

# **Medical Record Management System (MRMS)**

## **Software Design Document (SDD)**

**Version: 1.1**

**Team 9**

<b>Name</b>	<b>ID</b>	<b>E-mail</b>
鍾宜舫	108598005	t108598005@ntut.org.tw
鮑勇志	108598057	t108598057@ntut.org.tw
楊智成	108598043	t108598043@ntut.org.tw
白庭愷	108598042	t108598042@ntut.org.tw
游上逸	108598052	t108598052@ntut.org.tw
陳昭銘	108598065	t108598065@ntut.org.tw

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

**12/30/2019**

## 目錄 (Table of Contents)

<b>Section 1 System Models/System Architecture.....</b>	<b>4</b>
<b>1.1 Decomposition of the Functional Requirement to Subsystem Requirement and Interfaces .....</b>	<b>4</b>
1.1.1 Functional Requirement.....	4
1.1.2 System Interfaces .....	5
<b>1.2 Establish Technical Solution Criteria.....</b>	<b>5</b>
<b>1.3 Describe Alternative Solution .....</b>	<b>5</b>
<b>1.4 Select System Solution .....</b>	<b>7</b>
<b>1.5 System Internal/External Interface.....</b>	<b>8</b>
<b>Section 2 Design Issues and Solutions .....</b>	<b>9</b>
<b>2.1 Electronic Medical Record Management Subsystem .....</b>	<b>9</b>
2.1.1 Subsystem Characteristics .....	9
2.1.2 Establish Technical Solution Criteria.....	9
2.1.3 Selected Subsystem Solution .....	10
2.1.4 Error Detection and Recovery .....	10
<b>2.2 Reservation Subsystem.....</b>	<b>11</b>
2.2.1 Subsystem Characteristics .....	11
2.2.2 Establish Technical Solution Criteria.....	11
2.2.3 Selected Subsystem Solution .....	12
2.2.4 Error Detection and Recovery .....	12
<b>Section 3 Detailed of Subsystem and Interface Description .....</b>	<b>13</b>
3.1.1 Use Cases Analysis .....	13
3.1.2 User Interfaces Analysis .....	21
3.1.3 Static Model.....	26
3.1.4 Dynamic Models.....	28
<b>3.2 Subsystem Internal/External Interface Requirement and Design .....</b>	<b>34</b>
3.2.1 Internal Interface Requirements.....	34
3.2.2 External Interface Requirements.....	34
<b>Glossary .....</b>	<b>35</b>
<b>References.....</b>	<b>36</b>
<b>Appendix.....</b>	<b>37</b>

## Revisions

Version	Primary Author(s)	Description of Version	Date Completed
1.0	鍾宜舫 鮑勇志 楊智成 白庭愷 游上逸 陳昭銘	初期規劃	11/20/2019
1.1	鍾宜舫 鮑勇志 楊智成 白庭愷 游上逸 陳昭銘	因應 Demo 修改相關內容(System block diagram, use case diagram,use case 等)	12/30/2019

## Section 1 System Models/System Architecture

### 1.1 Decomposition of the Functional Requirement to Subsystem Requirement and Interfaces

#### 1.1.1 Functional Requirement

病歷管理系統(Medical Record Management System, MRMS)提供一個整合性的醫療資訊交換系統。本系統共分為二個子系統：

1. 電子病歷子系統(Electronic Medical Record Management System, EMRMS)提供管理所有帳號以及帳號使用者權限的介面，及所有帳號所對應到的相關個人醫療資訊。
2. 預約子系統(Reservation Subsystem, RS)可依據病患、醫療人員的不同介面管理預約資訊。

二個系統整合來完成我們的Medical Record Management System (MRMS)。

MRMS 的 Functional Requirement 如下：

需求編號	優先順序	需求描述
EMRMS -F-01	1	使用者必須先登入才可進行系統功能。
EMRMS -F-02	1	提供 Administrator 建立帳號功能。
EMRMS -F-03	1	提供使用者登入，進行身份辨識。
EMRMS -F-04	1	使用者可以設定/修改自己的個人資訊。
EMRMS -F-05	1	提供 Administrator 帳號管理功能，包括帳號的新增、設定權限、刪除和查詢。
EMRMS -F-06	1	提供使用者預約看診功能
EMRMS -F-07	1	提供醫療人員對病患病歷之新增、查詢功能。
EMRMS -F-08	1	自動紀錄病患之看診紀錄。
RS -F-01	1	使用者必須先登入才可進行預約相關事項。
RS -F-02	1	提供病患預約看診的功能。
RS -F-03	1	提供醫療人員查詢個人預約看診的資訊。

### 1.1.2 System Interfaces

圖 1-1 為 MRMS 的系統架構圖，說明了 MRMS 與其包含的數個子系統間的互動關係。

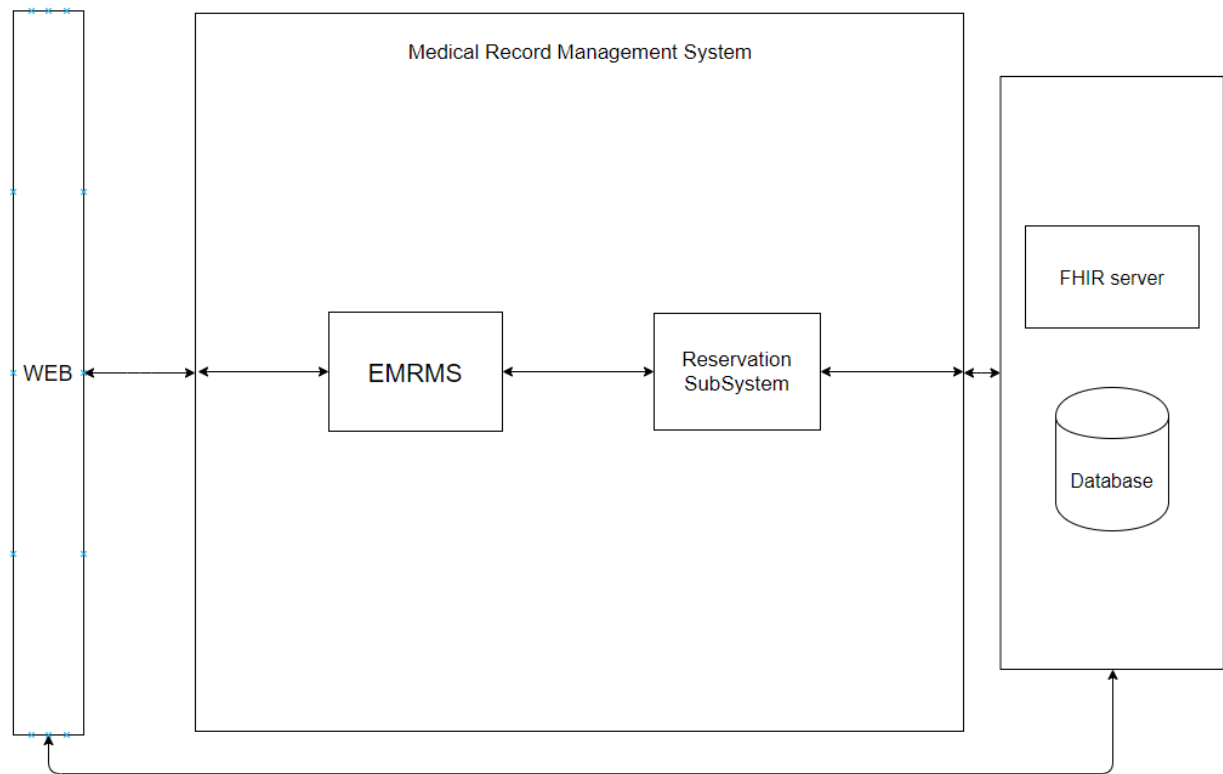


圖 1-1

本系統主要分為二個部分，分別為電子病歷子系統(Electronic Medical Record Management System, EMRMS)、預約子系統(Reservation Subsystem, RS)。

### 1.2 Establish Technical Solution Criteria

有關 Solution Criteria 這方面，專案針對 MRMS 擬定了將來會遇到的各種限制，包括：

- 易學性：考慮選擇的應用軟體是否容易上手。
- 可攜性：考慮能否在各個 OS 上執行。
- 安全性：考慮應用軟體設計上的安全保密性。
- 擴充性：考慮後續的擴充是否容易。
- 廠商支持程度：考慮是否有廣泛的使用者。
- 維護性：考慮後續的維護是否容易。

上述的限制問題，基本上與系統所用的軟體是否有關。

### 1.3 Describe Alternative Solution

在本系統設計前，要考慮各種可能影響系統架構的因素，包括系統架構、程式設計架構、資料庫軟體系統、與程式語言等的選擇。

(一) 由本專案的系統架構，開發者提供了可行的系統架構，分別為單機系統、主從系統、與網頁系統 架構，以下列出與上述限制比較表：

	單機系統	主從系統	網頁系統
易學性	高，開發時間較短	低，開發較長時間	低，開發較長時間
可攜性	低，必需因應每個平台而修改	中，必需額外開放使用	高，使用Browser連上即可
安全性	高，單機資料保護較容易	中，有被盜取資料的風險	中，有被盜取資料的風險
擴充性	彈性小，不能完全符合使用者需求	彈性大，可完全符合使用者需求	彈性大，可完全符合使用者需求
維護性	高，由開發者自行維護，且發展也極為不易。安裝較為簡單	高，由開發者自行維護。安裝較為複雜	低，由供應商提供維護作業。安裝容易

(二) 可用來開發本系統的程式語言約有 Python、Java、C++：

	Python	Java	C++
易學性	高，內建豐富的web相關API，易學	高，內建豐富的web相關Package(Servlet)，易學	中，需自行管理記憶體釋放問題，較難使用
可攜性	高，OS只需擁有python interpreter	高，JVM可支援各個平台	高，最早成熟的語言，有豐富的資源
安全性	中，物件導向語言可實現資料封裝	中，物件導向語言可實現資料封裝	中，物件導向語言可實現資料封裝
擴充性	彈性高，部份符合開發者需求	彈性大，可完全符合開發者需求	彈性小，不能完全符合開發者需求
維護性	高，物件結構較易維護	高，物件結構較易維護	中，需考量記憶體釋放問題

(三) 系統的 User Interface 所使用 AJAX 框架目前有三種選擇，分別為 Prototype、JQuery、EXT，以下列出與上述限制比較表：

	Prototype	JQuery	Ext JS
易學性	中，使用率最高，範例好找	高，每個API都有完整的例子	低，大量的UI框架，較為複雜
可攜性	無，無可攜性需求	無，無可攜性需求	無，無可攜性需求
安全性	無，無安全性考量	無，無安全性考量	無，無安全性考量
擴充性	低，無相關Plug-in	高，有大量用戶開發的Plug-in可供使用	高，模塊化實現，可擴展性強
維護性	高，每個Action都由單一檔案發佈	高，簡潔架構和簡短的語法，容易維護	高，100% 面向對象和組件化的思想，較易維護

## 1.4 Select System Solution

### (一) 系統架構選擇

	單機系統	主從系統	網頁系統
易學性	4	2	5
可攜性	2	3	4
安全性	4	2	3
擴充性	2	4	4
維護性	4	4	2
加總	16	15	18

Priorities {Scale: 1= Most bad(difficult), 5= Most good(easy), 0= Not necessary}

### (二) 開發語言選擇

	Pyhon	Java	C++
易學性	5	4	2
可攜性	3	5	4
安全性	3	3	3
擴充性	5	5	2
維護性	4	4	3
加總	20	21	14

Priorities {Scale: 1= Most bad(difficult), 5= Most good(easy), 0= Not necessary}

### (三) User Interface AJAX 框架選擇

	Prototype	JQuery	Ext JS
易學性	4	5	2
可攜性	0	0	0
安全性	0	0	0
擴充性	2	4	4
維護性	4	4	4
加總	10	13	10

Priorities {Scale: 1= Most bad(difficult), 5= Most good(easy), 0= Not Necessary}

因為 Ext 需要付費，所以可能的選擇只有 Prototype 和 JQuery，而其中 JQuery 為輕量級的開發，追求簡潔和高效，所以 JQuery 適合於快速開發，因此選擇 JQuery。

因此本系統將其特性來採用網頁系統的架構，並使用 JSP開發 GUI 介面，並用 Struts 分離系統介面與邏輯。

### 1.5 System Internal/External Interface

需求編號	優先順序	需求描述
MRMS -N-01	1	程式、UI 及資料庫連結均採用 Java、JavaScript 程式語言撰寫設計。
MRMS -N-02	1	資料庫採用 MySQL。
MRMS -N-03	1	使用者操作需基於 WEB 的 UI。
MRMS -N-04	1	因使用 MySQL，需注意資料庫連線之安全性設定，以降低遭受入侵的危險。
MRMS -N-05	1	登入介面。
MRMS -N-06	1	依據使用者身分建立介面。
MRMS -N-07	1	管理醫療資訊介面。
MRMS -N-08	1	EMRMS 與 FHIRserver、資料庫間必須能存取使用者相關資料。
MRMS -N-09	1	EMRMS 與 FHIRserver、資料庫間必須能存取個人醫療相關資料
MRMS -N-10	1	頁面切換必須小於 2 秒。
MRMS -N-11	1	報表產生速度小於 10 秒。
MRMS -N-12	1	通過 Use Case 所寫的 Test Case。
MRMS -N-13	1	系統需提供備份機制。
MRMS -N-14	1	系統需提供轉移機制。
MRMS -N-15	1	主機為 Intel P4 1.5GHZ 以上的機器。
MRMS -N-16	1	硬碟(Hard Disk)空間需要 200MB 以上。
MRMS -N-17	1	記憶體 512MB 以上。
MRMS -N-18	1	安裝建立 MySQL。
MRMS -N-19	1	資料庫建立資料表。
MRMS -N-20	1	交付的軟體必需依 MRMS 的需求規格來建造。
MRMS -N-21	1	由程式開發者提供維護服務。



## Section 2 Design Issues and Solutions

### 2.1 Electronic Medical Record Management Subsystem

#### 2.1.1 Subsystem Characteristics

電子病歷管理子系統(Electronic Medical Record Management Subsystem) 是提供系統管理者提供管理介面，並針對系統使用者做一病歷管理功能，隨時掌握病患身心狀況進行記錄的整合性系統。

1. 帳號管理(Account Management) 之功能為建立、修改、刪除所有使用者的登入帳戶。
2. 登入管理(Login Management)之功能為驗證系統使用者是否已註冊，若是以註冊則已登入。
3. 系統管理者之功能為建立、讀取、更新、刪除病歷名稱及病歷內容。
4. 系統使用者管理之功能為讓系統使用者查詢個人健康的詳細資料，變更個人資料(個人電話、住址...)。
5. 病歷管理(Medical Record Management)之功能為讓系統管理者新增、讀取、更新、刪除病歷，可依病歷內容屬性設定過濾器，且可供系統管理者上傳相關醫療資訊(檢驗報告、看診紀錄、預約看診)，系統管理者也可查看特定醫療項目的檢驗紀錄內容，以及病患的看診紀錄。

#### 2.1.2 Establish Technical Solution Criteria

使用者登入驗證方法比較表：

使用者的驗證機制有「以角色為基礎的存取控制 (Role-based access control, RBAC)」和直接實作兩種方法，以下為其特性。

	RBAC	直接實作登入驗證機制
易學性	低，RBAC使用較為複雜	高，直接實作角色權限不需另外學習
可攜性	中，必須處理安置SessionKey的儲存位置	高，沒有需要額外負擔的處理
安全性	高，RBAC控制權是直接賦予角色較安全	低，直接實作控制權限，漏洞很多
擴充性	中，必須依靠RBAC內建功能來擴充	高，不足部份直接實作即可
廠商支持程度	高，由美國國家標準局提出的一種新的存取控制機制	中，自行撰寫的權限控制方法
維護性	高，RBAC的角色權限較易維護	中，定義一個新的角色必須撰寫新的權限功能

### 2.1.3 Selected Subsystem Solution

使用者驗證方法選擇：

	RBAC	直接實作登入驗證機制
易學性	2	4
可攜性	3	5
安全性	5	2
擴充性	3	5
廠商支持程度	4	3
維護性	4	3
加總	21	22

Priorities {Scale : 1 = Most bad (difficult), 5 = Most good (easy) }

由於專案開發時間較短，因此決定放棄使用雖然較安全，且功能也較完善的 Role-based access control 方法，而採用較簡潔開發也較快速的方法當作 MRMS 的登入、權限驗證機制。

### 2.1.4 Error Detection and Recovery

Error	Detection & Recovery
使用者輸入密碼錯誤	跳出警告視窗，並警告使用者。
使用者輸入密碼長度錯誤	由系統檢測到不足/過長的密碼長度，將會警告使用者，並使其重新輸入。
使用者輸入資料錯誤	於使用者輸入資料錯誤的項目顯示訊息以供更正。
資料不一致(資料庫端)	本系統的主程式將於使用者新增資料的時候自動檢查 (1)帳號資料與群組資料的一致性。 (2)專案資料與開發人員帳號的一致性。 當系統偵測出不一致性時，會顯示相關訊息，且使用者無法新增任何資料。
資料不一致(使用者輸入端)	於前端的資料送至伺服端的應用程式時，再進行檢查。當子系統一偵測出不一致性時，相關的訊息將由本子系統由前端通知使用者。
資料庫損毀	資料庫的偵錯：使資料庫具偵錯機制，並定期產生LOG檔，供系統管理員查核。
	資料庫的備份：定期備份所有資料，確認資料庫安全不被入侵。
	資料庫的復原：系統發生問題、資料損毀時，可由之前備份的資料進行還原修復。

## 2.2 Reservation Subsystem

### 2.2.1 Subsystem Characteristics

預約子系統(RS, Reservation Subsystem)針對系統中的預約資訊相關的管理與操作。主要功能如下：

1. 預約管理(Reservation Management)之功能為可新增、刪除、查詢預約數量及預約內容，並且可自動排程預約順序。
2. 資料庫(Database)可儲存預約相關資料，包含各項屬性(Attribute)。

### 2.2.2 Establish Technical Solution Criteria

#### 2.2.2.1

資料庫系統儲存的比較表：

目前資料庫常見的有 Access、Microsoft SQL Server 與 MySQL 等。Access 的應用較偏重於個人使用，且執行速度並不佳；Microsoft SQL Server 與 MySQL 都屬於較大型的資料庫系統，兩者差別主要是在 MySQL 是免費的，而 Microsoft SQL Server 不是。

	Microsoft SQL Server	MySQL	MangoDB
易學性	中，原來的使用者已有基礎，但仍需再訓練	高，原來的使用者已有基礎	高，部分的使用者已有基礎
可攜性	低，MS SQL 與.NET 整合，必須使用 .NET FrameWork才能使用到完整的功能	高，為企業用免費的系統，容易安裝	高，為開源的系統，容易安裝
安全性	高，MS會定期提供安全性更新	中，My SQL安全性目前無重大漏失	中，MangoDB安全性目前無重大漏失
擴充性	高，資料量大，其SQL Server 在重負載下表現特別突出	中，儲存量適於中小型的資料量儲存	高，在處理大量資料時有先天優勢，非關聯式資料庫在擴充上比較不會受到硬體的限制
廠商支持程度	低，需付費	高，免費	中，免費，廠商的支持程度漸漸變多

### 2.2.3 Selected Subsystem Solution

資料庫系統儲存的選擇：

	Microsoft SQL Server	MySQL	MongoDB
易學性	2	4	3
可攜性	4	3	3
安全性	4	3	3
擴充性	4	3	4
廠商支持程度	1	5	4
加總	15	18	17

Priorities {Scale: 1 = Most bad (difficult), 5 = Most good (easy)}

根據上表評估，決定採用 My SQL 作為 RS 的儲存資料庫。

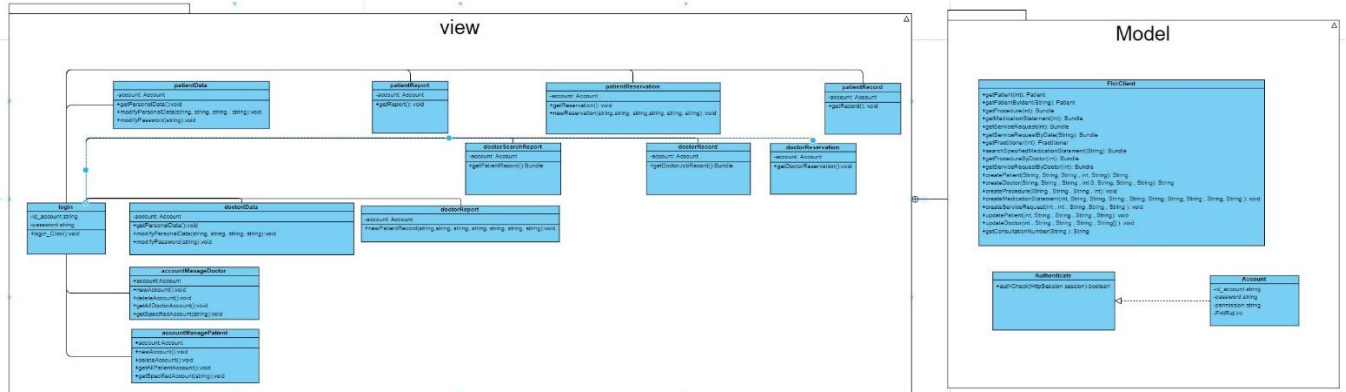
### 2.2.4 Error Detection and Recovery

Error	Detection & Recovery
使用者未登入	沒有客戶端的Section，不執行使用者操作並且導入MRMS登入畫面。
頁面產生速度過慢	重新審視子系統中模組的取得FHIR server資料的方式並加以改良。
資料不一致(使用者輸入端)	於前端的資料送至伺服器端的應用程式時，再進行檢查。 當子系統一偵測出不一致性時，相關的訊息將由本子系統由前端通知使用者。
資料庫損毀	資料庫偵錯機制:定期產生log檔供系統管理員查詢
	資料庫備份:定期備份資料庫資料
	資料庫復原發生問題及異常時，使用之前備份的資料庫資料復原資料庫

## Section 3 Detailed of Subsystem and Interface Description

### 3.1 Detailed System

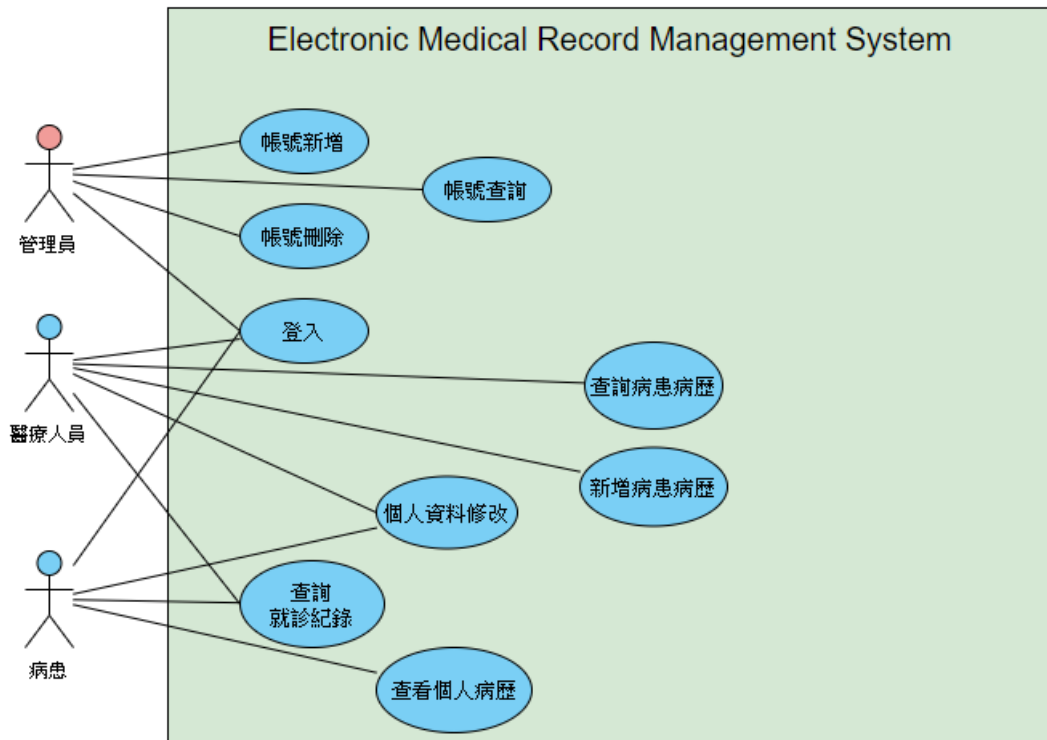
本系統因為其特性所以採用網頁系統的架構來實作本系統，並且使用 Class Diagram 來描述其中的細部設計。



#### 3.1.1 Use Cases Analysis

在這小節中我們將以 Use case 來描述此子系統中各個元件的使用流程和反應。

##### ● Electronic Medical Record Management Subsystem



<b>No</b>	EMRMS-UC01
<b>Use Case</b>	進行登入。
<b>Summary</b>	使用者可以登入系統。
<b>Actors</b>	All User。
<b>Preconditions</b>	連上 Internet。
<b>Description</b>	<ol style="list-style-type: none"> <li>1. 輸入系統網址。</li> <li>2. 進入系統登入頁面。</li> <li>3. 輸入帳號。</li> <li>4. 輸入密碼。</li> <li>5. 點選登入。</li> <li>6. 成功登入系統。</li> </ol>
<b>Extensions</b>	None
<b>Exceptions</b>	<ol style="list-style-type: none"> <li>1. 帳號不存在。</li> <li>2. 密碼錯誤。</li> </ol>
<b>Postconditions</b>	進入系統首頁。

<b>No</b>	EMRMS-UC02
<b>Use Case</b>	新增帳號。
<b>Summary</b>	管理者可以建立新帳號。
<b>Actors</b>	Administrator。
<b>Preconditions</b>	EMRMS-UC01
<b>Description</b>	<ol style="list-style-type: none"> <li>1. 進入帳號管理頁面</li> <li>2. 點選新增帳號</li> <li>3. 進入新增帳號頁面</li> <li>4. 輸入新增帳號資料</li> <li>5. 點選完成</li> <li>6. 帳號新增完畢</li> </ol>
<b>Extensions</b>	None
<b>Exceptions</b>	<ol style="list-style-type: none"> <li>1. 帳號重複</li> <li>2. 帳號長度不足</li> <li>3. 密碼長度不足</li> <li>4. 密碼內容無同時擁有數字及英文字</li> </ol>
<b>Postconditions</b>	回到帳號管理頁面。

<b>No</b>	EMRMS-UC03
<b>Use Case</b>	查詢帳號。
<b>Summary</b>	Administrator 可以查詢帳號資料。
<b>Actors</b>	Administrator。
<b>Preconditions</b>	UGMS-UC01
<b>Description</b>	<ol style="list-style-type: none"> <li>1. 進入帳號管理頁面。</li> <li>2. 輸入查詢帳號。</li> <li>3. 點選搜尋。</li> <li>4. 帳號搜尋完成。</li> </ol>
<b>Extensions</b>	None。
<b>Exceptions</b>	None。
<b>Postconditions</b>	回到帳號管理頁面。

<b>No</b>	EMRMS-UC04
<b>Use Case</b>	刪除帳號。
<b>Summary</b>	Administrator 可以刪除帳號。
<b>Actors</b>	Administrator。
<b>Preconditions</b>	UGMS-UC01
<b>Description</b>	<ol style="list-style-type: none"> <li>1. 進入帳號管理頁面。</li> <li>2. 勾選帳號。</li> <li>3. 點選刪除。</li> <li>4. 刪除前進行確認。</li> <li>5. 點選確定。</li> <li>6. 帳號刪除完成。</li> </ol>
<b>Extensions</b>	None。
<b>Exceptions</b>	None。
<b>Postconditions</b>	回到帳號管理頁面。

<b>No</b>	EMRMS-UC05
<b>Use Case</b>	修改個人資訊。
<b>Summary</b>	使用者可以修改本身的個人資料。
<b>Actors</b>	擁有帳號之使用者。
<b>Preconditions</b>	UGMS-UC01
<b>Description</b>	<ol style="list-style-type: none"> <li>1. 進入帳號管理頁面。</li> <li>2. 點選修改個人資料。</li> <li>3. 進入修改帳號資料頁面。</li> <li>4. 進行個人資料修改。</li> <li>5. 點選完成。</li> <li>6. 個人資料修改完成。</li> </ol>
<b>Extensions</b>	None。
<b>Exceptions</b>	None。
<b>Postconditions</b>	回到帳號管理頁面。

<b>No</b>	EMRMS-UC06
<b>Use Case</b>	查詢就診紀錄。
<b>Summary</b>	使用者可以查詢相關就診紀錄。
<b>Actors</b>	所有使用者。
<b>Preconditions</b>	UGMS-UC01
<b>Description</b>	<ol style="list-style-type: none"> <li>1. 進入醫療資訊管理介面</li> <li>2. 點選就診紀錄頁面。</li> <li>3. 輸入病歷編號。</li> <li>4. 點選查詢。</li> <li>5. 查詢紀錄完成。</li> </ol>
<b>Extensions</b>	None。
<b>Exceptions</b>	None。
<b>Postconditions</b>	顯示就診紀錄。

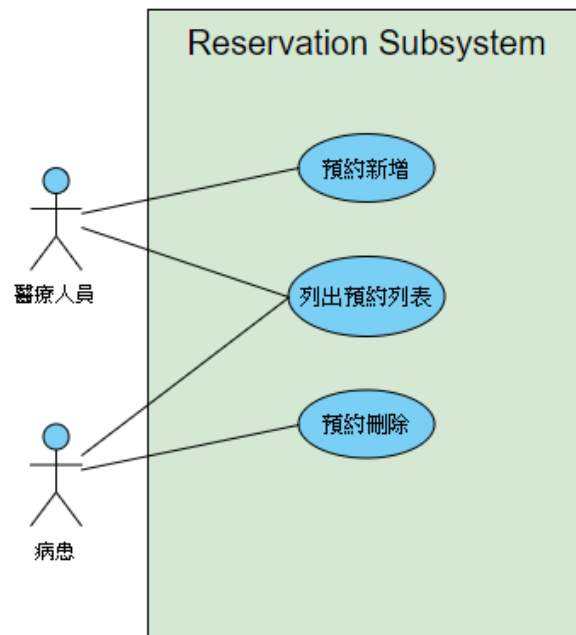


<b>No</b>	EMRMS-UC07
<b>Use Case</b>	查詢個人病歷。
<b>Summary</b>	使用者可以查詢相關就診報告。
<b>Actors</b>	病患。
<b>Preconditions</b>	UGMS-UC01
<b>Description</b>	1.進入醫療資訊管理介面 2.點選就診報告頁面。 3.輸入病歷編號。 4.點選查詢。 5.查詢紀錄完成。
<b>Extensions</b>	None。
<b>Exceptions</b>	None。
<b>Postconditions</b>	顯示就診報告。

<b>No</b>	EMRMS-UC08
<b>Use Case</b>	新增就診報告。
<b>Summary</b>	醫療人員可以新增相關就診報告。
<b>Actors</b>	醫療人員。
<b>Preconditions</b>	UGMS-UC01
<b>Description</b>	1.進入醫療資訊管理介面 2.點選就診報告頁面。 3.新增病歷資料。 4.點選完成。 5.新增報告完成。
<b>Extensions</b>	None。
<b>Exceptions</b>	4a. 病歷資料不完整。
<b>Postconditions</b>	回到就診報告頁面。

<b>No</b>	EMRMS-UC09
<b>Use Case</b>	查詢就診報告。
<b>Summary</b>	醫療人員可以查詢相關就診報告。
<b>Actors</b>	醫療人員。
<b>Preconditions</b>	UGMS-UC01
<b>Description</b>	<ol style="list-style-type: none"> <li>1.進入醫療資訊管理介面</li> <li>2.點選就診報告頁面。</li> <li>3.輸入病人身份證字號。</li> <li>4.點選完成。</li> <li>5.查詢報告完成。</li> </ol>
<b>Extensions</b>	None。
<b>Exceptions</b>	None。
<b>Postconditions</b>	回到就診報告頁面，顯示就診報告。

● Reservation Subsystem



<b>No</b>	RS-UC-01
<b>Use Case</b>	預約新增
<b>Summary</b>	新增一個新的預約
<b>Actors</b>	病患
<b>Preconditions</b>	病患必須擁有可登入此系統的帳號、密碼並且已登入
<b>Description</b>	1. 病患點選新增預約按鈕 2. 病患輸入預約科別、醫院、預約科別等...。 3. 病患點選確定按鈕 4. 將欲新增之預約資訊更新至 FHIR server。 5. 系統回到預約畫面，並顯示已新增之預約資料
<b>Extensions</b>	5a. 自動更新預約列表頁面
<b>Exceptions</b>	2a 病患未填入預約所需之必要內容。 4a 更新至 FHIR server 失敗
<b>Postconditions</b>	FHIR server 上新增一筆預約資料。

<b>No</b>	RS -UC-02
<b>Use Case</b>	刪除預約
<b>Summary</b>	刪除一個存在之預約
<b>Actors</b>	病患
<b>Preconditions</b>	病患必須擁有可登入此系統的帳號、密碼並且已登入，並且預約頁面上已有預約存在
<b>Description</b>	1.病患點選欲刪除預約按鈕 2.專案管理者點選確定按鈕 3.將欲刪除之預約資訊更新至 FHIR server 4.系統回到看板畫面，並顯示刪除預約後之預約頁面
<b>Extensions</b>	1a. 列出目前所有預約 5a. 自動更新預約列表頁面
<b>Exceptions</b>	4a 更新至 FHIR server 失敗
<b>Postconditions</b>	FHIR server 上刪除一筆任務資料

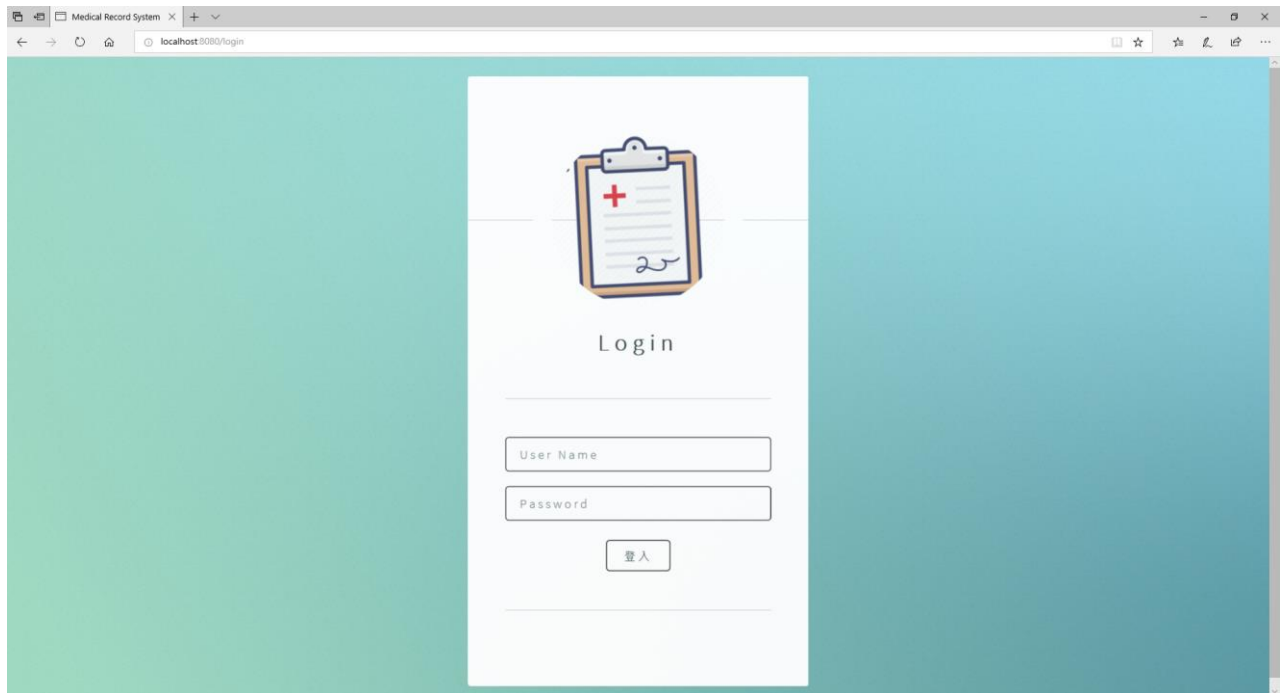
<b>No</b>	RS -UC-03
<b>Use Case</b>	列出預約列表
<b>Summary</b>	頁面顯示所有預約
<b>Actors</b>	一般使用者
<b>Preconditions</b>	使用者必須擁有可登入此系統的帳號、密碼並且已登入，並且 FHIR server 上已有預約資料存在
<b>Description</b>	1.使用者進入醫療資訊管理介面 2.使用者點選預約看診頁面 3.系統搜尋與預約者相關之預約紀錄。 4.將結果顯示於頁面上。 5.列表完成
<b>Extensions</b>	None
<b>Exceptions</b>	3a 與 FHIR server 連線失敗

Postconditions

頁面列出預約資訊

### 3.1.2 User Interfaces Analysis

#### 1. 登入介面



#### 2. 管理者帳號管理介面



### 3.個人資訊介面

[回首頁](#)登出/病患

個人資料

看診紀錄

診斷報告

預約看診

個人資料

姓名	性別	地址	生日	身分證字號
鮑博	FEMALE	新北市土城區三寶路87號8樓	2019/02/28	F111111111

修改個資

修改密碼

### 4.看診紀錄頁面

[回首頁](#)登出/病患

個人資料

看診紀錄

診斷報告

預約看診

看診紀錄

就醫日期	醫事機構	看診類別	看診醫師
2019-12-25	臺大醫院	內科	余光中
2019-12-25	臺大醫院	內科	余光中

## 5. 診斷報告頁面

[回首頁](#)
[登出/病患](#)

[個人資料](#)
[看診紀錄](#)
[診斷報告](#)
[預約看診](#)

### 診斷報告

就醫日期	醫事機構	看診醫師	疾病分類名稱	藥品代碼	藥品名稱	給藥日數	藥品用量	醫生囑咐
2019/12/25	臺大醫院	余光中	失眠	0483	STILNOX F.C. TAB. 10MG	3	一次一錠	避免不良的生活作息
2019/12/25	臺大醫院	余光中	失眠	0483	STILNOX F.C. TAB. 10MG	3	一次一錠	避免不良生活習慣

## 6. 預約看診介面

[回首頁](#)
[登出/病患](#)

[個人資料](#)
[看診紀錄](#)
[診斷報告](#)
[預約看診](#)

### 預約看診

身分證字號	看診日期	病例編號	醫事機構	看診科別	看診醫師	看診序號
F123456789	2020/01/15	584259	臺大醫院	內科	余光中	1
F123456789	2020/02/20	584260	臺大醫院	內科	余光中	1
F123456789	2020/01/15	584261	臺大醫院	內科	余光中	2
F123456789	2020/02/12	584332	臺大醫院	內科	余光中	1

新增預約

## 7.新增預約介面

預約看診						
身分證字號	看診日期	病例編號	醫事機構	看診科別	看診醫師	看診序號
F123456789	2020/01/15			內科	余光中	1
F123456789	2020/02/20			內科	余光中	1
F123456789	2020/01/15			內科	余光中	2
F123456789	2020/02/12			內科	余光中	1

新增預約

新增預約

姓名：

身分證字號：

預約日期：

年 / 月 / 日

醫事機構：

臺大醫院

看診科別：

內科

醫生：

余光中

取消

新增

可新增一筆預約紀錄於 FHIR server 上。

## 8.修改個人資訊頁面

個人資料				
姓名	性別	地址	生日	身分證字號
王小明	MALE	新北市板橋區中山路88號	1993/10/12	F123456789

修改個人資料

修改密碼

修改帳戶

姓名：

性別：

男

地址：

生日：

年 / 月 / 日

取消

修改

可修改個人資訊、密碼



## 9.新增診斷報告頁面

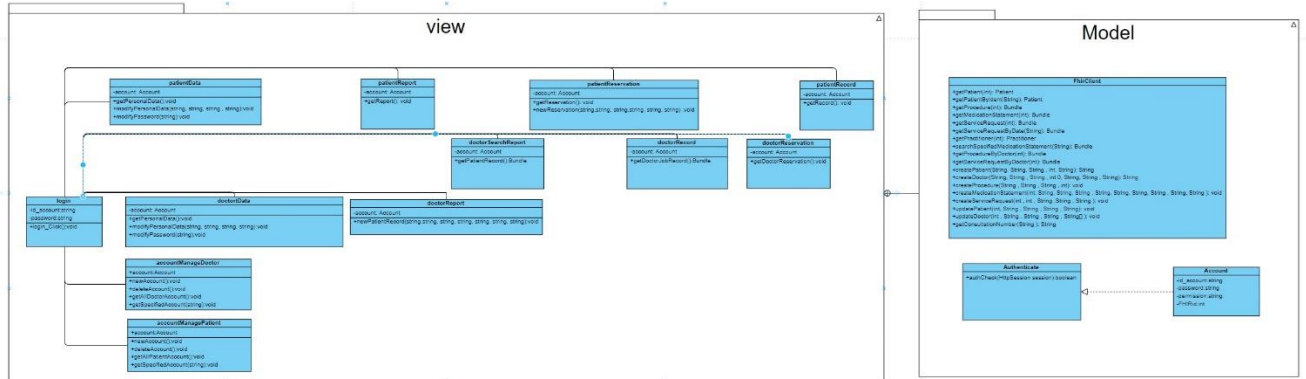
個人資料	看診紀錄	診斷報告	查詢病歷	預約看診
<p>病患姓名： <input type="text"/></p> <p>病患身分證字號： <input type="text"/></p> <p>疾病名稱： <input type="text"/></p> <p>藥品代碼： <input type="text"/></p> <p>藥品名稱： <input type="text"/></p> <p>給藥日數： <input type="text"/></p> <p>藥品用量： <input type="text"/></p> <p>醫生囑咐： <input type="text"/></p> <p><input type="button" value="新增"/></p>				

## 10.查詢診斷報告頁面

身分證字號：

醫生診斷報告								
病歷編號	病患姓名	病患身分證字號	疾病名稱	藥品代碼	藥品名稱	給藥日數	藥品用量	醫生囑咐

### 3.1.3 Static Model



Class No.	
Class Name	Login
Responsibility	負責處理使用者登入系統的動作
Collaboration	adminMain、patientMain、doctorMain
Related Subsystem	EMRMS Module

Class No.	
Class Name	adminMain
Responsibility	識別是否為管理員
Collaboration	None
Related Subsystem	EMRMS Module

Class No.	
Class Name	patientMain
Responsibility	識別是否為病患
Collaboration	None
Related Subsystem	EMRMS Module

Class No.	
Class Name	AccountManageDoctor
Responsibility	負責提供一個頁面來讓管理者管理醫生帳戶資訊
Collaboration	None
Related Subsystem	EMRMS Module

<b>Class No.</b>	
<b>Class Name</b>	AccountManagePatient
<b>Responsibility</b>	負責提供一個頁面來讓管理者管理病患帳戶資訊
<b>Collaboration</b>	None
<b>Related Subsystem</b>	EMRMS Module

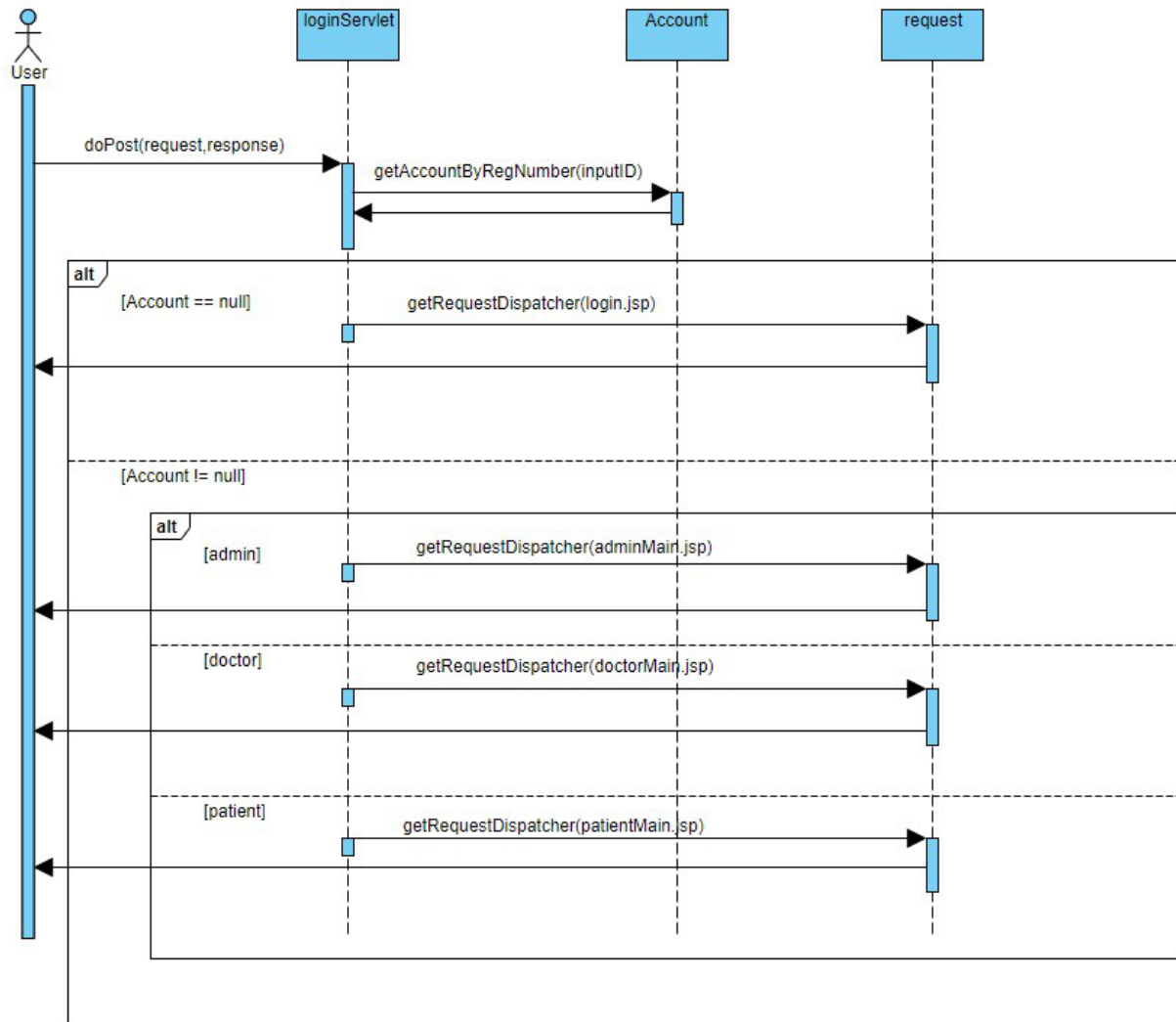
<b>Class No.</b>	
<b>Class Name</b>	FhirClient
<b>Responsibility</b>	負責處理系統與FHIRserver間的溝通
<b>Collaboration</b>	None
<b>Related Subsystem</b>	EMRMS Module

<b>Class No.</b>	
<b>Class Name</b>	doctorMain
<b>Responsibility</b>	識別是否為醫生
<b>Collaboration</b>	None
<b>Related Subsystem</b>	EMRMS Module

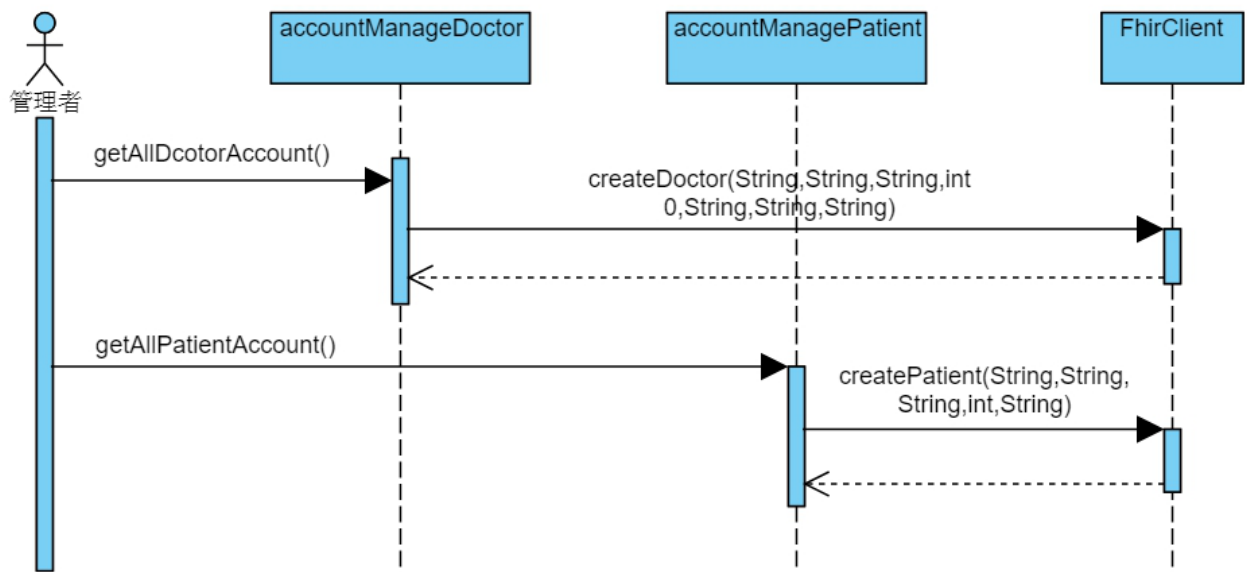
### 3.1.4 Dynamic Models

#### Electronic Medical Record Management Subsystem

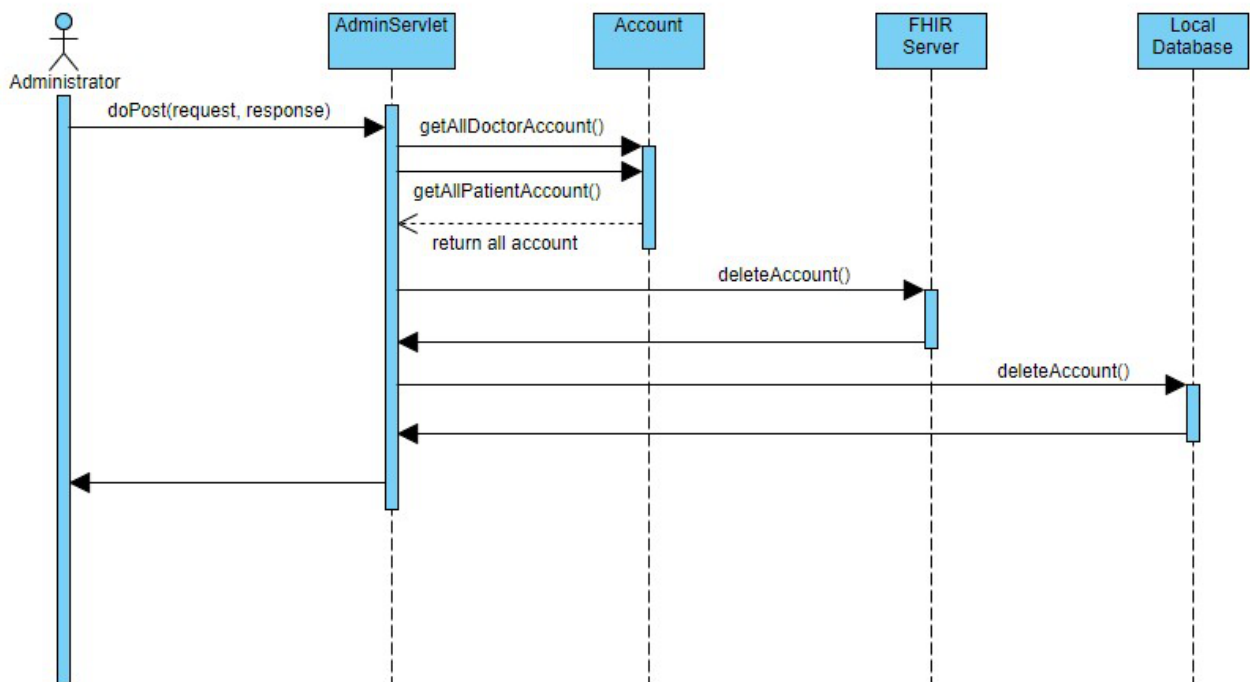
使用者登入：



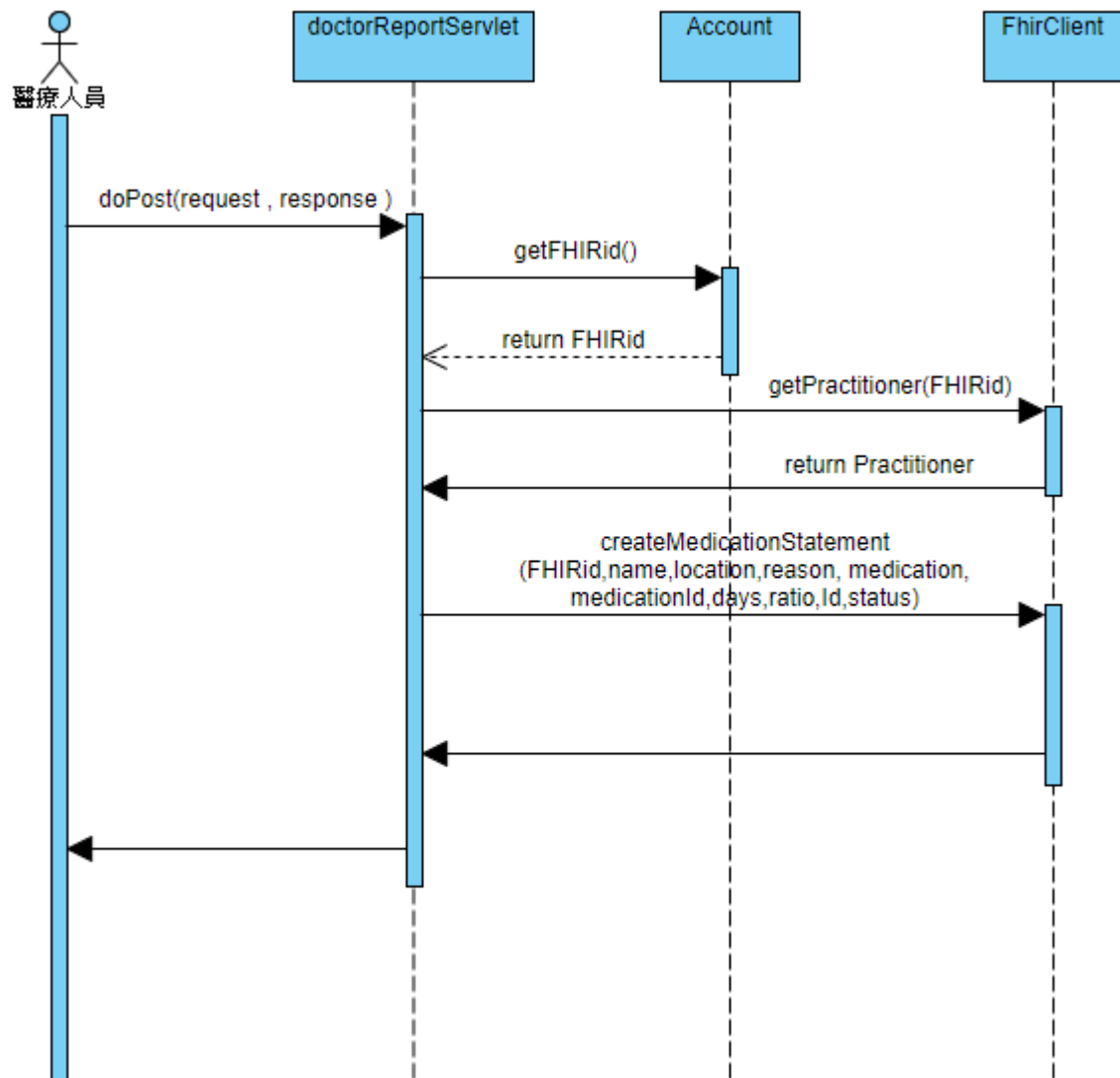
新增帳號：



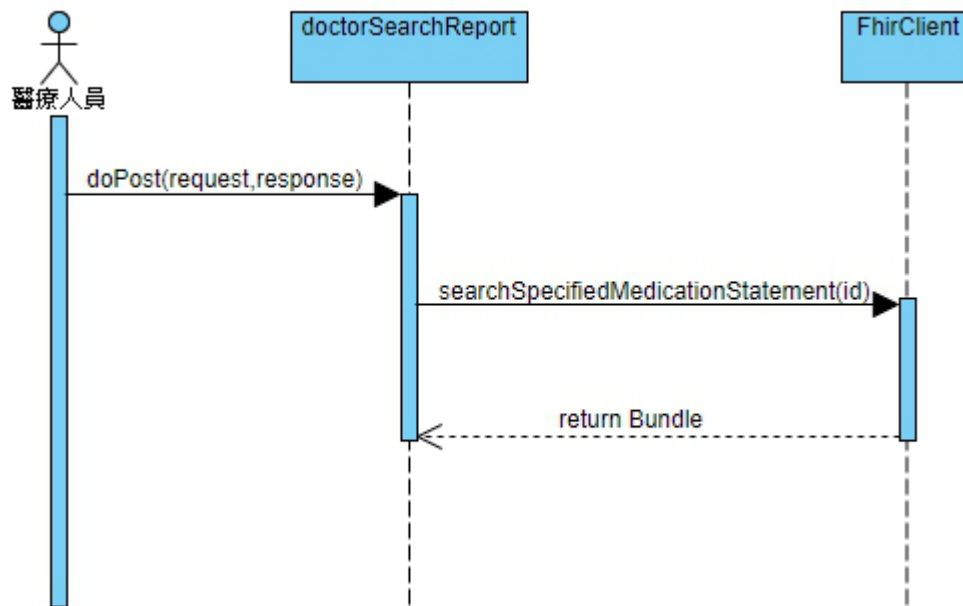
刪除帳號：



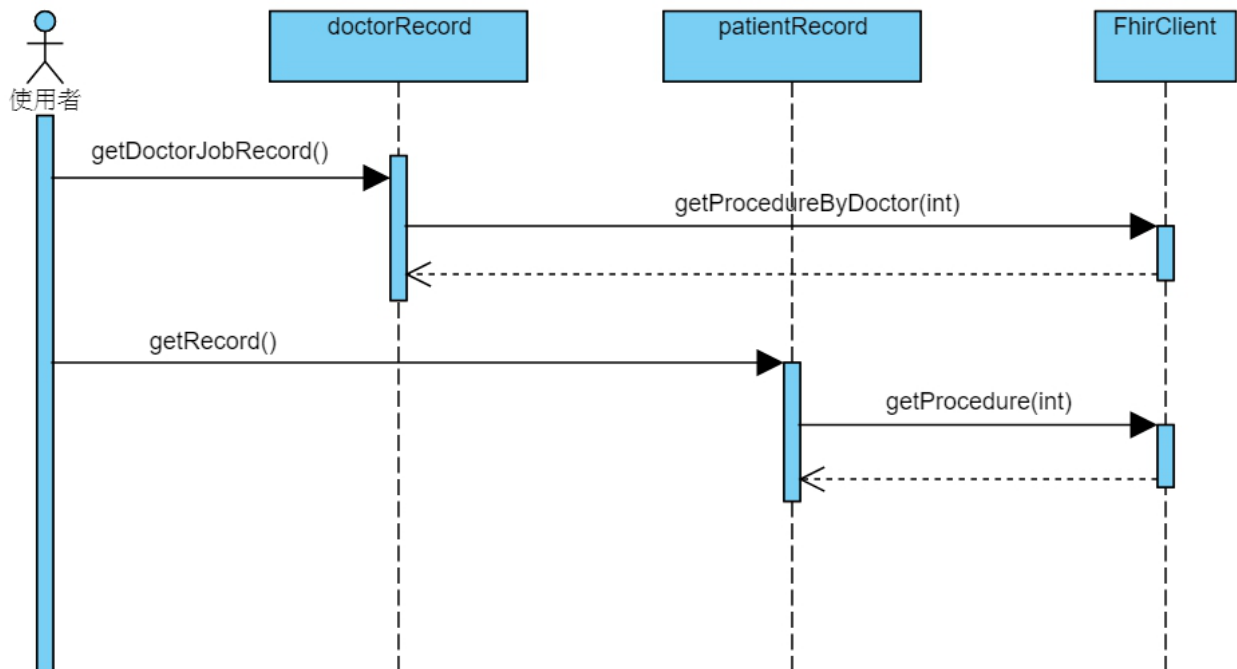
新增就診報告:



## 查詢就診報告

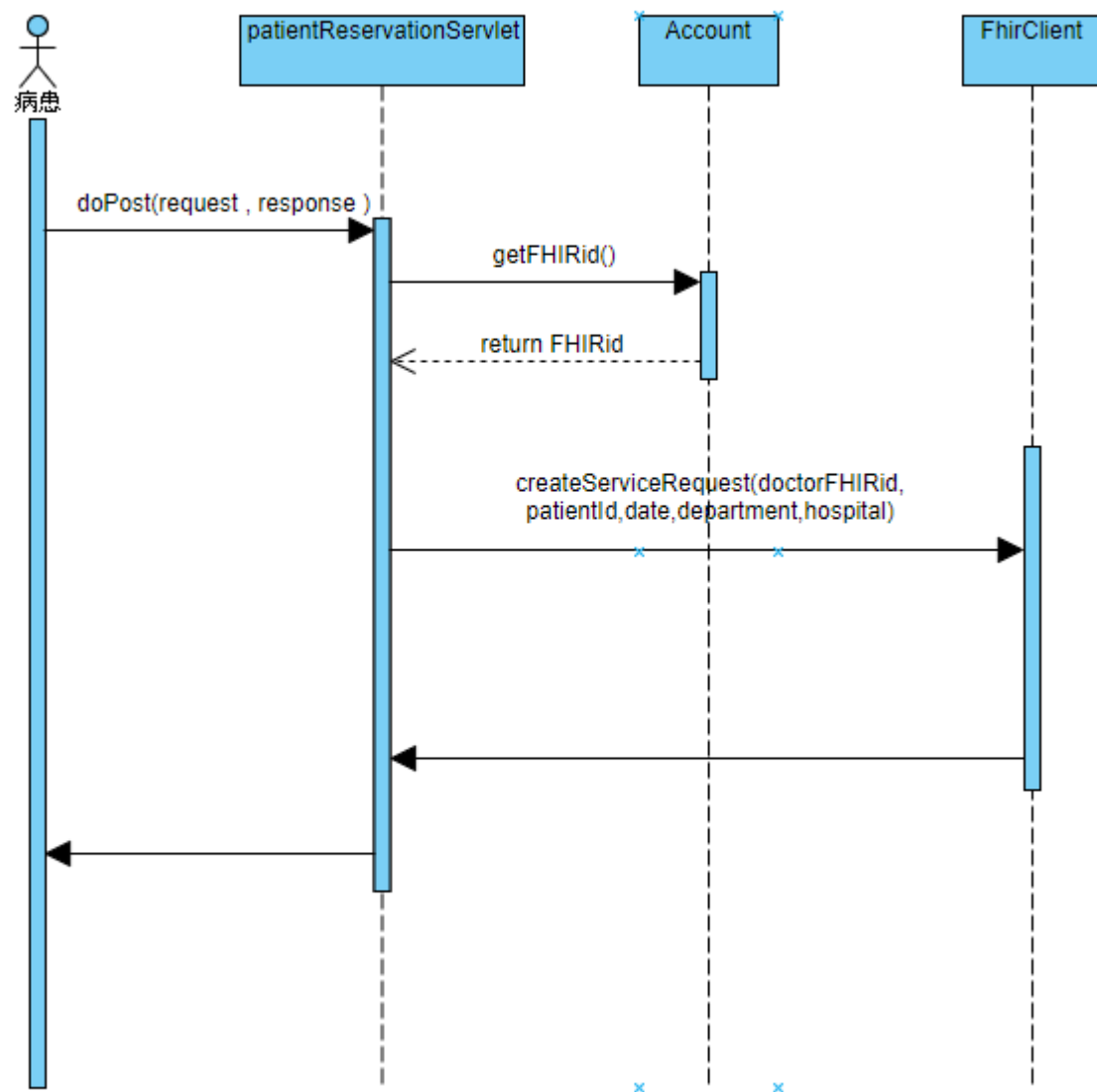


## 查詢就診紀錄



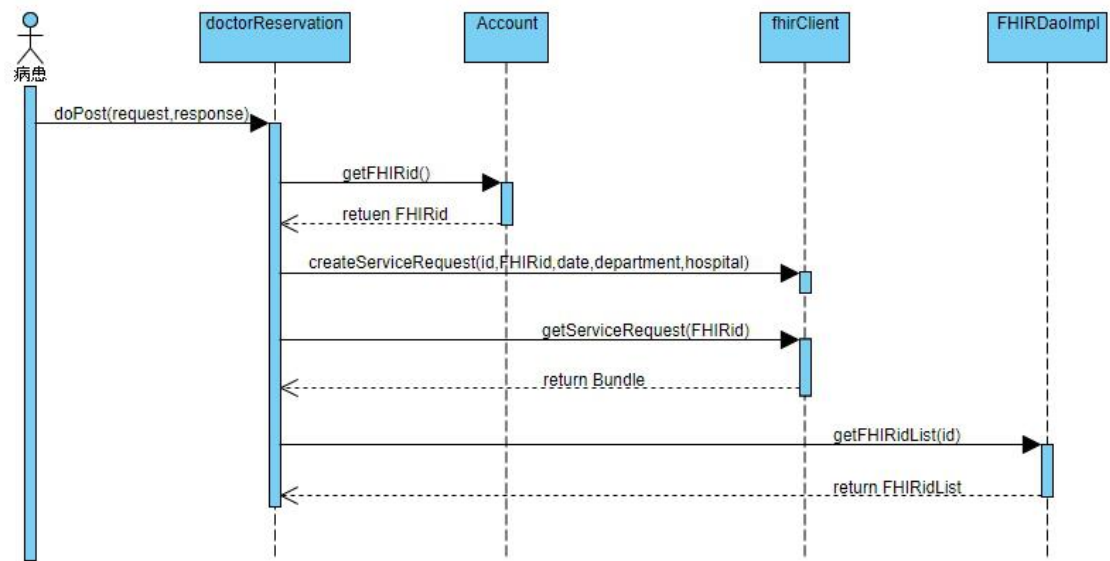
## Reservation Subsystem

預約新增:





## 列出預約列表



### 3.2 Subsystem Internal/External Interface Requirement and Design

#### 3.2.1 Internal Interface Requirements

需求編號	優先順序	需求描述
MRMS-I-01	1	EMRMS 與資料庫間必須能傳送與接收使用者資料

#### 3.2.2 External Interface Requirements

需求編號	優先順序	需求描述
MRMS -E-01	1	EMRMS 提供 CUI
MRMS -E-02	1	EMRMS 允許使用者登入，更改帳號資料
MRMS -E-03	1	EMRMS 提供透過網路來傳送資料
MRMS -E-04	1	EMRMS、RS 提供 GUI 供使用者操作
MRMS -E-05	1	EMRMS 允許管理者新增、刪除、查詢帳號
MRMS -E-06	1	EMRMS 允許使用者修改個人資料、密碼
MRMS -E-07	1	EMRMS 允許使用者查詢看診紀錄
MRMS -E-08	1	EMRMS 允許 醫療人員查詢、新增就診報告
MRMS -E-09	1	RS 允許 醫療人員查詢預約看診排程
MRMS -E-10	1	RS 允許 病患新增預約看診項目
MRMS -E-11	1	RS 允許 病患新增預約看診項目

## Glossary

Database	記錄資料的地方，提供新增、刪除、更新記錄的功能。
JAVA	Java 程式語言的風格十分接近 C++ 語言。繼承了 C++ 語言物件導向技術的核心，Java 捨棄了 C++ 語言中容易引起錯誤的指標、運算符過載、多重繼承等特性，增加了垃圾回收器功能用於回收不再被參照的物件所佔據的內部記憶體空間。Java 伴隨著網際網路的迅猛發展而發展，逐漸成為重要的網路程式語言。
MySQL	MySQL 是一個開放原始碼的關聯式資料庫管理系統，由於性能高、成本低、可靠性好，已經成為最流行的開源資料庫，被廣泛地應用在 Internet 上的中小型網站中。隨著 MySQL 的不斷成熟，它也逐漸用於更多大規模網站和應用。
SQL	Structured Query Language，關連式查詢語言，用來定義資料庫的結構，或是利用 SQL 對資料庫執行一些查詢、增加、刪除、更新記錄，到目前為止，SQL 是第一個，也是唯一的標準資料庫語言，受到廣泛的接受。
UI	用戶介面(User Interface)是介於使用者與硬體而設計彼此之間互動溝通相關軟體，目的在使得使用者能夠方便有效率地去操作硬體以達成雙向之互動，完成所希望借助硬體完成之工作，用戶介面定義廣泛，包含了人機互動與圖形使用者介面，凡參與人類與機械的信息交流的領域都存在著用戶介面。
JSP	全稱是 Java Server Page，JSP 的面向是處理 View 的部份，也就是實際把動態產生的內容呈現給客戶端。這樣的做法是我們能夠把邏輯和顯示層分開，達到更容易維護。

## References

<https://www.hl7.org> - FHIR v4.0.0

<https://ez-fhir.netlify.com/#/search-fhir-resoure> - HAPI FHIR 公開測試伺服器

## Appendix

### A. Traceability Matrix Use Cases v.s. Classes

	login	patient Data	patient Report	patient Reservat ion	patient Record	doctor Data	doctor Report	doctor Search Report	doctor Record	doctor Reservat ion	account Manage Doctor	account Manage Patient
EMRMS-UC01	O											
EMRMS-UC02											O	O
EMRMS-UC03											O	O
EMRMS-UC04											O	O
EMRMS-UC05		O				O						
EMRMS-UC06					O				O			
EMRMS-UC07			O									
EMRMS-UC08							O					
EMRMS-UC09								O				
RS-UC-01				O								
RS-UC-02				O								
RS-UC-03				O						O		

### B. Traceability Matrix Classes v.s. Classes

	login	patient Data	patient Report	patient Reservati on	patient Record	doctor Data	doctor Report	doctor Search Report	doctor Record	doctor Reservati on	account Manage Doctor	account Manage Patient
login	O	O	O	O	O	O	O	O	O	O	O	O
patient Data	O	O										
patient Report	O		O									
patient Reservation	O			O								
patient Record	O				O							
doctor Data	O					O						
doctor Report	O						O					

doctor Search Report	O							O				
doctor Record	O								O			
doctor Reservation	O									O		
account Manage Doctor	O										O	
account Manage Patient	O											O