

醫療資訊整合系統
(Medical Information Integration
System, MII)

Project Execution Plan (PEP)
Version: 2.0

Team# 4

NAME	ID	EMAIL
邱文煜	107598015	t107598015@ntut.edu.tw
謝翰誼	107599003	hyxie.000@outlook.com
吳俊青	108598014	t108598014@ntut.edu.tw
方聖華	108598058	t108598058@ntut.edu.tw
簡子祺	108598063	t108598063@ntut.edu.tw

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

11/21/2019

目錄

目錄.....	2
版次變更記錄 (Change Log)	5
1. 專案規劃及里程碑 (Project Planning and Milestone Checking)	6
1.1. 專案工作內容 (Project Work Description).....	6
1.1.1. 技術方法(Technical Approach)	6
1.1.2. 分工結構圖 (Work Breakdown Structure).....	7
1.1.3. 工作分包與工作項目估算模型與方法 (Establish Estimates of Project Attributes).....	8
1.1.4. 工作分包與工作項目總表 (List of Work Packages and Tasks)	9
1.1.5. 工作分包與工作項目內容說明 (Descriptions of Work Packages and Tasks)	10
1.1.6. 專案生命週期定義 (Project Life Cycle).....	17
1.2. 預定時程及里程碑 (Schedule and Milestone Checking)	18
1.2.1. 預訂里程碑說明 (Milestone Checking Description).....	18
1.2.2. 預定時程 (Schedule)	19
1.2.3. 時程與進度審查監控機制說明 (Schedule & Progress Monitor and Control Mechanism).....	20
2. 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)	22
2.3. 調整專案成員 (Adjustments).....	23
2.4. 專案專業知識與技能需求 (Requirements of Knowledge and Skills)	24

2.5.	訓練計畫表 (Training Plan)	24
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).....	24
3.	資源需求 (Resources)	25
3.1.	計畫經費預算說明 (Budget)	25
3.2.	人事費用估算 (Estimations of Personnel Fee)	26
3.4.	計畫經費預估表 (Project Cost Estimation).....	27
3.5.	預算監控機制說明 (Budget Monitor and Control Mechanism).....	27
4.	資料管理規劃 (Data Management Plan)	28
4.1.	資料管理計畫 (Data Management Plan)	28
4.2.	列管資料總表 (List of Managed Data).....	28
4.3.	列管資料監控機制說明 (Monitor and Control Mechanism)	29
5.	風險評估 (Risk Management)	30
5.1.	風險項目評估 (Risks Assessment)	30
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).....	30
6.	建構管理計畫 (Configuration Management Plan)	31
6.1.	目的 (Purpose).....	31
6.2.	建立基準 (Establish Baselines).....	31
6.2.1.	標示建構管理項目 (Identify Configuration Items).....	31
6.2.2.	運用建立建構管理系統 (Establish a CM System)	32
6.2.3.	建立基準 (Create or Release Baselines)	32
6.3.	異動追蹤與控制 (Track and Control Changes).....	32
6.3.1.	異動追蹤 (Track Change)	32
6.3.2.	建構控制小組 (Configuration Control Board)	32
6.3.3.	異動控制 (Control Change)	32

6.3.4. 版本控制程式 (The Version Control Tool)	32
6.4. 達成完整性 (Establish Integrity)	33
6.4.1. 建構管理記錄 (Establish Configuration Management Records).....	33
6.4.2. 建構審核 (Perform Configuration Audits).....	33
7. 度量與分析計畫 (Measurement and Analysis Plan)	34
7.1. 目的 (Purpose).....	34
7.2. 蒐集資訊的目的與資訊需求 (Information Needs and Objectives).....	34
7.3. 基礎度量 (Base Measurement)	35
7.4. 度量與分析工具 (Measurement and Analysis Tool).....	35
8. 流程與產品品質保證計畫 (PPQA Plan)	36
8.1. 目的 (Purpose).....	36
8.2. 客觀檢視流程與產品 (Objectively Evaluate Process and Work Packages)	36
8.3. 專案目標洞察 (Project Objective Insight)	36
8.3.1. 管理架構 (Management Architecture)	37

版次變更記錄 (Change Log)

Revisions			
Version	Primary Author(s)	Description of Version	Date Completed
1.0	邱文煜、謝翰誼、吳俊青、 方聖華、簡子祺	初期規劃	2019/10/06
2.0	邱文煜、謝翰誼、吳俊青、 方聖華、簡子祺	第一次修改	2019/11/21

1. 專案規劃及里程碑 (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)

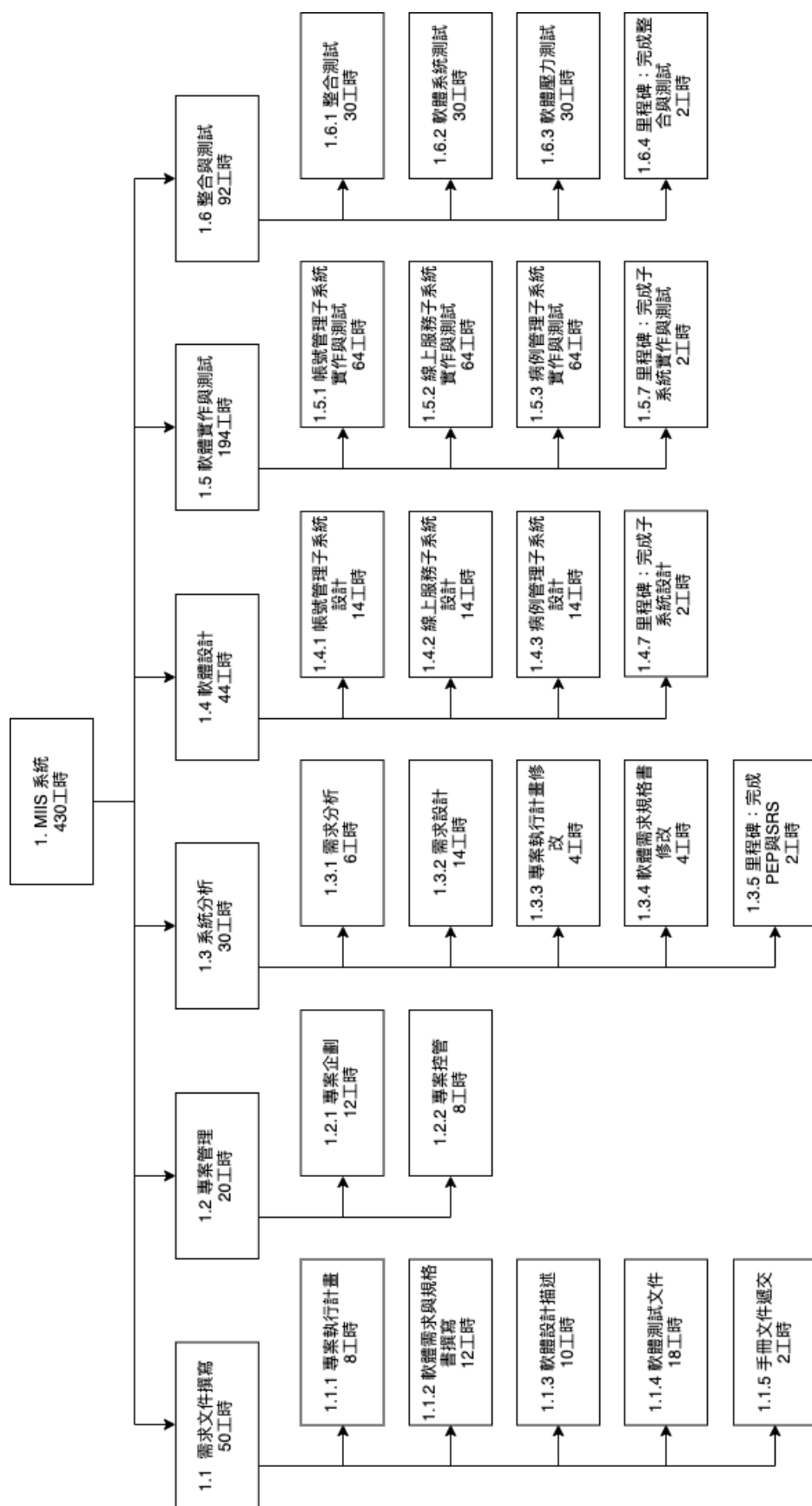
本系統採用 Web Application 設計方法，不但可以讓使用者進行簡單的網頁平台操作此系統，並且也不會受限於同一台電腦設備，可於不同的電腦設備中進行操作。

此系統在於提供一個可以讓不同的醫院使用者進行不同的操作，並且也可以在適當時機進行 Email 的通知提醒。

本系統採用 MVC 架構以及物件導向程式設計方式，完成 MII 系統中的各項子系統的開發、設計與功能測試。

資料庫採用 FHIR Server，使用的程式語言為 TypeScript，且採用軟體工程方法，將整個專案分成：Project Management（專案管理）、System Engineering（系統工程）、Software Development（軟體開發）、System Integration and Test（系統整合測試）、Documentation and Deployment（文件處理）Project Support（專案支援），以取得先前計劃資源等主要工作分類（Package）。

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)

- COCOMO
- COCOMOII
- 單元估算累加法 (Bottom-up, 將工作劃分為較小單元進行估算, 再行累加)
- 專家法(透過個人專業判斷, 進行估算)
- 經驗法(根據歷史資料)

■ 其他估算方式

我們採用 Poker 法進行估算, 開發團隊的成員每個人皆會有一副 Poker, 裡面會有 0、1、2、5、8、10、13、20, 點數是以近似費式數列成長, 當數字越大時, 代表實際上花費的時間也會較為長, 相對困難度也較高, 較容易產生不可預期的誤差。

在剛開始團隊進行開發前會利用此估算法討論出每一樣子系統的難易度, 間接估算出需要的時間以及困難度, 當估算過程中有分數差別時, 最小及最大的數字講解出自己的看法, 經過討論修正後再次出牌, 最後得出最平均的預估級數。

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

主題	起始日期	完成日期	耗時
1.MIIS 系統			430
1.1 需求文件撰寫			50
1.1.1 專案執行計畫	2019/9/26	2019/9/30	8
1.1.2 軟體需求規格書撰寫	2019/10/1	2019/10/7	12
1.1.3 軟體設計描述	2019/11/1	2019/11/6	10
1.1.4 軟體測試文件	2019/11/11	2019/11/20	18
1.1.5 手冊文件遞交	2019/12/24	2019/12/25	2
1.2 專案管理			20
1.2.1 專案企劃	2019/10/5	2019/10/11	12
1.2.2 專案控管	2019/10/15	2019/12/23	8
1.3 系統分析			30
1.3.1 需求分析	2019/10/7	2019/10/10	6
1.3.2 需求設計	2019/10/9	2019/10/16	14
1.3.3 專案執行計畫修改	2019/11/6	2019/11/8	4
1.3.4 軟體需求規格書修改	2019/11/6	2019/11/8	4
1.3.5 里程碑：完成 PEP 與 SRS	2019/10/29	2019/10/30	2
1.4 軟體設計			44
1.4.1 帳號管理子系統設計	2019/10/23	2019/10/30	14
1.4.2 線上服務子系統設計	2019/10/23	2019/10/30	14
1.4.3 病歷管理子系統設計	2019/10/23	2019/10/30	14
1.4.4 里程碑：完成子系統設計	2019/10/30	2019/10/31	2
1.5 軟體實作與測試			194
1.5.1 帳號管理子系統實作與測試	2019/11/1	2019/12/3	64
1.5.2 線上服務子系統實作與測試	2019/11/1	2019/12/3	64
1.5.3 病歷管理子系統實作與測試	2019/11/1	2019/12/3	64
1.5.4 里程碑：完成子系統實作與測試	2019/12/3	2019/12/4	2
1.6 整合與測試			92
1.6.1 整合測試	2019/12/5	2019/12/20	30
1.6.2 軟體系統測試	2019/12/5	2019/12/20	30
1.6.3 軟體壓力測試	2019/12/5	2019/12/20	30
1.6.4 里程碑：完成整合與測試	2019/12/22	2019/12/23	2

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		撰寫系統相關的執行企劃文件	工作內容說明		撰寫軟體相關的規格書
工作產品/格式		PEP / 文件	工作產品/格式		SRS / 文件
與其他工作的相依性			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	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	工作時程		12
資源	人力	4 工作日	資源	人力	6 工作日
	硬體設備	PC		硬體設備	PC
	軟體工具	Microsoft Office		軟體工具	Microsoft Office

工作項目 編號/名稱		1.1.3 軟體設計描述	工作項目 編號/名稱		1.1.4 軟體測試文件
工作內容說明		描述軟體設計的架構	工作內容說明		描述如何測試軟體的文件
工作產品/格式		SDD / 文件	工作產品/格式		STD / 文件
與其他工作的相依性			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	需求技能與知識		軟體架構、軟體測試、專業技術
工作時程		10	工作時程		18
資源	人力	5 工作日	資源	人力	9 工作日
	硬體設備	PC		硬體設備	PC
	軟體工具	Microsoft Office、Draw.io		軟體工具	VSCode、Robot Framework

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
資源	人力	1 工作日
	硬體設備	
	軟體工具	Microsoft Office

1.2 專案管理					
工作項目 編號/名稱		1.2.1 專案企劃	工作項目 編號/名稱		1.2.2 專案控管
工作內容說明		規劃專案發展的企劃	工作內容說明		專案執行過程中的版本控制
工作產品/格式		PEP / 文件	工作產品/格式		Github / Code
與其他工作的相依性		1.1.1	與其他工作的相依性		1.4、1.5、1.6
需求技能與知識		專案分析、管理、軟體架構、軟體工程	需求技能與知識		程式設計
工作時程		12	工作時程		8
資源	人力	6 工作日	資源	人力	69 工作日
	硬體設備	PC		硬體設備	PC server
	軟體工具	Microsoft Office		軟體工具	VSCode

1.3 系統分析					
工作項目 編號/名稱		1.3.1 需求分析	工作項目 編號/名稱		1.3.2 需求設計
工作內容說明		確認專案需求，並做評估	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	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	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	需求技能與知識		軟體設計、軟體架構、專業技術
工作時程		6	工作時程		14
資源	人力	3 工作日	資源	人力	7 工作日
	硬體設備	黑板		硬體設備	PC
	軟體工具	Microsoft Office		軟體工具	Microsoft Office、Draw.io

工作項目 編號/名稱		1.3.3 專案執行計畫修改	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		第一次修改 PEP	工作內容說明		第一次修改 SRS
工作產品/格式		PEP / 文件	工作產品/格式		SRS / 文件
與其他工作的相依性		1.3.1、1.3.2	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	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		4	工作時程		4
資源	人力	2 工作日	資源	人力	2 工作日
	硬體設備	PC		硬體設備	PC
	軟體工具	Microsoft Office		軟體工具	Microsoft Office

工作項目 編號/名稱		1.3.5 里程碑：完成 PEP 與 SRS
工作內容說明		完成 PEP 與 SRS
工作產品/格式		技術文件 / 文件
與其他工作的相依性		1.3.1、1.3.2、 1.3.3、1.3.4
需求技能與知識		軟體設計、軟體架構、專業技術
工作時程		2
資源	人力	1 工作日
	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 帳號管理子系統設計	工作項目 編號/名稱		1.4.2 線上服務子系統設計
工作內容說明		設計與規劃 1. UI 介面設計 2. 註冊帳號 3. 驗證帳號 4. 更改帳號權限 5. 修改帳號資料	工作內容說明		設計與規劃 1. UI 介面介面設計 2. 提醒健檢時間 3. 預約看診 4. 查看預約資訊 5. 預約設備租借 6. 管理設備租借
工作產品/格式		技術文件 / 文件	工作產品/格式		技術文件 / 文件
與其他工作的相依性			與其他工作的相依性		
需求技能與知識		軟體功能設計	需求技能與知識		軟體功能設計
工作時程		14	工作時程		14
資源	人力	7 工作日	資源	人力	7 工作日
	硬體設備	PC		硬體設備	PC
	軟體工具	Microsoft Office、Draw.io		軟體工具	Microsoft Office、Draw.io

工作項目 編號/名稱		1.4.3 病歷管理子系統設計	工作項目 編號/名稱		1.4.4 里程碑：完成子系統設計
工作內容說明		設計與規劃 1. UI 介面設計 2. 新增病例 3. 查看病例 4. 修改病例 5. 產生圖表	工作內容說明		完成各子系統設計
工作產品/格式		技術文件 / 文件	工作產品/格式		技術文件 / 文件
與其他工作的相依性			與其他工作的相依性		1.4.1、1.4.2、1.4.3
需求技能與知識		軟體功能設計	需求技能與知識		軟體設計、軟體架構、專業技術
工作時程		14	工作時程		2
資源	人力	7 工作日	14 工作日 PC	人力	14 工作日
	硬體設備	PC		硬體設備	PC
	軟體工具	Microsoft Office、Draw.io		軟體工具	Microsoft Office、Draw.io

1.5 軟體開發					
工作項目 編號/名稱		1.5.1 帳號管理子系統實作與測試	工作項目 編號/名稱		1.5.2 線上服務子系統實作與測試
工作內容說明		實做與測試 1. UI 介面實做 2. 註冊帳號 3. 驗證帳號 4. 更改帳號權限 5. 修改帳號資料	工作內容說明		實做與測試 1. UI 介面實做 2. 提醒健檢時間 3. 預約看診 4. 查看預約資訊 5. 預約設備租借 6. 管理設備租借
工作產品/格式		Code / Code	工作產品/格式		Code / Code
與其他工作的相依性			與其他工作的相依性		1.5.1、1.5.3
需求技能與知識		Typescript、UI design、FHIR Server	需求技能與知識		Typescript、UI design、FHIR Server
工作時程		64	工作時程		64
資源	人力	32 工作日	資源	人力	32 工作日
	硬體設備	PC		硬體設備	PC
	軟體工具	VSCode		軟體工具	VSCode

工作項目 編號/名稱		1.5.3 病歷管理子系統實作與測試	工作項目 編號/名稱		1.5.3 里程碑：完成子系統實作與測試
工作內容說明		實做與測試 1. UI 介面實做 2. 新增病例 3. 查看病例 4. 修改病例 5. 產生圖表	工作內容說明		完成各子系統實作與測試
工作產品/格式		Code / Code	工作產品/格式		Code / 文件
與其他工作的相依性		1.5.1、1.5.2	與其他工作的相依性		1.5.1、1.5.2、1.5.3
需求技能與知識		Typescript、UI design、FHIR Server	需求技能與知識		Typescript、UI design、FHIR Server
工作時程		64	工作時程		2
資源	人力	32 工作日	資源	人力	1 工作日
	硬體設備	PC		硬體設備	PC
	軟體工具	VSCode		軟體工具	VSCode、Microsoft Office

1.6 整合與測試					
工作項目 編號/名稱		1.6.1 整合測試	工作項目 編號/名稱		1.6.2 軟體系統測試
工作內容說明		測試 1. 使用者登入測試 2. 管理員權限測試 3. 使用者權限測試 4. 各子系統頁面測試	工作內容說明		測試 1. Web application 加強測試 2. Server Loading 加強測試
工作產品/格式		Code /測試報告	工作產品/格式		Code /測試報告
與其他工作的相依性		1.5	與其他工作的相依性		1.5
需求技能與知識		Typescript、Node.JS	需求技能與知識		Typescript、Node.JS
工作時程		30	工作時程		30
資源	人力	15 工作日	資源	人力	15 工作日
	硬體設備	PC		硬體設備	PC
	軟體工具	Testing Unit、Robot Framework		軟體工具	Testing Unit、Robot Framework

工作項目 編號/名稱		1.6.3 軟體壓力測試	工作項目 編號/名稱		1.6.4 里程碑：完成整合與測試
工作內容說明		測試 1. 多個使用者測試 2. 確認軟體穩定度	工作內容說明		完成各子系統實作與測試
工作產品/格式		Code /文件	工作產品/格式		Code / 文件
與其他工作的相依性		1.5	與其他工作的相依性		1.6.1、1.6.2、1.6.3
需求技能與知識		Apache Jmeter	需求技能與知識		Typescript、Node.JS
工作時程		30	工作時程		2
資源	人力	15 工作日	資源	人力	1 工作日
	硬體設備	PC		硬體設備	PC
	軟體工具	Microsoft Office		軟體工具	Testing Unit、Microsoft Office、Robot Framework

1.1.6. 專案生命週期定義 (Project Life Cycle)

- 瀑布(Waterfall)模式

- 快速雛型(Prototype)模式
- 往覆式(Iterative)模式/螺旋模式
- 漸進(Incremental)模式/螺旋模式
- 演化雛型(Evolutionary)模式 (演化法)
- V-Shaped Model
- 同步模式(Concurrent)
- 其他自訂生命週期

1. 需求分析

- a. 了解相關領域背景，例如：醫院進行病歷 CRUD 相關流程。
- b. 挖掘需求，例如：尋找客戶進行需求討論。
- c. 定義需求，例如：依據客戶需求，定義出將提供給客戶的功能，並製作相關文件。

2. 架構設計

- a. 相關技術研究，例如：Typescript、FHIR Server、FHIR Api 使用。
- b. 功能區分，例如：不同的功能進行相對應設計。
- c. 畫架構圖，例如：產生 Class Diagram、Domain Diagram。

3. 開發，例如：藉由 TDD 方式進行開發。

4. 測試，例如：針對整合與系統進行測試。

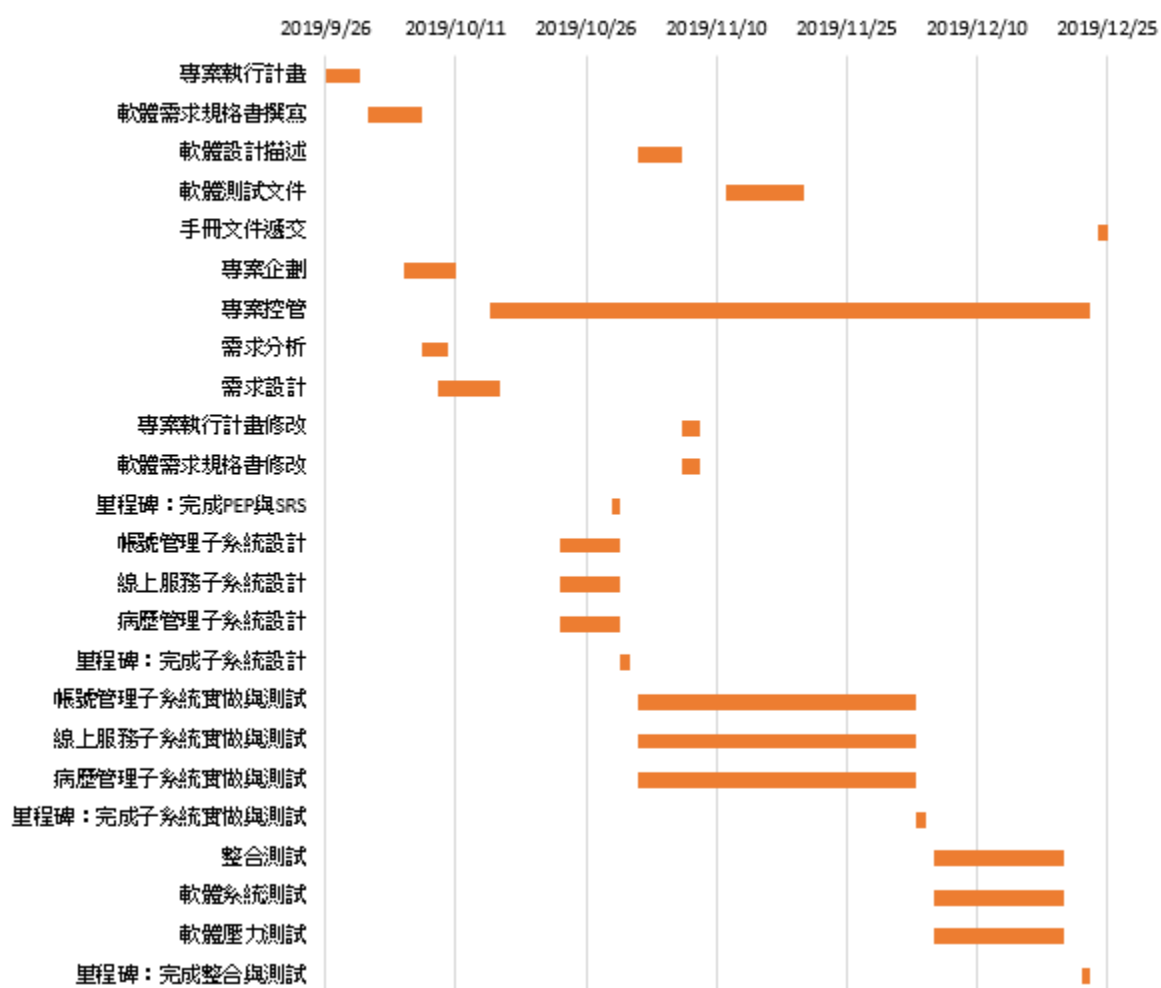
5. 維護，依據客戶檢視成果後，進行相對維護及修改。

1.2. 預定時程及里程碑 (Schedule and Milestone Checking)

1.2.1. 預訂里程碑說明 (Milestone Checking Description)

里程碑	預定時間	里程碑概述	技術文件/產出物
M1	2019/10/05	PEP&SRS 完成	PEP&SRS
M2	2019/10/27	子系統設計完成	子系統設計架構
M3	2019/11/23	子系統實作與測試完成	子系統程式碼
M4	2019/12/21	整合與測試完成	整合與測試報告

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)

本專案對於持續進行中的工作採用兩個禮拜至三個禮拜監控一次，由組長針對每人負責項目所完成的工作比例進行進度審視，而當進度落後時，則實施矯正措施，其基準為：

(1)2019/10~2019/11 矯正基準為 15%，當進度落後超過 15%時則必須進行矯正措施。

(2)2019/11~2020/01 矯正基準為 10%，當進度落後超過 10%時則必須進行矯正措施。

矯正措施由組長與相關人員召開會議，討論未完成的查核點，並且該如何改善落後進度，進而實施。

以下為矯正措施預設範例

(1) 系統文件完成

監控時間	矯正基準	矯正措施
2019/10/30	PEP 完成	列出未完成部分，後續每隔兩天進行進度確認。
2019/10/30	SRS 完成	列出未完成部分，後續每隔兩天進行進度確認。

(2) 軟體開發完成

監控時間	矯正基準	矯正措施
2019/12/04	系統全功能開發完成	列出未完成部分，後續每隔兩天進行進度確認。

2. 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)

※ 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：

a. 文件：1 頁 / 1 人時

b. 系統功能：1 個 / 8 人時

c. 假設條件：以人事行政局公佈的年度上班時間為工作日

d. 一日工作時數為 2 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
系統需求文件撰寫	2
軟體功能設計	2

專業技能	需求人數
軟體開發	4
整合與測試	4

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

姓名	縮寫
邱文煜	MK
謝翰誼	HY
吳俊青	GC
方聖華	SH
簡子祺	TC

工作編號	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	MK、HY、GC 、SH、TC	專案分析、軟體架構、軟體工程
1.1.2	軟體需求規格書撰寫	MK、HY、GC 、SH、TC	專案分析、軟體架構、軟體工程
1.1.3	軟體設計描述	MK、HY、GC 、SH、TC	專案分析、軟體架構、軟體工程
1.1.4	軟體測試文件	MK、HY、GC 、SH、TC	專案分析、軟體架構、軟體工程
1.1.5	手冊文件遞交	MK	溝通與交流
1.2.1	專案企劃	MK	專案分析、軟體架構、軟體工程
1.2.2	專案控管	MK	專案分析、軟體架構、軟體工程
1.3.1	需求分析	HY、GC	專案分析、軟體架構、軟體工程
1.3.2	需求設計	HY、GC	專案分析、軟體架構、軟體工程
1.3.3	專案執行計畫修改	MK、SH、TC	專案分析、軟體架構、軟體工程

1.3.4	軟體需求規格書修改	MK、SH、TC	專案分析、軟體架構、軟體工程
1.3.5	里程碑： 完成 PEP 與 SRS	MK	再次確認是否完整
1.4.1	帳號管理子系統設計	MK、HY	TypeScript、PrimeNG、 軟體架構、軟體工程
1.4.2	線上服務子系統設計	GC	TypeScript、PrimeNG、 軟體架構、軟體工程
1.4.3	病歷管理子系統設計	SH、TC	TypeScript、PrimeNG、 軟體架構、軟體工程
1.4.4	里程碑： 完成子系統設計	MK	再次確認是否有不完整處
1.5.1	帳號管理子系統實作 與測試	MK、HY	TypeScript、PrimeNG、 軟體架構、軟體工程
1.5.2	線上服務子系統實作 與測試	GC	TypeScript、PrimeNG、 軟體架構、軟體工程
1.5.3	病歷管理子系統實作 與測試	SH、TC	TypeScript、PrimeNG、 軟體架構、軟體工程
1.5.4	里程碑： 完成子系統實作與測試	MK	再次確認是否完整
1.6.1	整合測試	MK、HY、GC 、SH、TC	Robot Framework
1.6.2	軟體系統測試	MK、HY、GC 、SH、TC	Robot Framework
1.6.3	軟體壓力測試	MK、HY、GC 、SH、TC	Apache Jmeter
1.6.4	里程碑： 完成整合與系統測試	MK	確認系統測試無誤

2.3. 調整專案成員 (Adjustments)

人員將不會有調度問題，專案成員皆會積極努力且發揮各自專長來完成系統。

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

專業技能及知識	預估需要人數	預計受訓人員	說明
TypeScript	5	5	本專案成員將會利用課餘時間進行研究相關技術
FHIR Server	5	5	本專案成員將會利用課餘時間進行研究相關技術
UI design(PrimeNG)	5	5	本專案成員將會利用課餘時間進行研究相關技術
Robot Framework	5	0	已有相關實作經驗
Apache Jmeter	4	4	本專案成員將會利用課餘時間進行研究相關技術

2.5. 訓練計畫表 (Training Plan)

本團隊針對每個成員的技術知識採取自行學習的方式，並且在後續較為熟悉的成員可以互相教導，因此不另擬定特別訓練計畫。

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)

專案在進行中時，每兩到三個禮拜皆會有一次監控確認，了解各成員所負責的部分是否有按照規劃進行。當執行進度有所延後，將採用 1.2.3 的基準進行是否實施矯正措施，藉由此方法，可以使整體專案確實地往前並有所進度。

3. 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	81,000	1,000		82,000
1.1.2	軟體需求規格書撰寫		1,000		1,000
1.1.3	軟體設計描述		1,000		1,000
1.1.4	軟體測試文件		1,000		1,000
1.1.5	手冊文件遞交		1,000		1,000
1.2.1	專案企劃		1,000		1,000
1.2.2	專案控管		1,000		1,000
1.3.1	需求分析		1,000		1,000
1.3.2	需求設計		1,000		1,000
1.3.3	專案執行計畫修改		1,000		1,000
1.3.4	軟體需求規格書修改		1,000		1,000
1.3.5	里程碑： 完成 PEP 與 SRS		1,000	500	1,500
1.4.1	帳號管理子系統設計		1,000		1,000
1.4.2	線上服務子系統設計		1,000		1,000
1.4.3	病歷管理子系統設計		1,000		1,000
1.4.4	里程碑： 完成子系統設計		1,000		1,000
1.5.1	帳號管理子系統實作與測試	2,000	1,500		3,500
1.5.2	線上服務子系統實作與測試	2,000	1,500		3,500

1.5.3	病歷管理子系統實作與測試	2,000	1,500		3,500
1.5.4	里程碑： 完成子系統實作與測試	2,000	1,500		3,500
1.6.1	整合測試	3,000	1,500		3,500
1.6.2	軟體系統測試	3,000	1,500		3,500
1.6.3	軟體壓力測試		1,000		1,000
1.6.4	里程碑： 完成整合與系統測試		1,000	500	1,500
合計		99,000	34,000	1,000	132,000

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

工作計畫需求人力：375 人時	單位(小時)	總人事費用：5 員	
職級		人事費概算	備註
1321 研究測試員	375	348,000	
其他(加班費)		0	沒這種東西

3.4. 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	99,000	筆記型電腦及其他相關耗材
業務費	25,000	用於一般業務，如文件及消耗品。
人事費	348,000	專案研究人員共五名之研究經費
管理費	34,000	專案管理以及其他經費
合計	506,000	

3.5. 預算監控機制說明 (Budget Monitor and Control Mechanism)

專案預算監控機制：

- a. 監控頻率：每周一次。
- b. 實施矯正之基準及其措施：所有預算盡量依據預算控制花費，若支出超出預算20%，須立即實施矯正。
- c. 矯正措施：開會決定如何從其他預算調動款項，或由組長負擔。

4. 資料管理規劃 (Data Management Plan)

4.1. 資料管理計畫 (Data Management Plan)

本計畫資料管理與儲存方式將分為三種：

- a. 原始程式碼：程式碼由 Github 負責做版本控制，而程式碼將會在此伺服器上做管理。
- b. 電子文件及可執行檔：由臺北科技大學軟體系統實驗室所提供的 ezScrum 系統，可以在記錄板上做上傳 file 的動作，因此只要將報表結果或是圖檔之類的文件，直接利用系統上做管理。
- c. 紙本文件或光碟資料：由 Master 負責做管理。

4.2. 列管資料總表 (List of Managed Data)

資料名稱	是否交付	資料型態	媒介	機密等級	資料提供者	資料使用者
專案執行規劃書	否	文件	電子	密	團隊	團隊
系統需求規格書	否	文件	電子	密	團隊	團隊、使用者
系統設計規格書	否	文件	電子	密	團隊	團隊
整合測試計畫書	否	文件	電子	密	團隊	團隊
系統測試報告	否	文件	電子	密	團隊	團隊
原始程式碼	否	程式碼	電子	密	團隊	團隊
使用手冊	是	文件	電子	密	團隊	團隊、使用者

4.3. 列管資料監控機制說明 (Monitor and Control Mechanism)

專案監控列管資料機制：

- a. 監控頻率：每兩周一次。
- b. 實施矯正之基準：所有列管資料都必須按照資料管理計畫進行，若發現任何的資料未按計畫保管或備份，須立即矯正。
- c. 矯正措施：立即補上未完成或遺失的資料。

5. 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	風險發生之處理或避免方法
人員異動	5%	低	每週固定 Meeting, 定期 Code Review, 使組員資訊同步
版本衝突	5%	低	利用 Github 版控系統, 解決版本不一致的情況
人員技術不足	30%	低	多補充相關技術之知識, 組員間彼此互相交流
版控伺服器損毀	5%	高	Client 端之間資料進行比對
需求改變	20%	高	盡快召開會議, 重新設計架構
專案系統伺服器被入侵	5%	高	做好資料加密傳輸等網路安全設定

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)

人員技術不足：

解決方法：除針對專案使用技術充實自我外，由於專案為團隊開發，各團隊成員皆有各自擅長領域與項目，可彈性調整人員以及互相合作完成開發項目。

需求改變：

解決方法：盡快召開臨時會議，討論此需求改變的優先程度以及造成的影響，依照優先程度調整投入的人力變動，並且以此投入的變動對專案時程規劃與各工作單元進度進行調整。

6. 建構管理計畫 (Configuration Management Plan)

6.1. 目的 (Purpose)

本專案系統開發為釋出讓醫院及使用者去使用，所以需要有很長的時間去做維護，當需求變更時，早期發現修改後造成的問題，可以降低修改後造成的風險，也可以作為專案規劃與維護管理依據，因此才需要做此計畫。

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	Yes	規格書	Event	團隊	團隊
2	系統需求規格書	Yes	規格書	Event	團隊	團隊、醫院
3	系統設計規格書	Yes	規格書	Event	團隊	團隊
4	系統測試計畫書	Yes	規格書	Event	團隊	團隊、測試人員
5	原始程式碼	Yes	原始碼	Monthly	團隊	團隊
6	系統測試報告	Yes	報告資料	Event	團隊、測試人員	團隊、測試人員

6.2.2. 運用建立建構管理系統 (Establish a CM System)

本專案透過 Jenkins 方式進行建構管理。

6.2.3. 建立基準 (Create or Release Baselines)

隨著系統開發更新，需求會不斷改變，因此，可以透過版本管理可以幫助我們回顧檢視過去編輯的內容以及誰更動了這些內容，我們可以輕易選擇回到上一步狀態，或是恢復到先前編輯的版本方式來進行建構。

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

6.3.1. 異動追蹤 (Track Change)

1. 提出異動需求申請單。
2. 由建構管理員或 master 評估影響層面，並且通知所有組員。
3. 由 master 邀請受影響單位進行評估，並決定是否異動。
4. 追蹤異動狀態。

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

小組團隊本身自行監控。

6.3.3. 異動控制 (Control Change)

1. 開出一個分支。
2. 再次確認正確性。
3. 確認異動後必須對異動的地方和異動的原因做紀錄。

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

Github 為分散式版本控制系統，可以把檔案的狀態作為更新歷史記錄保存起來。

6.4. 達成完整性 (Establish Integrity)

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

主要控制及紀錄，來達到建立與維護工作產品的完整性。紀錄資料可以使用 Github 所提供欄位來觀看修改記錄。

6.4.2. 建構審核 (Perform Configuration Audits)

1. 準備一份報告，其中列出基準目標的每個要求，標記測試結果(通過/失敗)。
2. 確認每個需求都有一個或多個測試，並且該需求的所有測試均已通過。

7. 度量與分析計畫 (Measurement and Analysis Plan)

7.1. 目的 (Purpose)

本項目目的主要是透過各種層面來評估本專案，如針對開發流程、開發程式碼、開發過程文件或是開發成果等，收集不同指標度量提供作為分析使用，並達到追求最大客戶滿意度、最小成本和時程以及最小風險。

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

- 客戶滿意度：為分析客戶滿意度，針對此項目收集以下度量資訊
 - 問卷調查
 - 系統易用性量表
- 成本和時程：為掌控並分析專案成本和時程，收集以下度量資訊
 - WBS
 - 預定時程與查核點規劃
 - 各項資源支出費用與投入人力數
 - 工作單元進度
- 風險：為避免專案開發過程與結果發生意外，定期對以下資訊進行分析
 - 程式碼控管紀錄
 - 電子文件控管紀錄
 - 系統測試報告

7.3. 基礎度量 (Base Measurement)

序號	度量	因子
1	用戶體驗滿意度	透過用戶直接反應、問卷調查以及系統易用性量表
2	專案查核點完成度	透過 WBS 與工作單元進度評估完成進度
3	各資源支出與支援程度	依照開發進度計算實際支出與投入人力數
4	程式碼版本控管紀錄	透過版控系統定期對程式碼進行版本管理，並記錄各次異動資訊
5	電子文件控管紀錄	透過雲端文件管理平台紀錄文件編輯歷程
6	系統功能品質	為確保系統品質，開發過程進行單元測試、整合測試、以及程式碼覆蓋率等，並撰寫測試報告

7.4. 度量與分析工具 (Measurement and Analysis Tool)

本專案使用臺北科技大學軟體系統實驗室所提供之 ezScrum 系統以及 Github 平台管控之資料作為分析上述各項資訊的工具。

8. 流程與產品品質保證計劃 (PPQA Plan)

8.1. 目的 (Purpose)

本系統的流程與產品品質保證計畫是為了提供參與人員和管理者客觀的洞察流程及相關的工作產品，以告知使用者關於本系統的品质趨向。

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

執行方法：

客觀地評鑑工作產品方面：

- 從軟體測試、系統整合測試報告中，評定系統是否有達到標準
- 於里程碑時，檢視專案產品完成度是否符合預期
- 在系統交付前，評估軟體系統品質是否符合標準

客觀地評鑑流程方面

- 於里程碑時，檢視專案進度是否符合預期
- 以 CMMI 為準則，檢視流程是否有依 CMMI 標準來完成

8.3. 專案目標洞察 (Project Objective Insight)

執行方法：

- 專案負責人必須與使用者一同討論和確定需求，使得能更了解使用者預期的結果為何
- 專案負責人將需求細節告知於其他組員，並確保每位組員完全了解需求內容及最後預期結果為何
- 將需求紀錄於 ezScrum 上，讓每個組員可以隨時確保完成的功能如同預期

8.3.1. 管理架構 (Management Architecture)

執行方法：

- 每兩週進行一次會議，報告現階段之完成度，加以檢視專案進度
- 利用 Jenkins 實現測試自動化及持續整合，並每日建置一次，確保系統穩定性
- 由於此專案沒有 Product Owner，故在例行會議上，每個組員分別展示完成的部分，並由其他組員檢視是否符合需求