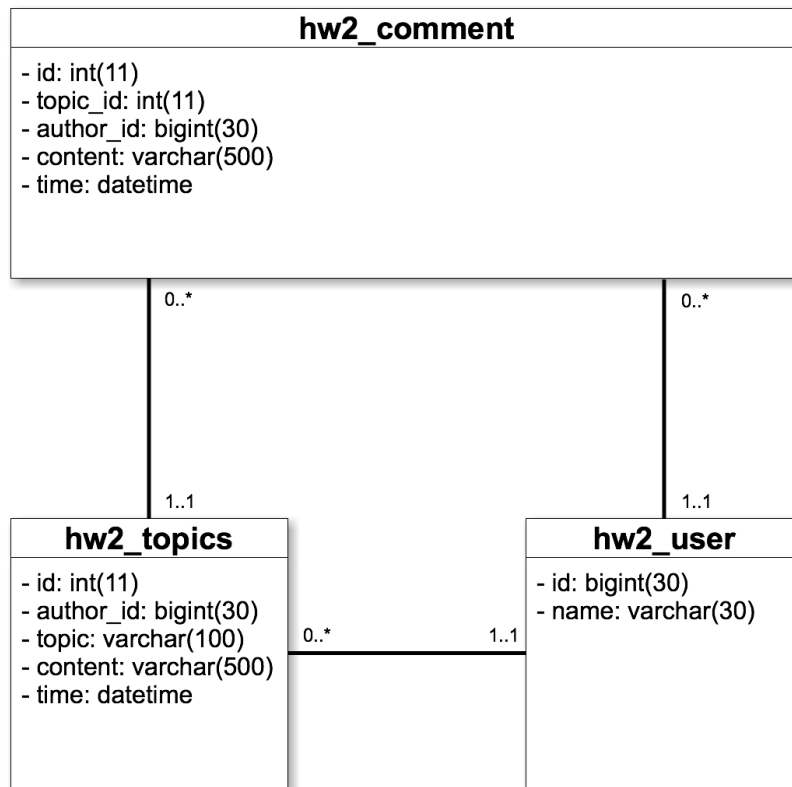


# SDM HW2

R03725029 資管碩一 何均晏

## 一、資料庫結構（以UML圖呈現）



資料庫裡主要含有三個資料表，也是這個論壇系統主要的角色，分別是儲存所有討論主題的 `hw2\_topics`、儲存所有評論的 `hw2\_comment` 與儲存使用者資訊的 `hw2\_user`。

### (a) Entity：

- **`hw2\_topics`**：記錄了討論主題的 id (primary and unique)、該討論串的作者 id (`author\_id` — foreign key from `hw2\_user`.id)、主題名稱 (topic)、討論內容 (content) 與討論發佈的時間 (time — 以server時間為主)。
- **`hw2\_comment`**：記錄了評論的 id (primary and unique)、該評論屬於的討論主題的 id (topic\_id — foreign key from `hw2\_topics`.id)、該評論的作者 id (author\_id — foreign key from `hw2\_user`.id)、評論的內容 (content) 與評論發佈的時間 (time)。
- **`hw2\_user`**：記錄了使用者的 id (primary and unique) 與使用者名稱(name)。

### (b) Relation :

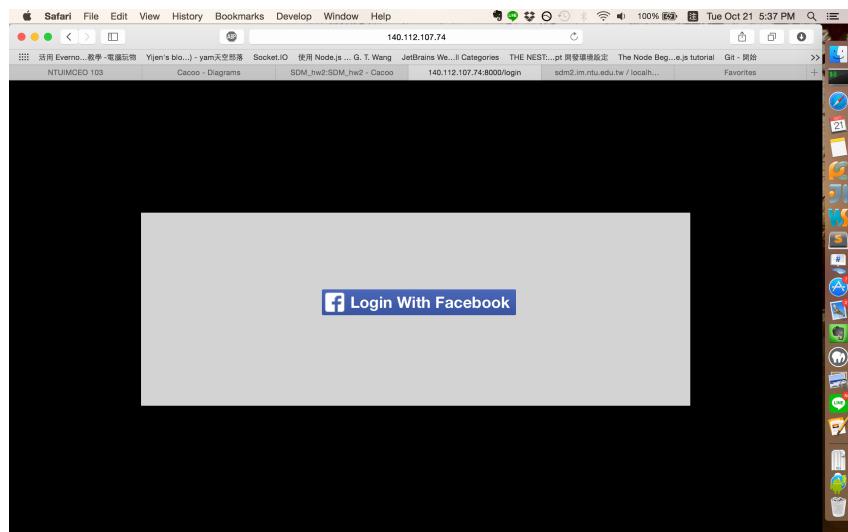
- 每個討論主題中，一定會有作者，而且只會有一個作者；對於該主題的評論，可能不會有，也可能會收到許多不同使用者的評論。
- 每個評論只會對應到某一篇主題（唯一），且只會有一個作者。
- 每個使用者可能會有多篇主題，也可能會有多個評論，也可能沒有主題也沒有評論。
- 藉由以上的關係，我們可以將資料庫裡的資料表，以UML的形式呈現如上圖。

## 二、系統設計理念

此系統主要的設計是為了解使用者能夠在網頁上發佈主題，並且能自由地給予所有相關的評論，並且能夠即時的與其他使用者互動。

### • 登入系統

登入系統主要透過外嵌 Facebook 的 API 來實作，因為現在大部份的使用者都擁有至少一個的 Facebook 帳號，為了讓使用者能有更迅速與簡單的方式就能夠登入使用這個網頁論壇，登入系統選擇以 Facebook 作為媒介以拉近與使用者之間的距離。



（圖）系統登入畫面，以簡潔聚焦的方式呈現。

### • 操作介面

一登入（經過 Facebook 與伺服器的驗證之後）就會馬上進到即時論壇網頁系統的首頁，此論壇的使用 UI 設計上非常的直覺，讓使用者一眼就能夠掌握各項所需的基本功能。

舉例來說，網頁最上方很清楚的顯示發佈新主題的提示，以及各篇主題下面都有直接的輸入框能夠讓使用者快速的回應主題。而網頁的最右上角”首頁”的按鈕讓使用者能夠快速地回到最新的主題，“登出”的按鈕確保使用者能夠正確的登出。

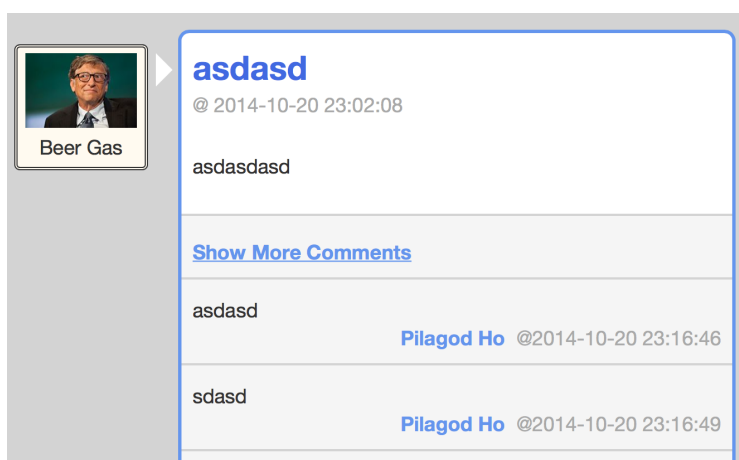


(圖) 系統首頁

- 動態讀取主題與評論

為了讓使用者能有最佳的瀏覽體驗，使用者一登入頁面只會先讀取前10篇的主題，當使用者將視窗往下滾到網頁底部時，系統會判斷並且自動再讀取額外新的10篇主題。

在評論的方面也是，每一篇主題最初只能看到底下最新的五則評論，如果想要看到更多的評論，可以透過按下頁面上“Show More Comments”的提示按鈕，讀取更多相關的評論。



(圖) 動態讀取評論示意圖

- 使用者能夠刪除自己的主題與評論

屬於自己的主題或是評論，都能在右上角找到“X”的按鈕，並在按下後會與使用者確認是否刪除後，若是，會一併將資料庫內的資料刪除，並且移除畫面上的顯示，讓使用者能更有彈性的運用。

### 三、實作細節

在登入網頁時，server會透過 `page_load(...)` 函式，將 `topic` 相關資料讀取出來，為了資訊操作的方面性，我在資料庫額外建立了 ``topic_data`` (combine

`hw2\_topics` and `hw2\_user`) 與 `comment\_data`(`hw2\_comment` and `hw2\_user`) 的 view，負責所有與 SELECT 有關的 query。

新增主題與評論時分別透過 `add_topic(...)` 與 `add_comment(...)` 先將資料新增至資料庫，再回傳給網頁做即時的更新。在新增時必須將資料 INSERT 到資料庫的實體資料表 (無法對VIEW新增資料)，並且會有許多foreign key的限制要注意，如下面以粗體表現的地方。

- 以下 **code** 節錄至 `server.js:199-201`

```
function add_topic(...){
  ...
  connection.query(
    "INSERT INTO `hw2_topics`(topic, author_id, content, time) VALUES" +
    "(" + data.topic + ", (SELECT id FROM `hw2_user` WHERE `id`=" +
      data.author_id + "), " + data.content + ", " + time + ")",
    ...
  );
  ...
}
```

#### 四、備註

- 實作環境：  
Server : nodejs  
Client : html + javascript + css + jQuery  
Database : MySQL (on sdm2.im.ntu.edu.tw)
- Reference :  
FB API - <https://developers.facebook.com>  
Socket.io - <http://socket.io>  
Express - <http://expressjs.com>