

# **Project Name**

## **Programming Assignment Submission System (PASS)**

**Version: 1.0**

**Team#1**

<b>Name</b>	<b>ID</b>	<b>E-mail</b>
楊雅雯	105598069	s7130457@gmail.com
劉彥麟	105598003	cubemail88@gmail.com
陳韻文	106598053	ingridchen82@gmail.com
李柏霖	106598038	morris12345678987654321@gmail.com
郭士銓	106598048	chenny2169@gmail.com

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

**10/02/2017**

## 目錄 (Table of Contents)

版次變更記錄 (Change Log).....	1
專案規劃及查核點 (Project Planning and Milestone Checking).....	2
1.1 專案工作內容 (Project Work Description).....	2
1.1.1 技術方法(Technical Approach).....	2
1.1.2 分工結構圖 (Work Breakdown Structure).....	2
1.1.3 工作分包與工作項目估算模型與方法 (Establish Estimates of Project Attributes) .....	3
1.1.4 工作分包與工作項目總表 (List of Work Packages and Tasks).....	3
1.1.5 工作分包與工作項目內容說明 (Descriptions of Work Packages and Tasks).....	5
1.1.6 專案生命週期定義 (Project Life Cycle).....	18
1.2 預定時程及查核點 (Schedule and Milestone Checking).....	19
1.2.2 預定時程 (Schedule).....	20
1.2.3 時程與進度審查監控機制說明 (Schedule & Progress Monitor and Control Mechanism).....	21
Section2 專案成員工作指派 (Personnel).....	21
2.1 工作項目或工作分包預估需求與估算之假設條件 (Task Estimation Assumptions)	21
2.2 計畫成員指派 (Roles and Responsibilities).....	21
Section3 資源需求 (Resources).....	24
3.1 計畫經費預算說明 (Budget).....	24
3.2 人事費用估算 (Estimations of Personnel Fee).....	25
3.3 計畫經費預估表 (Project Cost Estimation).....	25
3.4 預算監控機制說明 (Budget Monitor and Control Mechanism).....	25
Section4 資料管理規劃 (Data Management Plan).....	25
4.1 資料管理計畫 (Data Management Plan).....	25
4.2 列管資料總表 (List of Managed Data).....	26
4.3 列管資料監控機制說明 (Monitor and Control Mechanism) (此項目為必要監控項目) 26	
5.1 風險項目評估 (Risks Assessment).....	26
5.2 風險監控機制說明 (Risk Monitor and Control Mechanism) (此項目為必要監控項目) 27	
Section 6 建構管理計畫 (Configuration Management Plan).....	27
6.1 目的 (Purpose).....	27
6.2 建立基準 (Establish Baselines).....	27
6.2.1 標示建構管理項目(Identify Configuration Items).....	27
6.2.2 運用建立建構管理系統 (Establish a CM System).....	27
6.2.3 建立基準 (Create or Release Baselines).....	27

<b>6.3 異動追蹤與控制 (Track and Control Changes)</b> .....	27
6.3.1 異動追蹤 (Track Change) .....	27
6.3.2 建構控制小組 (Configuration Control Board) .....	28
6.3.3 異動控制 (Control Change) .....	28
6.3.4 版本控制程式 (The Version Control Tool) .....	28
<b>6.4 達成完整性 (Establish Integrity)</b> .....	28
6.4.1 建構管理記錄 (Establish Configuration Management Records).....	28
6.4.2 建構審核 (Perform Configuration Audits).....	28
<b>7.1 目的 (Purpose)</b> .....	28
<b>7.2 蒐集資訊的目的與資訊需求 (Information Needs and Objectives)</b> .....	28
<b>7.3 基礎度量 (Base Measurement)</b> .....	29
<b>7.4 度量與分析工具 (Measurement and Analysis Tool)</b> .....	29
<b>Section8 流程與產品品質保證計劃 (PPQA Plan)</b> .....	<b>29</b>
<b>8.1 目的 (Purpose)</b> .....	29
<b>8.2 客觀檢視流程與產品 (Objectively Evaluate Process and Work Packages)</b> .....	29
<b>8.3 專案目標洞察 (Project Objective Insight)</b> .....	29
<b>8.4 管理架構 (Management Architecture)</b> .....	29

## 版次變更記錄 (Change Log)

### Revisions

Version	Primary Author(s)	Description of Version	Date Completed
1.0	劉彥麟 楊雅雯 郭士銓 陳韻文 李柏霖	專案初期規劃	10/02/2017

## 專案規劃及查核點 (Project Planning and Milestone Checking)

### 1.1 專案工作內容 (Project Work Description)

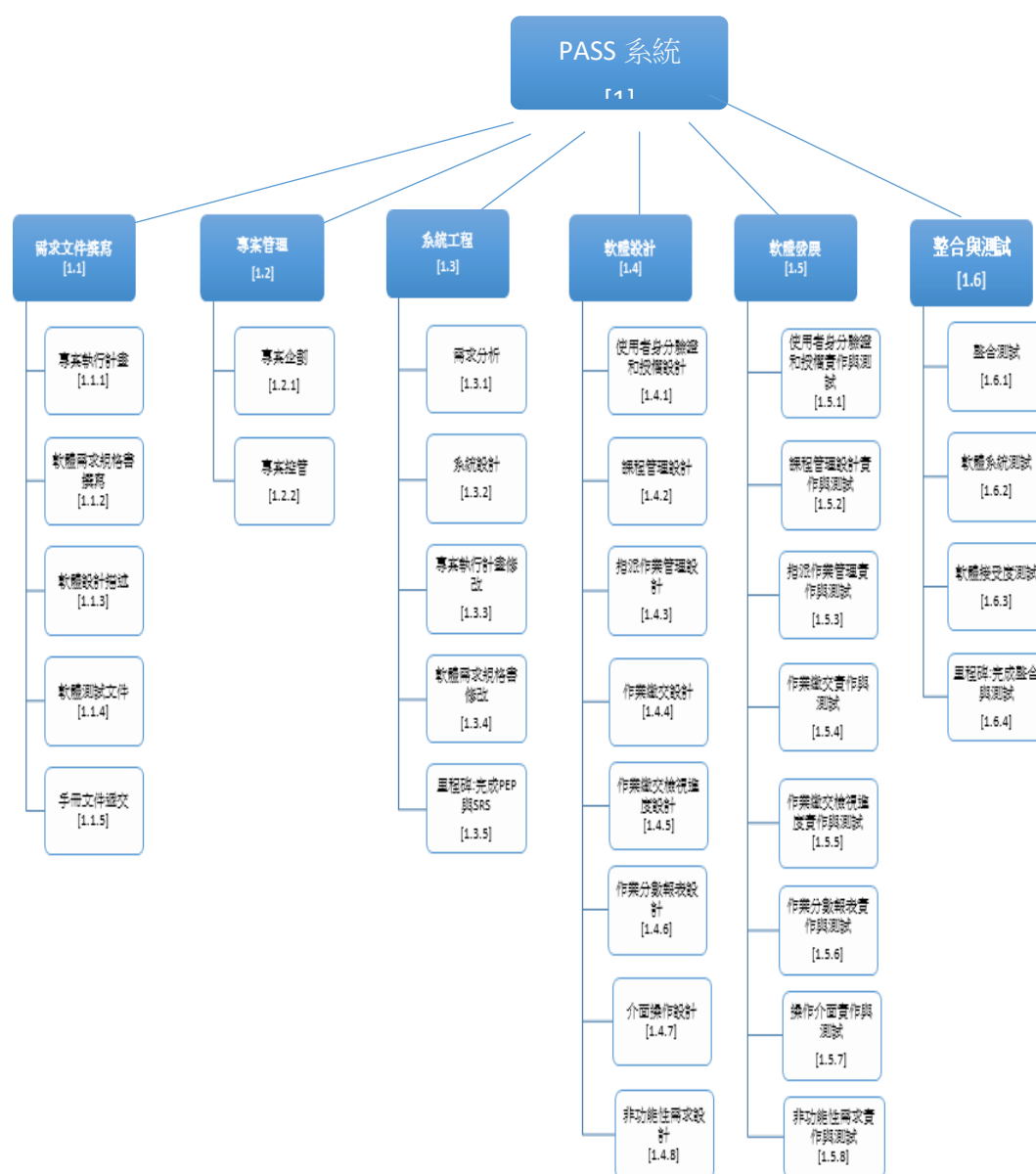
#### 1.1.1 技術方法(Technical Approach)

本系統採用 Web MVC 設計方法，讓使用者和管理者可以直覺的方法使用此作業繳交系統。此系統是基於 Frame (<https://jedireza.github.io/frame/>) 下去開發，Frame 是 Node.js 的一個使用者系統 API，此 api 提供了：『註冊、登入、使用者和角色、管理者群組、權限』，因此我們基於這個 user api 開始向上開發我們的 PASS 系統，以下為將會使用到技術

##### 1. 資料庫 MongoDB：

MongoDB 是一個高性能，開源，無模式的文檔型資料庫，是當前 NoSql 資料庫中比較熱門的一種。它在許多場景下可用於替代傳統的關聯式資料庫或鍵/值存儲方式。

#### 1.1.2 分工結構圖 (Work Breakdown Structure)



### 1.1.3 工作分包與工作項目估算模型與方法 (Establish Estimates of Project Attributes)

Ans:我們將採用 Scrum Poker 來進行估算。

估算描述:

開發團隊每幾個禮拜開會決定這次週期要做哪些需求(Story)，並針對每個 Story 細切成數個小塊的 task，接著發給每位成員一副 Poker，上面的點數有：0、1、2、3、5、8、13、20、40，點數是以費式數列遞增，越大的數值代表實際的誤差也越大。團隊成員彼此選擇一張自己認為適合的點數進行估算，若彼此點數有落差時，則由點數最高和最低的成員解釋出此點數的考量，藉由互相討論來降低彼此對工作需求的差距，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，直到彼此估算的點數接近即可。

### 1.1.4 工作分包與工作項目總表 (List of Work Packages and Tasks)

	任務名稱	工期	開始時間	完成時間	工時
1	-1 PASS 系統	68 工作日	2017/10/02	2018/01/07	68 工作小時
2	-1.1 需求文件撰寫	70 工作日	2017/10/02	2018/01/07	70 工作小時
3	1.1.1 專案執行計畫	7 工作日	2017/10/02	2017/10/10	7 工作小時
4	1.1.2 軟體需求規格書撰寫	30 工作日	2017/10/02	2017/11/02	30 工作小時
5	1.1.3 軟體設計描述	30 工作日	2017/10/19	2017/11/19	30 工作小時
6	1.1.4 軟體測試文件	30 工作日	2017/11/16	2018/01/01	30 工作小時
7	1.1.5 手冊文件遞交	2 工作日	2018/01/06	2018/01/07	2 工作小時
8	-1.2 專案管理	68 工作日	2017/10/02	2018/01/07	68 工作小時
9	1.2.1 專案企劃	8 工作日	2017/10/02	2017/10/11	8 工作小時
10	1.2.2 專案控管	60 工作日	2017/10/13	2018/01/07	60 工作小時
11	-1.3 系統工程	50 工作日	2017/10/02	2017/11/7	50 工作小時
12	1.3.1 需求分析	4 工作日	2017/10/02	2017/10/05	4 工作小時
13	1.3.2 系統設計	9 工作日	2017/10/06	2017/10/14	9 工作小時
14	1.3.3 專案執行計畫修改	8 工作日	2017/10/14	2017/10/21	8 工作小時
15	1.3.4 軟體需求規格書修改	8 工作日	2017/10/21	2017/10/28	8 工作小時
16	1.3.5 里程碑:完成 PEP 與 SRS	11 工作日	2017/10/28	2017/11/7	11 工作小時
17	-1.4 軟體設計	40 工作日	2017/10/12	2017/12/6	40 工作小時
18	1.4.1 使用者身分驗證和授權設計	8 工作日	2017/10/12	2017/10/23	8 工作小時

19	1.4.2 課程管理設計	8 工作日	2017/10/12	2017/10/23	8 工作小時
20	1.4.3 指派作業管理設計	8 工作日	2017/10/12	2017/10/23	8 工作小時
21	1.4.4 作業繳交設計	10 工作日	2017/10/31	2017/11/13	10 工作小時
22	1.4.5 作業繳交檢視進度 設計	10 工作日	2017/10/31	2017/11/13	10 工作小時
23	1.4.6 作業分數報表設計	10 工作日	2017/10/31	2017/11/13	10 工作小時
24	1.4.7 介面操作設計	10 工作日	2017/10/31	2017/11/13	10 工作小時
25	1.4.8 非功能性需求設計	8 工作日	2017/11/27	2017/12/6	8 工作小時
26	1.5 軟體發展	39 工作日	2017/10/15	2017/11/23	39 工作小時
27	1.5.1 使用者身分驗證和 授權實作與測試	17 工作日	2017/10/15	2017/11/1	17 工作小時
28	1.5.2 課程管理設計實作 與測試	17 工作日	2017/10/15	2017/11/1	17 工作小時
29	1.5.3 指派作業管理實作 與測試	17 工作日	2017/10/16	2017/11/2	17 工作小時
30	1.5.4 作業繳交實作與測 試	17 工作日	2017/11/1	2017/11/17	17 工作小時
31	1.5.5 作業繳交檢視進度 實作與測試	17 工作日	2017/11/1	2017/11/17	17 工作小時
32	1.5.6 作業分數報表實作 與測試	17 工作日	2017/11/1	2017/11/17	17 工作小時
33	1.5.7 操作介面實作與測 試	17 工作日	2017/11/1	2017/11/17	17 工作小時
34	1.5.8 非功能性需求 實作與測試	5 工作日	2017/11/18	2017/11/23	5 工作小時

35	1.6 整合與測試	8 工作日	2017/12/21	2017/12/30	8 工作小時
36	1.6.1 整合測試	3 工作日	2017/12/21	2017/12/23	3 工作小時
37	1.6.2 軟體系統測試	3 工作日	2017/12/24	2017/12/27	3 工作小時
38	1.6.3 軟體接受度測試	2 工作日	2017/12/28	2017/12/30	2 工作小時
39	1.6.4 里程碑:完成整合與 測試	0 工作日	2017/12/30	2017/12/30	0 工作小時

### 1.1.5 工作分包與工作項目內容說明 (Descriptions of Work Packages and Tasks)

1.1 需求文件撰寫		
工作項目 編號/名稱		1.1.1 專案執行計畫
工作內容說明		撰寫系統相關的執行企劃記錄文件
工作產品/格式		PEP / 文件
與其他工作的相依性		
需求技能與知識		專案分析、管理、軟體工程
工作時程		7 工作日 共7小時
資源	人力	5 人
	硬體設備	PC
	軟體工具	Microsoft Office、Draw.io

工作項目 編號/名稱		1.1.2 軟體需求規格書撰寫
工作內容說明		撰寫軟體相關的規格書
工作產品/格式		SRS / 文件
與其他工作的相依性		
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		30 工作日 共30小時
資源	人力	5 人
	硬體設備	PC



	軟體工具	Microsoft Office
--	------	------------------

工作項目 編號/名稱		1.1.3 軟體設計描述
工作內容說明		描述軟體設計的架構
工作產品/格式		SDD / 文件
與其他工作的相依性		
需求技能與知識		專案分析、軟體架構、軟體工程
工作時程		30 工作日 共30小時
資源	人力	5 人
	硬體設備	PC
	軟體工具	Microsoft Office、Visio

工作項目 編號/名稱		1.1.4 軟體測試文件
工作內容說明		描述如何測試軟體的文件
工作產品/格式		STD / 文件
與其他工作的相依性		
需求技能與知識		軟體架構、軟體測試、專業技術
工作時程		30 工作日 共30小時
資源	人力	5 人
	硬體設備	PC
	軟體工具	Microsoft Office、Visio

工作項目 編號/名稱		1.1.5 手冊文件繳交
工作內容說明		軟體使用說明文件
工作產品/格式		說明文件 / 文件
與其他工作的相依性		
需求技能與知識		溝通與談判
工作時程		2 工作日 共2小時
資源	人力	1 工作日
	硬體設備	5 人

	軟體工具	Microsoft Office、Visio
--	------	------------------------

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	規劃專案發展的企劃	
工作產品/格式	PEP / 文件	
與其他工作的相依性	1.1.1	
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	8 工作日 共8小時	
資源	人力	5 人
	硬體設備	PC
	軟體工具	Microsoft Office

工作項目 編號/名稱	1.2.2 專案控管	
工作內容說明	專案執行過程中的版本控制	
工作產品/格式	Git / code	
與其他工作的相依性	1.4、1.5、1.6	
需求技能與知識	程式設計	
工作時程	60 工作日 共60小時	
資源	人力	5 人
	硬體設備	PC server
	軟體工具	Git控制、Github託管

1.3 系統工程		
工作項目 編號/名稱	1.3.1 需求分析	
工作內容說明	確認專案需求，並做評估	
工作產品/格式	技術文件 / 文件	
與其他工作的相依性	1.1.1、1.2.1	
需求技能與知識	軟體設計、軟體架構、專業技術	
工作時程	4 工作日 共 4 工作小時	
	人力	5 人

資源	硬體設備	PC
	軟體工具	Microsoft Word

工作項目 編號/名稱		1.3.2 系統設計
工作內容說明		根據需求分析後的結果，設計專案
工作產品/格式		技術文件 / 文件
與其他工作的相依性		1.3.1
需求技能與知識		軟體設計、軟體架構、專業技術
工作時程		9 工作日 共 9 工作小時
資源	人力	5 人
	硬體設備	PC
	軟體工具	Microsoft Office

工作項目 編號/名稱		1.3.3 專案執行計畫修改
工作內容說明		第一次修改 PEP
工作產品/格式		PEP / 文件
與其他工作的相依性		1.3.1、1.3.2
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		8 工作日 共 8 工作小時
資源	人力	8 工作日
	硬體設備	PC
	軟體工具	Microsoft Office

工作項目 編號/名稱		1.3.4 專案需求規格書修改
工作內容說明		第一次修 SRS
工作產品/格式		SRS / 文件
與其他工作的相依性		1.3.1、1.3.2
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		8 工作日 共 8 工作小時
資源	人力	5 人
	硬體設備	PC
	軟體工具	Microsoft Office

## 1.4 軟體設計

工作項目 編號/名稱		1.4.1 使用者身份驗證和授權設計
工作內容說明		1. 允許管理者創立不同身分的使用者帳號(管理者、老師、學生、助教) 2. 提供安全的登入認證機制 3. 提供使用者管理自己的基本資料(姓名、密碼、e mail、角色) 4. 系統必須提供同時有多個身分
工作產品/格式		技術文件 / 文件
與其他工作的相依性		1.4.2、1.4.3
需求技能與知識		軟體架構、軟體設計、軟體系統
工作時程		40 工作日 共 40 工作小時
資源	人力	5
	硬體設備	PC
	軟體工具	UML、Text、Visio、Word

工作項目 編號/名稱		1.4.2 課程管理設計
工作內容說明		1. 系統提供老師課程管理功能(CRUD) 2. 系統提供老師可以指派該堂課的 TA 3. 老師可以檢視該堂課學生的基本資料並且該堂課的學生才可以繳交作業

		4. 系統提供一個機制給老師或 TA
工作產品/格式		技術文件 / 文件
與其他工作的相依性		1.4.1、1.4.3
需求技能與知識		軟體架構、軟體設計、軟體系統
工作時程		8 工作日 共 8 工作小時
資源	人力	5 人
	硬體設備	PC
	軟體工具	UML、Text、Visio

工作項目 編號/名稱		<b>1.4.3 指派作業管理設計</b>
工作內容說明		1.允許老師/TA CRUD 指派的作業  2.老師/TA 設定作業截止日期和遞交規定
工作產品/格式		技術文件 / 文件
與其他工作的相依性		1.4.1、1.4.2
需求技能與知識		軟體架構、軟體設計、軟體系統
工作時程		8 工作日 共 8 工作小時
資源	人力	5 人
	硬體設備	PC
	軟體工具	UML、Text、Visio

工作項目 編號/名稱		<b>1.4.4 作業繳交設計</b>
工作內容說明		1. 系統允許該堂課學生在作業繳交截止日期前上傳作業  2. 系統要有能力可以處理多次上傳作業  3. 系統在學生上傳作業後提供郵件通知上傳成功  4. 系統可以提供學生上傳 zip 檔並且 unzip
工作產品/格式		技術文件 / 文件
與其他工作的相依性		1.4.5、1.4.6、1.4.7

需求技能與知識		軟體架構、軟體設計、軟體系統
工作時程		10 工作日 共 10 工作小時
資源	人力	5 人
	硬體設備	PC
	軟體工具	UML、Text、Visio

工作項目 編號/名稱		1.4.5 作業繳交檢視進度設計
工作內容說明		1. 系統提供給老師下載學生作業的功能 2. 系統可以顯示該堂課學生繳交作業狀態和繳交時間 3. 系統可以顯示特定學生繳交作業狀態和繳交時間 4. 系統允許學生檢查自己的繳交作業狀態和繳交時間
工作產品/格式		技術文件 / 文件
與其他工作的相依性		1.4.4、1.4.6、1.4.7
需求技能與知識		軟體架構、軟體設計、軟體系統
工作時程		10 工作日 共 10 工作小時
資源	人力	5 人
	硬體設備	PC
	軟體工具	UML、Text、Visio

工作項目 編號/名稱		1.4.6 作業分數報表設計
工作內容說明		1. 系統提供特定作業的分數報表和平均成績 2. 系統提供該堂課全部的分數總平均報表
工作產品/格式		技術文件 / 文件
與其他工作的相依性		1.4
需求技能與知識		軟體架構、軟體設計、軟體系統
工作時程		10 工作日 共 10 工作小時
資源	人力	5 人
	硬體設備	PC
	軟體工具	UML、Text、Visio

工作項目 編號/名稱		1.4.7 操作介面設計
工作內容說明		1. 希望用 WEBUI 或 APP 的方式來呈現
工作產品/格式		技術文件 / 文件
與其他工作的相依性		1.4
需求技能與知識		軟體架構、軟體設計、軟體系統
工作時程		5 工作日 共 5 工作小時
資源	人力	5 人
	硬體設備	PC
	軟體工具	UML、Text、Visio

工作項目 編號/名稱		1.4.8 非功能性需求設計
工作內容說明		1. 提供負載至少 100 位使用者時系統處理能力每秒要有 1/10 的效能 2. 電腦維持每秒一個單位量的處理能力 3. 在尖峰時刻 99% 的回應時間不得多於 2 秒且平均回應時間最多 1 秒
工作產品/格式		技術文件 / 文件
與其他工作的相依性		1.4
需求技能與知識		軟體架構、軟體設計、軟體系統
工作時程		5 工作日 共 5 工作小時
資源	人力	5 人
	硬體設備	PC
	軟體工具	UML、Text、Visio

## 1.5 軟體發展

工作項目 編號/名稱		1.5.1 使用者身份驗證實作與測試
工作內容說明		1. 使用者登入畫面 2. 驗證使用者身份 3. 判斷使用者權限

		4. 顯示使用者可以使用的功能
工作產品/格式		Code / Code
與其他工作的相依性		1.5.2、1.5.3
需求技能與知識		Node.js
工作時程		17 工作日 共 17 小時
資源	人力	5 人
	硬體設備	PC
	軟體工具	VS code、source tree

工作項目 編號/名稱	1.5.2 課程管理設計實作與測試	
工作內容說明	實作與測試 1. 老師可以對課程 CRUD 2. 老師可以指派該堂課的 TA 3. 老師可以檢視該堂課修課同學資料並規定該堂學生才可以教作業 4. 該系統提供多種程式語言繳交機制 5. 設計課程管理頁面	
工作產品/格式	Code / Code	
與其他工作的相依性	1.5.1、1.5.3	
需求技能與知識	Node.js	
工作時程	17 工作日 共 17 小時	
資源	人力	5 人
	硬體設備	PC
	軟體工具	VS code、source tree

工作項目 編號/名稱	1.5.3 指派作業管理實作與測試	
工作內容說明	1. 老師和 TA 可以對指派的作業做 CRUD 的操作 2. 提供設定作業截止時間	



工作產品/格式		Node.js
與其他工作的相依性		1.5.1、1.5.2
需求技能與知識		軟體架構、軟體設計、軟體系統
工作時程		17 工作日 共 17 小時
資源	人力	5 人
	硬體設備	PC
	軟體工具	VS code、source tree

工作項目 編號/名稱		1.5.4 作業繳交實作與測試
工作內容說明		1. 作業繳交畫面設計 2. 系統提供截止日前可以繳交作業 3. 可以處理多次上傳作業能力 4. 寄發上傳成功通知信 5. 系統提供上傳 zip 並且 unzip 功能
工作產品/格式		Code / Code
與其他工作的相依性		1.5.5、1.5.6、1.5.7
需求技能與知識		Node.js
工作時程		17 工作日 共 17 小時
資源	人力	5 人
	硬體設備	PC
	軟體工具	VS code、source tree

工作項目 編號/名稱	1.5.5 作業繳交檢視進度實作與測試
------------	---------------------

工作內容說明		<ol style="list-style-type: none"> <li>1. 作業檢視進度畫面設計</li> <li>2. 提供老師下載學生上傳的作業</li> <li>3. 系統提供該堂課特定作業狀態和繳交時間和平均成績的報表</li> <li>4. 系統提供該堂課特定學生所有繳交狀況和平均成績</li> <li>5. 提供學生檢視自己的作業繳交狀態和上傳時間</li> </ol>
工作產品/格式		Node.js
與其他工作的相依性		1.5.4、1.5.6、1.5.7
需求技能與知識		Node.js
工作時程		17 工作日 共 17 小時
資源	人力	5 人
	硬體設備	PC
	軟體工具	VS code、source tree

工作項目 編號/名稱		1.5.6 作業分數報表實作與測試
工作內容說明		<ol style="list-style-type: none"> <li>1. 系統提供特定作業的分數報表和平均作業成績</li> <li>2. 系統提供該堂課全部的總平均報表</li> <li>3. 報表呈現設計</li> </ol>
工作產品/格式		Node.js
與其他工作的相依性		1.5.4、1.5.5、1.5.7
需求技能與知識		Node.js
工作時程		17 工作日 共 17 小時
資源	人力	5 人
	硬體設備	PC
	軟體工具	VS code、source tree

工作項目 編號/名稱		1.5.7 操作介面實作與測試
------------	--	-----------------

工作內容說明		1. 介面與後端操作相關連接 2. 驗證所呈現的資料跟後端是否相符
工作產品/格式		Node.js
與其他工作的相依性		1.5.4、1.5.5、1.5.6
需求技能與知識		Node.js
工作時程		17 工作日 共 17 小時
資源	人力	5 人
	硬體設備	PC
	軟體工具	VS code、source tree

工作項目 編號/名稱		<b>1.5.8 非功能性需求實作與測試</b>
工作內容說明		1. 提供處理能力再多個使用者下也要維持一定的處理能力 2. 在尖峰時刻 99%的回應時間不得多於 2 秒，平均回應最多為一秒
工作產品/格式		Node.js
與其他工作的相依性		1.5
需求技能與知識		軟體架構、軟體設計、軟體系統
工作時程		17 工作日 共 17 小時
資源	人力	5 人
	硬體設備	PC
	軟體工具	VS code、source tree

<b>1.6 整合與測試</b>	
工作項目 編號/名稱	<b>1.6.1 整合測試</b>
工作內容說明	1. 使用者登入測試 2. 課程管理 CRUD測試

		3. 課程管理權限測試 4. 作業繳交上傳和通知信測試 5. 作業繳交檢視進度設計測試 6. 產生作業分數報表驗證 7. 介面測試
工作產品/格式		Code / Code
與其他工作的相依性		1.5
需求技能與知識		Node.js
工作時程		3 工作日 共3小時
資源	人力	5 人
	硬體設備	PC
	軟體工具	Microsoft Office

工作項目 編號/名稱		<b>1.6.2 軟體系統測試</b>
工作內容說明		1. Web application 加強測試 2. Server Loading 加強測試
工作產品/格式		Code / Code
與其他工作的相依性		1.5
需求技能與知識		Node.js
工作時程		3 工作日 共3小時
資源	人力	5 人
	硬體設備	PC
	軟體工具	Testing Unit

工作項目 編號/名稱		<b>1.6.3 軟體接受度測試</b>
工作內容說明		1. 不同瀏覽器下的操作測試 2. 不同解析度下畫面呈現比例測試 3. 供多個使用者測試，統計使用心得
工作產品/格式		Code / Code
與其他工作的相依性		1.5

需求技能與知識		溝通與談判
工作時程		2 工作日 共2小時
資源	人力	5 人
	硬體設備	PC
	軟體工具	Text

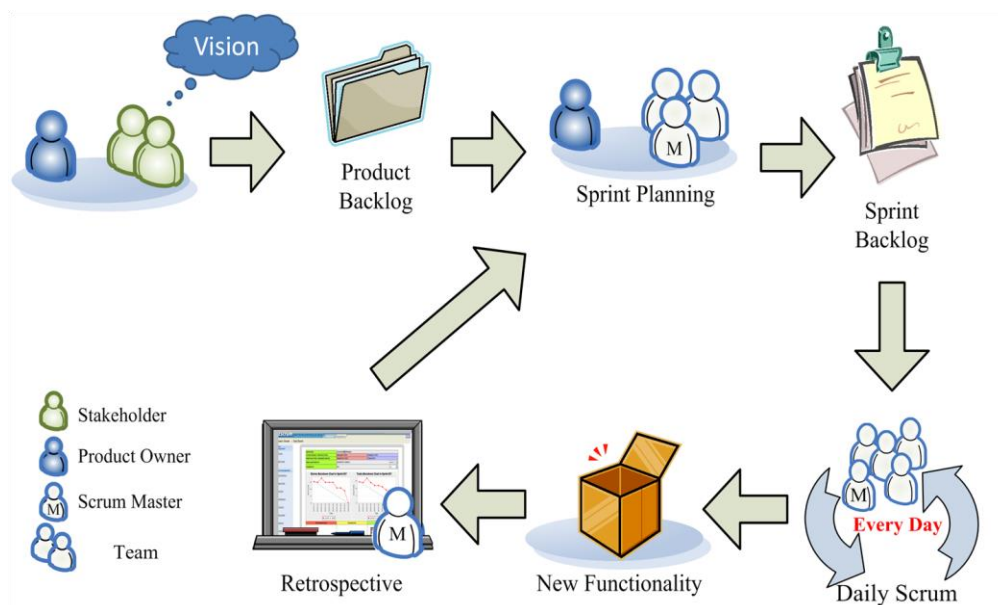
### 1.1.6 專案生命週期定義 (Project Life Cycle)

此專案採用 Scrum 的生命週期方式，我們將一次 Sprint 的週期定在兩周，所以每兩個禮拜我們會把 sprint 的產品進行驗收，驗收完畢後，在選擇對使用者價值高的需求，做為下次 spint 的工作。

#### Scrum 介紹：

將開發時程分為若干個次固定期間的**衝刺(sprint)**，長度介於1-4週。每一個衝刺開始前，由產品擁有者與Scrum團隊會同召開一個**衝刺規劃會議(sprint planning)**，自產品待辦目錄中選擇若干個故事，由Scrum團隊一起將選中的每一個故事分割成為數個工作(task)，將之紀錄於該次衝刺的待辦目錄(sprint backlog)之中。這些工作將由scrum隊員領取並在該次衝刺中完成。在當次衝刺即將結束前，由Scrum團隊對產品擁有者**展演已完成的機能及品質特性 (sprint demo)**。

每一個衝刺的最後一項活動為**回顧(sprint retrospective)**，由隊長主持檢討該次衝刺的好的(good)與待改進事項(improvements)。如此週而復始，直到專案結束或完成所有故事。



## 1.2 預定時程及查核點 (Schedule and Milestone Checking)

### 1.2.1 預訂查核點說明 (Milestone Checking Description)

查核點	預定時間	查核點概述	技術文件/產出物
M1	106.10.10	完成 PEP	專案執行規劃書
M2	106.10.31	完成 Prototyping 和 SRS	系統需求規格書 Prototyping
M3	106.12.4	設計並修正 SRS、PEP 設計並完成 SDD increment 1 設計並完成 Use case	軟體設計描述、專案執行規劃書 系統需求規格書、Use case
M4	106.12.26	設計並完成 STD、 設計並完成 SDD increment 2	系統測試文件
M5	107.1.1	完成系統	專案程式碼、系統測試文件、SDD

1.2.2 預定時程 (Schedule)



### 1.2.3 時程與進度審查監控機制說明 (Schedule & Progress Monitor and Control Mechanism)

本專案對於進行中的工作採用Scrum每兩個禮拜demo的驗收方式進行監控，由專案負責人針對所完成的工作進行進度審視的方式。若此次 Sprint有沒有做完的工作，我們會視情況並且看有沒有繼續做的需要，若要延續實做的話，我們將未完成的部分放進下次的 sprint backlog。

## Section2 專案成員工作指派 (Personnel)

### 2.1 工作項目或工作分包預估需求與估算之假設條件 (Task Estimation Assumptions)

其他估算法

※ 參數：

- a. 文件：1 頁 / 1 人
- b. 系統功能：1 個 / 8 人時
- c. 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	2
專案管理	2
系統工程	2
軟體設計	4
軟體發展	4
整合與測試	4

### 2.2 計畫成員指派 (Roles and Responsibilities)

姓名	縮寫
楊雅雯	Jeni
劉彥麟	Leo
郭士銓	Chenny
陳韻文	Ingrid
李柏霖	Morris

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	專案執行計畫	全體成員	專案分析、軟體架構、軟體工程
1.1.2	軟體需求規格書撰寫	全體成員	專案分析、管理、軟體架構、軟體工程
1.1.3	軟體設計描述	全體成員	專案分析、軟體架構、軟體工程



1.1.4	軟體測試文件	全體成員	軟體架構、軟體測試、專業技術
1.1.5	手冊文件遞交	全體成員	溝通與談判
1.2.1	專案企劃	全體成員	專案分析、管理、 軟體架構、軟體工程
1.2.2	專案控管	全體成員	程式設計
1.3.1	需求分析	全體成員	軟體設計、軟體架構、專業技術
1.3.2	系統設計	全體成員	軟體設計、軟體架構、專業技術
1.3.3	專案執行計畫修改	全體成員	專案分析、管理、 軟體架構、軟體工程
1.3.4	軟體需求規格書修改	全體成員	專案分析、管理、 軟體架構、軟體工程
1.3.5	里程碑： 完成PEP與SRS	全體成員	check
1.4.1	使用者身份驗證設計	全體成員	軟體架構、軟體設計、軟體系統
1.4.2	課程管理設計	全體成員	軟體架構、軟體設計、軟體系統
1.4.3	指派作業管理設計	全體成員	軟體架構、軟體設計、軟體系統
1.4.4	作業繳交設計	全體成員	軟體架構、軟體設計、軟體系統
1.4.5	作業繳交檢視進度設計	全體成員	軟體架構、軟體設計、軟體系統
1.4.6	作業分數報表設計	全體成員	軟體架構、軟體設計、軟體系統
1.4.7	操作介面設計	全體成員	UX、UI 設計想法
1.4.8	非功能性需求設計	全體成員	壓力測試Test case
1.5.1	使用者身份驗證實作與測試	全體成員	軟體架構、Node.js、MongoDB、 Authentication login
1.5.2	課程管理設計實作與測試	全體成員	軟體架構、Node.js、MongoDB
1.5.3	指派作業管理實作與測試	全體成員	軟體架構、Node.js、MongoDB

1.5.4	作業繳交實作與測試	全體成員	軟體架構、Node.js、FileAPI
1.5.5	作業繳交檢視進度實作與測試	全體成員	軟體架構、Node.js
1.5.6	作業分數報表實作與測試	全體成員	軟體架構、D3.js
1.5.7	操作介面實作與測試	全體成員	BootStarp、CSS、Mocha
1.5.8	非功能性需求實作與測試	全體成員	JMeter
1.6.1	整合測試	全體成員	Node.js
1.6.2	軟體系統測試	全體成員	Node.js
1.6.3	軟體接受度測試	全體成員	溝通與談判
1.6.4	里程碑： 完成整合與測試	Leo	check

### 2.3 調整專案成員 (Adjustments)

因為採用 Scrum 的方式，大家都會參與此專案，故不需要特別調整。

### 2.4 專案專業知識與技能需求 (Requirements of Knowledges and Skills)

※ 專業技能：

專業技能及知識	預估需要人數	預計受訓人員	說明
Node.js	5	0	本專案的成員已有相關經驗
MongoDB	5	3	本專案的成員已有相關經驗
RESTful api	5	3	本專案的成員已有相關經驗
UI design	5	0	本專案的成員已有相關經驗

### 2.5 訓練計畫表 (Training Plan)

技術方面團員均有相當的開發經驗，因此只需要在實作時一邊開發一邊發掘問題並討論即可，若成員有進度上的問題，我們會使用 pair programming 的方式來共同開發，不需要特別訓練。

### 2.6 成員參與情況監控機制說明 (Monitor and Control Mechanism) (此項目為必要監控項目)

專案進行時，每天進行 daily scrum 來了解每位成員目前的進度及遇到的困難。當成員執行的工作遇到瓶頸且重要性比其他工作更高時，其他有空的成員要放下手邊的工作先來幫忙解決問題。如果工作進度超前，則團隊將再開一次 Sprint planning 會議，從 product backlog 中挑選對使用者價值性較高的工作來開發。

### Section3 資源需求 (Resources)

#### 3.1 計畫經費預算說明 (Budget)

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	專案執行計畫	30000	1000		31000
1.1.2	軟體需求規格書撰寫	500	1000		1500
1.1.3	軟體設計描述	500	1000		1500
1.1.4	軟體測試文件	500	1000		1500
1.1.5	手冊文件遞交		1000		1000
1.2.1	專案企劃	500	1000		1500
1.2.2	專案控管	500	1000		1500
1.3.1	需求分析	500	1000		1500
1.3.2	系統設計	500	1000		1500
1.3.3	專案執行計畫修改		1000		1000
1.3.4	軟體需求規格書修改		1000		1000
1.3.5	里程碑： 完成PEP與SRS		1000		1000
1.4.1	使用者身份驗證設計	500	1000		1500
1.4.2	課程管理設計	500	1000		1500
1.4.3	指派作業管理設計	500	1000		1500
1.4.4	作業繳交設計	500	1000		1500
1.4.5	作業繳交檢視進度設計	500	1000		1500
1.4.6	作業分數報表設計	500	1000		1500
1.4.7	介面操作設計	500	1000		1500
1.4.8	非功能性需求設計	500	1000		1500
1.5.1	使用者身分驗證和授權實作與測試	5000	2000		7000
1.5.2	課程管理設計實作與測試	5000	2000		7000
1.5.3	指派作業管理實作與測試	5000	2000		7000
1.5.4	作業繳交實作與測試	5000	2000		7000
1.5.5	作業繳交檢視進度實作與測試		500		500
1.5.6	作業分數報表實作與測試	5000	500		5500
1.5.7	操作介面實作與測試	5000	500		5500

1.5.8	非功能性需求實作與測試	5000	500		5500
1.6.1	整合測試	5000	500		5500
1.6.2	軟體系統測試	5000	500		5500
1.6.3	軟體接受度測試		500		500
1.6.4	里程碑： 完成整合與測試		500		500
合計		80000	30000	0	110000

### 3.2 人事費用估算 (Estimations of Personnel Fee)

工作計畫需求人力：754 人時		總人事費用：5 員	
職級	單位(時)	人事費概算	備註
研究生(5人)	754小時	0	學習固無費用
其他(加班費)		0	

### 3.3 計畫經費預估表 (Project Cost Estimation)

經費項目	預定金額	說明
研究設備費	80000	個人電腦以及筆記型電腦
業務費	30000	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	0	專案研究人員共五名之研究經費
管理費	0	專案管理以及其他經費
合計	110000	

### 3.4 預算監控機制說明 (Budget Monitor and Control Mechanism)

預算監控機制為：

- (a) 監控頻率：每隔兩周Sprint結束時監控一次
- (b) 實施矯正之基準及其措施：預算使用超過 5%則實施矯正措施，立即開會討論如何減少花費和如何取得資金。

## Section4 資料管理規劃 (Data Management Plan)

### 4.1 資料管理計畫 (Data Management Plan)

本專案的資料儲存方式有：

- a. 原始程式碼：程式碼由 git 負責做版本控制，並放到 github 託管。
- b. 電子文件：放在 google drive 上由團隊成員共同管理。

c. 紙本文件：由團隊成員共同管理。

#### 4.2 列管資料總表 (List of Managed Data)

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	是	是	密	Event	紙本	團隊	團隊
系統需求規格書	是	是	密	Event	紙本	團隊	團隊、使用者
系統設計規格書	是	是	密	Event	紙本	團隊	團隊
原始程式碼	是	是	公開	Daily	git	團隊	團隊
整合測試計畫書	是	是	密	Event	電子檔	團隊	團隊、測試者
系統測試報告	是	是	密	Event	紙本	團隊、測試者	團隊、測試者
系統接受度報告	是	是	密	Event	電子檔	測試、使用者	測試、使用者

#### 4.3 列管資料監控機制說明 (Monitor and Control Mechanism) (此項目為必要監控項目)

(說明監控列管資料之實施矯正措施基準及機制)

本專案列管資料監控機制為：

- 監控頻率：每隔兩周，sprint 結束時監控一次。
- 實施矯正之基準及措施：資料管理列管的所有資料要按照管理計畫的方式執行，如發現有未按照管理計畫的方式保管，則馬上進行矯正。

### Section5 風險評估 (Risk Management)

#### 5.1 風險項目評估 (Risks Assessment)

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	0%	低	每日固定時間 Meeting，了解組員彼此情況
版本衝突	10%	低	利用git版本控管，以解決版本不一致情況
人員的訓練不足	10%	低	開發時pair programming
資料庫需求變更	50%	高	降低程式的相依度
資料庫伺服器毀損	10%	高	定期備份、異地備份
版控伺服器毀損	10%	高	Client 端之間交叉比較資料
Scrum 伺服器毀損	10%	低	做一個Scrum板貼上牆上、通知Scrum開發人員盡快修復

## 5.2 風險監控機制說明 (Risk Monitor and Control Mechanism) (此項目為必要監控項目)

本專案針對高風險議題提出的處理方案：

- 資料庫需求變更：開會討論，畫出資料庫結構圖
- 資料庫伺服器毀損：隨時在 github 上做備份
- 版控伺服器毀損：定期做備份

## Section 6 建構管理計畫 (Configuration Management Plan)

### 6.1 目的 (Purpose)

本系統開發為了釋出讓使用者使用，因此需要有長時間的維護，或當新需求被提出時，會需要開發的工作，為了有良好的建構管理，本專案在開發的同時也針對早期版本發現的問題做修改，因此需要此計畫。

### 6.2 建立基準 (Establish Baselines)

#### 6.2.1 標示建構管理項目 (Identify Configuration Items)

ID	資料名稱	版本控管	建構類別	產生週期	資料提供者	資料使用者
1	專案執行規劃書	是	規格書	Event	團隊	團隊
2	系統需求規格書	是	規格書	Event	團隊	團隊、使用者
3	系統設計規格書	是	規格書	Event	團隊	團隊
4	整合測試計畫書	是	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	是	報告資料	Event	團隊、測試者	團隊、測試者
7	系統接受度報告	是	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 (Establish a CM System)

本專案系統採用 git 進行建構管理

#### 6.2.3 建立基準 (Create or Release Baselines)

書面資料電子檔及原始碼每次完成後，都要上傳到 github 上進行版本控制

### 6.3 異動追蹤與控制 (Track and Control Changes)

#### 6.3.1 異動追蹤 (Track Change)

- 在 Daily Scrum 時提出異動問題
- 由團隊成員評估影響層面，並決定是否准許異動。

c. 在Daily Scrum持續追蹤異動狀態。

### 6.3.2 建構控制小組 (Configuration Control Board)

由團隊成員自行監控

### 6.3.3 異動控制 (Control Change)

- 確認異動部分的正確性
- 針對異動部分對該版本開出一個 branch
- 記錄此異動原因
- 異動結束後，將此 branch 與 master merge

### 6.3.4 版本控制程式 (The Version Control Tool)

GitHub，是一個透過 Git 進行版本控制的軟體原始碼代管服務。

<<參考資料:<https://zh.wikipedia.org/wiki/GitHub>>>

## 6.4 達成完整性 (Establish Integrity)

### 6.4.1 建構管理記錄 (Establish Configuration Management Records)

此管理記錄為建立與維護用來進行版本控制的紀錄。

### 6.4.2 建構審核 (Perform Configuration Audits)

為達成對於建構系統中的分支擴充性，團隊們必須將子系統開發到一定的整合後，必須再推出一分支，以利後期的除錯。

## Section7 度量與分析計畫 (Measurement and Analysis Plan)

### 7.1 目的 (Purpose)

度量分析主要蒐集本專案的各項資訊，提供各種分析之用。

### 7.2 蒐集資訊的目的與資訊需求 (Information Needs and Objectives)

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	里程碑完成狀況、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 (Base Measurement)

序號	度量	因子
1	客戶滿意度度量	經由客戶問題的反應：與客戶互動的時間
2	里程碑完成狀況、工作單元進度	里程碑完成的時間、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、使用者界面的良劣	系統或功能之錯誤數、使用者反應介面問題
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 (Measurement and Analysis Tool)

使用臺北科技大學軟體系統實驗室所提供之 ezScrum 系統作為分析上述各項資訊的工具。

## Section8 流程與產品品質保證計劃 (PPQA Plan)

### 8.1 目的 (Purpose)

讓團隊可以藉由 PPQA 對整個程式碼的流程和步驟更加了解，除此之外也讓程式穩定度增加，也可以當作系統品質確認的步驟，還可以讓使用者瞭解本系統的穩定度。

### 8.2 客觀檢視流程與產品 (Objectively Evaluate Process and Work Packages)

- 由 Product Owner 檢視系統執行流程，將每個子系統運作過程依照使用者手冊以及系統規格書去複查。
- 確認需求皆為 Product Owner 所期望的。
- 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 (Project Objective Insight)

- Team Member 與 Product Owner 討論並確認需求，來了解 Product Owner 期望的結果。
- Team Member 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 (Management Architecture)

本系統由 Leo 為 Master，其餘成員為 Member，但由於並沒有 Product Owner 存在，因此團隊成員本身也是 Product Owner 的一部份，當系統實作過程中，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。