

Software Requirements Specification

for

Virtual Room Reservation Assistant

Version 1.0 approved

Prepared by :

1. **B10504028** **林哲豪**
2. **B10601010** **黃士哲**
3. **B10601108** **陳奕憲**
4. **B10801120** **許世佑**

CS3025301 Software Engineering

2020/10/31

Table of Contents

| | |
|---|-----------|
| Table of Contents | ii |
| Revision History | ii |
| 1. Introduction | 1 |
| 1.1 Purpose | 1 |
| 1.2 Glossary | 1 |
| 1.3 Intended Audience and Reading Suggestions | 1 |
| 1.4 Product Scope | 2 |
| 1.5 References | 2 |
| 2. Overall Description | 2 |
| 2.1 System Environment | 2 |
| 2.2 Functional Requirements Definition | 3 |
| 2.3 User Interface Specifications | 6 |
| 2.4 Non-Functional Requirements | 13 |
| 3. Requirements Specifications | 13 |
| 3.1 External interface Requirement | 13 |
| 3.2 Functional Requirements | 13 |
| 4. Other Nonfunctional Requirements | 16 |
| 4.1 Performance Requirements | 16 |
| 4.2 Safety Requirements | 16 |
| 4.3 Security Requirements | 16 |

Revision History

| Date | Reason For Changes | Author | Version |
|----------|---|---------------|---------|
| 2021/1/6 | Google reminder / Database description change | B10632011 陳彥瑜 | 1.0 |
| | | | |

1. Introduction

1.1 Purpose

This document describes the detail of virtual room reservation system. It includes the system feature, interface, constraints and also shows how people interact with this system.

This document is for both the developers and stakeholders of the system, and is used to outline the virtual room reservation system.

1.2 Glossary

| Term | Definition |
|-----------------|--|
| Physic Room | Meeting or discussing room in physics. |
| Virtual Room | Meeting or discussing room create by web conference software ,such as Zoom, Cisco Webex, Google Meet, which allow users to communicate and make discussion online. |
| Online calendar | An online application which can edit what, when and where an event will going to happen by user or other application. |
| Web browser | The application which can make users access to a web page or a web app. |
| Database | An organized collection of data which allow user or other application access electronically. |
| Web API | Application Programming Interface for a web service. |

1.3 Intended Audience and Reading Suggestions

All readers of this document will find use in reading section 1 of this document if they plan on continuing to read other sections. For what sections to read after section one please read the following guidelines:

Developers / Testers / Maintainers - Developers, Testers, and Maintainers who did not already write this document will find use in reading the detailed specification of the features the systems must include and how they function in sections 3.

Users - Although this document is not meant for direct users of the software they may find value in reading section 2 to understand all of the features available to them.

1.4 Product Scope

This software will be a virtual room reservation system that provides users with reliable information so that they can track events, record messages, and manage meeting information. More specifically, this system is designed to allow a user to manage and communicate with members and publish meeting information to members by Email.

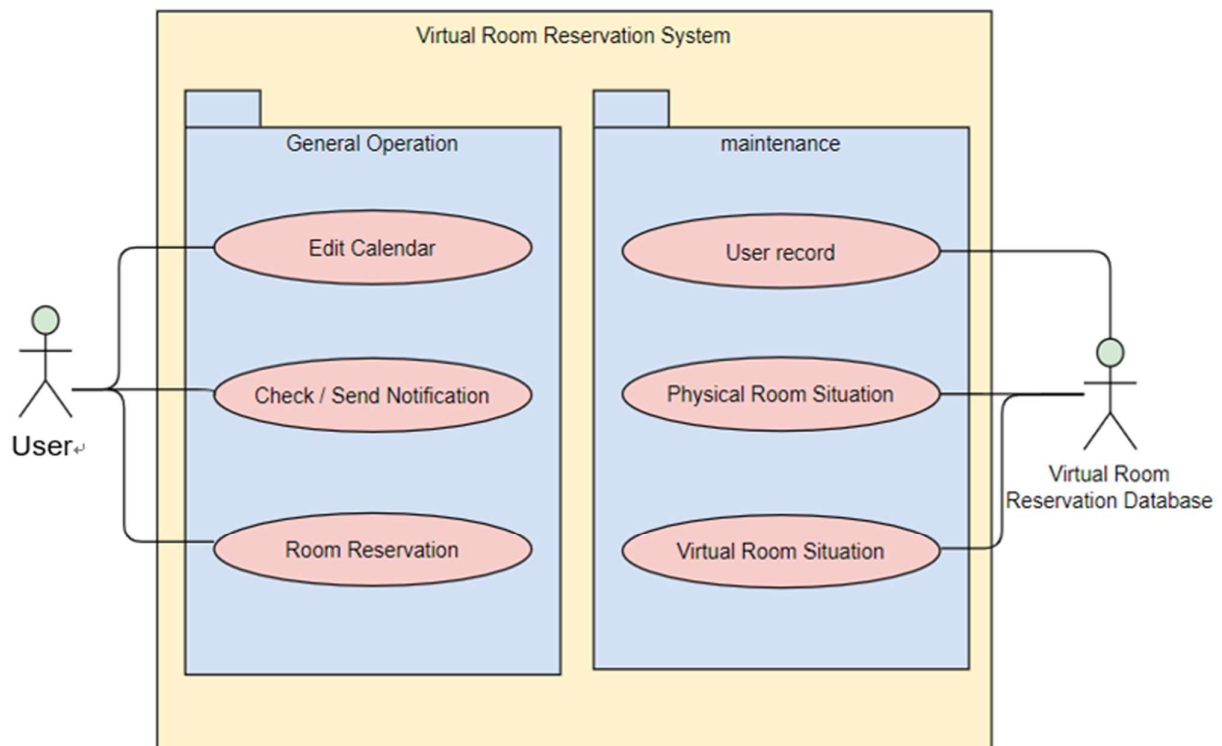
1.5 References

IEEE Template for System Requirement Specification Documents

SW Engineering Project List by Professor Kaliski

2. Overall Description

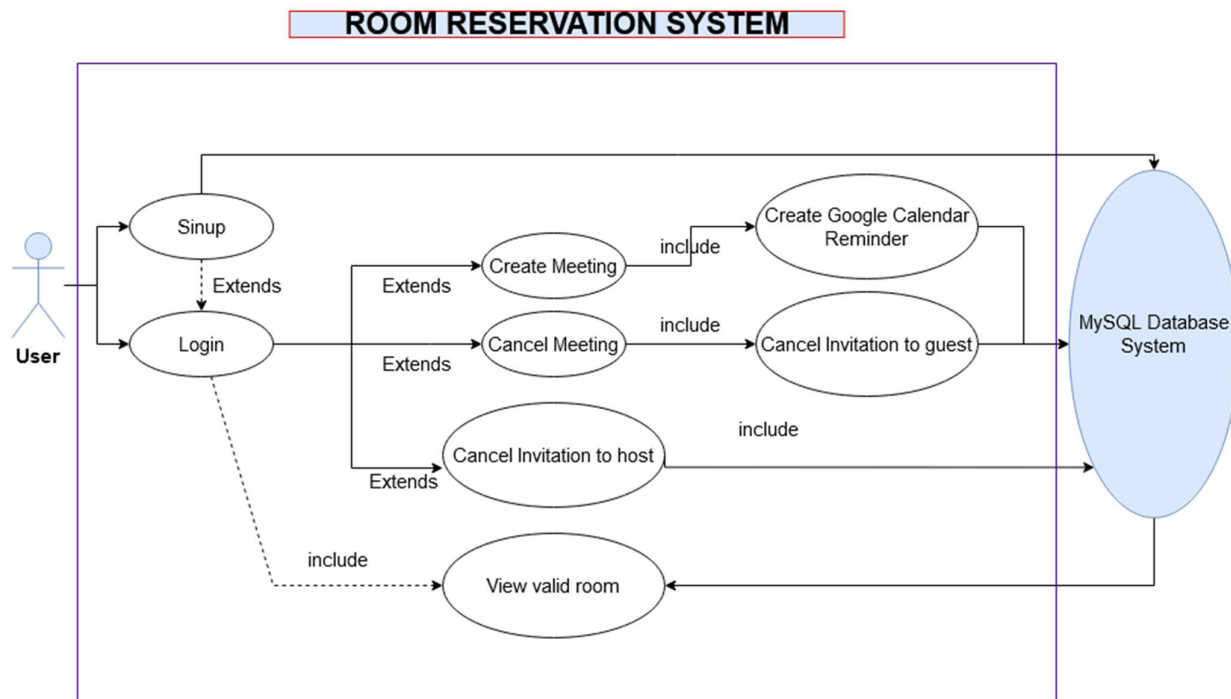
2.1 System Environment



The Virtual Room Reservation System has one active actors and one cooperating system. The User accesses the entire system directly. The user can Edit Calendar, Check/ Send Notification, and Room Reservation. There is a link to the (existing) Virtual Room Reservation Database.

<< The division of the Virtual Room Reservation System into two component parts, the General Operation and the Maintenance, is an example of using domain classes to make an explanation clearer. >>

2.2 Functional Requirements Definition



The product itself is a website application. And it provides the on-line reservation to users. First, we let user create their own account. Then save it to the SQLite database. Second, we provide three main function (**Create meeting, Email, View meeting room**) to users. Each Function would be related to the User's information and Room information.

| Class of Use Case | Use Case Name | Description of Use Cases |
|------------------------------------|-------------------|---|
| 1) Use cases related to user input | 1.1) Click button | User triggers the selection of a button using either a mental command or through blinking. After the action command is done for a sufficient amount of time, the action of the button is triggered. |

| | | |
|--|-----------------------------|--|
| | 1.2) Keyboard Input | User types through an on-screen keyboard that is optimized for typing using a single mental command or through blinking. These keyboards can either be alphabetical keyboards that allow users to create custom messages, or keyboards that have previously created canned responses. |
| 2) Use cases related to access authority | 2.1) Sign Up | User create a unique account based on email. After user click the “建立” button, we save the user info (name, e-mail, account, password) to the database. |
| | 2.2) Login | Let User type their account and password. Detect the account and the password if exist in database. If it's not, return error message. If it's in database, jump to home page. |
| 3) Use cases related to editing. | 3.1) Create Meeting | Let User create the meeting, detect the date if it is valid. If the date user choose is not valid, refresh the page and print “The date is not valid !”. If the date user choose is valid, save the meeting info to database (topic, host, starting date, end date, room id), then jump to create Google Calendar Reminder page. |
| | 3.2) Cancel Meeting | Let User Cancel the meeting. After canceling the meeting. Delete the meeting info (topic, host, starting date, end date, room id) from database. |
| | 3.3) Create Google Calendar | User input the reminder time to set when to send an email message to remind the |

| | | |
|--|---------------------------------|---|
| | Reminder | user. And click “建立” button. |
| | 3.4) Cancel Invitation to host | When user want to cancel the meeting which is been invited, they could click “取消”. |
| | 3.5) Cancel Invitation to guest | After Cancel Meeting, jump to the new page. Click ”寄送” which will cancel the invitation to guest. |
| | 3.6) View valid room | Click “View Meeting Room”, User could see which room is valid to reserve. |

2.3 User Interface Specifications

2.3.1 Signup page



The screenshot shows a web browser window with a blue title bar. The address bar contains the text "signup". The main content area displays a signup form with the following fields and labels:

- 姓名 (Name) [text input field]
- e-mail [text input field]
- 帳號 (Account) [text input field]
- 密碼 (Password) [text input field]
- 確認密碼 (Confirm Password) [text input field]

At the bottom right of the form is a button labeled "建立" (Create).

1. Name text field: user name
2. Email text field: email of user
3. Account text field: account wanted by user, some account validation should be conducted to ensure every user is using unique account
4. Password text field: password used to login.
5. Password confirm text field: make user type in last field's value again to ensure that they know what their password exactly is.

2.3.2 Login page

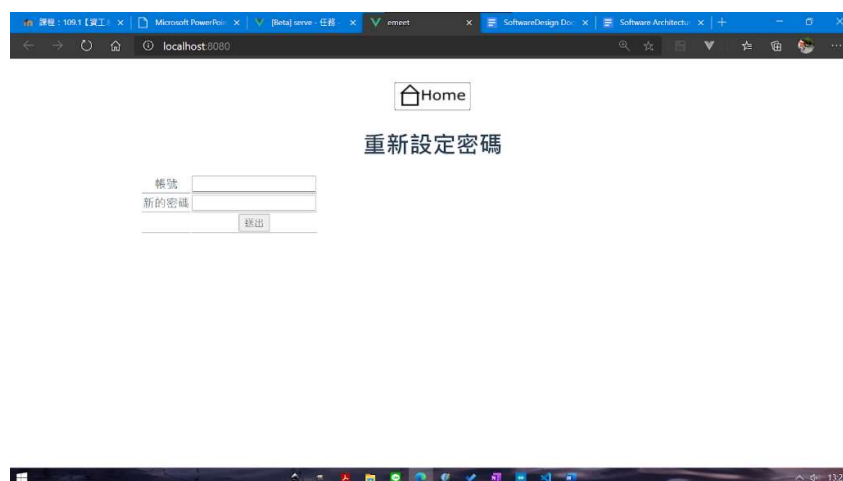


The screenshot shows a web browser window with the address bar displaying 'login'. The main content area contains a login form with the following elements:

- A label '帳號' (Account) followed by a text input field.
- A label '密碼' (Password) followed by a text input field.
- Three buttons arranged horizontally: '註冊' (Sign up), '忘記密碼' (Forgot password), and '登入' (Login).

1. Account text field: user account
2. Password text field: user password
3. Signup button: lead to signup page
4. Forget password button: system would send a email to the email of input account, if the account exist.
5. Login: validate and authorize user to login their account.

After the user clicks “忘記密碼” button, jump to reset password page.

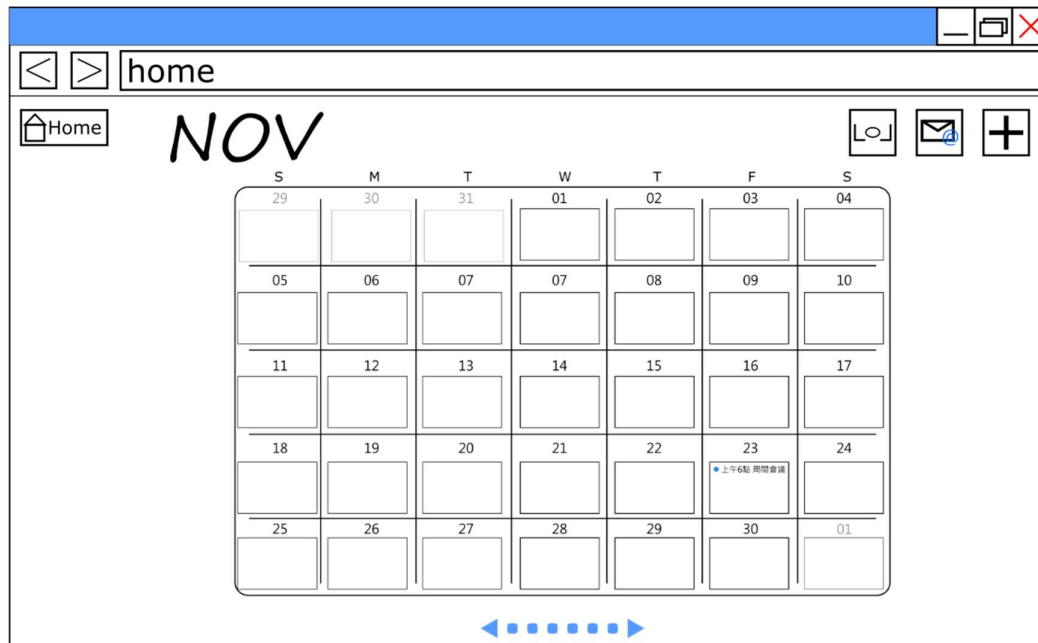


The screenshot shows a web browser window with the address bar displaying 'localhost:8080'. The main content area contains a reset password form with the following elements:

- A 'Home' button at the top.
- A heading '重新設定密碼' (Reset Password).
- A label '帳號' (Account) followed by a text input field.
- A label '新的密碼' (New Password) followed by a text input field.
- A '送出' (Submit) button.

1. Account text field: user account
2. New password text field: new user password

2.3.3 Home page



1. Home button: appear on every page after user logged in. lead to this main page.
2. Calendar: show every meeting for a specific month.
3. Meeting object: lead to the meeting information page for a specific meeting.
4. Left most button on top right corner: lead to meeting room information for every meeting room.
5. Middle button on top right corner: lead to user's email.
6. Right most button on top right corner: lead to create meeting page.

2.3.4 Create meeting page

The screenshot shows a web browser window titled 'creatMeeting'. The browser's address bar shows the page title. The page has a blue header bar with standard window controls (minimize, maximize, close). Below the header, there is a navigation bar with a 'Home' button and the page title 'creatMeeting'. The main content area contains a form with six text input fields, each preceded by a label in Chinese: '主題' (Topic), '主持人' (Host), '開始' (Start), '結束' (End), '參與人' (Participant), and '會議室' (Meeting Room). A '建立' (Create) button is located at the bottom right of the form.

1. Title text field: title of the meeting
2. Host text field: host of the meeting
3. Start time field: start time of the meeting
4. End time field: end time of the meeting
5. Participant text field: participant of the meeting. Reminder would be sent to all of the user in this field. Some data validation is required to ensure every user in this field is existed.
6. Meeting room text field: if the room is physical room, user could choose meeting room from a list. While the room is virtual room, user must create one.
7. Create button: meeting saved to the database and lead to the create google calendar reminder page.

2.3.5 Create google calendar reminder page



createGoogleReminder

Home

建立Google Calendar Reminder

主題

開始

結束

地點

提醒

確認 取消

1. Title text field: automatically filled by system after user create a meeting.
2. Start time field: automatically filled by system after user create a meeting.
3. End time field: automatically filled by system after user create a meeting.
4. Location: automatically filled by system after user create a meeting.
5. Remind time field: pick by user to remind user this meeting at the corresponding time.

2.3.6 Meeting information page

1. Title text field: show title for a specific meeting.
2. Host text field: show host for a specific meeting.
3. Start time field: show start time for a specific meeting.
4. End time field: show end time for a specific meeting.
5. Meeting room text field: show meeting room for a specific meeting.
6. Participant text field: show participant for a specific meeting. Click on the cross button in the beginning of every name of user would disinvite the user, and lead to a meeting cancel page to inform the user that they have been disinvite.

2.3.7 Cancel meeting page

cancel meeting

Home

發送取消通知

主題 _____

開始 _____

結束 _____

發送

1. Title text field: show the title of a specific meeting.
2. Start time field: show the start time of a specific meeting.
3. End time field: show the end time of a specific meeting.

2.3.8 Room Information page

roomInformation

Home

S 11/12 M 11/13 T 11/14 W 11/15 T 11/16 F 11/17 S 11/18

| | | | | | | | |
|--------|--|--|--|--|-----------|--|--|
| ROOM 1 | | | | | | | |
| ROOM 2 | | | | | | | |
| ROOM 3 | | | | | | | |
| ROOM 4 | | | | | 早上8點 房間會議 | | |
| ROOM 5 | | | | | | | |

Navigation arrows: < >

1. Calendar widget: the first column shows the name of each room, the first row shows the date. Clicking the meeting object would lead to the meeting information page of a specific meeting.

2.4 Non-Functional Requirements

This service would be installed on local machine for testing. The database should be installed on cloud environment for testing. User's password must be encrypted before store into database to ensure the security of user's personal data. The webpage should be compatible with common web browser like Google Chrome, Apple Safari, Microsoft Edge.

3. Requirements Specifications

3.1 External interface Requirement

The only external system of this service is online calendar, the app should connect to user's online calendar service, such as Google calendar, outlook etc., with web API provided by these cloud services. Web API usually comes as REST style, which a standard http request is made and a json response is return, this service should help user make the request and handle with the return data.

3.2 Functional Requirements

3.2.1 Create Account

| | |
|--------------------------|--|
| Use Case Name | Create Account |
| Priority | Essential |
| Trigger | The User selects to create a new account to the database. |
| Precondition | The User has accessed the login page. |
| Basic Path | <ol style="list-style-type: none">1. Click "Signup" button.2. Fill out the User name.3. Fill out the User's email.4. Fill out the account number of the User.5. Fill out the password of the User.6. Fill out the password again.7. Click "Create" button. |
| Alternative Paths | None. |
| Postcondition | The User's account has been added to the database. |
| Exception Paths | The User may abandon the operation at any time. |

3.2.2 Login

| | |
|--------------------------|--|
| Use Case Name | Login |
| Priority | Essential |
| Trigger | The User selects to login his / her account. |
| Precondition | The User has accessed the login page. |
| Basic Path | <ol style="list-style-type: none"> 1. Fill out the account number of the User. 2. Fill out the password of the User. 3. Click “Login” button. |
| Alternative Paths | None. |
| Postcondition | The User has logged in his / her account. |
| Exception Paths | The User may abandon the operation at any time. |

3.2.3 Create Meeting

| | |
|--------------------------|--|
| Use Case Name | Create Meeting |
| Priority | Essential |
| Trigger | The User selects to create the meeting. |
| Precondition | The User has accessed the Home page. |
| Basic Path | <ol style="list-style-type: none"> 1. Click right most button on the top right corner. 2. Fill out title of the meeting. 3. Fill out start time of the meeting. 4. Fill out end time of the meeting. 5. Fill out participants of the meeting. 6. Choose an available room or create a virtual room. 7. Click “Create” button. |
| Alternative Paths | None. |
| Postcondition | The meeting information has been added to the database. |
| Exception Paths | The User may abandon the operation at any time. |

3.2.4 Cancel Meeting

| | |
|--------------------------|---|
| Use Case Name | Cancel Meeting |
| Priority | Essential |
| Trigger | The User selects to cancel the meeting. |
| Precondition | The User has accessed the meeting information page. |
| Basic Path | <ol style="list-style-type: none"> 1. Select unwanted meeting. 2. Click “Cancel” button. 3. Fill out the title of the unwanted meeting. 4. Fill out the start time of the unwanted meeting. 5. Fill out the end time of the unwanted meeting. 6. Click “Send” button. |
| Alternative Paths | None. |
| Postcondition | The meeting information has been removed from the database and it will be sent the notification to participants. |
| Exception Paths | The User may abandon the operation at any time. |

3.2.5 Create Google Calendar Reminder

| | |
|--------------------------|--|
| Use Case Name | Create Google Calendar Reminder |
| Priority | Essential |
| Trigger | The User selects to create Google Calendar Reminder. |
| Precondition | The User has created the meeting. |
| Basic Path | <ol style="list-style-type: none"> 1. Fill out the reminder time. 2. Click “Confirm” button. |
| Alternative Paths | None. |
| Postcondition | The Google Calendar Reminder has been created and it will remind User the meeting at the corresponding time. |
| Exception Paths | The User may abandon the operation at any time. |

4. Other Nonfunctional Requirements

4.1 Performance Requirements

Our system is available through web browser. The system may have multiple online users using the system simultaneously. All action about connecting to users' online calendar account should be done without too much lag. The notification of successfully booking a room should be sent to attendee and host in 30 minutes after the users successfully book a room. A room reservation must be upgrade to the database and show in calendar immediately without obviously lag.

4.2 Safety Requirements

Our system will ask user to link their online calendar account. The system can't access any information that users don't want to provide and also can't require any unnecessary authority. Our system may ask user shared their calendar, so the other user can know if a user will be free for meeting.

4.3 Security Requirements

Our system will not store the information that user provides include their email address, phone number, calendar etc. for improper usage. The virtual room is created by external web conference software, therefore our system won't use any method to monitor users' meeting. However, we can't guarantee that user's online meeting won't be monitor through the web conference software.