Hello,
so you found the documentation of the Join project? Great!
We hope you'll find what you're looking for.
If not, feel free to send us a message (see 'impressum.html').
Thank you.

JavaScript Documentation:

Members

`color`

color defines the color assigned to the category of each task;

Source:

board.js, line 4

Methods

`(async) addBacklog()`

addBacklog: Creates arranged HTML code based on alltasks array.

Source:

backlog.js, line 5

`addCurrentUserProfile()`

This function adds name and profile pic from the current user to the myprofile.html

Source:

myprofile.js, line 42

addMyProfileBox()

This function adds information about the current User to the myprofile.html

Source:

[myprofile.js, line 16](myprofile.js)

assignuser()

assignuser: shows available users to assign task to

Source:

[addtask.js, line 20](addtask.js)

(async) BuildMyProfile()

this function initiates building the html and runs the other building functions

Source:

[myprofile.js, line 6](myprofile.js)

checklatesttaskid()

ID counters

Source:

[Server_functions.js, line 159](Server_functions.js)

checkpassword(a, b)

checkpassword: checks, if password matches selected user

Parameters:

Name	Type	Description

a		: selected user that exists.
b		: selected user's id'

Source:

checkUseralreadyexists(name)

function checks if username already exists in database and runs addUser() if not true

Parameters:

Name	Type	Description
name		: name from input field

Source:

checkUsersImage()

This function checks if the user uploaded a profile picture, else this user gets the default picture

Source:

ClearBoard()

ClearBoard: sets all innerHTML to ''

Source:

deleteActiveUser()

deletes activeuser id from localstorage after logout

Source:

deleteAll()

ClearServer: Deletes all saved data from server.

Source:

deleteTask(i)

deleteTask: "Deletes" selected task (via trashbin symbol) from view. Done by assigning new status 'TaskDeleted'. Task is still stored in alltasks!

Parameters:

Name	Type	Description
i	*	: task id

Source:

deleteuserfromtask()

deleteuser: deletes user from currenttasks and initiates view update; Important: This function does not use userid, instead it uses current position in array

Source:

getActiveUserId(account:)

Parameters:

Name	Type	Description
account:		useraccount of user who logged in

Source:

active_user.js, line 7

(async) initBoard()

These functions update the sidebar on the current html

Source:

sidebar.js, line 75

InitLocal()

InitLocal: creates localstorage items (only used at Login)

Source:

localstorage_functions.js, line 49

LoadLocal()

LoadLocal: updates localstorage items

Source:

localstorage_functions.js, line 57

(async) loadSideBar()

Menu is loaded on every HTML file

Source:

LocalLoadUsers()

LocalLoad/LocalSave: Each function specifies targets for setArray() and getArray() and runs them. Those functions are included in other javascripts.

Source:

login()

searchuser(): checks if input user exists

Source:

logout()

Logout: logs user out and resets to login page

Source:

menuClose()

mobile menu close

Source:

menuOpen()

mobile menu is opening

Source:

openMobileMenu()

When click on the 3 stripes the mobile menu is opening

Source:

pickcolor(currenttask)

Color for each Category is choosen

Parameters:

Name	Type	Description
currenttask	*	

Source:

PushTask(id, tasktarget:)

PushTask: initiated by TaskMoveLeft / TaskMoveRight; changes taskstatus to newly assigned status.

Parameters:

Name	Type	Description
id	*	: task id

tasktarget:	*	new task status

Source:

saveActiveUser(activeuser:)

Parameters:

Name	Type	Description
activeuser:		//recieve the activeuser id

Source:

selectuserassign(id:)

selectuserassign: pushes users' id into currenttaskusers array

Parameters:

Name	Type	Description
id:		provided by the image that starts this function

Source:

ServerDeleteAndUpload()

NOT USED: replaces old alltasks with new one from temporary storage

Source:

setActiveUser()

used in server_functions while adding user

Source:

setArray(key, array)

Save to and load arrays from local storage

Parameters:

Name	Type	Description
key		name of array
array	*	name of array (same as key)

Source:

SetLocal()

SetLocal: overwrites old arrays with current arrays in local storage

Source:

SetServer()

first setup only. Not used in standard application.

Source:

TaskMoveLeft(id)

onclick="TaskMoveLeft('taskid')" : task is backrolled in board to next panel to the left, initiates pushfunction to alltasks

Parameters:

Name	Type	Description
id	*	task id

Source:

TaskMoveRight(id)

onclick="TaskMoveRight('taskid')" : task progresses in board to next panel to the right, initiates pushfunction to alltasks

Parameters:

Name	Type	Description
id	*	task id

Source:

TasksAssignedToUser()

this function displays the tasks the user is assigned to at the moment

Source:

TasksCreatedByUser()

this function displays all active tasks the user has created

Source:

myprofile.js, line 60

toBase64()

The file is displayed, so the user can see which picture is uploaded

Source:

adduser.js, line 36

updateBoard()

updateBoard: This function updates all panels

Source:

board.js, line 38

updatenewid()

Update ID functions (might be redundant if functions above are updated!)

Source:

Server_functions.js, line 171

updatetaskusers()

updateusers: displays those users who are assigned to task.

Source:

addtask.js, line 4

UpdateTodo(currenttask:, currentid)

Update"STATUS": the following 4 functions update the assigned divs in view and fill their HTML with tasks from alltasks, if existing

Parameters:

Name	Type	Description
currenttask:	*	task-element that is being processed
currentid	*	id of task that is being processed

Source:

board.js, line 67

(async) uploadImage()

This function converts the full path of the uploaded file and save it to the backend

Source:

adduser.js, line 6