

Pràctica
CFGS DAW – M06 – Desenvolupament web entorn client
UF2
Entregables: Fitxer en format .zip : M06UF2_Practica_[Nombre_Apellido].zip
DATA LÍMIT D'ENTREGA: Mirar el pou de salenet

Consideracions inicials:

- No poseu el vostre nom entre [], aquests símbols són per indicar quina part del nom del fitxer s'ha de modificar.
- Necessitareu el Xampp o un altre programa similar per a poder executar un servidor local.
- Feu servir el fitxer sql que hi ha al mateix zip per a crear una BBDD dins del gestor PhpMyAdmin.
- Per comprovar la api podeu fer servir l'eina POSTMAN.
- Feu servir la api_chat com a servidor per a interactuar amb la BBDD.
- Aquesta pràctica es fa per parelles.
- La pràctica es puntua sobre 6, però el 5 segueix sent l'aprobat. La resta de punts fins arribar al 10 seran repartits de forma següent:
 - La bona presentació visual (fer servir sass i algun framework de css) de la pràctica contarà 1 punt.
 - La qualitat del codi i comentar-lo correctament contarà 1 punt.
 - Utilitzar correctament git i afegir en l'entrega una captura dels commits realitzats durant la pràctica, contarà 1 punt.
 - Fer funcionalitats extres contarà 1 punt.
- Aquests 4 punts no es sumaran si no s'obté mínim un 4 en la pràctica.
- Hi haurà un seguiment setmanal, si cada setmana no s'ha avançat suficient, restarà 0,25 de la nota.

Pràctica - Xat

Crea una web que contingui un xat. Per fer-ho utilitzeu la API adjuntada en mateix zip que aquest enunciat, la qual farà la funció de servidor, només caldrà fer la part del client.

Funcionalitats a implementar:

Mínim (6p)

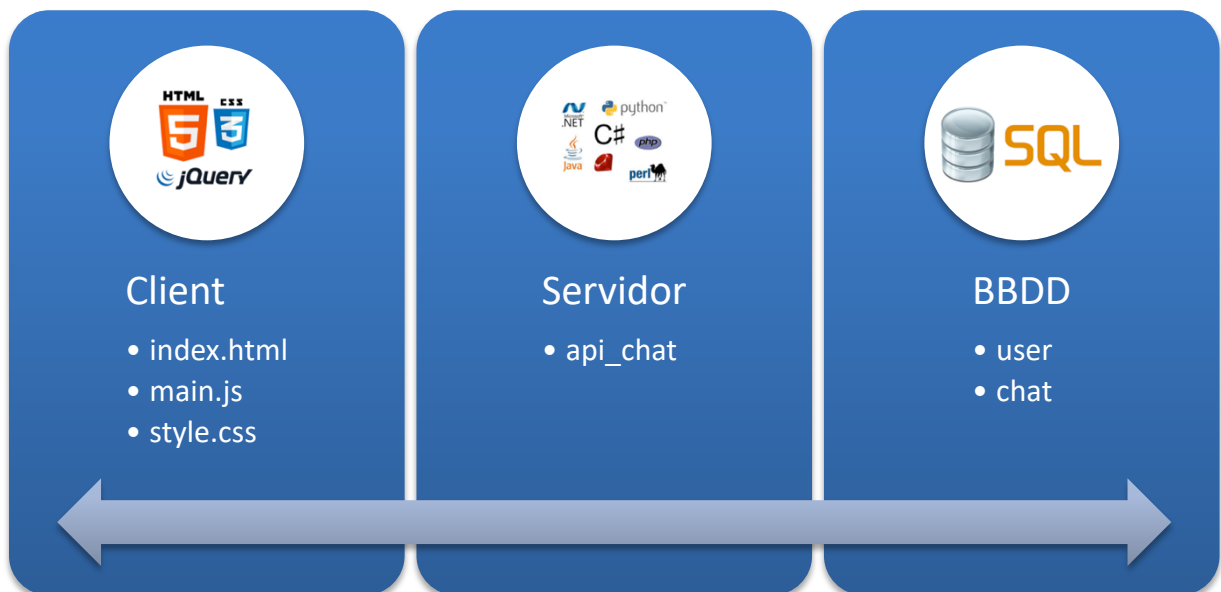
1. Crea un div que contingui un camp de text, que serà on escriurem el nom del nostre usuari (com si fos un login).
2. Crear un div que contingui un buscador d'usuaris.
 - a. Ha de ser un camp de text que mentre es vagi escrivint el nom de l'usuari, ha de mostrar tots els usuaris que coincideixin amb el nom escrit.
 - b. Es igual com mostreu les opcions, en un div, en el mateix camp de text, ... mostreu-ho com us sembli millor.
3. Crear un div lateral que tindrà una llista de totes les converses (chats) del nostre usuari.

- a. En la llista s'ha de mostrar l'altre usuari amb qui està oberta la conversa, i un resum (tros de text acabat en "...") de l'últim missatge.
4. Crear un div que contingui els missatges enviats i rebuts amb l'usuari, i un camp de text per enviar nous missatges.
 - a. Quan l'usuari cliqui un botó o el "enter" del teclat, s'ha d'enviar el missatge al servidor, guardar-lo a la bbdd i mostrar-lo als missatges enviats i rebuts.
5. Fer que la funció ajax demani al servidor cada pocs segons si hi ha alguna actualització dels missatges del xat. Si hi és s'ha d'actualitzar els missatges enviats i rebuts.

Possibles extres

6. Que no es mostri el formulari d'escriure usuari a qui enviar missatge fins que no t'has loggejat, i que no es mostri el formulari d'escriure cap missatge fins que no has seleccionat una conversa.
7. Millorar el login, posar-lo en una altra pàgina i que el comprovi amb un password.
8. Afegir un camp a la taula de missatge que sigui "vist" i si l'usuari ha vist el missatge que es mostri a la pàgina.

Anomeneu els fitxers tal com surt al diagrama i creeu-ne més si ho considereu oportú.



Funcionament de la API

La api tindrà les següents rutes que ens retornaran la següent informació:

- **Ruta de test**
 - URL: `.../api_chat/public/`
 - Tipus de petició: GET
 - Retorn:
 - Mostra per pantalla "Hello world!"
 - Variable json `{"message": "Hello world!"}`
- **Ruta per exportar un id d'un usuari**
 - URL: `.../api_chat/public/getId/{name}`
 - Tipus de petició: GET
 - Dades a enviar: No s'han d'enviar dades amb el mètode GET, sinó que s'ha d'escriure com si fossin part de la URL
 - `{name}`: nom de l'usuari del qual volem saber l'id (sense les claus `{}`)
 - Retorn:
 - Variable json `{"status": "OK", "result": id de l'usuari}`
 - Variable json `{"status": "ERROR", "result": missatge d'error}`
- **Ruta per exportar una llista d'usuaris a través d'una part del nom**
 - URL: `.../api_chat/public/getUsersByName/{name}`
 - Tipus de petició: GET
 - Dades a enviar: No s'han d'enviar dades amb el mètode GET, sinó que s'ha d'escriure com si fossin part de la URL
 - `{name}`: nom o part del nom de l'usuari que estem buscant (sense les claus `{}`)
 - Retorn:
 - Variable json `{"status": "OK", "result": array de variables json amb els ids i els noms dels usuaris}`
 - Variable json `{"status": "ERROR", "result": missatge d'error}`
- **Ruta per comprovar si un usuari i password són correctes**
 - URL: `.../api_chat/public/checkPassword/{id}/{password}`
 - Tipus de petició: GET
 - Dades a enviar: No s'han d'enviar dades amb el mètode GET, sinó que s'ha d'escriure com si fossin part de la URL
 - `{id}`: id del usuari que es vol comprovar (sense les claus `{}`)
 - `{password}`: contrassenya del usuari que es vol comprovar (sense les claus `{}`)
 - Retorn:
 - Variable json `{"status": "OK", "result": id de l'usuari trobat}`
 - Variable json `{"status": "ERROR", "result": missatge d'error}`
- **Ruta per enviar missatge**
 - URL: `.../api_chat/public/sendMessage`
 - Tipus de petició: POST
 - Dades a enviar:
 - `"sender"`: id de l'usuari que envia el missatge
 - `"receiver"`: id de l'usuari que rebrà el missatge
 - `"message"`: Text del missatge

- Retorn:
 - Variable json {"status": "OK", "result": missatge de confirmació}
 - Variable json {"status": "ERROR", "result": missatge d'error}
- **Ruta per exportar una conversa (chat)**
 - URL: .../api_chat/public/getChat/{sender}/{receiver}
 - Tipus de petició: GET
 - Dades a enviar: No s'han d'enviar dades amb el mètode GET, sinó que s'ha d'escriure com si fossin part de la URL
 - {sender}: id de l'usuari que envia el missatge (sense les claus {}), o sigui l'usuari loguejat.
 - {receiver}: id de l'usuari que rebrà el missatge (sense les claus {})
 - Retorn:
 - Variable json {"status": "OK", "result": variable json amb tota la conversa}
 - Variable json {"status": "ERROR", "result": missatge d'error}
- **Ruta per exportar una llista dels últims missatges de totes converses (chats) de l'usuari**
 - URL: .../api_chat/public/getAllResumeChats/{user}
 - Tipus de petició: GET
 - Dades a enviar: No s'han d'enviar dades amb el mètode GET, sinó que s'ha d'escriure com si fossin part de la URL
 - {user}: id del nostre usuari (sense les claus {})
 - Retorn:
 - Variable json {"status": "OK", "result": array de variables json amb l'últim missatge de totes les converses i els usuaris}
 - Variable json {"status": "ERROR", "result": missatge d'error}
- **Ruta per exportar una llista dels usuaris**
 - URL: .../api_chat/public/getAllUsers
 - Tipus de petició: GET
 - Retorn:
 - Variable json {"status": "OK", "result": array de variables json amb tots els ids i noms dels usuaris}
 - Variable json {"status": "ERROR", "result": missatge d'error}
- **Ruta per exportar els nous missatges d'una conversa (chat), només exporta els que no estan vistos.**
 - URL: .../api_chat/public/getChatNotSeen/{sender}/{receiver}
 - Tipus de petició: GET
 - Dades a enviar: No s'han d'enviar dades amb el mètode GET, sinó que s'ha d'escriure com si fossin part de la URL
 - {sender}: id de l'usuari que envia el missatge (sense les claus {}), o sigui l'usuari loguejat.
 - {receiver}: id de l'usuari que rebrà el missatge (sense les claus {})
 - Retorn:
 - Variable json {"status": "OK", "result": variable json amb tota la conversa}
 - Variable json {"status": "ERROR", "result": missatge d'error}