**Names:**

Aya aly saad aly hassan abouzeid

(2)

Amira nabil shaaban abdelnaby haiba (22)

Web page link

https://to-do-list-aya-abouzeid.c9users.io/interface

**How code is organized**

An Interface webpage with a log in and a signup button ,

Together with the ToDoList tasks webpage , each has their own :

Style

Script contains all functions

Html body

In phase 2 , a .JS file was added , this enabled operating the webpage through the server and dealing with the backend side of the server ,

A JSON file is also included to save all the webpage and signed up users data so we can recall it at anytime when the user log in again

Ajax was also used to keep all the users information and save them when he refreshes the page or open a different tab in the browser ,in general , it saves the users action instantaneously after each operation done to keep the users data saved.

Validations were added to the interface webpage which validates the user’s input while signup and check if his information is valid for a new account or not.

Signing up saves the users info in the Json file ,together with all the info in his webpage , which can be retrieved later when signing in.

A Logout button was added to phase1 webpage which enables the user to signout of his account and return to the interface webpage.

**Main functions**

1. Show()
2. Hide()
3. Add()
4. Newtask()
5. Deletefromall()
6. CtoA()
7. CtoP()
8. PtoA()
9. PtoC()
10. AtoP()
11. AtoC()
12. Deletearch()
13. Deleteprogress()
14. Deletecompleted()
15. Drawall()
16. Drawprogress()
17. Drawarch()
18. Drawcompleted()
19. Editarch2()
20. Editall2()
21. Editprogress2()
22. Editcompleted2()
23. Edit()
24. Editarch(i)
25. Editprogress(i)
26. Editcompleted(i)
27. Edit1(i)
28. Edit2(i)
29. Edit3(i)
30. Show2()
31. Hide2()
32. Sortbyname ()
33. Sortbydate ()
34. AllToP(i)
35. AllToC(i)
36. AlltoA(i)
37. $(document).ready(function()
38. function save()
39. function newtask(id,name,date,description,status)

**Show function  :**

It shows the dialog where we enter the name of the task , the date and the description after clicking on “ add task “ button

**Hide function :**

It hides the dialog when clicking on “submit” button

**Add function :**

It saves the data token from the dialog and calls and adds it in All tasks table and In progress table

**Newtask function :**

It gives the data an id , name , date , description and status

**Deletefromall function :**

It deletes a task from “ All Tasks “ table when clicking on “delete task “

It also deletes the task from in progress table or archived table or completed table

**CtoA(i) :**

 Replaces the task from completed table to archived table

**CtoP(i)**

 Replaces the task from completed table to in progress table

**PtoA(i)**

 Replaces the task from in progress table to archived table

**PtoC(i)**

 Replaces the task from in progress table to completed table

**AtoP(i)**

 Replaces the task from archived table to in progress table

**AtoC(i)**

 Replaces the task from archived table to completed table

**Deletearch(i)**

Delete element from archived table

**Deleteprogress(i)**

Delete element from progress table

**Deletecompleted(i)**

Delete element from completed table

**Drawall()**

It draws the table that contains all tasks

**Drawprogress()**

It draws the table that contains tasks in progress only

**Drawarch()**

It draws the table that contains archived tasks only

**Drawcompleted()**

It draws the table that contains completed tasks only

**Edit()**

Append the new values after editing to the specified table

**Editall2()**

Edits any task at the all tasks table and it's original table as well

**Editarch2()**

Edits archived tasks from archived table and all tasks table as well

**Editprogress2()**

Edits tasks in progress from in progress table and all tasks table as well

**Editcompleted2()**

Edits completed tasks from completed table and all tasks table as well

**Editarch(i)**

**Editprogress(i)**

**Editcompleted(i)**

**Edit1(i)**

**Edit2(i)**

**Edit3(i)**

All of them call the function show2() and assign the variable I to a global variable

**Show2()**

It shows the edit dialog

**Hide2()**

It hides the edit dialog

**Sortbyname ()**

It sorts the elements of the current table by name

**Sortbydate ()**

It sorts the elements of the current table by date

**AlltoP()**

Moves the task if it's archived or completed to progress table from the all tasks table

**AlltoC()**

Moves the task if it's archived or In progress to completed from the all tasks table

**AlltoA()**

Moves the task if it's completed or in progress to archived from the all tasks table

**$(document).ready(function()**

it redirects to the url: '/datain' and if success function data is executed which equalize the new data entered to the json object

**function save()**

it saves the added tasks of specific user in json file

**function newtask(id,name,date,description,status)**

gives the values of the id , name , date , description , status the equivalent current information

Main Functions phase II :

**app.post('/datain', function (req, res)**

a .JS function which reads the data in the .JSON file and parse it ,then responds with the data stored when for example refreshing the web page or opening a new tab.

**app.post('/dataout', function (req, res)**

a .JS function which takes all the data sent from the webpage including the arrays containg tasks and every thing related to the user , and then parse the data in the .JSON file then edit it and rewrite the new data in it then stringfy it again.

**app.post('/getout' , function (req,res)**

deletes the session ID and email and logs the user out of his account

**app.get('/interface',function(req,res) & app.get('/',function (req,res)**

if the session is still valid for the user , respond with the taskslist webpage , else respond with the interface webpage.

**app.post('/addUser', function (req, res)**

create a .JSON object for a new user containing his name , email and password , and for every user 4 arrays ,alltasks,inprogress,archived,completed , to save his data in the .JSON file,

the function also validates the new user’s email and name ,checking if there is no user already having these info.

**app.post('/toDoList', function (req, res)**

**Functions written by Aya Aly**

Show()

Hide()

Add()

Newtask()

Deletefromall()

Deletearch()

Deleteprogress()

Deletecompleted()

Drawall()

Drawprogress()

Drawarch()

Drawcompleted()

Editarch2()

Editall2()

Editprogress2()

Editcompleted2()

Edit()

Editarch(i)

Editprogress(i)

Editcompleted(i)

Edit1(i)

Edit2(i)

Edit3(i)

Show2()

Hide2()

AllToP(i)

AllToC(i)

AlltoA(i)

**Functions written by Amira Nabil**

Sortbyname ()

Sortbydate ()

AtoC()

AtoP()

PtoA()

PtoC()

CtoA()

CtoP()

CSS of the project

Writing the report