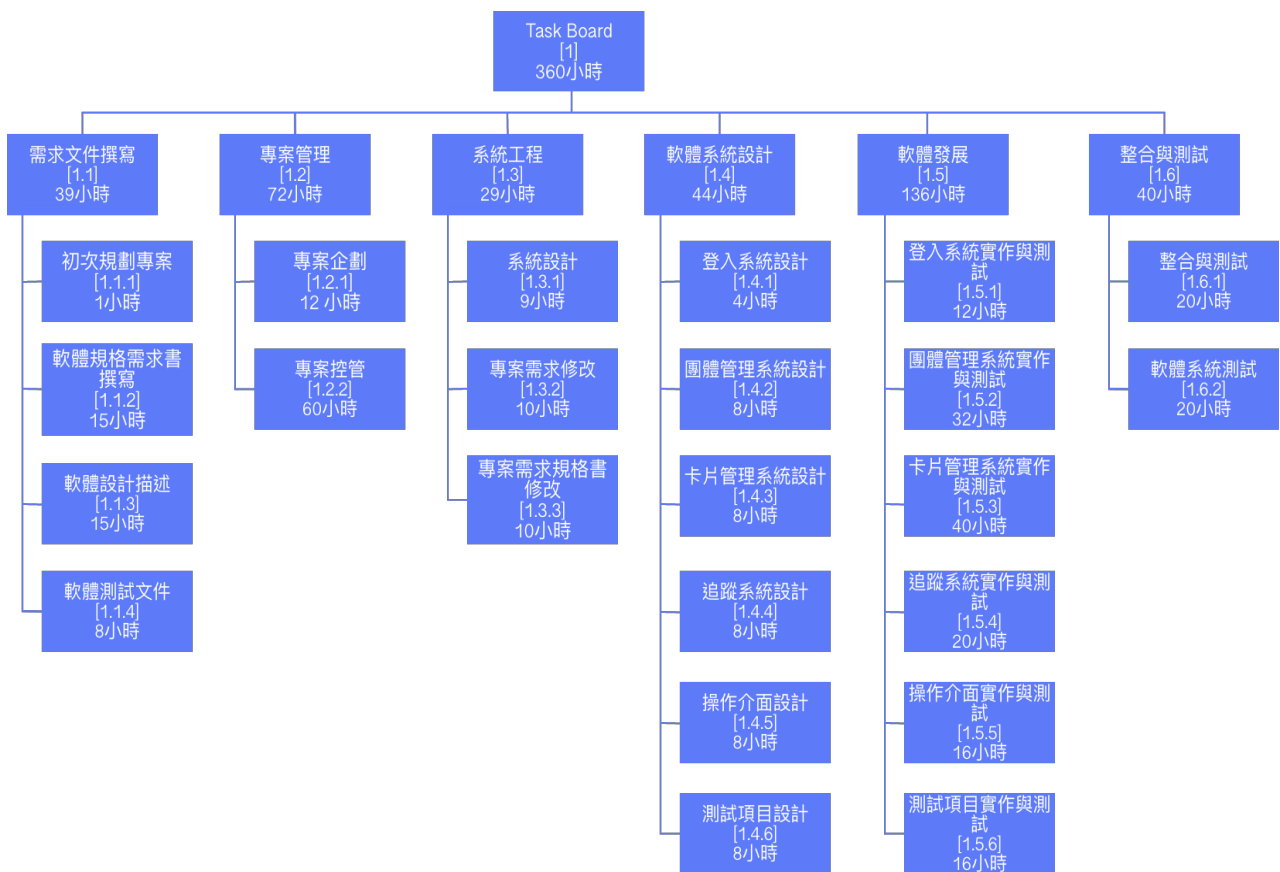
## 1.1 專案工作內容

### 1.1.1 技術方法

本系統採用Web Application 設計方法，讓使用者可以透過此網頁，計畫並且追蹤團隊的進度，可以記錄開發或是工作的任務，使用waterfall process model 跟 incremental的結合，增加系統開發的效率！

系統開發上，網頁頁面使用Vue.js框架，串連後端Nodejs server，資料庫使用mysql，前端的部分透過ajax技術，達到豐富使用者體驗，使用者不用一直重載頁面，另外版本控制使用git，分開前端開發跟後端開發，以方便後面系統整合。

### 1.1.2 分工結構圖



### 1.1.3 工作分包與工作項目估算模型與方法

我們採用 Poker 方式進行估算，彼此開發團隊每個禮拜固定開會議討論工作需求，並且 確認此需求如何去實踐，接著彼此會有一副 Poker，但上面只有點數：0、0.5、1、2、4、5、10、20、40，點數是以近似費氏數列成長，用意在於當估算點數越大時，其實際估 算誤差可能也會越大，因此不宜拘泥在小點數的爭執；當點數準備妥善時，彼此選出認為最合適的點數，此時點數若無落差即可確認工作點數，但當點數有落差呈現時，必須以最高以及最低點數者負責說明原因，之後組員再次重新分析剛剛評估的理由後再重新 選出點數，直到彼此間點數誤差到正負一即可。

#### 1.1.4工作分包與工作項目總表

	任務名稱	工期	開始時間	完成時間	工時
1	1 T.B.系統	45工作日	107/10/03	108/01/02	360小時
2	1.1 需求文件專寫	5工作日	107/10/03	107/10/08	39小時
3	1.1.1 初次規劃專案	0.1工作日	107/10/03	107/10/03	1小時
4	1.1.2 軟體規格需求書撰寫	2工作日	107/10/04	107/10/06	15小時
5	1.1.3 軟體設計描述	2工作日	107/10/05	107/10/07	16小時
6	1.1.4 軟體測試文件	1工作日	107/10/07	107/10/08	8小時
7	1.2 專案管理	9工作日	107/10/09	107/10/18	72小時
8	1.2.1 專案企劃	2工作日	107/10/18	107/10/20	16小時
9	1.2.2 專案控管	8工作日	107/10/20	107/10/28	64小時
10	1.3 系統工程	4工作日	107/10/28	107/11/02	32小時
11	1.3.1 系統設計	1工作日	107/11/02	107/11/03	8小時
12	1.3.2 專案需求修改	1工作日	107/11/03	107/11/04	8小時
13	1.3.3 專案需求與規格書修	2工作日	107/11/04	107/11/06	16小時
14	1.4 軟體系統設計	6工作日	107/11/06	107/11/12	48小時
15	1.4.1 登入系統設計	1工作日	107/11/12	107/11/13	8小時
16	1.4.2 團體管理系統設計	1工作日	107/11/13	107/11/14	8小時
17	1.4.3 卡片管理系統設計	1工作日	107/11/14	107/11/15	8小時
18	1.4.4 追蹤系統設計	1工作日	107/11/15	107/11/16	8小時
19	1.4.5 操作介面設計	1工作日	107/11/16	107/11/17	8小時
20	1.4.6 測試項目設計	1工作日	107/11/17	107/11/18	8小時
21	1.5 軟體發展	17工作日	107/11/18	107/12/05	136小時
22	1.5.1 登入系統實作與測試	2工作日	107/12/05	107/12/07	16小時
23	1.5.2 團體管理系統實作	4工作日	107/12/07	107/12/11	32小時
24	1.5.3 卡片管理系統	5工作日	107/12/11	107/12/16	40小時
25	1.5.4 追蹤系統	3工作日	107/12/16	107/12/19	24小時
26	1.5.5 操作介面與實作測試	2工作日	107/12/19	107/12/21	16小時
27	1.5.6 測試項目實作與測試	2工作日	107/12/21	107/12/23	16小時
28	1.6 整合與測試	5工作日	107/12/23	107/12/28	40小時
29	1.6.1 整合與測試	3工作日	107/12/28	107/12/31	24小時
30	1.6.2 軟體系統測試	2工作日	107/12/31	108/01/02	16小時

### 1.1.5工作分包與工作項目內容說明

1.1 需求文件撰寫		
工作項目/編號名稱		1.1.1 初次規劃專案
工作內容說明		撰寫系統企劃文件
工作產品/格式		PEP/文件
其他工作相依性		
需求技能與知識		專案分析、管理、軟體架構、軟體工程
工作時程		1小時
資源	人力	4
	硬體設備	PC
	軟體工具	Office

1.1 需求文件撰寫		
工作項目/編號名稱		1.1.2 軟體規格需求書撰寫
工作內容說明		撰寫軟體規格書
工作產品/格式		SRS/文件
其他工作相依性		
需求技能與知識		專案分析、管理、軟體架構、軟體工程。
工作時程		15小時
資源	人力	4
	硬體設備	PC
	軟體工具	Office

1.1 需求文件撰寫		
工作項目/編號名稱		1.1.3 軟體設計描述
工作內容說明		描述軟體架構
工作產品/格式		SDD/文件
其他工作相依性		
需求技能與知識		軟體架構
工作時程		15小時
資源	人力	4
	硬體設備	PC
	軟體工具	Office

1.1 需求文件撰寫		
工作項目/編號名稱		1.1.4 軟體測試文件
工作內容說明		撰寫測試文件
工作產品/格式		STD/文件
其他工作相依性		
需求技能與知識		軟體測試
工作時程		8小時
資源	人力	4
	硬體設備	PC
	軟體工具	Office

1.2 專案管理		
工作項目/編號名稱	1.2.1 專案企劃	
工作內容說明	撰寫規劃文件	
工作產品/格式	PEP/文件	
其他工作相依性		
需求技能與知識	軟體工程	
工作時程	12小時	
資源	人力	4
	硬體設備	PC
	軟體工具	Office

1.2 專案管理		
工作項目/編號名稱	1.2.2 專案控管	
工作內容說明	專案版本控制	
工作產品/格式	Git/code	
其他工作相依性		
需求技能與知識	版本控制	
工作時程	60小時	
資源	人力	4
	硬體設備	PC server
	軟體工具	Git

1.3 系統工程		
工作項目/編號名稱	1.3.1 系統設計	
工作內容說明	系統設計	
工作產品/格式	系統架構圖	
其他工作相依性		
需求技能與知識	系統架構	
工作時程	9小時	
資源	人力	4
	硬體設備	PC
	軟體工具	Office

1.3 系統工程		
工作項目/編號名稱	1.3.2 專案需求修改	
工作內容說明	修改軟體需求	
工作產品/格式	PEP/文件	
其他工作相依性		
需求技能與知識	軟體工程	
工作時程	9小時	
資源	人力	4
	硬體設備	PC
	軟體工具	Office



1.3 系統工程			1.4 軟體系統設計		
工作項目/編號名稱		1.3.3 專案需求規格書修改	工作項目/編號名稱		1.4.1 登入系統設計
工作內容說明		修改軟體需求規格書	工作內容說明		1.登入系統 2.註冊系統 3.登出系統
工作產品/格式		PEP/文件	工作產品/格式		code
其他工作相依性			其他工作相依性		
需求技能與知識		軟體工程	需求技能與知識		軟體工程, coding
工作時程		8小時	工作時程		4小時
資源	人力	4	資源	人力	4
	硬體設備	PC		硬體設備	PC
	軟體工具	Office		軟體工具	Node js

1.4 軟體系統設計			1.4 軟體系統設計		
工作項目/編號名稱		1.4.2 團體系統設計	工作項目/編號名稱		1.4.3 卡片管理系統設計
工作內容說明		團體CRUD	工作內容說明		卡片CRUD
工作產品/格式		code	工作產品/格式		Code
其他工作相依性			其他工作相依性		
需求技能與知識		軟體工程, coding	需求技能與知識		軟體工程, coding
工作時程		8小時	工作時程		8小時
資源	人力	4	資源	人力	4
	硬體設備	PC		硬體設備	PC
	軟體工具	Node js		軟體工具	Node js

1.4 軟體系統設計			1.4 軟體系統設計		
工作項目/編號名稱		1.4.4 追蹤系統設計	工作項目/編號名稱		1.4.5 操作介面設計
工作內容說明		追蹤系統	工作內容說明		使用者介面
工作產品/格式		Code	工作產品/格式		Code
其他工作相依性			其他工作相依性		
需求技能與知識		軟體工程, coding	需求技能與知識		軟體工程, coding
工作時程		8小時	工作時程		8小時
資源	人力	4	資源	人力	4
	硬體設備	PC		硬體設備	PC
	軟體工具	Node js		軟體工具	Node js

1.4 軟體系統設計			1.5 軟體發展		
工作項目/編號名稱		1.4.6 測試項目設計	工作項目/編號名稱		1.5.1 登入系統發展
工作內容說明		測試項目	工作內容說明		登入系統擴充
工作產品/格式		Code	工作產品/格式		Code
其他工作相依性			其他工作相依性		
需求技能與知識		軟體工程, coding	需求技能與知識		軟體工程, coding
工作時程		8小時	工作時程		12小時
資源	人力	4	資源	人力	4
	硬體設備	PC		硬體設備	PC
	軟體工具	Node js		軟體工具	Node js

1.5 軟體發展			1.5 軟體發展		
工作項目/編號名稱		1.5.2 團體管理系統發展	工作項目/編號名稱		1.5.3 卡片管理系統發展
工作內容說明		團體管理系統擴充	工作內容說明		卡片管理系統擴充
工作產品/格式		Code	工作產品/格式		Code
其他工作相依性			其他工作相依性		
需求技能與知識		軟體工程, coding	需求技能與知識		軟體工程, coding
工作時程		32小時	工作時程		40小時
資源	人力	4	資源	人力	4
	硬體設備	PC		硬體設備	PC
	軟體工具	Node js		軟體工具	Node js

1.5 軟體發展			1.5 軟體發展		
工作項目/編號名稱		1.5.4 追蹤系統發展	工作項目/編號名稱		1.5.5 操作介面系統發展
工作內容說明		追蹤系統擴充	工作內容說明		操作介面系統擴充
工作產品/格式		Code	工作產品/格式		Code
其他工作相依性			其他工作相依性		
需求技能與知識		軟體工程, coding	需求技能與知識		軟體工程, coding
工作時程		20小時	工作時程		16小時
資源	人力	4	資源	人力	4
	硬體設備	PC		硬體設備	PC
	軟體工具	Node js		軟體工具	Node js

1.5 軟體發展			1.6 整合與測試		
工作項目/編號名稱		1.5.6 測試項目系統發展	工作項目/編號名稱		1.6.1 整合與測試
工作內容說明		測試項目系統擴充	工作內容說明		整合與測試
工作產品/格式		Code	工作產品/格式		Code
其他工作相依性			其他工作相依性		
需求技能與知識		軟體工程, coding	需求技能與知識		軟體工程, coding
工作時程		16小時	工作時程		20小時
資源	人力	4	資源	人力	4
	硬體設備	PC		硬體設備	PC
	軟體工具	Node js		軟體工具	Node js

1.6 整合與測試		
工作項目/編號名稱		1.6.2 系統軟體測試
工作內容說明		確定系統符合需求
工作產品/格式		Code
其他工作相依性		
需求技能與知識		軟體工程, coding
工作時程		20小時
資源	人力	4
	硬體設備	PC
	軟體工具	Node js

### 1.1.6專案生命週期定義

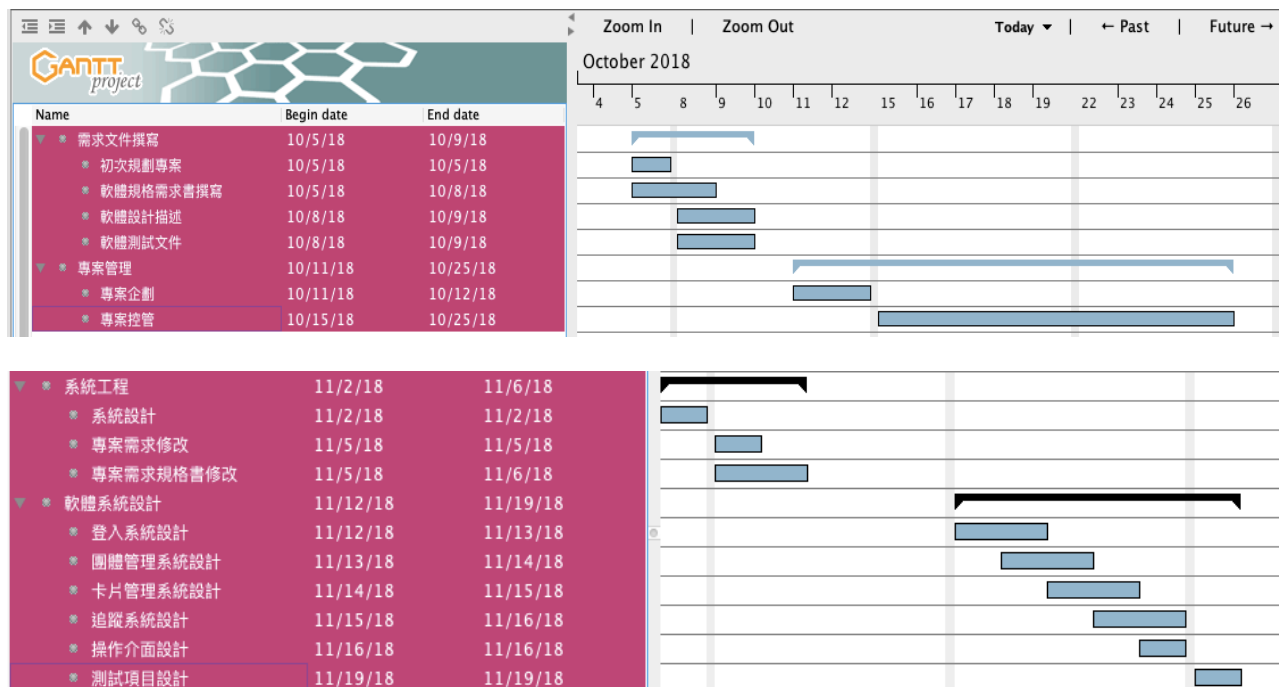
本系統一開始從需求分析時，團員們必須針對各個需求要有徹底的認知，之後再事先設計系統雛型，並在分析過程中探討如何實做的問題，接著才開始進入軟體設計以及軟體實做的階段。在實做過程中會一邊撰寫測請文件，並在軟體開發快結束的階段，重新修改執行計畫文件以及軟體需求規格書，再接續著軟體開發設計以及實做，因此就專案執行順序來看會是：文件→需求→設計→實做→文件→設計→實做(整合)→全部整合→釋出 所以，我們自定了結合瀑布(Waterfall)與迭代的生命週期模式。

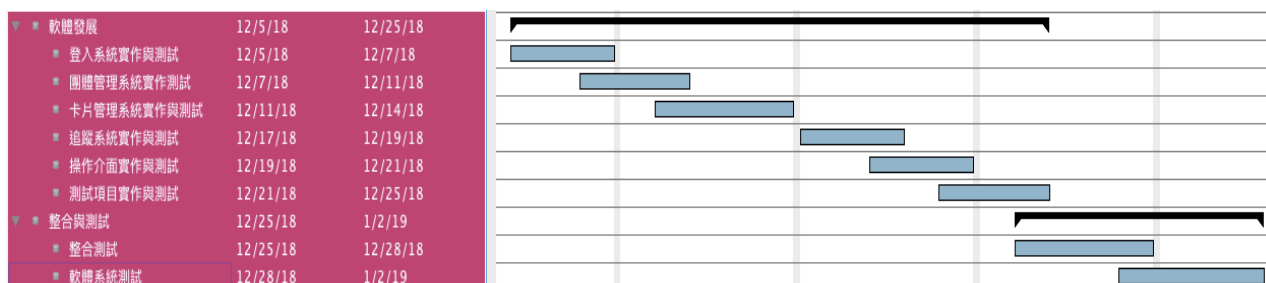
## 1.2預定時程及查核點

### 1.2.1預訂查核點說明

查核點	預定時間	查核點敘述	技術文件/產出物
M1		完成PEP與SRS	專案執行規劃書 系統需求規格書
M2		完成軟體設計	系統規劃設計書
M3		完成軟體發展	各系統的程式碼
M4		完成整合測試	系統的釋出

### 1.2.2預定時程 (Schedule)





### 1.2.3時程與進度審查監控機制說明

本專案對於進行中的工作採用每兩個禮拜監控一次，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。而需要實施矯正措施時，其基準為

- (1) 107/10~107/11矯正基準定為20%，當進度落後超過20%必預實施矯正措施。
- (2) 107/11~107/12矯正基準定為15%，當進度落後超過15%必預實施矯正措施。
- (3) 107/12~108/01矯正基準定為10%，當進度落後超過10%必預實施矯正措施。

矯正措施為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）

#### (1)系統規格分析完成

監控時間	矯正基準	矯正機制
M1	SRS與PEP完成	完成PEP與SRS
M2	SDD完成	完成軟體設計

#### (2)軟體開發完成

監控時間	矯正基準	矯正機制
M1	完成所有子系統開發	若有未完成部分，成員協力完成，排除困難

## 2. 專案成員工作指派

### 2.1 工作項目或工作分包預估需求與估算之假設條件

#### ◇ 工作分包預估方式

- 歷史資料法
- 專家法(透過個人專業判斷，進行估算)
- 其他估算法

#### ◇ 參數

1. 文件: 1頁 / 1人時
2. 系統功能: 1個 / 8人時
3. 假設條件: 以人事行政局公佈的年度上班時間為工作日
4. 一日工作時數為8小時(加班視專案及課業程度而自行調整)

#### ◇ 專業技能需求

專業技能	需求人數
需求文件撰寫	4
專案管理	4
系統工程	4
軟體設計	4
軟體發展	4
整合與測試	4

### 2.2 計畫成員指派

姓名	縮寫
劉柏宏	BH
鄭鴻仁	HJ
陳巧宜	JT
洪子軒	ZX

專案人員需求規格表			
WBS	活動與交付項目	負責人員	所需的知識與技能
1.1.1	初次規劃專案	BH, HJ, JT, ZX	專案分析、軟體架
1.1.2	軟體規格需求書撰寫	BH, HJ, JT, ZX	專案分析、管理、軟
1.1.3	軟體設計描述	BH, HJ, JT, ZX	專案分析、軟體架
1.1.4	軟體測試文件	BH, HJ, JT, ZX	軟體架構、軟體測
1.2.1	專案企劃	BH, HJ, JT, ZX	專案分析、管理、軟
1.2.2	專案控管	BH, HJ, JT, ZX	程式設計
1.3.1	系統設計	BH, HJ, JT, ZX	軟體設計、軟體架
1.3.2	專案需求修改	BH, HJ, JT, ZX	專案分析、管理、軟
1.3.3	專案需求規格書修改	BH, HJ, JT, ZX	專案分析、管理、軟
1.4.1	登入系統設計	BH, HJ, JT, ZX	軟體架構、軟體設
1.4.2	團體管理系統設計	BH, HJ, JT, ZX	軟體架構、軟體設
1.4.3	卡片管理系統設計	BH, HJ, JT, ZX	軟體架構、軟體設
1.4.4	追蹤系統設計	BH, HJ, JT, ZX	軟體架構、軟體設
1.4.5	操作介面設計	BH, HJ, JT, ZX	軟體架構、軟體設
1.4.6	測試項目設計	BH, HJ, JT, ZX	軟體架構、軟體設
1.5.1	登入系統實作與測試	BH, HJ, JT, ZX	軟體架構、軟體設
1.5.2	團體管理系統實作與測試	BH, HJ, JT, ZX	軟體架構、軟體設
1.5.3	卡片管理系統實作與測試	BH, HJ, JT, ZX	軟體架構、軟體設
1.5.4	追蹤系統實作與測試	BH, HJ, JT, ZX	軟體架構、軟體設
1.5.5	操作介面實作與測試	BH, HJ, JT, ZX	軟體架構、軟體設
1.5.6	測試項目實作與測試	BH, HJ, JT, ZX	軟體架構、軟體設
1.6.1	整合與測試	BH, HJ, JT, ZX	JavaScript
1.6.2	軟體系統測試	BH, HJ, JT, ZX	JavaScript

## 2.3調整專案成員

人員基本上不會有調度問題，本團隊將會盡可能發揮各自專長來完成此系統。



## 2.4 專案專業知識與技能需求

專業技能及知識	預估需要人數	預計受訓人員	說明
HTML	4	0	本專案的成員已有相關經驗
Javascript	4	0	本專案的成員已有相關經驗
CSS	4	0	本專案的成員已有相關經驗
DBMS	4	0	本專案的成員已有相關經驗
Ajax	4	0	本專案的成員已有相關經驗
UI design	4	0	本專案的成員已有相關經驗

## 2.5 訓練計畫表

除了使用者頁面設計概念比較不著重於技術層面，因此只能靠成員個人審美觀念去設計，而其餘技術方面的技術，團員間均有相當的開發經驗，因此只需要在實做時一邊開發一邊發掘問題並討論即可，不需要特別訓練。

## 2.6 成員參與情況監控機制說明

專案進行中，每個開發階段子系統的子任務皆會做一次監控確認，以瞭解進度是否有按照規劃日期運行。當執行結果若有延後，將採取篩選需求的作法，將之後要實做系統的需求以重要性來取捨，藉此來趕上下一個篩選日期所期盼的結果；當執行結果有超前，將採取回顧需求，把以往因為重要性關係而被忽略的任務，再次加回來實做。

### 3. 資源需求

#### 3.1 計畫經費預算說明

工作編號	經費描述	設備費用	管理費用	其他費用	合計
1.1.1	初次規劃專案	21000	500		21500
1.1.2	軟體規格需求書撰寫		500		500
1.1.3	軟體設計描述		500		500
1.1.4	軟體測試文件		500	500	1000
1.2.1	專案企劃		500		500
1.2.2	專案控管		500	500	1000
1.3.1	系統設計		500		500
1.3.2	專案需求修改		500		500
1.3.3	專案需求規格書修改		500	500	1000
1.4.1	登入系統設計		500		500
1.4.2	團體管理系統設計		500		500
1.4.3	卡片管理系統設計		500		500
1.4.4	追蹤系統設計		500		500
1.4.5	操作介面設計		500		500
1.4.6	測試項目設計		500		500
1.5.1	登入系統實作與測試	5000	2000		7000
1.5.2	團體管理系統實作與	5000	2000		7000
1.5.3	卡片管理系統實作與	5000	2000		7000
1.5.4	追蹤系統實作與測試	5000	2000		7000
1.5.5	操作介面實作與測試	5000	2000		7000
1.5.6	測試項目實作與測試	5000	2000		7000
1.6.1	整合與測試		500	500	1000
1.6.2	軟體系統測試		500	500	1000
合計		51000	20500	2500	74000

### 3.2人事費用估算

工作計畫需求人力：人時		總人事費用：4人	
職級	單位(時)	人事費概算	備註
萬能研究生工作員(4人)	492小時	492000	萬能比較貴
其他(加班費)		0	

### 3.3計畫經費預估表

經費項目	預定金額	說明
研究設備費	51000	個人電腦以及筆記型電腦
業務費	2500	一般業務或特殊業務之用，例如文件的產生所使用的資源費用
人事費	492000	專案人員共四名之研究經費
管理費	20500	專案管理及其他研究經費
合計	566000	

### 3.4預算監控機制說明

本專案有關於預算的監控機制為：

- 一. 監控頻率：每天監控一次。
- 二. 實施矯正之基準及其措施：

預算使用超過10%即必預實施矯正措施。矯正措施為開會決定如何取得資金，或是刪減專案活動。

## 4. 資料管理規劃

### 4.1 資料管理計畫

本計畫資料管理與儲存方式分為三種：

- 一. 原始程式碼：程式碼使用github負責版本控制與管理。
- 二. 電子文件及可執行檔：由臺北科技大學開放式網路教室儲存和管理。
- 三. 紙本文件或光碟資料：由Master負責管理。

### 4.2 列管資料總表

資料名稱	版控	建構管理	機密等級	產生週期	儲存方式	資料提供者	資料使用者
專案執行規劃書	否	否	密	Event	Google Drive	團隊	團隊
系統需求規格書	否	否	密	Event	Google Drive	團隊	團隊、測試者
系統設計規格書	否	否	密	Event	Google Drive	團隊	團隊
原始程式碼	是	是	密	Daily	Google Drive	團隊	團隊
整合測試計畫書	否	否	密	Event	Google Drive	團隊	團隊、測試者
系統測試報告	否	否	密	Event	Google Drive	團隊、測試者	團隊、測試者
系統接受度報告	否	否	密	Event	Google Drive	測試、使用者	測試、使用者

### 4.3 列管資料監控機制說明

◇ 監控頻率：每月監控一次。  
實施矯正之基準及其措施：

資料管理所列管的所有資料都必須按照資料管理計畫的方式進行，如果發現任何的資料未按資料管理計畫保管或備份，都必須立刻進行矯正，矯正措施為立即增補。

## 5. 風險評估

### 5.1 風險項目評估

風險項目	發生可能性	影響程度	風險發生處理或避免方法
人事變動	15%	低	固定時間討論，了解組員狀況
版本衝突	10%	中	利用Git進行版控，記錄所有變更
人員的訓練不足	10%	中	平日多充實自我能力、組員間互相幫忙
資料庫需求變更	50%	高	降低程式的相依度
資料庫毀損	10%	低	定期備份資料庫
版控伺服器毀損	5%	低	各組員都保留一份紀錄，協助後續修復

### 5.2 風險監控機制說明 (此項目為必要監控項目)

#### ◇ 資料庫需求變更—

解決方法：盡可能讓程式間相依性變低，當需求異動或需求變更，只需要修改少數資料表的可能，或者新增資料表但不影響其它資料表的運作。

#### ◇ 資料庫伺服器毀損—

解決方法：每週固定備份資料庫資料，以及當需求有重大變更時也要做一次資料庫備份。

#### ◇ 版控伺服器毀損—

解決辦法：使用Git進行版控，各組員都會有一份完整的Local Repository，在伺服器上重新建立新的Repository，並把本地的Repository重新上傳。

## 6. 建構管理計畫

### 6.1 目的

本專案系統開發為釋出讓使用者去使用，因此可能需要有長時間維護的工作，或是當新的需求被提出要加入時，可能會有開發的工作，而一個良好的建構管理，即可在一邊開發的同時也可以針對早期版本發現的問題做個別修改，因此才需要此計畫。

### 6.2 建立基準

#### 6.2.1 標示建構管理項目

ID	資料名稱	版本	建構類別	產生週	資料提供者	資料使用者
1	專案執行規劃書	否	規格書	Event	團隊	團隊
2	系統需求規格書	否	規格書	Event	團隊	團隊、使用者
3	系統設計規格書	否	規格書	Event	團隊	團隊
4	整合測試計畫書	否	規格書	Event	團隊	團隊、測試者
5	原始程式碼	是	原始碼	Monthly	團隊	團隊
6	系統測試報告	否	報告資料	Event	團隊、測試者	團隊、測試者
7	系統接受度報告	否	報告資料	Event	測試、使用者	測試、使用者

#### 6.2.2 運用建立建構管理系統

本專案系統採用Github進行版本管理，使團隊能夠同時開發。

#### 6.2.3 建立基準

由表 6.2.1 得知，沒有版本控管的書面資料，是為隨時跟著系統的開發更新，但企劃書以及一些測請報告書確認後將不會異動，因此皆不需要版本控制，唯獨原始程式碼必須要依靠Git，來達到 6.1 所提及的目的。

#### 6.2.4 版本控制程式

Git is a free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency.

資料來源: <https://git-scm.com/>

## 6.3 異動追蹤與控制

### 6.3.1 異動追蹤

1. 提出異動申請(異動申請單)。
2. 由建構管理人員以及 Master 評估影響層面，並通知 Member。
3. 由 Master 邀集受影響單位進行評估，並決定是否准予異動。
4. 追蹤異動的狀態(例如異動時間)。

### 6.3.2 建構控制小組

由組員自行監控。

### 6.3.3異動控制

- 1.對於異動的項目對該版本提出一個分支（branch）。
- 2.再次確認其正確性。
- 3.確認異動後，必須對此次做異動紀錄以及異動原因。

## 6.4達成完整性

### 6.4.1建構管理記錄

此管理記錄為建立與維護用來描述建構管理項目的紀錄。而紀錄項目以及資料僅採用Git本身紀錄的欄位。

### 6.4.2建構審核

未來每開發一個功能都必須獨立出feature分支，功能完成時才合併回develop分支，而有bug要修復時也必須另外分出一個hotfix分支去處理，當子系統開發到一定程度後在必要時可以合併到release，直到系統穩定後可以推到master分支，並增加一個小版號。

## 7. 度量與分析計畫

### 7.1 目的

度量分析主要在蒐集本專案的各項資訊，以提供各種分析之用。

### 7.2 蒐集資訊的目的與資訊需求

序號	目的	資訊需求
1	客戶滿意度	客戶的反應、支援客戶的狀況
2	時程與進度	需求完成狀況、工作單元進度
3	資源與成本	支出、各項資源支援的程度
4	產品品質	系統或功能品質、使用者介面的良劣
5	客戶需求的穩定程度	客戶需求的異動
6	產品大小	每個子系統的大小、功能多寡

### 7.3 基礎度量

序號	度量	因子
1	客戶滿意度度量	經由客戶問題的反應
2	需求完成狀況、工作單元進度	需求完成的時間、階層中工作單元完成度
3	支出、各項資源支援的程度	專案人員投入的工作時數、實際支出數
4	系統或功能品質、使用者介面的良劣	系統或功能之錯誤數、使用者反應介面問
5	客戶需求的異動	客戶需求異動個數、個數、無法修改個數
6	每個子系統的大小、功能多寡	每個子系統的程式行數、功能數

### 7.4 度量與分析工具

本專案使用自行記錄之資料作為分析上述各項結果。



## 8. 流程與產品品質保證計劃

### 8.1 目的

本系統的流程與產品品質保證計畫主要是提供專案所有人員對於產品的品質能深入瞭解其流程，亦可當作系統接受度測誦的檢視，以告知使用者關於本系統的品質趨向。

### 8.2 客觀檢視流程與產品

步驟如下：

- 1.Team Master與Product Owner共同檢視系統執行流程，將每個頁面運作過程依照使用者手冊以及系統規格書去複查。
- 2.確認需求皆為 Product Owner 所期望的。
- 3.檢視四份文件(PEP、SRS、SDD 及 STD)，一邊展示的同時一邊複查。

### 8.3 專案目標洞察

- 1.Team Master 必預與 Product Owner一邊討論一邊確認需求，以讓自己本身更瞭解 Owner想要的結果。
- 2.Team Master 將個人對於需求的認知詳細告知於Team Member，而團員若是有任何不確定的疑問均可向 Master詢問。
- 3.將需求紀錄於ezScrum系統上，並且每個sprint demo均可視察任務的執行結果。

### 8.4 管理架構

本系統由劉柏宏為 Master，其餘洪子軒、鄭鴻仁、陳巧宜為 Member，我們四個本身也是 Owner 的一部份，當系統實做過程中，覺得系統有任何可以改善的地方，均可提出作為一個系統需求來做評估。專案執行上採用 Scrum 執行精神，於每個 Sprint 工作階段，皆是團員們互相分工合作，並不需要特別指派工作分工細項。