

The Report for Term Project

The Introduction to Database System

Hotel Management System

Group Members:

孫聖 0340249

車曉宇 0340299

劉雨薇 0340247

Introduction

這是一個酒店管理系統，能實現對酒店職工及房間的管理，和客戶訂房退房的功能。

資料夾 Design 中，放有初始的 attribute 的設計和 ER Diagram。

資料夾 src 中，放有 python 源碼和初始的 sql 代碼。

System description

1. 建立步驟

1.1 運行 XAMPP，在 phpMyAdmin 中的 test 資料庫中執行 src 中的 Schema.sql 語句。

1.2 運行 src 中的 Main.py 即進入主界面。

2. 程序運行

主界面有 Administrator 和 Employee 兩個選項。

2.1 Administrator

簡介：Adminstrator 主要是酒店的管理員，他能夠對職工和房間的資料進行修改。

進入 Administrator 中需要輸入用戶名和密碼（用戶名:a，密碼:a），並有錯誤檢查。完成驗證後，能進行與 Employee 和 Room 相關的操作。

2.1.1 Employee

2.1.1.1 add employee

增加一个 Employee，需要设置密码，会返回 UserId。

2.1.1.2 reset employee password

输入 UserId 及所需要更改的密码。

2.1.1.3 delete employee

输入 UserId 并将其删除。

2.1.2 Room

2.1.2.1 add room

能够分别增加三种不同的房间。

2.1.2.2 changeprice

输入 RoomId 可以修改该 Room 的价格。

2.1.2.3 delete room

输入 RoomId 可以删除该 Room。

2.2 Employee

簡介：Employee 是酒店的職工，他能為酒店顧客登入資料、查詢空房、訂房及退房。注：部分功能如 Service 的訂購尚未開發。

同樣，進入 Employee 中需要輸入用戶名和密碼（但並沒有初始的 Employee，需要通過 Administrator 來創建），並有錯誤檢查。完成驗證後，能進行與 Employee 和 Room 相關的操作。

2.2.2 InfoInput

輸入客戶資料，客戶分為兩種，一種是 Person，另一種是 Company。

2.2.2.1 Person

需要輸入 SSN，年齡，性別，姓名及一個或多個電話號碼。如果是 VIP，可以輸入 Vid 及其狀態(status)。

2.2.2.2 Company

需要輸入公司名、所在地址和電話。公司默認不是 VIP。

2.2.3 QueryEmptyRoom

能夠查詢三種不同房間是否有空房，如果有會返回第一間的編號，沒有則返回沒有空房的訊息。

2.2.4 Reserve

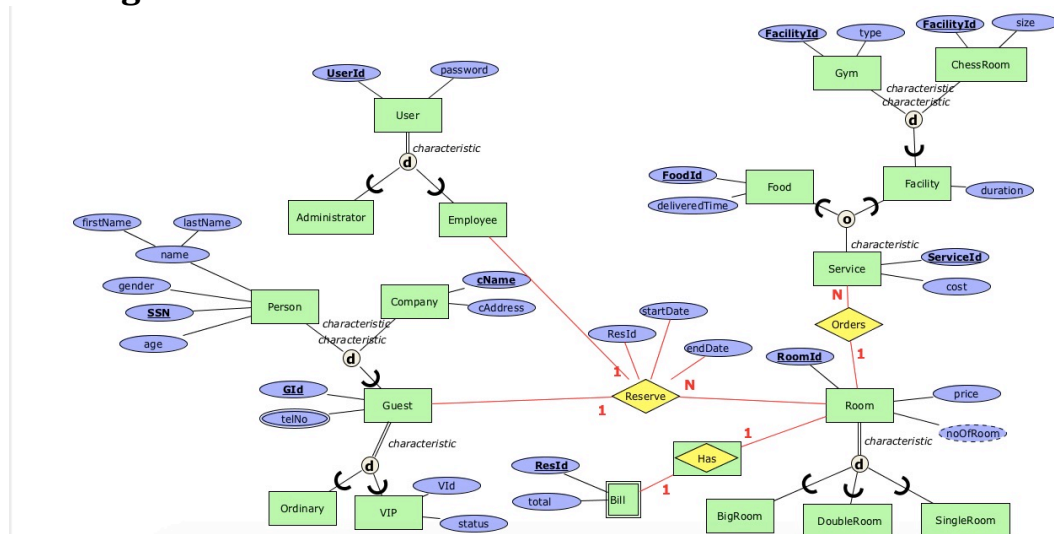
先要輸入做此次訂房功能的員工的帳號，之後輸入顧客編號，及要訂的房間的編號。最後輸入入住日期和退房日期（格式:yyyy-mm-dd）

2.2.5 Check Out

直接輸入預訂的 ID (ResId)，便可實現退房，退房時會顯示金額。

注：Project 備份的網路資料夾 <https://github.com/camouflage/Database>

ER Diagram



（ER Diagram 的圖片檔在資料夾 Design 內）

注：1. Has, 綠色矩形包含黃色菱形的圖標是 Identifying relationship。

2. 由於該套軟體沒有 UNION，所以用 d 來代替：Person 和 Company UNION 成 Guest，Gym 和 ChessRoom UNION 成 Facility。
3. telNo 為 multivalued attribute。
4. noOfRoom 為 derived attribute

Schema

Person

<u>SSN</u>	age	gender	firstName	lastName	GId
-------------------	-----	--------	-----------	----------	-----

SSN -> all.

SSN(key)

BCNF

Company

<u>cName</u>	cAddress	GId
---------------------	----------	-----

cName -> all.

cName(key)

BCNF

Guest

<u>GId</u>	guestType	VId
-------------------	-----------	-----

GId -> all. GId(key)

BCNF

VIP

<u>VId</u>	status
-------------------	--------

VId -> status.

VId(key)

BCNF

注：Guest 的 union type 的實現方式是 surrogate key.

Guest 的 specialization 的實現方式是 c. single relation with type attribute t.

Gym

<u>FacilityId</u>	type
--------------------------	------

FacilityId -> type. FacilityId(key)

BCNF

ChessRoom

<u>FacilityId</u>	size
--------------------------	------

FacilityId -> size.

FacilityId(key)

BCNF

Facility 預訂的設備服務

<u>FacilityId</u>	<u>ServiceId</u>	duration
--------------------------	-------------------------	----------

FacilityId, ServiceId -> duration.

FacilityId, ServiceId(key)

BCNF

注：Facility 的 union type 的實現方式是 inherit the same key.

Food 預訂的食物服務

FoodId	ServiceId	deliveredTime
--------	------------------	---------------

ServiceId -> all.

ServiceId(key)

注：Specialization Food & Facility 的實現方式是 a. Multiple Relations – superclass and subclass.

BCNF

Service

ServiceId	cost	RoomId
------------------	------	--------

ServiceId -> all.

ServiceId(key)

注：Relationship Orders 的實現方式是 Foreign key.

BCNF

Room

RoomId	available	price	BigFlag	DoubleFlag	SingleFlag
---------------	-----------	-------	---------	------------	------------

RoomId -> all.

RoomId(key)

注：Specialization BigRoom & DoubleRoom & SingleRoom 的實現方式是 d.

Single relation with multiple type attributes.

BCNF

Administrator

UserId	password
---------------	----------

UserId -> password.

UserId(key)

BCNF

Employee

UserId	password
---------------	----------

UserId -> password.

UserId(key)

BCNF

注：Specialization User 的實現方式是 b. Multiple relations – subclass relations only.

Bill

ResId	total
--------------	-------

ResId -> total.

ResId(key)

BCNF

Contact

GId	telNo
------------	--------------

BCNF

注：telNo 是 Multivalued attribute.

EmpId	GId	RoomId	<u>ResId</u>	startDate	endDate
-------	-----	--------	---------------------	-----------	---------

ResId -> all.

ResId(key)

BCNF

注：Reserve 的實現方式是 Relationship relation.

注：為了減少 join 從而提高效率，部分 relation 在原先 ER Diagram 轉換而來的基礎上有所修改。