HTML file

index.html- registration page user interface

login.html- login page

JavaScript file

|  |  |
| --- | --- |
| file name | purpose |
| login.js | after login,   1. send credential to login.php 2. active getVerify(), if login, we should pass user to addnote.php, where a platform to add/delete, edit note |
| addNote.js | 1. control logic of addnote.php, which is an user interface to note management platform 2. CRUD and presentation of the note are controlled from here use for loop to generate html code that present the note and button underneath in a dynamic way |
| js.js | 1. data validation in captial letter, password length, confirm password check 2. fetch of verified data to register.php by using a POST method 3. writing a greeting message with username, so getusername() first fetch a request to getUsername.php, a php that retrieve username from database, and then getusername() pass the username to success to do simple amendment on greeting msg and print it out |
|  |  |
|  |  |
|  |  |

-

PHP file

|  |  |
| --- | --- |
| file name | purpose |
| connect.php | it is a PDO that connect the database |
| login.php | 1. Filter the incoming message 2. validate if there is any matching email and pw in db, if match, throw back message to login.js 3. if verification success, assign email to session |
| addnote.php | user is bought from login.js, where they are all verified as user.   * this is a platform to manage your notes. it shows a html interface for user. Buttons’ logic is controlled by addNote.js * php part of the code open session and verify the session validity |
| Register.php | 1. filter the incoming msg 2. validate if there is any matching email in db before insert credentials to db 3. writing a greeting message with username, so getusername() first fetch a request to getUsername.php, a php that retreive username from database, and then getusername() pass the username to success to do simple amendment on greeting msg and print it out |
| getUsername.php | 1. after receive request from js.js, use SQL command to inquire the db for username 2. send username after json encode |
| throwNote.php | php part verify the session is correct, then start. this page receive message from addNote.js, line 78, a fetch function active after save button pressed, then we throw newly created note to db, and reply |
| updateNote.php | session verification and then find the matching note to update |
| Logout.php | Log out and end the session |
| getNote.php | \* 1. to retreive exisitng id and content by searching notes all under the email in session  \* 2. throw back the notes in [] to loadNotes() in addNote.js, where raise the request |
| DeleteNote.php | delete the note according to the id received vs id in db |