Software Design Document

for

Virtual Room Reservation Assistant

Version 2.0 approved

Prepared by Group 26
B10715044 Kuo-Yen Chang
B10715029 Yen-Wei, Chen
B10730226 Wen-Yen, Tseng
B10715024 Yin-Cheng, Wang

Submitted in partial fulfillment
Of the requirements of
CS3025301 Software Engineering

2021/01/08

Table of Contents

1. Introduction

1.1 Purpose

The purpose of the Software Design Document is to provide a description of the design of our VRRA system. Our intended audience are those who want to know the details of our system, so we expect that you already have the basic knowledge of software development. It is enough to allow for software development to proceed with an understanding of what is to be built and how it is expected to be built. I think if you read our document, you will know our system design idea clearly.

1.2 Scope

This document provides the architecture and design of Release of the VRRA. It will show how the design will accomplish the functional and non-functional requirements detailed in the VRRA Software Requirements Specification (SRS) document.

Our VRRA system is a room reservation assistant system. It helps people search for appropriate meeting rooms, then reserve it. Those who make the reservation can also manage his or her reservations, whether modify the detail or just delete it. Also, we use Google's APIs, especially Calendar and Gmail API, to notify the host and participate.

1.3 Overview

The Software Design Document is divided into 8 main sections and several subsections. You can also look at the Table of Contents section above.

In section 1, we illustrate the detailed information of this document.

In section 2 and 3, we describe a fully system architecture.

In section 4 ~ 6, we take a look at the details of the system, including the data representation, operate flow, GUI, etc.

In section 7, we link the components and data structures to the functional requirement we mention in the SRS document.

In section 8, we add some useful references for extra reading.

1.4 Definitions and Acronyms

Acronyms	Definitions
VRRA	Virtual Room Reservation Assistant
GUI	Graphical User Interface
SRS	Software Requirements Specification

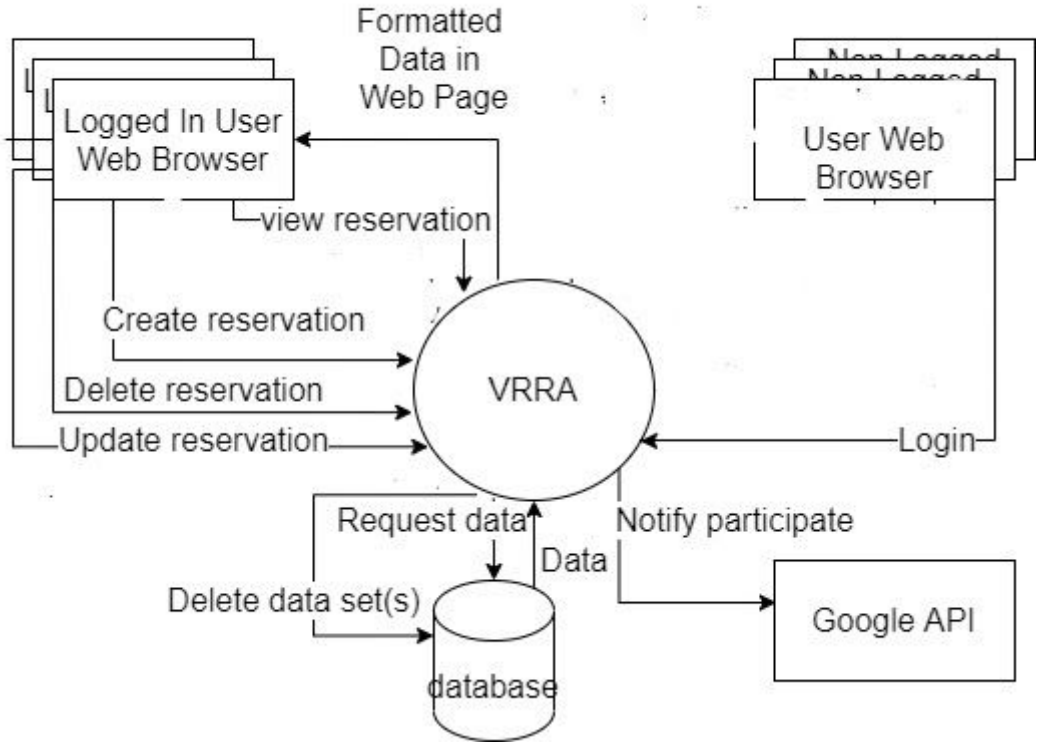
2. System Overview

The project named VRRA is a room reservation system accessible over a network to be built. The project provides an easier control way to reserve rooms for room users, and let the room provider not need to care about the room.

The scope of the project is to make a client-server architecture for about 2,000 users who can connect, and about 500 users who can access, search room uselist and make room reservation at the same time.

In order to provide a concise system, some specialty functions will not be considered, and some function will be combined to make the user interface easy to use. The context diagram (2.1) illustrates the external entities and system interfaces for release version.

2.1 System Context Diagram

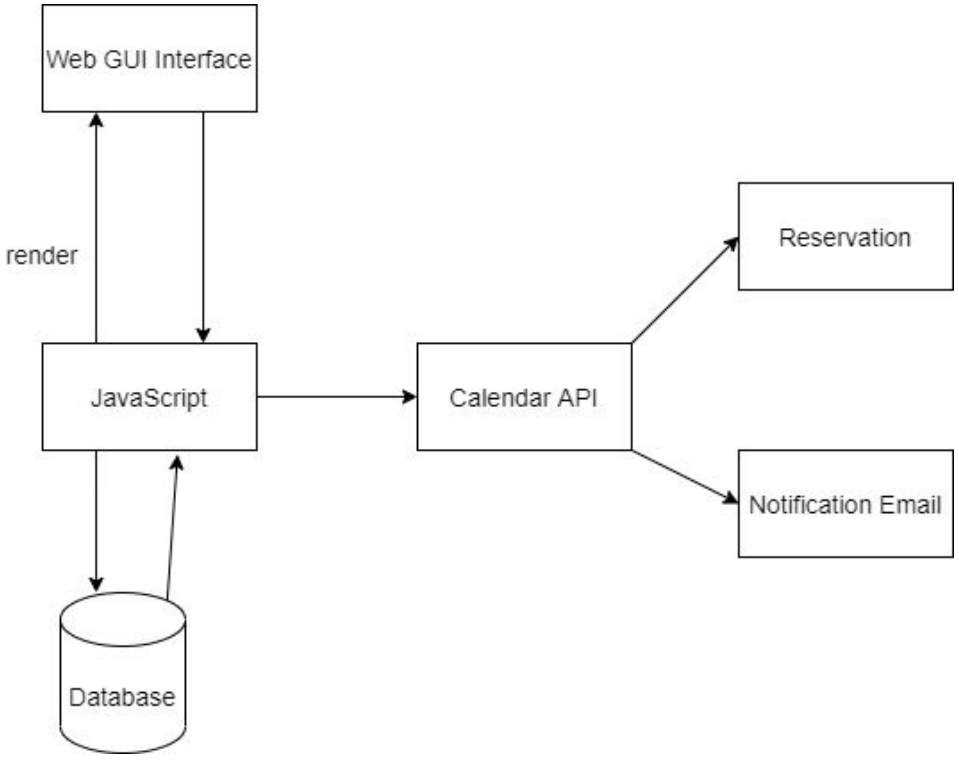


3. System Architecture

3.1 Architectural Design

VRRA uses JavaScript as the frontend to render the Web GUI Interface.

For backend. VRRA use mySQL-like server for Database server. And use the Calendar API provided by Google to create/modify/delete the reservation and send the notification mail to the user via Calendar API.



3.2 Decomposition Description

VRRA uses JavaScript programming as the frontend in correspondence with the skills of the development team, for the backend, VRRA use database server and Google API to control the calendar. Google Cloud Platform is a bundle of tools for API Tools. Among all the other supplement tools, VRRA relies heavily on Google's Calendar API for creating/ modifying/ deleting and sending the notification mail to users.

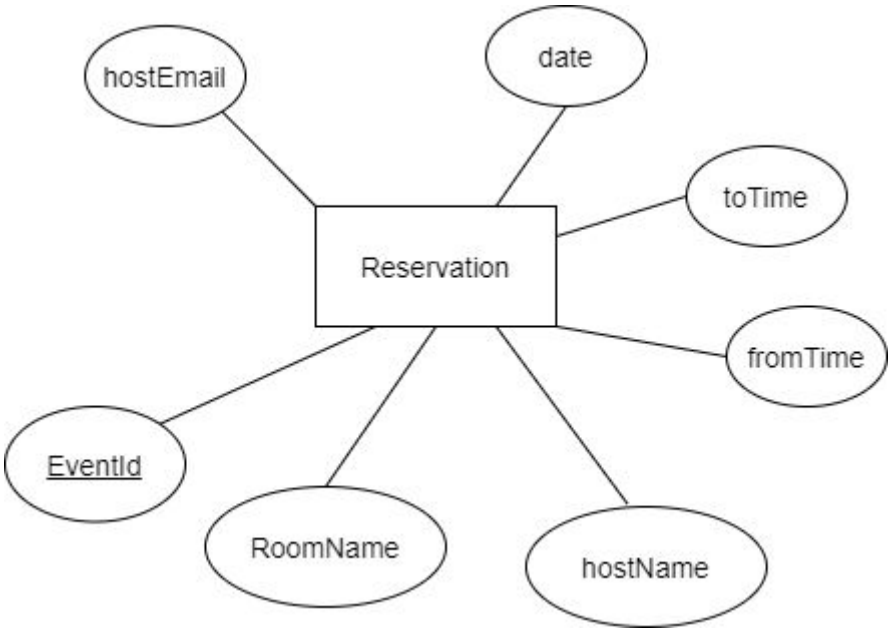
3.3 Design Rationale

Google provides design documentation for Calendar which allows developers to do a lot of things like creating / modifying / deleting an event. And more importantly. we can send the notification mail to users via Calendar API. To reduce a chance of our notification mail getting flagged as spam by mail provider

4. Data Design

4.1 Data Description

ER Diagram of database:



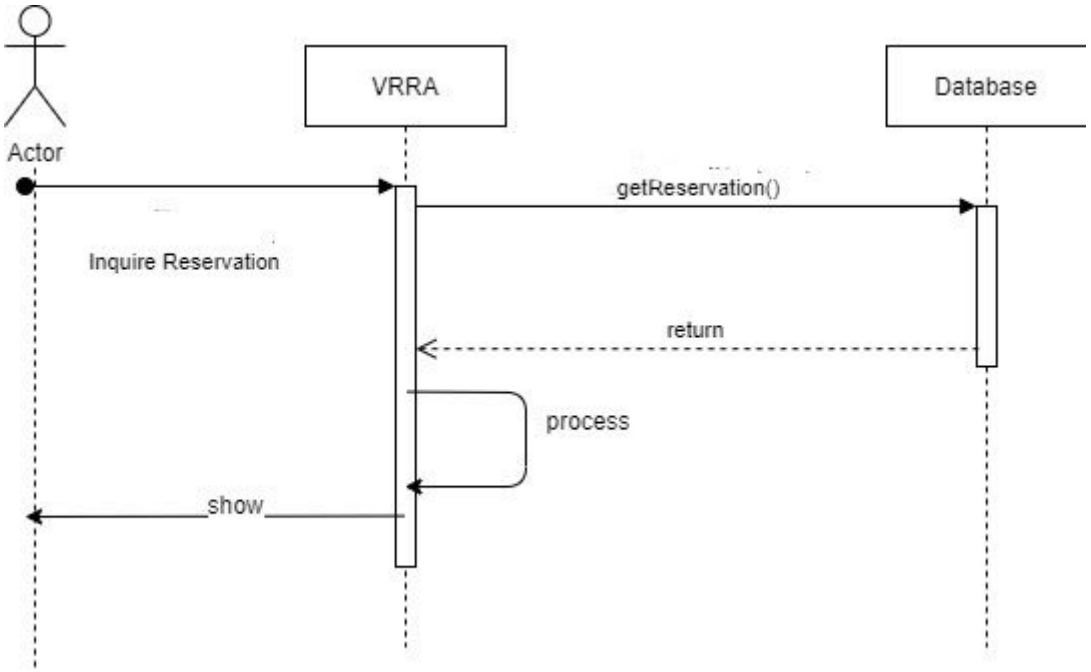
Reservation entity store the reservation info. It has Room, Time, Host attributes, and so on. For real, you can see a example as follow.

room number	host name	from time	to time	date	status	id	email
116	testt	5:00	8:00	2021-01-20		knkt7g1eti4gov8uqqmcuqigi4@google.com	109se.group26@gmail.com
111	ttt	16:41	18:41	2021-01-08		mnus3l22hrrugfqhnesh2om6hc@google.com	example@aaa.com
102	frank	18:14	19:14	2021-01-22		0mb0ju9fj2klgvm5lgd33erm14@google.com	frank1314168@gmail.com
306	123	20:14	23:04	2021-01-08		gio7d8839u5aa9unfghi0kp19c@google.com	frank1314168@gmail.com
234	joywang	19:25	20:25	2021-01-30		3559om2mlqnk3ohur3c9bn7vck@google.com	12joywang@gmail.com
2354	ffgg	19:33	19:36	2021-01-08		kak6do6lqffteadt4vivtge8ag@google.com	frank1314168@gmail.com
12341234	12341234	20:55	21:55	2021-01-08		iol5i7bg8pk76sp6clt2d5k05o@google.com	wynn1212@gmail.com
12341234	1234123123121	20:55	21:55	2021-01-08		48tfrd86nsc9kmteej4f3pkmpg@google.com	wynn1212@gmail.com
1231231231231	123	21:02	22:02	2021-01-08		fhrp0btqkjukri6jm1sss083c4@google.com	wynn1212@gmail.com
282828	282828	21:05	22:05	2021-01-08		s4ddgekfkhs20jlti49i8ubi7o@google.com	wynn1212@gmail.com
123111	123	22:20	23:20	2021-01-08		j3h8dph552jn4di4gp0rep0eak@google.com	wynn1212@gmail.com

5. Component Design

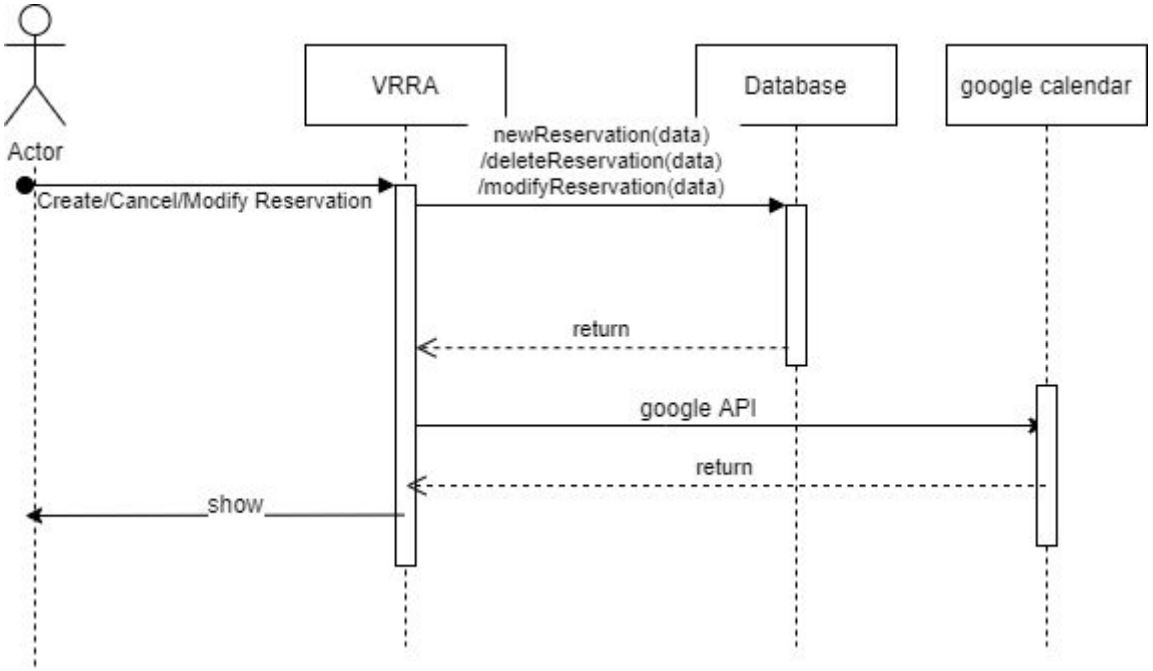
In this section, we take a closer look at what each component does to give a more clear understanding. Refer to the following document to understand how the components for VRRA interact with one another.

- User Inquire Reservations




```
inquireReservation():
    displayImage("waiting")
    result = queryReservation(userInput)
    displayQuery(result)
```

- User Create A Reservation
- User Cancel The Reservation
- User Modify The Reservation



```
createReservation():
    displayImage(“waiting”)
    userInput, errCode = checkUserInput([rRoom, rTime, rInvite])
    if (userInput == valid) {
        reserveConflict = checkReserveConflict(userInput.rRoom, userInput.rTime)

        if (not reserveConflict) {
            createReservation(userInput.rRoom, userInput.rTime, userInput.rInvite)
            calendarAPI.add(“Reservation in ”+userInput.rRoom,
                    userInput.rTime, userInput.Invite)

            showMsg(“Created reservation”)
        }
        else {
            displayErr(Error.conflectReservation)
        }
    }
    else {
        displayErr(errCode)
    }
```

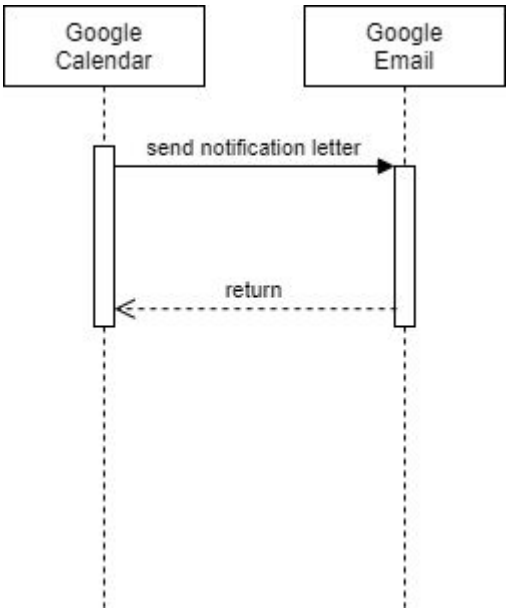
```
deleteReservation():
    displayImage(“waiting”)
    deleteReservation(userInput.rRoom, userInput.rTime)
    calendarAPI.delete(“Reservation in ”+userInput.rRoom, userInput.rTime)
    showMsg(“Deleted reservation”)
```

```
modifyReservation():
    displayImage(“waiting”)
    userInput, errCode = checkUserInput([rRoom, rTime, rInvite])
    if (userInput == valid) {
        reserveConflict = checkReserveConflict(userInput.rRoom, userInput.rTime)

        if (not reserveConflict) {
            updateReservation(userInput.rRoom, userInput.rTime, userInput.rInvite)
            calendarAPI.change(“Reservation in ”+userInput.rRoom,
                    userInput.rTime, userInput.Invite)

            showMsg(“Modified reservation”)
        }
        else {
            displayErr(Error.conflectReservation)
        }
    }
    else {
        displayErr(errCode)
    }
```

- System Notification



There is no code here because this process is done by Google Calendar automatically.

6. Human Interface Design

6.1 Overview of User Interface

The interface will be implemented by web pages, which include several buttons to let users do the operations. The main method is using mouse to control the command ,and users may ask to be familiar with Google Calendar and Gmail, which are important for users to get the information.

6.2 Screen Images

- Homepage:

Logged in user

這個應用程式是由其他使用者建立，而非 Google。

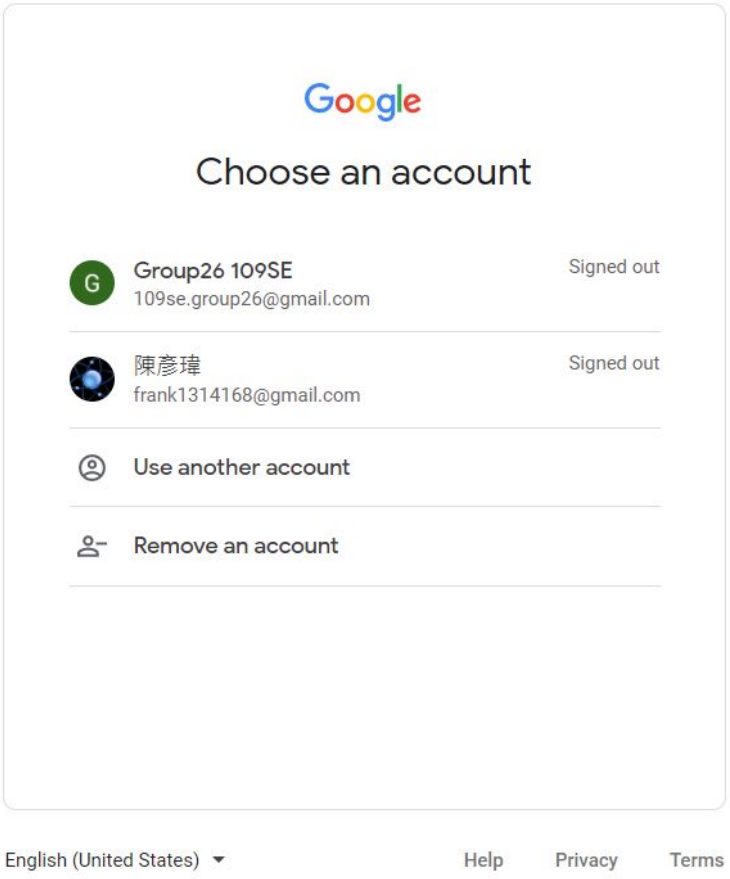
服務條款

VRRA

Inquire Reservation | Create Reservation | Reservation Management | frank1314168@gmail.com

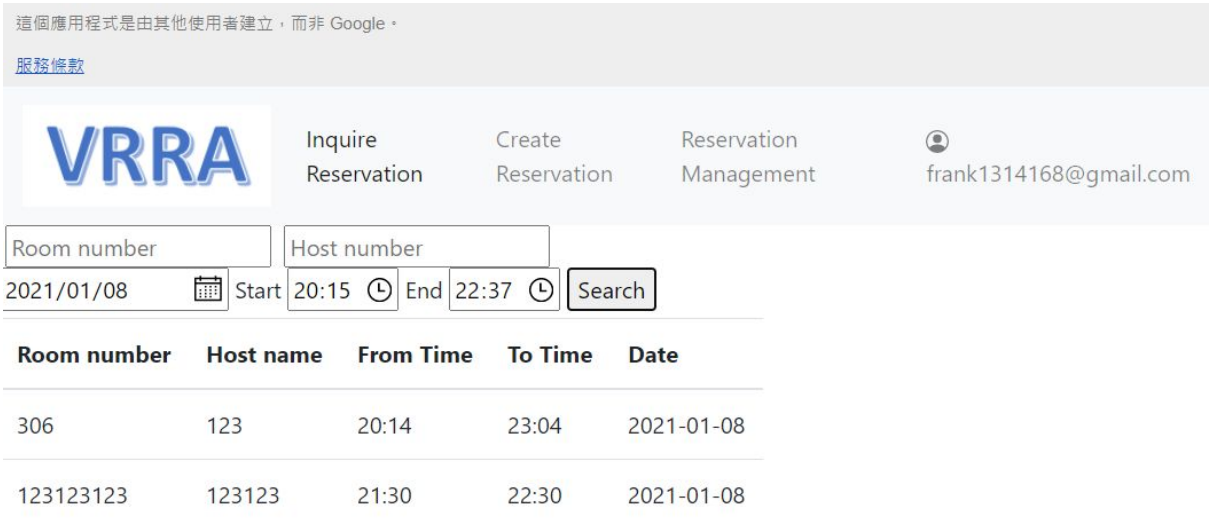
Contact: 109SE.group26@gmail.com

- Login:

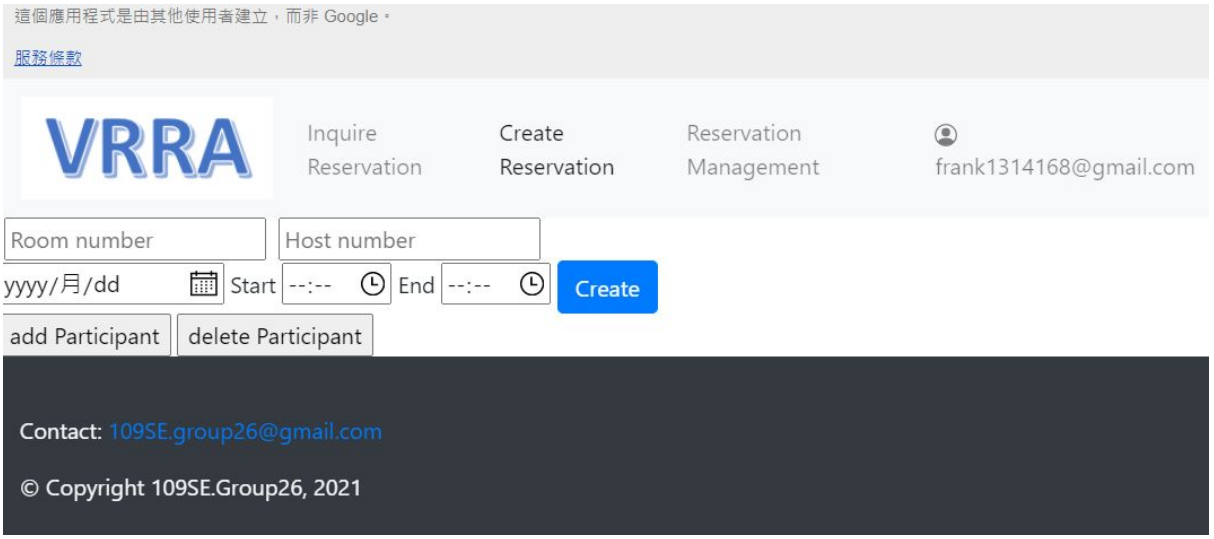

Google
Choose an account
G
Group26 109SE
109se.group26@gmail.com
Signed out
陳彥瑋
frank1314168@gmail.com
Signed out
Use another account
Remove an account
English (United States)
Help
Privacy
Terms

- Inquire Reservation:

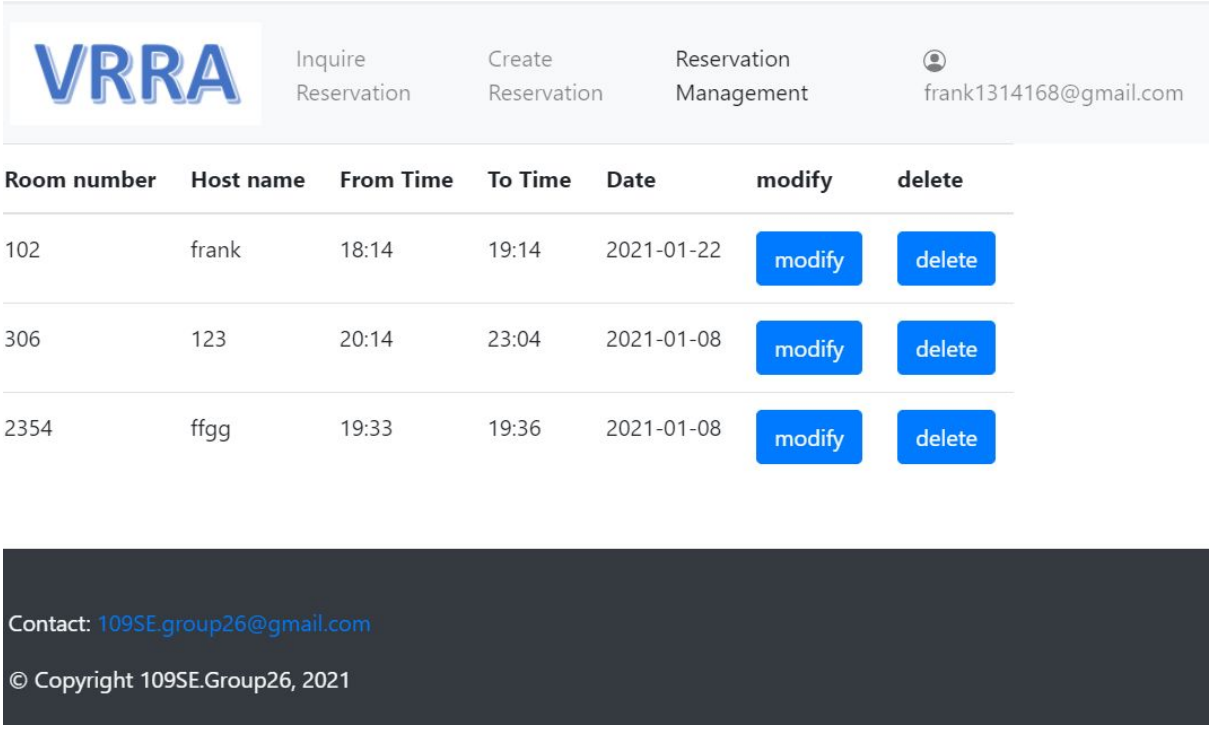
Logged in user



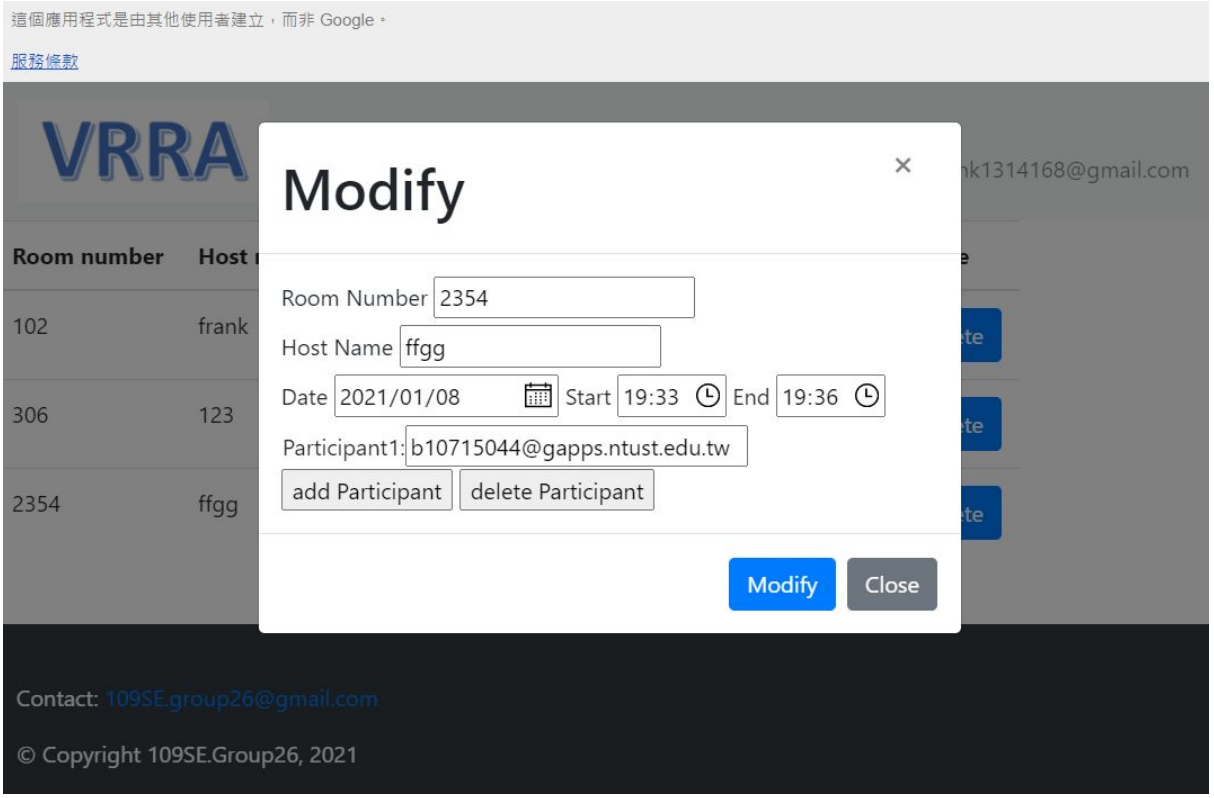
- Create Reservation:



- Reservation Management:



- Modify Reservation:



6.3 Screen Objects and Actions

A discussion of screen objects and actions associated with those objects.

- Homepage:

Logged in user will see this:

1) Logo:
   Our services' logo.
2) Inquire Reservation button:
   If a user presses this button, he will go to Inquire Reservation page.
3) Sign in button:
   If a user presses this button, he will go to Login page.
4) Sign up button:
   If a user presses this button, he will go to Sign Up page.
5) Our service banner:
   A banner to publicize our service.
6) Footer:
   Copyright notice.
7) Create Reservation button:

If a user presses this button, he will go to Create Reservation page.

8) Reservation Management button:

If a user presses this button, he will go to Reservation Management page.

9) User Account:

Show user google account.

- Login:

This part is provided by Google.

- Inquire Reservation:

Logged in user will see this:

More than Homepage components:

1) Date:

A Date selection to select when the date wanders to search reservation.

2) Room number:

If a user wants to search by room number, he can enter data.

3) Host name:

If a user wants to search by host name, he can enter data.

4) From time:

If a user wants to search by the reservation start time, he can enter data.

5) To time:

If a user wants to search by the reservation end time, he can enter data.

6) Search button:

If Input conditions are complete, users need to press to search data.

7) Room number:

The Results of Room number.

8) Host name:

The Results of Host name.

9) From time:

The Results of From time.

10) To time:

The Results of To time.

11) Date:

The Results of Date.

12) Other Results:

Other result show like the first row

More than non logged user the function is mentioned in Homepage Logged in user part.

- Create Reservation:

Logged-in user will see this:(And Non logged user can’t access)

More than Homepage components:

1) Date:

A Date selection selects when the reservation is created.

2) Room number:

Enter the room number which the user wants to borrow.

3) From time:

Enter the start time of reservation when the user wants to borrow.

3) Host name:

Enter the host name of reservation who wants to borrow.

4) To time:

Enter the end time of reservation when the user wants to borrow.

5) Participants:

If the user wants to change the participants of reservation, change the participants' email.

6) Add Participants:

If the participant input box need more, push it.

7) Delete Participants:

If the participant input box so more, push it.

8) Create button:

If the users complete the requirement, users need to press it to create data and to create a reservation.

- Reservation Management:

Logged in user will see this:(And Non logged user can’t access)

More than Homepage components:

7) Room number:

The Reservation of Room number which belongs to the user.

8) Host name:

The Reservation of Host name which belongs to the user.

9) From time:

The Reservation of From time which belongs to the user.

10) To time:

The Reservation of To time which belongs to the user.

11) Date:

The Reservation of Date which belongs to the user.

12) Modify button:

If the users want to modify the reservation, users need to press to modify button.

13) Delete button:

If the users want to delete the reservation, users need to press to delete button.

14) Other Results:

Other Reservations show like the first row or the last row.

- Modify Reservation:

  Logged in user will see this:(And Non logged user can’t access)

More than Homepage components:

1) Date:

It will show the old date.

If the user wants to change the date, entering the new one.

2) Room number:

It will show the previous room number.

If the user wants to change the room number, entering the new one.

3) From time:

It will show the previous start time of reservation.

If the user wants to change the start time of reservation, entering the new one.

4) To time:

It will show the previous end time of reservation.

If the user wants to change the end time of reservation, entering the new one.

5) Participants:

If the user wants to change the participants of reservation, change the participants' email.

6) Add Participants:

If the participant input box need more, push it.

7) Delete Participants:

If the participant input box so more, push it.

8) Modify button:

If the users completely revise the data, users need to press to modify data.

7. Requirements Matrix

System Component	Relevant Component
Logged-in user	SRS.2.2.1
Non-logged-in user	SRS.2.2.2
Google API	SRS.2.2.3

8. Appendices

[1] Get Started with the Calendar API https://developers.google.com/calendar/overview

[2] Introduction · Bootstrap v4.5
https://getbootstrap.com/docs/4.5/getting-started/introduction/

[3] jQuery API Documentation https://api.jquery.com/