1. 設計目的

日常生活中人們習慣用手機的記事本或電腦的txt檔來記錄各種事物，但這樣紀錄的資訊很分散，沒有一個統一的格式，若是想尋找之前的紀錄，有時需要從手機裡找，有時要從電腦找，紀錄格式的不同也增加了閱讀的難度，若是要找圖片還要從相簿一頁一頁查找當天拍的照片。而我平常習慣紀錄自己吃過好吃的美食、好喝的飲料，所以想到能用美食當作主題建立一個美食評價網站來方便我記錄與查找各個美食的資訊。

1. 資料來源

自己評論手動輸入

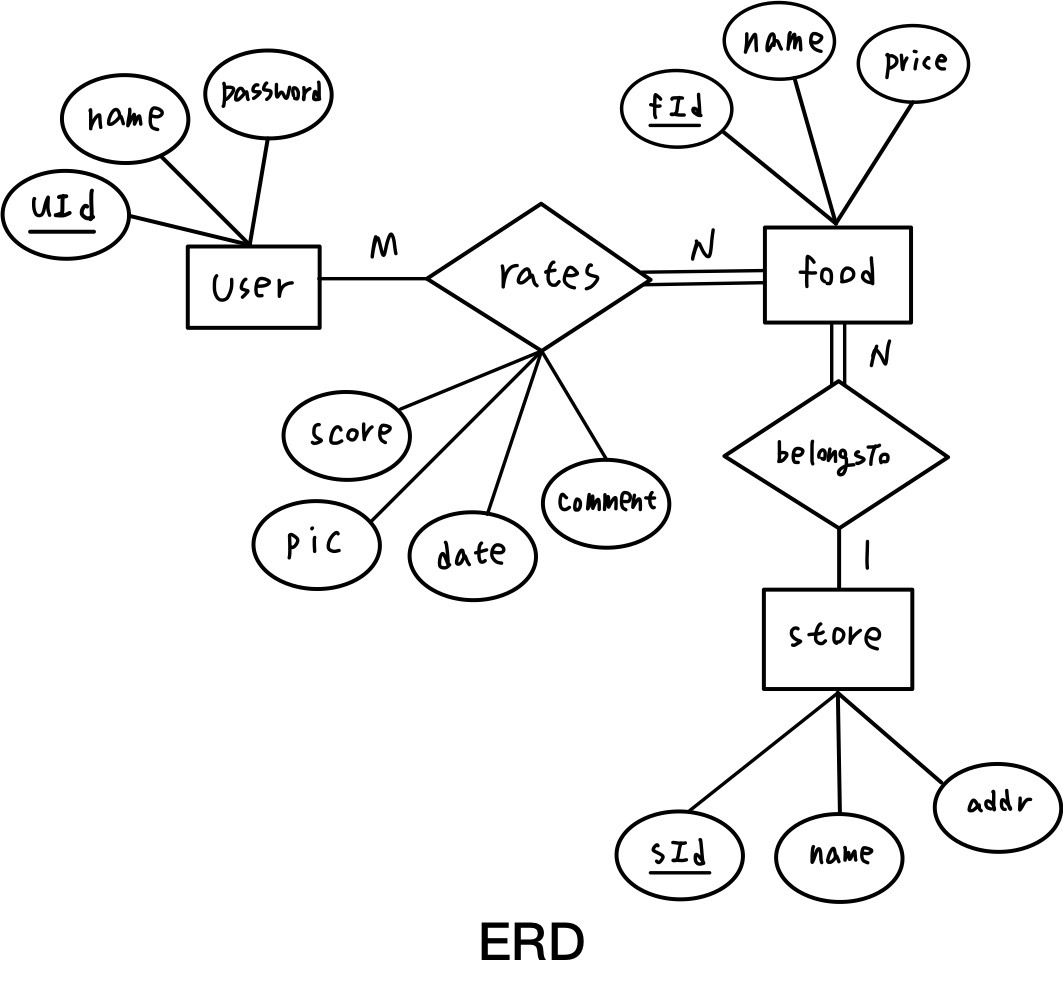
1. 系統功能說明

系統有三種身分，遊客、會員、管理員。

1.遊客:可以查看所有美食和所有使用者的評論，還有查看最熱門美食排行榜，且遊客可以註冊成為會員

2.會員:會員可以查看所有美食和所有使用者的評論、最熱門美食排行榜，除此之外還能對美食新增評論，新增的評論可被所有人看到，使用者還可以單獨查看自己的評論，若不滿意可以進行刪除或修改

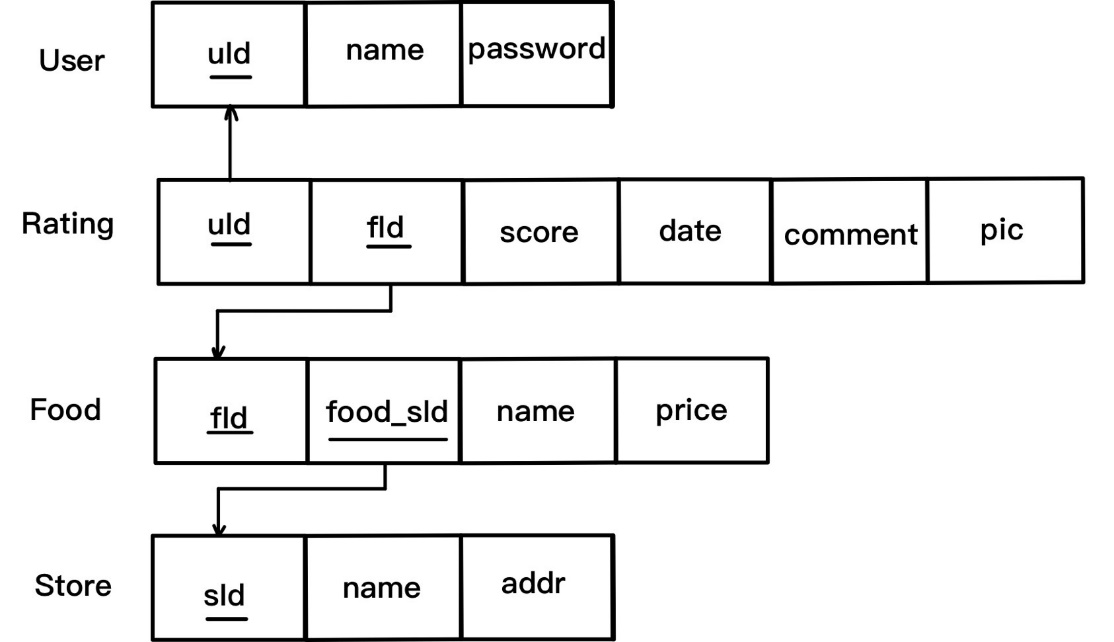
3.管理員:管理員可以查看所有美食和所有使用者的評論、最熱門美食排行榜，也能對美食新增評論，可以單獨查看自己的評論，並進行刪除或修改，除此之外管理員可以查看每張table，且能刪除table裡的row

1. 系統之ERD和解說設計概念

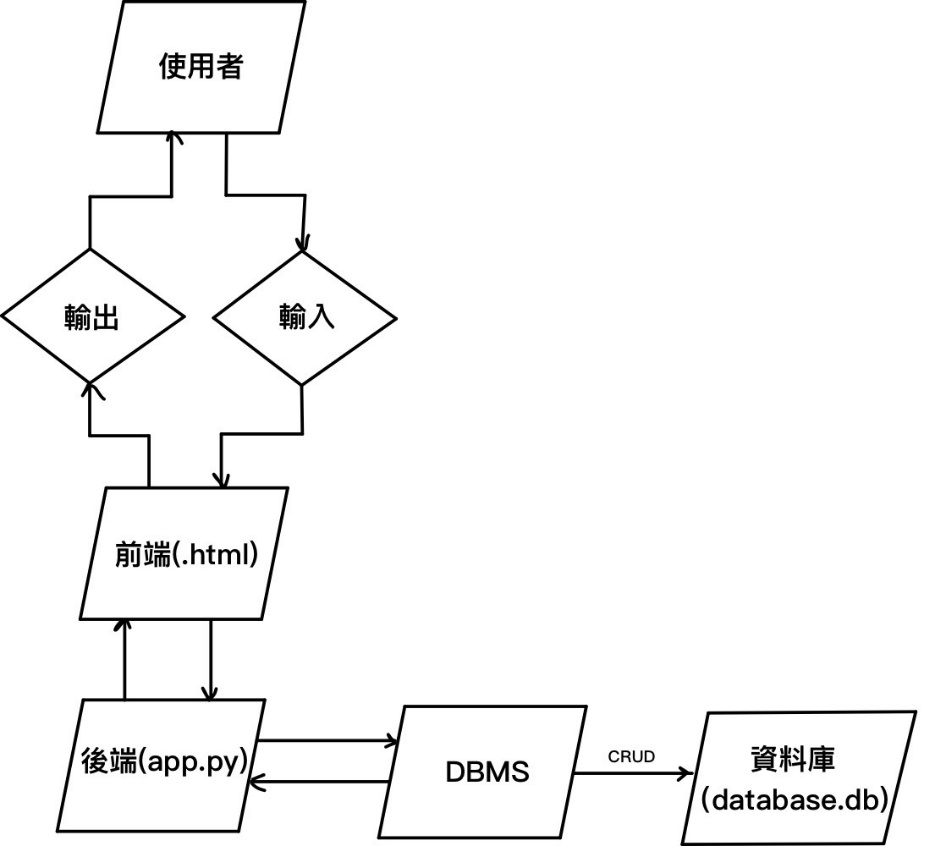
圖一、系統之ERD

我設計的ERD包含三個實體，使用者(User)、美食(Food)、店家(Store)，考量到一個使用者可以評論多種美食，一種美食也可以被多個使用者評論，所以我採用多對多的關係，且被評價的美食一定有使用者評論故美食對使用者是完全參與，而使用者不一定要評論美食所以使用者對美食是部分參與。而我假設一個店家會擁有多種美食，但同種的美食只會屬於一個店家，故店家對美食會是一對多的關係，且被評價的美食一定有店家販賣所以美食對店家是完全參與，而店家不一定有販賣被評論的美食，所以店家對美食是部分參與。

1. 關聯綱目

圖二、系統之關聯綱目

根據ERD繪製出來有四個綱目，其中ERD有三個實體，故有三列綱目分別是使用者(User)、美食(Food)、店家(Store)，由於多對多的關係，此時需要多加一列關係綱目，於是就有了評價(Rating)這條綱目分別指向使用者以及美食

1. 系統之架構圖和所用的工具

圖三、系統之架構圖

開發這個資料庫系網站所使用的是python的flask搭配sqlite3的資料庫系統

1. 系統所使用之SQL敘述以及功能說明

cur.execute("SELECT rowid, name, password FROM User WHERE name = '"+name+"' AND password = '"+password+"'")

用於比對輸入的名稱、密碼是否有在User table裡

cur.execute("INSERT INTO User VALUES (?, ?)", (name, password))

將註冊的會員名稱、密碼插入User table

cur.execute("SELECT rowid FROM Store WHERE (name, addr) = (?,?)", (store\_name, addr))

取出符合輸入的店名ID

cur.execute("INSERT INTO Store VALUES (?,?)", (store\_name, addr))

將輸入的店名、店家地址插入到Store table

cur.execute("SELECT rowid FROM Food WHERE (name, price) = (?,?)", (food\_name, price))

取出符合輸入的美食ID

cur.execute("INSERT INTO Food VALUES (?,?,?)", (food\_sId, food\_name, price))

將輸入的美食名稱、價格與先前得到的food\_sId插入到Food table

cur.execute("INSERT INTO Rating VALUES (?,?,?,?,?,?)", (uId, fId, score, date, comment, pic))

將輸入的評分、造訪日期、評論、相片與之前得到的uId、插入到Rating table

cur.execute("SELECT rowid, \* FROM User")

取出所有User table中的row

cur.execute("SELECT rowid, \* FROM Store")

取出所有Stor table中的row

cur.execute("SELECT rowid, \* FROM Rating")

取出所有Rating table中的row

cur.execute("SELECT rowid, \* FROM Food")

取出所有Food table中的row

cur.execute("DELETE FROM User WHERE rowid = (?)", rowid)

刪除User table中指定的row

cur.execute("DELETE FROM Store WHERE rowid = (?)", rowid)

刪除Store table中指定的row

cur.execute("DELETE FROM Rating WHERE rowid = (?)", rowid)

刪除Rating table中指定的row

cur.execute("DELETE FROM Food WHERE rowid = (?)", rowid)

刪除Food table中指定的row

cur.execute("SELECT \* FROM Food, Rating, Store WHERE Store.rowid = Food.food\_sId AND Food.rowid = Rating.fId AND Food.name LIKE (?)", ('%' + food\_name + '%',))

輸入的food\_name與Food table中的name比對後列出比對到的美食們的評價、 店家資訊

cur.execute("SELECT \* FROM Food, Rating, User WHERE User.rowid = Rating.uId AND Food.rowid = Rating.fId AND User.name LIKE (?)", ('%' + user\_name + '%',))

輸入的user\_name與User table中的name比對後列出比對到的會員們評價的美食、 店家資訊

cur.execute("SELECT Rating.rowid, \* FROM Food, Rating, User WHERE User.rowid = (?) AND User.rowid = Rating.uId AND Food.rowid = Rating.fId AND Food.name LIKE (?)", (uId, '%' + food\_name + '%'))

用先前得到的uId與輸入的user\_name與User table中的name比對的結果列出美食、 評價資訊

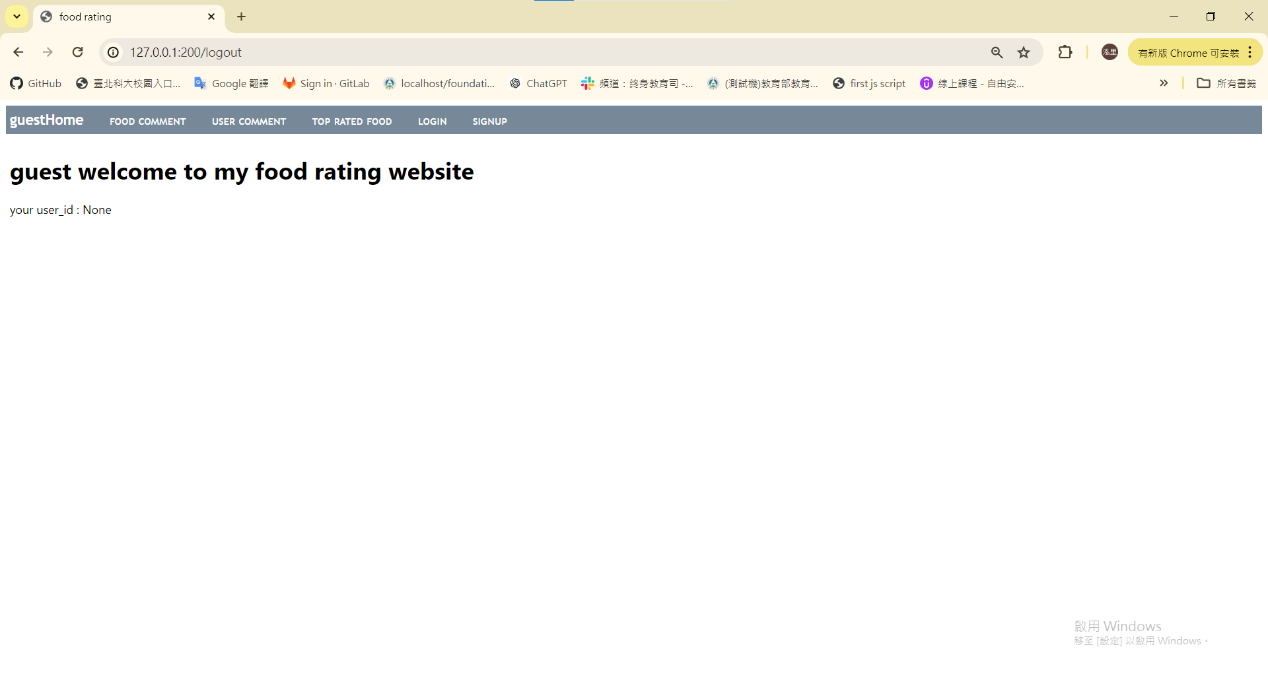
cur.execute("UPDATE Rating SET score=?, comment=?, date=?, pic=? WHERE rowid=?", (score, comment, date, pic, rId))

根據rId更新Rating table中的那一row的評分、評論、日期、照片

cur.execute("SELECT \*,COUNT(Rating.rowid), SUM(Rating.score) FROM Food, Rating, Store WHERE Store.rowid = Food.food\_sId AND Food.rowid = Rating.fId GROUP BY Food.rowid")

根據美食id分組並計算每一組內的總數以及加總評分並列出相關店家資訊

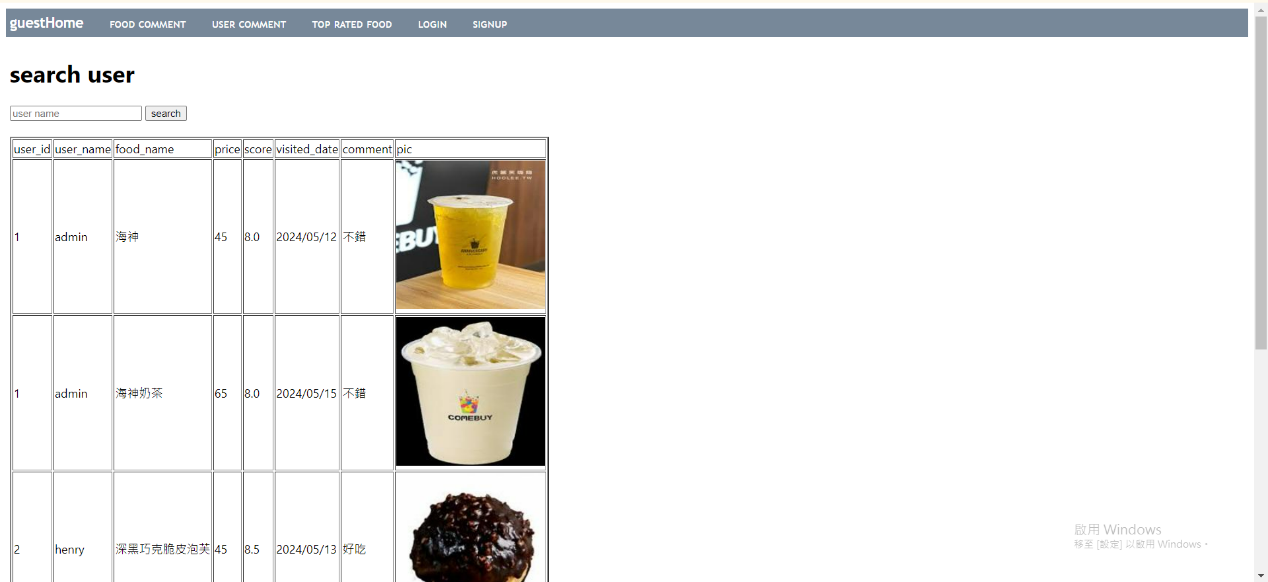
1. 程式執行的螢幕截圖與功能解說

 歡迎頁面、搜尋美食評論之頁面、搜尋使用者評論頁面、熱門美食排行榜頁面於遊客、會員、管理員三個介面大同小異所以由遊客介面來代表展示

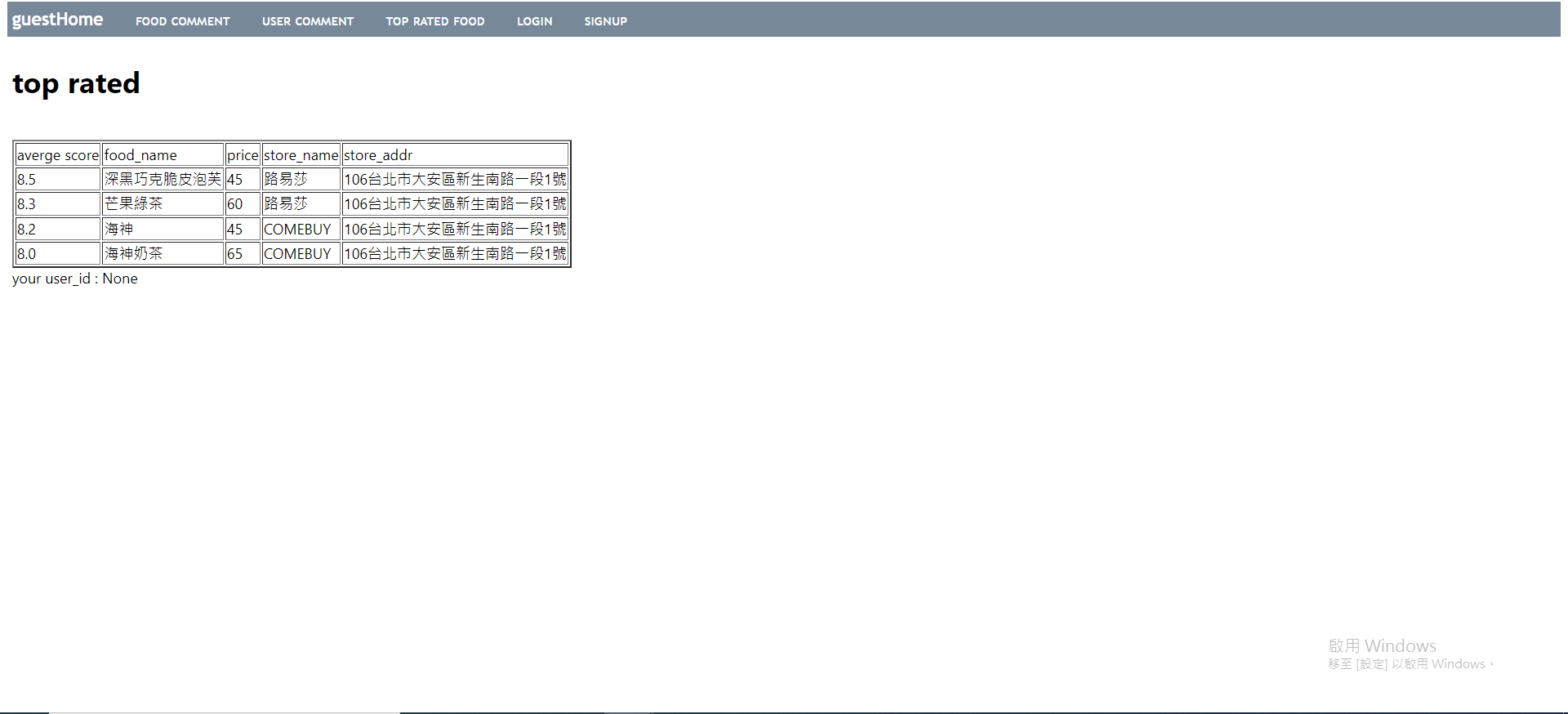
圖四、歡迎頁面

 在美食查詢的部分，使用者可以看到所有美食的評論且在最上方有一個搜尋欄，使用者可以輸入美食的關鍵字搜尋想尋找的美食的評論

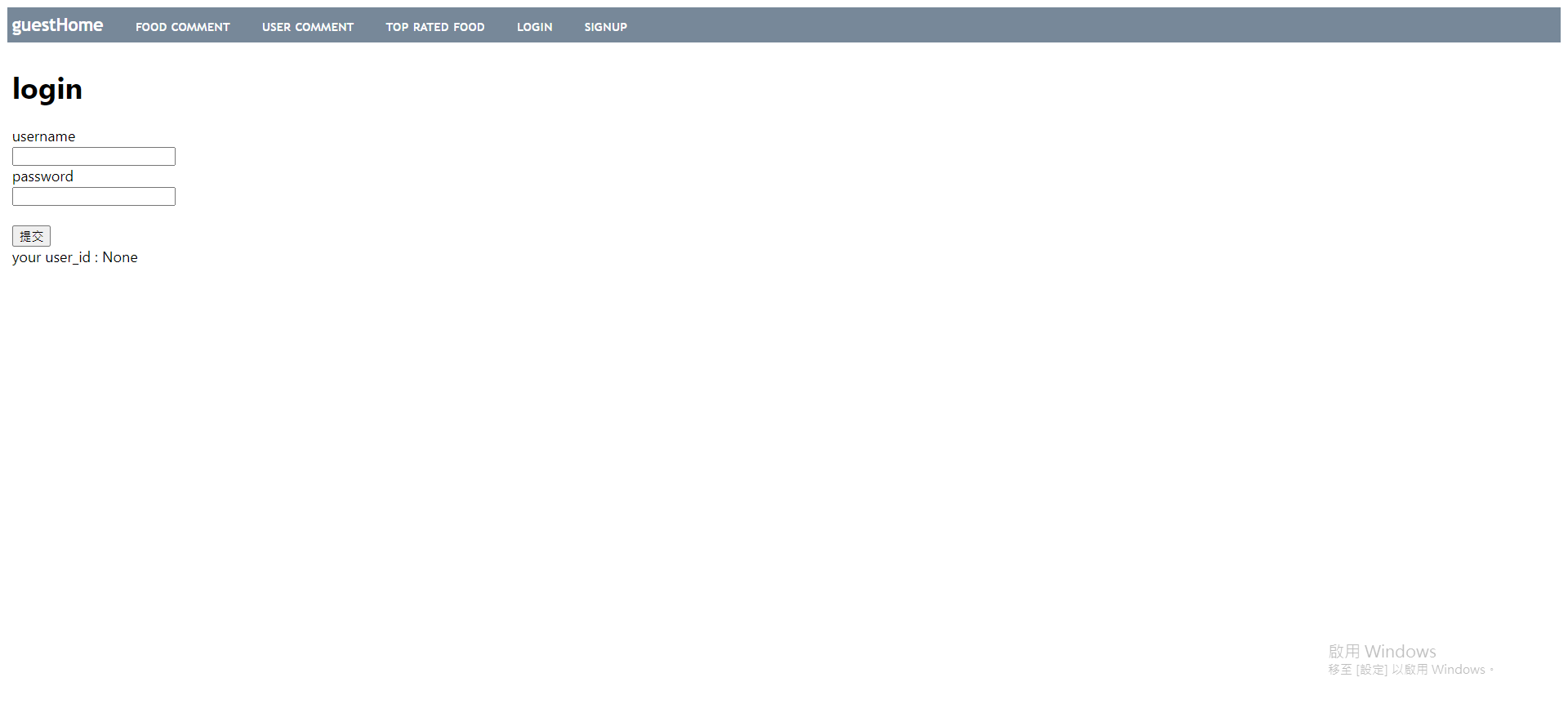
圖五、搜尋美食評論頁面

 在使用者查詢的部分，使用者可以看到所有使用者的評論且在最上方有一個搜尋欄，使用者可以輸入使用者部分名稱搜尋想找的使用者的評論

圖六、搜尋使用者評論頁面

 在最熱門美食排行榜的部分，系統會把帶有評分的美食由高分到低分自上而下列出來，若是有兩位以上使用者對同一種美食評分，系統則會將所有使用者的評分做平均再進行評分排列

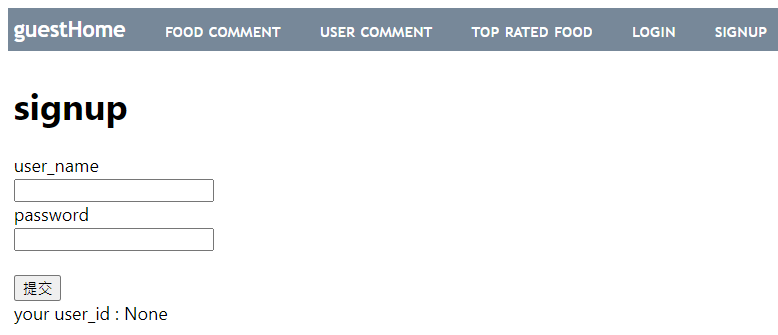
圖七、熱門美食排行榜頁面

 在登入頁面若輸入正確的使用者名稱與密碼可以進入到會員頁面，若登入的是管理員帳號則會進入到管理員介面

圖八、登入頁面

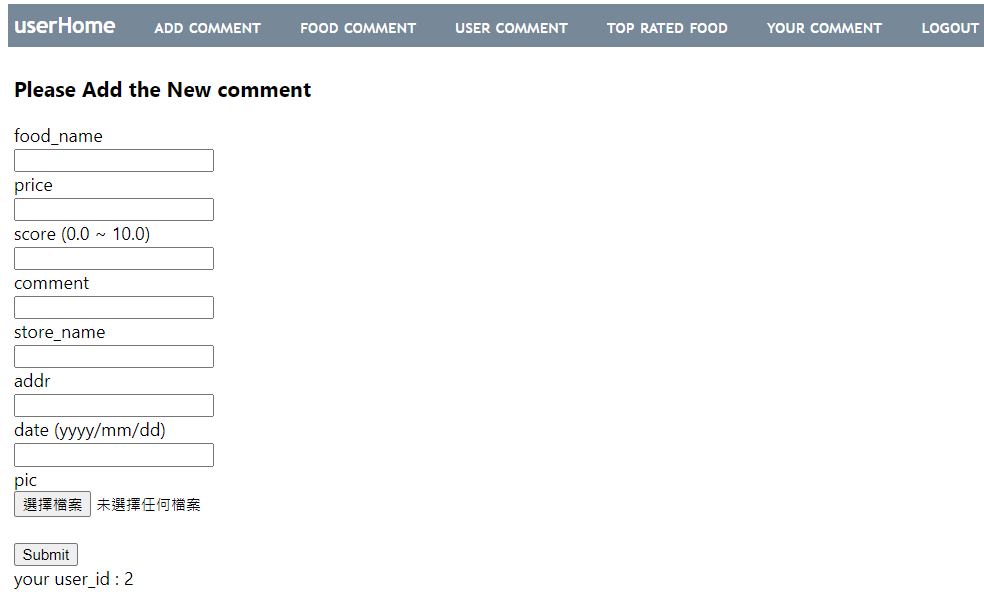
 若輸入的使用者名稱與密碼不在資料庫內會跳出請重新登入或請先註冊

圖九、登入錯誤頁面

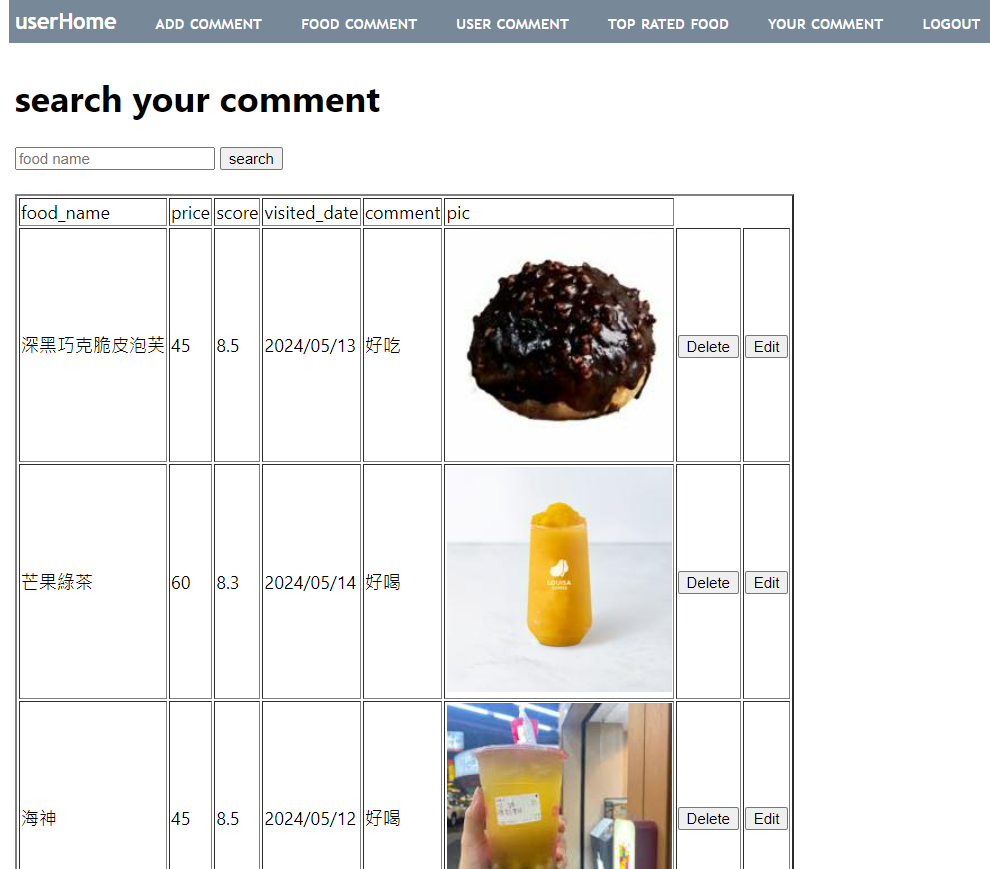
在註冊頁面遊客可以註冊成為會員

圖十、註冊頁面

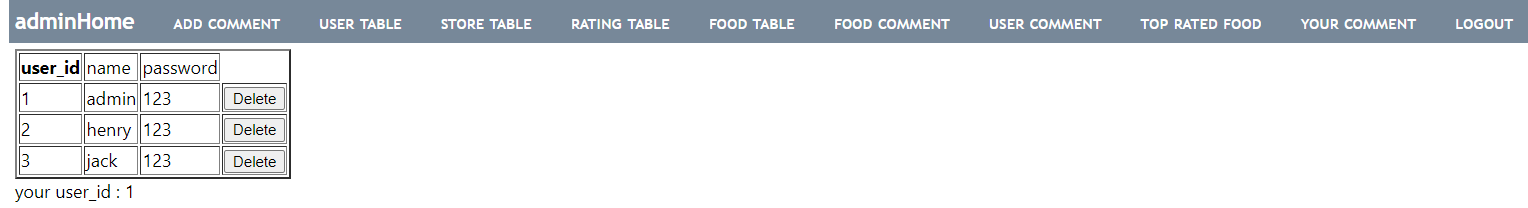
新增美食評論頁面、搜尋自己評論頁面於會員、管理員二個介面基本一樣所以就由會員介面來代表展示。

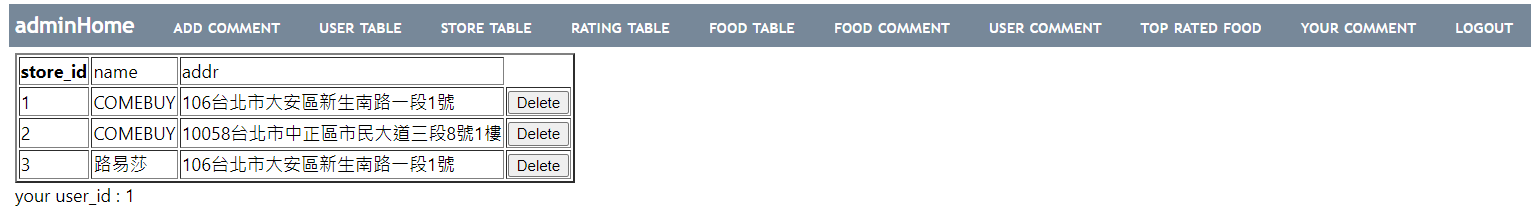
 在新增評論的部分，使用者可以新增的資訊有美食的名稱、價格、評分、評論、店家名稱、店家地址、造訪日期、照片

圖十一、新增美食評論頁面

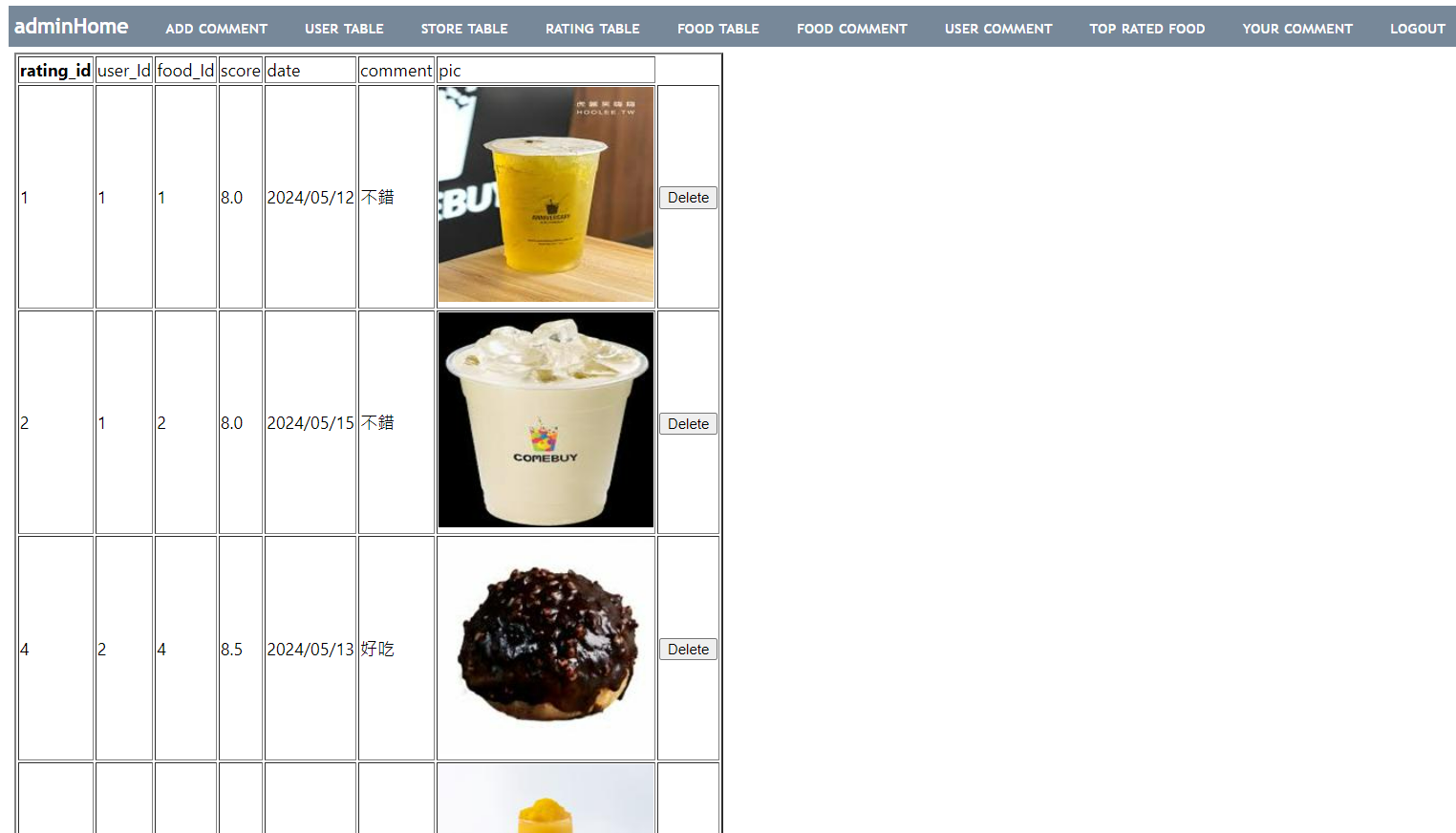
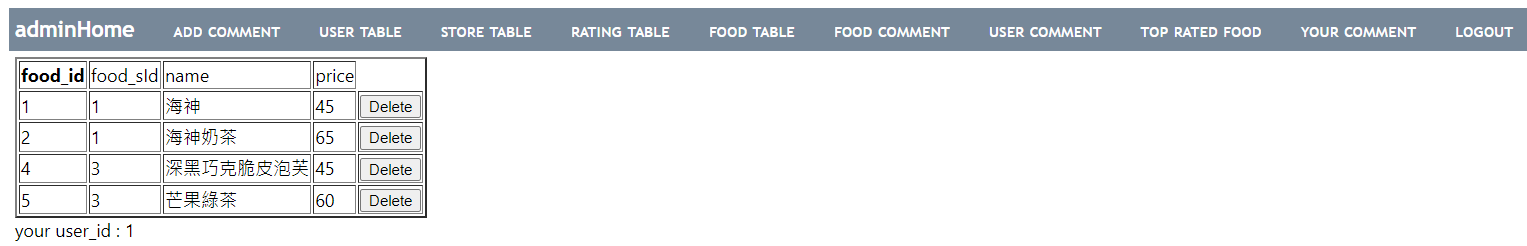
 在搜尋自己評論頁面，使用者可以看到所有自己過往的評論且在最上方有一個搜尋欄，使用者可以輸入美食關鍵字搜尋想尋找曾經評論過的美食，且若不滿意可以進行刪除或修改評論

圖十二、搜尋自己評論頁面

 查看Table頁面，是管理員特有的權限，管理員也能刪除table中的row

圖十三、會員(User)table展示頁面

圖十四、店家(Store)table展示頁面

圖十五、評分(Rating)table展示頁面

圖十六、美食(Food)table展示頁面

若會員或管理員登出後按上一頁想繼續操作會挑出存取遭拒，必須再登入以獲得權限繼續操作

圖十七、存取遭拒頁面