Collaberative Notes Application

Front end flow diagram

Start

index.html
(Login page)

FORGOT PW:
route: /forgot
method: POST

Send to Backend:
username:<string>
email:<email>

LOGIN:
route: /login
method: POST

Send to Backend:
username:<string>
email:<email>
password:<string>

REGISTER:
route: /register
method: POST

Send to Backend:
username:<string>
email:<email>
password:<string>

REMEMBER ME:
route: /remember
method: UPDATE

Send to Backend:
remember:<boolean>

Exists? — Yes → Email new random password & alert user to check email
Exists? — No → No such account alert

Successful? — Yes → Profile created alert
Successful? — No → Username already in use alert
Successful? — No → Email already in use alert

HTML script
Elements that will be displayed in home page on startup and built immediately

GET AVATAR:
route: /avatar/<user_id>
method: GET

Return from Backend:
user_avatar:<string>

GET NOTES LIST:
route: /notes/list/<user_id>
method: GET

Return from Backend:
Array
note_id: <int>
title: <string>
created_on: <timestamp>
last_update: <timestamp>
catagory: <string>

home.html
(homepage)

update home.html

FILTER & SORT NOTE LIST:
route: /notes//list/<user_id>
method: GET

Send to Backend:
user_id:<int>
filterbytitle: <boolean>
filter:<string>
sortbylatest: <boolean>
sortbycatagory: <boolean>

CREATE NOTE:
route: /notes/create
method: POST

Send to Backend:
user_id: <int>

EDIT NOTE:
route:
/notes/edit/<note_id>
method: GET

Return from Backend:
note_id: <int>
title: <string>
content: <string>

DELETE NOTE:
route:
/notes/delete<note_id>
method: DELETE

Send to Backend:
note_id: <int>

GET PROFILE:
route: /profile/<userid>
method: GET

Return from backend:
user_id: <int>
username: <string>
user_avatar:<string>
email: <string>

Return to home page

GO TO HOME:
route: /home
method: GET

Successful? — No → Does not exist Alert
Successful? — Yes → Successful Alert

profile.html
(profile page)

Return to home page

GO TO HOME:
route: /home
method: GET

notes.html
(blank notes page)

notes.html
(blank notes page)

HTML script
Keep refreshing avatars

GET ACTIVE AVATARS:
route:
/notes/avatars/<notes_id>
method: GET

Return from Backend:
Array of users using note
user_avatar:<string>

UPDATE NOTE:
route: /notes/<note_id>
method: UPDATE

Send to Backend:
title: <string>
catagory: <string>
*Do not update field that are null

SHARE NOTE:
route: /notes/<note_id>
method: UPDATE

Send to Backend:
user_id: <int>

UPDATE profile:
route: /profile/<userid>
method: UPDATE

Send to backend:
username: <string>
email: <string>
avatar: <string>
password: <string>
*Do not update field that are null

Successful? — Yes → Successful Alert
Successful? — No → Errorr Alert

Successful? — No → Does not exist Alert
Successful? — Yes → Successful Alert