

Software Requirements Specification

Virtual Room Reservation Assistant

<version 1.0>

NOVEMBER 10 , 2021

Group:25

Group Members:

B10704101 王柏鈞

B10815043 鄭鴻翔

B10815052 洪翊方

B10832030 陳政亮(NO contribution to this assignment)

**Submitted in partial fulfillment
Of the requirements of
CS3025301 Software Engineering**

目錄:

1.0 Introduction	4
1.1 purpose	4
1.2 scope of project	4
1.3 Glossary	4
1.4 References	4
1.5 overview	5
2.0 overall Description	5
2.1 System Environment	5
2.2 functional requirement definition	6
2.2.1 未註冊人使用情況	6
2.2.2 使用者	7
2.2.3 發起人	10
2.3 User Interface Specification	12
2.3.1 登入介面	12
2.3.2 註冊介面	13
2.3.3 首頁	14
2.3.4 選擇日期	15
2.3.5 選擇時段介面	16
2.3.6 輸入會議資訊介面	17
2.3.7 輸入受邀請人介面	18
2.3.8 已預約 \ 已參與介面	19
2.3.9 會議室資訊介面	19
2.4 Non-Functional Requirements	20
3.0 Requirements Specification	20
3.1 External Interface Requirement	20
3.2 Function Requirement	20
3.2.1 註冊	20
3.2.2 登入	21
3.2.3 更新	22
3.2.4 查看預約的會議室	22
3.2.5 查看參與的會議室	22
3.2.6 自動更新	23
3.2.7 預約	23
3.2.8 取消會議	24
3.2.9 修改會議資訊	25
3.3 Detailed Non-Functional Requirement	25
3.3.1 資料庫上的資料結構	25

3.3.2 使用者 entity	26
3.3.3 會議室 entity	27
4.0 Others	27
4.1 system requirement	27

1.0 Introduction

1.1 purpose

本文件的目的是詳細介紹與解釋本次期末專案"Virtual Room Reservation Assistant"
內容會提到: 開發環境、開發流程、使用方式、特點、使用者介面、以及預期成果等。
本文件適用於本專案開發者與相關教學單位人士。

1.2 scope of project

此軟件系統設定為"私人企業與小型機關"的會議室預定輔助系統，設計過程主要考量部分為"已註冊使用者"的便利性與可讀性，透過系統提供的工具以及較高可讀性的 UI 建立一個與本地資料庫互動的應用程式，並且連結使用者的 e-mail 提供個人的通知。

主要為避免發生兩個或以上同時執行的會議發生於同一空間，將各個團隊的時間或地點錯開，避免所有會議集中到特定時段，以此讓辦公室空間與開放時間得到最大效率的使用。

具體解釋則為，使用者透過已連結到外部資料庫的應用程式，決定發起會議的地點與時間以及邀請的對象，而會議新增或改動影響到的關係人都會收到從應用程式端，發出的格式化電子郵件通知。

1.3 Glossary

未註冊人: 未經過註冊手續的人。

使用者: 已經過註冊手續並成功的人。

受邀請人: 必須是使用者。受邀請使用會議室的人。

發起人: 必須是使用者。成功申請會議室使用的人。

參與人: 必須是使用者。受發起人邀請，使用會議室的人。

資料庫: 存放帳密資訊、會議室資訊資料儲存的外部系統。

1.4 References

Software Requirements Specification Version 1.0 <> April 15, 2004

1.5 overview

在下一章節會詳細解釋產品功能，主要提供給使用者查閱，列出使用本產品的方式，以及主要功能的使用流程，並列出非功能性需求，同時替下一章節建立產品規範。

第三章節則根據前面規範詳敘開發環境、使用語言、實現流程、執行環境和方式等，主要提供給本專案開發者參考，並且將需求功能分段解釋。

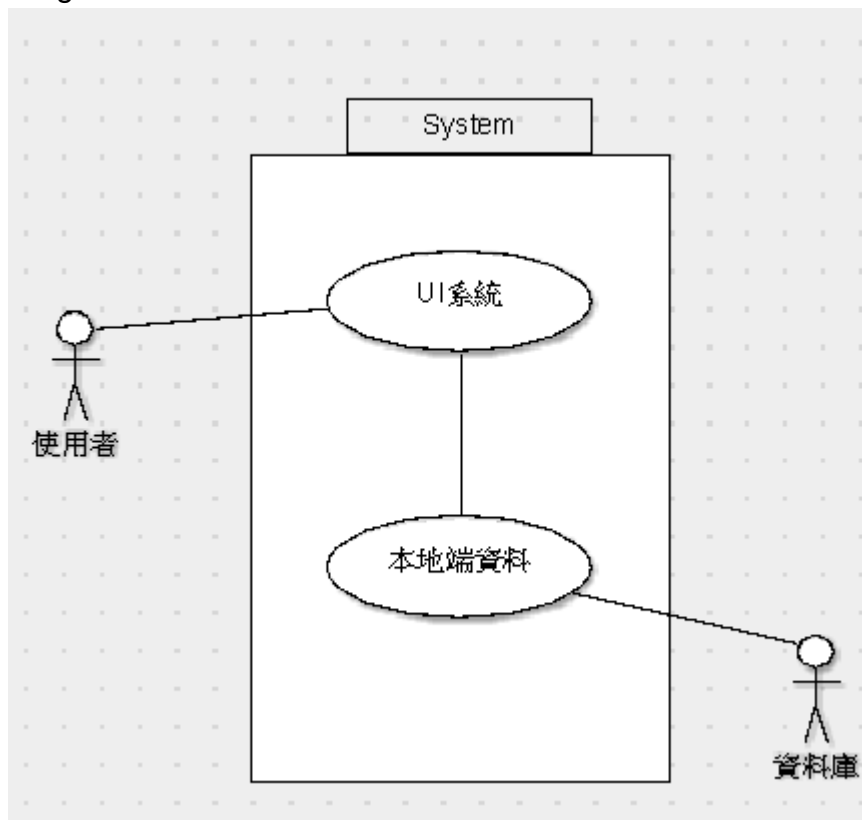
各章節皆描述相同產品，但針對不同受眾，採用不同的方式表達內容。

2.0 overall Description

2.1 System Environment

此系統包含 UI 系統，本地端資料，及外部資料庫。由 UI 系統提供使用者操作流程。通過一系列操作流程，給予、獲取資料，同時確認輸入資料的正確性，最後提供給外部資料庫。本地端資料透過"更新"來獲取外部資料庫資料。

Diagram



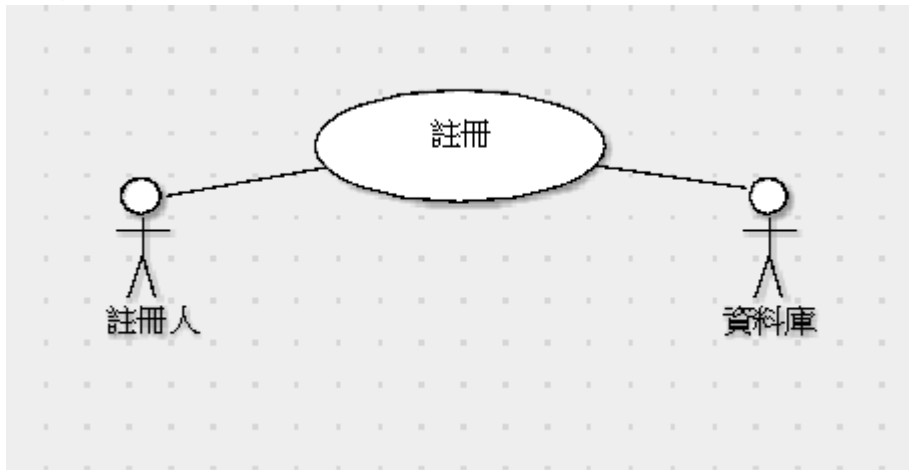
2.2 functional requirement definition

這部分將會一一描述所有軟體使用者的應用情況，並確切定義出每個步驟。確認每個身份能執行的行為。

2.2.1 未註冊人使用情況

Use case: 註冊

Diagram



Brief Description

填入名稱、密碼、Email，資料經過確認手續，確認無誤後將資料存入資料庫中。

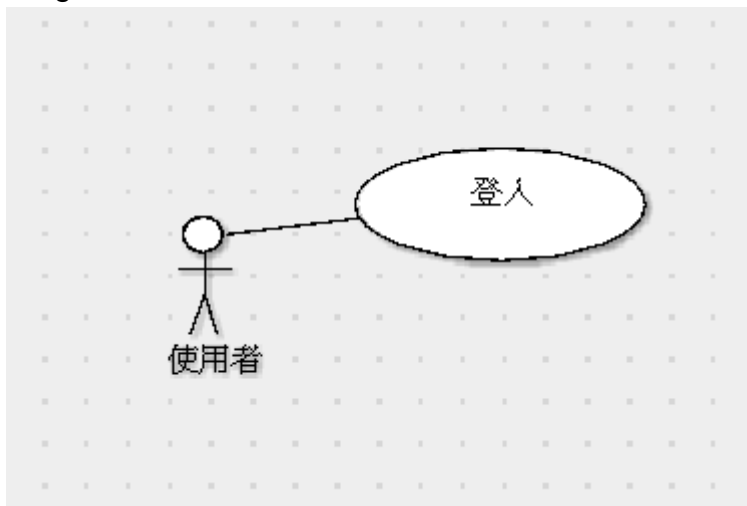
Initial Step-By-Step Description

1. 點選"註冊"按鈕
2. 未註冊人在名稱、密碼、Email 相應欄位輸入資料
3. 點選"確認"按鈕
4. 驗證使用者 email，失敗則回到註冊畫面
5. 搜尋資料庫，找尋是否有重複 Email、重複名稱
6. Email 驗證通過、沒有重複資料則存入資料庫，反之，顯示註冊失敗提示，回到註冊畫面

2.2.2 使用者

Use case: 登入

Diagram



Brief Description

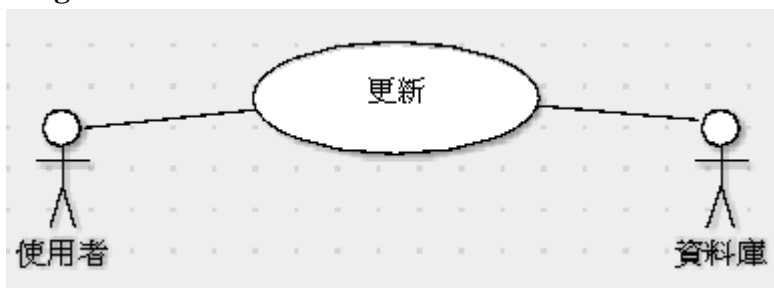
填入名稱、密碼，經過確認手續，確認無誤後以註冊身份進入程式。

Initial Step-By-Step Description

1. 使用者在名稱、密碼相應欄位輸入資料
2. 點選"登入"按鈕
3. 搜尋資料庫，找尋是否有該名稱；核對密碼是否正確
4. 資料正確進入程式，反之，顯示登入失敗提示

Use case: 更新

Diagram



Brief Description

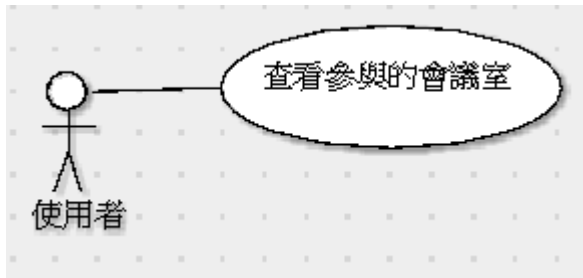
使用者更新會議室資訊，。

Initial Step-By-Step Description

1. 點選"更新"按鈕
2. 載入資料

Use case: 查看參與的會議室

Diagram



Brief Description

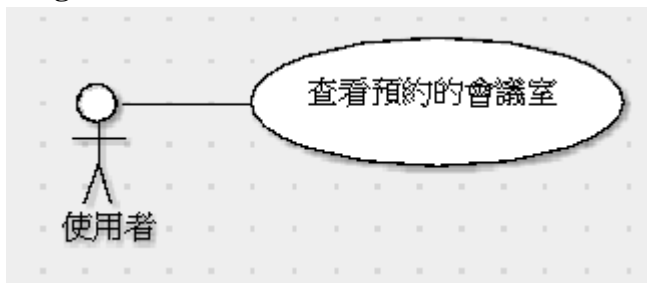
使用者查看目前參與的會議室，包含自己發起的。

Initial Step-By-Step Description

1. 點選"已參與"按鈕
2. 載入資料
3. UI 顯示目前參與的會議室

Use case: 查看預約的會議室

Diagram



Brief Description

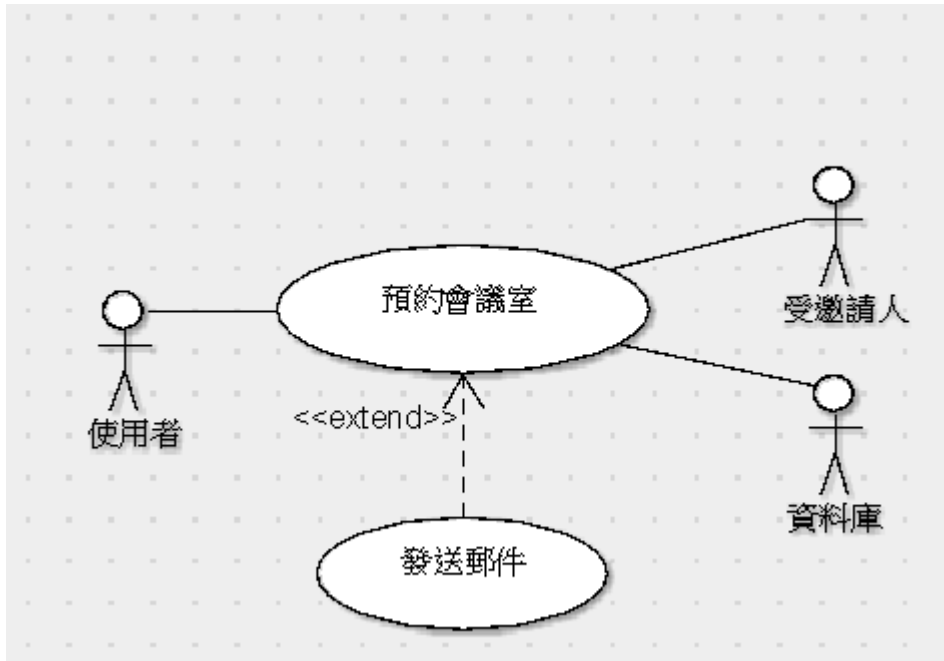
使用者查看目前預約的會議室

Initial Step-By-Step Description

1. 點選"已預約"按鈕
2. 載入資料
3. UI 顯示目前預約的會議室

Use case: 預約會議室

Diagram



Brief Description

使用者依據自己的需求，借用會議室一段時間。

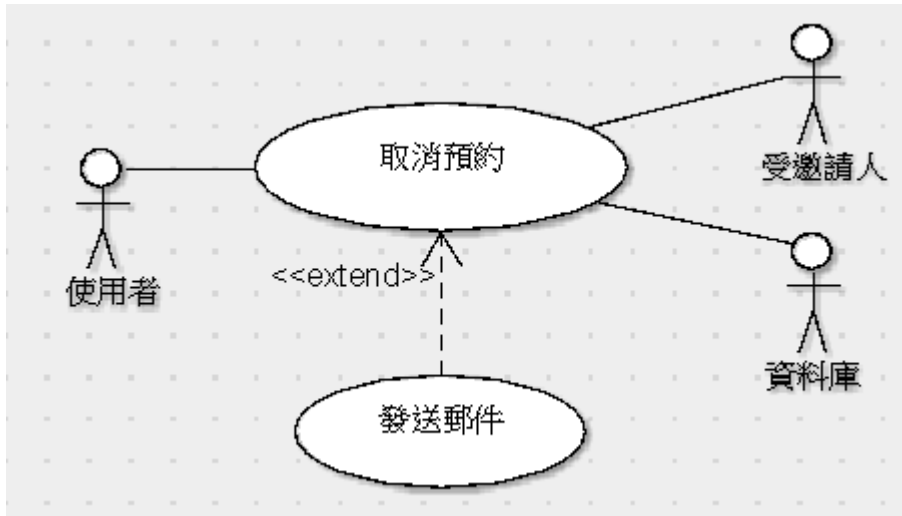
Initial Step-By-Step Description

1. 選擇會議室
2. 選擇日期
3. 選擇時段
4. 輸入會議名稱、會議內容
5. 輸入受邀請人
6. 點選"預約"按鈕
7. 檢查會議名稱、會議內容是否為空白，若有空白項，回到步驟 4
8. 檢索資料，若沒有衝突則申請成功，反之，顯示申請失敗提示，回到步驟 1
9. 申請成功則對所有受邀請人送出邀請郵件

2.2.3 發起人

Use case: 取消預約

Diagram



Brief Description

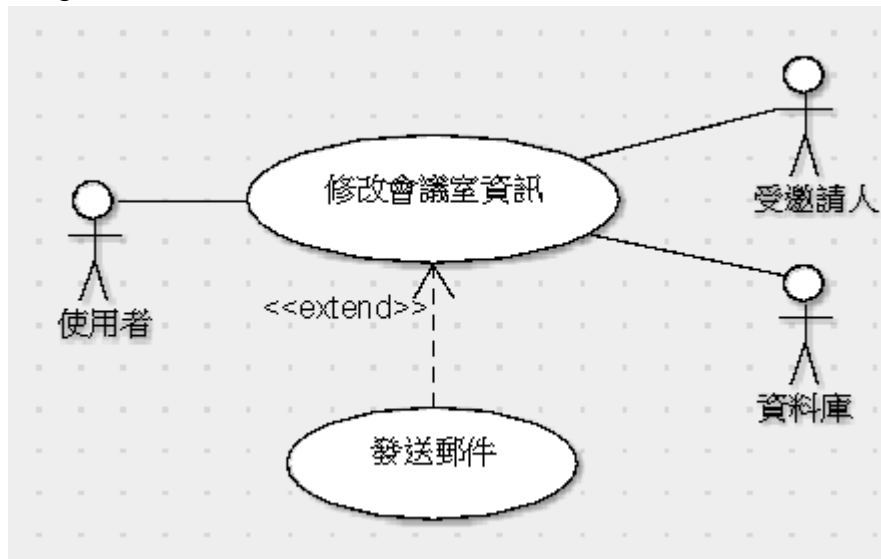
發起人取消已預約的會議室

Initial Step-By-Step Description

1. 選擇已預約的會議室
2. 按下"取消預約按鈕"
3. 點選"確認"按鈕
4. 更新資料庫
5. 發送電子郵件給受邀請人

Use case: 修改會議資訊

Diagram



Brief Description

發起人修改會議日期、會議資訊、受邀請人

Initial Step-By-Step Description

1. 發起人點選想修改的欄位
2. 輸入資訊
3. 點選"確認"按鈕
4. 檢查資料庫，確認修改資料是否可用
5. 更新資料庫
6. 發送 Email 給受邀請人

2.3 User Interface Specification

此章節主要是提供 UI 設計的雛形、UI 操作定義，方便使用者迅速了解該軟體的操作。由於設計與實作軟體不同，UI 顯示上可能存在稍微差異。

2.3.1 登入介面



The image shows a login interface mockup. It features a light gray header bar with the text "登入介面" (Login Interface) on the left and a "結束" (End) button on the right. The main content area is white. It contains two labels, "名稱" (Name) and "密碼" (Password), each followed by a "Text Box" input field. Below these fields is a gray "登入" (Login) button. In the bottom right corner, there is a gray "註冊" (Register) button.

- 1)結束按鈕: 按下後會跳出確認\取消介面，按下確認即關閉程式，之後的亦同。
- 2)名稱\密碼 **Text Box**: 使用者可輸入相應資訊
- 3)登入按鈕: 執行"登入"動作
- 4)註冊按鈕: 轉移至註冊介面

2.3.2 註冊介面



The screenshot displays a registration form within a window titled "註冊介面" (Registration Interface). The window has a light gray header bar with the title on the left and a "結束" (End) button on the right. The main content area is white and contains three vertically stacked input fields. Each field is preceded by a label: "名稱" (Name), "密碼" (Password), and "Email". Each input field is a rectangular box with a thin black border and the placeholder text "Text Box". Below these fields, centered horizontally, is a gray button with the white text "確認" (Confirm).

1)名稱 \ 密碼 \Email Text Box: 使用者可輸入相應資訊

2)確認按鈕: 執行"註冊"動作

2.3.3 首頁



- 1)會議室按鈕: 移至選擇日期介面
- 2)已預約按鈕: 執行"查看預約的會議室"動作；移至已預約介面
- 3)已參與按鈕: 執行"查看參與的會議室"動作；移至已參與介面
- 4)更新按鈕: 執行更新

2.3.4 選擇日期

選擇日期						結束
<		March 2017				>
Mo	Tu	We	Th	Fr	Sa	Su
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

上一步

- 1) < \ > 按鈕: 選擇 上 \ 下 月分
- 2) 日期按鈕: 選擇日期，移至選擇時段介面。
- 3) 上一步按鈕: 移至上一個使用的介面，之後亦同。

2.3.5 選擇時段介面

Time	Date	Action	Status
10	00	show available	<input checked="" type="checkbox"/> Selected
11	00		<input checked="" type="checkbox"/> Selected
12	00		<input checked="" type="checkbox"/> Selected
13	00		<input checked="" type="checkbox"/> Selected

上一步 下一步

- 1)可用顯示: 顯示該時段會議室可否借用
- 2)勾選: 點擊選擇借用時段，必須為連續時段
- 3)strollbar: 上下移動，顯示不同時段
- 4)下一步按鈕: 移至 輸入會議資訊介面

2.3.6 輸入會議資訊介面

輸入會議資訊

結束

會議名稱

會議內容

上一步

下一步

- 1)會議名稱 **Text Box**: 輸入會議名稱，不可空白
- 2)會議內容 **Text Box**: 輸入會議內容，不可空白
- 3)下一步: 移至輸入受邀請人介面

2.3.7 輸入受邀請人介面

輸入受邀請人

結束

Q

受邀請人1 X

受邀請人2 X

受邀請人3 X

上一步

預約

- 1)搜索: 輸入使用者名稱，按下 **Enter** 鍵加入
- 2)受邀請人顯示欄: 顯示目前已邀請的人員
- 3)X 按鈕: 刪除已邀請的人員
- 4)預約按鈕: 執行"預約"動作

2.3.8 已預約\已參與介面

已預約\參與		結束
<div>返回</div>	<div><input type="checkbox"/> 會議1</div> <div>date</div>	
	<div><input type="checkbox"/> 會議2</div> <div>date</div>	
	<div><input type="checkbox"/> 會議3</div> <div>date</div>	

- 1)會議按鈕: 以日期排序，移至會議室資訊介面。
- 2)返回按鈕: 移至首頁

2.3.9 會議室資訊介面

會議室資訊						結束
12 May 2016					鐘	
10	00	~	12	00	✎	
會議名稱			✎			
會議內容			受邀請人1 ✎			
			受邀請人2			
			受邀請人3			
返回			取消預約			

- 1)編輯按鈕: 按下後移至上述相應資訊編輯介面
- 2)取消預約按鈕: 執行"取消預約"動作
- 3)返回按鈕: 移至首頁

2.4 Non-Functional Requirements

會議室需求

1. 會議室租借衝突是由時間先後順序決定，如果遇到時間衝突則以隨機方式決定，以保持公平性。
2. 只有發起人才有修改會議資訊的權限，且發起人身分不得修改，防止租借人權益受到損害。
3. 隱私部分，只有參與人才能看到會議資訊，其餘情況只能看到會議室是否可用。

資料庫

1. 只有使用者才有存取、修改權
2. 連線、資料讀取速度由資料庫方決定

3.0 Requirements Specification

3.1 External Interface Requirement

此預約系統需要一個線上資料庫來儲存與提取會議室的相關資料。在身份資訊部分包含使用者名稱、信箱、密碼，在會議部分包含會議名稱、會議時段、會議內容、發起人名稱、參與人名稱。

主要的 **use case** 在身份方面有註冊和登入，都會回傳 **boolean** 判別是否成功。而在管理會議室內容方面，更新、自動更新、查詢參與的會議、查詢預約的會議、預約、取消、修改，皆會回傳結果讓本地端得以顯示變動後的資訊給使用者。

3.2 Function Requirement

3.2.1 註冊

Use case:註冊

Trigger:註冊按鈕

Precondition:在註冊介面

Step:

1. 傳入資料庫時 call 檢查信箱 function 回傳一個 boolean
 - 0->進入 step 2
 - 1->回傳("此信箱已被註冊"), 結束連線
2. 傳入資料庫時 call 檢查帳號 function 回傳一個 boolean
 - 0->回傳("帳號建立成功"), 並儲存內容至資料庫, 結束連線
 - 1->回傳("此帳號已被註冊"), 結束連線

Postcondition:回到登入介面 or 註冊介面

3.2.2 登入

Use case:登入

Trigger: 登入按鈕

Precondition:在登入介面

Step:

1. 傳入資料庫時 call 檢查帳號 function 比對資料庫中儲存的帳號, 回傳一個 boolean
 - 0->回傳("此帳號尚未註冊"), 結束連線
 - 1->step2
2. call 檢查密碼 function 比對資料庫中帳號對應的密碼, 回傳一個 boolean。
 - 0->回傳("密碼錯誤"), 結束連線
 - 1->取得資料庫中會議室情況, 回傳至程式, 結束連線

Postcondition:回到登入介面(0)or 進入會議室選擇 UI(1)

3.2.3 更新

Use case:更新

Trigger:點擊更新按鈕

Precondition:使用者登入成功

Step:

1. Call 更新 function
2. 取得資料庫中會議室情況回傳至程式，結束連線

Postcondition:資料已更新

3.2.4 查看預約的會議室

Use case:查看預約的會議室

Trigger:點擊已預約按鈕

Precondition:使用者登入成功

Step:

1. Call 更新 function
2. 取得資料庫中會議室情況回傳至程式，結束連線

Postcondition: 進入已預約介面

3.2.5 查看參與的會議室

Use case:查看參與的會議室

Trigger:點擊已參與按鈕

Precondition:使用者登入成功

Step:

1. Call 更新 function
2. 取得資料庫中會議室情況回傳至程式，結束連線

Postcondition: 進入已參與介面

3.2.6 自動更新

Use case:自動更新

Trigger:超過一定時間(5min)資料庫沒有被 call 更新 function

Precondition:使用者登入成功

Step:

1. Call 更新 function
2. 取得資料庫中會議室情況回傳至程式，結束連線

Postcondition:資料已更新

3.2.7 預約

Use case:預約

Trigger:點擊預約按鈕

Precondition:使用者於預約介面

step:

1. 傳入預約的會議室名稱、信息、時間、受邀請人帳號
2. Call 檢查會議室時段 function，回傳一個 boolean
 - 0->回傳("該時段已被借用，預約不成立")，結束連線
 - 1->step3
3. 針對所有受邀請人 Call 檢查帳號 function，回傳一個 boolean
 - 0->於輸出存入("(帳號)不是使用者，預約不成立")，繼續檢查

- 1->繼續檢查其他人
4. 針對所有受邀請人 call 檢查帳號時段 function，回傳一個 boolean
 - Call 查詢 function 查詢該帳號的使用時間
 - 0->於輸出存入("(帳號)已被邀請至其他會議，預約不成立")，繼續檢查
 - 1->繼續檢查其他人
 5. Call 最終檢查 function，拿取 step3、step4 結果，回傳一個 boolean，若結果中有 0 輸出 0，皆為 1 輸出 1
 - 0->輸出結果，結束連線
 - 1->將預約的會議室名稱、信息、時間、受邀請人帳號存入資料庫，回傳("預約成功")與當次預約的內容，結束連線
 6. 預約成功，執行寄出 Email function

Postcondition:顯示結果

3.2.8 取消會議

Use case:取消會議

Trigger:按下取消預約按鈕

Precondition:發起人於會議室資訊介面中

Step:

1. 傳入欲取消之會議室名稱、時間
2. Call 刪除 function 刪除相關資訊
3. 回傳("取消預約成功")，結束連線
4. 取消預約成功，執行寄出 Email function

Postcondition:顯示結果

3.2.9 修改會議資訊

Use case:修改會議資訊

Trigger: 按下修改按鈕

Precondition:發起人於會議室資訊介面中

Step:

1. 傳入修改會議的名稱、時間、內容、參與人資訊
2. Call 檢查會議室時段 function，回傳一個 boolean
 - 0->回傳(“時段衝突，修改不成立”)，結束連線
 - 1->step3
3. Call 修改內容 function，修改相關訊息
4. 回傳(“修改成功”)，結束連線
5. 修改成功，執行寄出 Email function

Postcondition:顯示結果

3.3 Detailed Non-Functional Requirement

3.3.1 資料庫上的資料結構

會議 entity

Data Item	Type	Description
日期	date	西元年-月-日
時段	integer	表示使用的小時

會議室編號	pointer	指向會議室 entity
會議名稱	text	會議的名稱
發起人帳號	pointer	指向使用者 entity
發起人信箱	text	發起人的 email address
參與人帳號	pointer	指向使用者 entity
參與人信箱	text	受邀請人的 email address
內容	text	會議的相關資訊

3.3.2 使用者 entity

Data Item	Type	Description
信箱	text	申請帳號所用的信箱
名稱	text	登入名稱
密碼	text	登入密碼

預約的會議	pointer	指向會議 entity
參與的會議	pointer	指向會議 entity

3.3.3 會議室 entity

Data Item	Type	Description
日期	date	西元年-月-日
時段	integer	表示使用的小時
會議	pointer	指向會議 entity

4.0 Others

4.1 system requirement

開發者電腦需求

OS	Windows 7 (SP1+), Windows 10 and Windows 11, 64-bit versions only.
CPU	X64 architecture with SSE2 instruction set support
G API	DX10, DX11, and DX12-capable GPUs
Additional	Hardware vendor officially supported drivers

使用者電腦需求

OS	Windows	Mac OS	Linux
OS version	Windows 7 (SP1+) , Windows 10 and Windows 11	High Sierra 10.13+	Ubuntu 20.04, Ubuntu 18.04, and CentOS 7
CPU	x86, x64 architecture with SSE2 instruction set support.	Apple Silicon, x64 architecture with SSE2.	x64 architecture with SSE2 instruction set support.
G API	DX10, DX11, DX12 capable.	Metal capable Intel and AMD GPUs	OpenGL 3.2+, Vulkan capable.
Additional	Hardware vendors officially supported drivers. For development: IL2CPP scripting back-end requires Visual Studio 2015 with C++ Tools component or later and Windows 10 SDK.	Apple officially supported drivers. For development: IL2CPP scripting back-end requires Xcode. Targeting Apple Silicon with IL2CPP scripting back-end requires macOS Catalina 10.15.4 and Xcode 12.2 or newer.	Gnome desktop environment running on top of X11 windowing system Other configuration and user environment as provided stock with the supported distribution (such as Kernel or Compositor) Nvidia and AMD GPUs using Nvidia official proprietary graphics driver or AMD Mesa graphics driver.