
Message Board

Embedded System Lab1 Report

第8組 電機四 b01901160 蘇上育 電機三 b02901102莊世聿

Message Board

Nickname

0/10

Password

0/10

REGISTER

LOGIN

Message

0/150

SEND

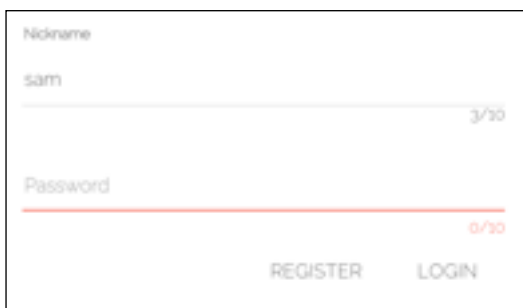
SHOW

select ▼

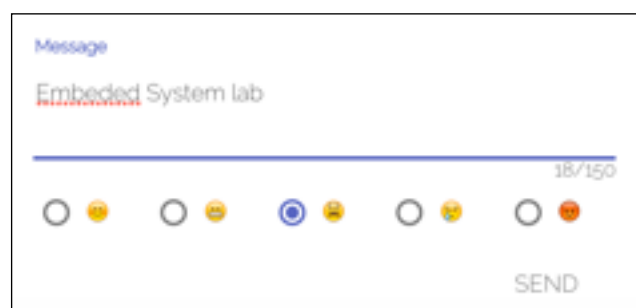
報告

在前端的部分，我們將助教原本的程式碼整合進AngularJS這個框架中，一開始會想要整合進這個框架其實是因為想要美化UI，那剛好在survey的過程中看到Google 在推廣的Material Design的概念，所以我們除了整合進AngularJS這個框架外，在介面設計的部分我們也導入Angular Material這個framework。

雖然在之前有寫過Angular的經驗，但是這是第一次使用Angular Material，在使用的過程中還是遇到一些無法預期的問題，官方的document也不算健全，一些attribute 都不太知道其功能，後來靠著一些測試還有搜尋網路上的demo才慢慢做得比較好一點。



A screenshot of a web form for user registration and login. It features two input fields: 'Nickname' with the value 'sam' and a character count '3/30'; and 'Password' with a character count '0/30'. Below the fields are two buttons labeled 'REGISTER' and 'LOGIN'.



A screenshot of a web interface for sending a message. It shows a 'Message' input field with the text 'Embedded System lab' and a character count '18/150'. Below the input is a row of six circular mood selection buttons, each with a different colored face. A 'SEND' button is located at the bottom right.

首先整個網頁大概分成兩個部分："request-zone"，"response-zone"。前者"request-zone"是負責向後台發送request，所包含的元件有："userLogin"包含讓使用者註冊以及登入的部分，並且在前端檢查使用者名稱是否合法，合法才能夠註冊以及登入，呼叫http_post此函數發送'/register' '/query'這兩種request。"LeaveMessage"讓使用者輸入信息內容，按下send會經過檢查內容是否合法以及是否已經登入。"moodContainer"讓使用者選擇一種心情。

後者"response-zone"包含顯示後台response的東西，除了訊息內容外，還有右上方select可以看到目前已經註冊的使用者們，展開list會呼叫http_post此函數並發送'/list'這個request。選擇一個使用者後按下Show會呼叫http_post此函數發送'/showOnly'這個request，可以單獨顯示此使用者先前在留言板上留下的內容。最右上方的refresh button在點擊後會呼叫http_post此函數發送'/refresh'這個request可以顯示目前資料庫中所有留言。



A screenshot of a web interface showing a list of users. At the top, there is a 'SHOW' button, a 'select' dropdown menu, and a refresh icon. Below these is a large rectangular area, likely a list or message display, which is currently empty.

加分項目中我們總共實做了註冊登入以及單獨顯示某使用者先前留過的留言。

後端

後端的工作相當繁多，大概可以粗略地分成資料庫內容存取、send http_post的response兩個類別。

(一) 資料庫存取

先從資料庫存取講起，我們程式有data.db以及user.db兩個資料庫，一個是負責存放使用者送出的訊息，另一個則是負責存放已經註冊的使用者資料。在app.js裡面的253~281行的code做的事情就是把這兩個檔案的資料讀取出來並分別儲存到DataBase、UserData這兩個變數(陣列)裡面。而在283行的save_data這個函式則是負責把資料存到檔案裡。

(二) send http_post response

Send http_post的response則包含了這份程式大部分的工作，針對不同的request server會做出相對應的response，在app.js的一開頭部分就已經設定，其中echo、submit、read_all、refresh是屬於基本要求的部分，register、query、showOnly、list則是針對額外加分的功能所設定的。

(1) echo

echo是收到前端的request後，將前端送進來的request parse成一個JSON的物件，其中包含nickname、emoji、message、timestamp等member，其中timestamp一開始會是空字串，在app.js收到這個request後才將當前的時間附加上去，之後這一個完整的JSON物件會先放進DataBase這個陣列裡面、再存放到data.db中，並且送回附有timestamp的JSON到前端。

(註:這邊並不用做任何error的handle，因為在前端已經做過了)

(2) submit

submit所做的事情大概和echo差不多，比較不同的部分就是需要做error的handle，如果格式不正確就要response error的訊息，如果格式正確就把資料放進DataBase這個陣列裡面並存進data.db中。

(3) read_all

read_all就是把DataBase裡面的東西response出去。

(4) refresh

功能和read_all相似。

(5) register

這個是我們實作的註冊功能，收到這個request後會把request parse成一個JSON物件，裡面包含nickname以及password這些內容(string)，註冊的時候我們會做一些檢查，如果使用者名稱已經註冊過的話就會註冊失敗，如果輸入密碼的長度為零的話也會註冊失敗，無論失敗成功都會送一個response給前端，是一個包含ok (是個bool)、reason(是個string)的JSON物件，如果ok是true的話reason不會有任何訊息，如果ok是false的話reason會給出相對應的error message，註冊成功後會把使用者的資料放進UserData並存進user.db中。

(6) query

query是針對前端要確認這個使用者有沒有註冊過、能不能登入的response，前端會送進一組使用者資料，我們會先進行使用者名稱比對，如果這個使用者名稱不存在資料庫中的話就會給出please sing up first這個error message，如果這個使用者名稱存在於資料庫中但是密碼不正確就會給出password error的訊息，如果使用者名稱密碼都正確就會讓使用者正確登入。和register一樣無論成功與否都會送出一個response包含JSON物件內存ok、reason。

(7) showOnly、list

這兩個request handler是必須一起存在的，list是要讓前端能獲得現在的使用者名單，而showOnly則是根據前端request的使用者名稱、給出那個使用者的所有訊息，兩者的response內容都是一個陣列。