

Telehealth app:

Description: This app will allow doctors to have online consultations with patients and keeps records of patient medicine prescriptions. Will focus on building functionalities for doctors.

Features:

- 1) online consultations in the form of a forum (exactly like the post-class exercise on comments)
- 2) sign up and login for doctors and users.
- 3) patient can schedule/cancel a date to consult a doctor and view past consultation records. (Many-to-many: users-to-doctors)
- 4) doctor can accept, start, cancel and end a consultation and view past consultations.
- 5) forum-like chatbox for patient and doctor to interaction during consultation.
- 6) doctor can prescribe medicine for the patient (not sure how to prescribe more than 1 medicine without reloading the webpage due to the large variety of medicines out there so checkboxes might not be suitable. Might have to reload the page each time a new medicine is added to the prescription).

Routes:

signup/login:

/login: (GET) to render login form and 2 signup buttons (GET) (/signup?=patient and /signup?identity=doctor)

/login: (POST) ask for email and password to login. Also put cookies for userId and loggedInHash and identity(patient or doctor).

/signup?identity=patient: (GET) to render signup form.

/signup: (POST) ask for name, email, password, allergies isdoctor=false to create a patient.

/signup?identity=doctor: (GET) to render signup form.

/signup: (POST) ask for name, email, password, clinics to create a doctor

/logout: (DELETE) Log a user out. Get rid of their cookie.

doctors:

/doctor-dashboard: (GET) to show the dashboard: ongoing consultations (/consultation/:id/edit and /consultations?status=ongoing), consultations to accept(/consultation/:id and /consultations?status=requested), upcoming consultations(/consultation/:id and /consultations?status=upcoming), history btn (/consultations?status=ended), profile btn(/profile/:id)

/consultation/:id/edit: (GET) to render the consultation details: 1. End consult btn (PUT to /consultation/:id) (only show btn when status is ongoing). 2. Update/ send message btn to update consult details (PUT to /consultation/:id then redirect to /consultation/:id/edit).

/consultation/:id: 1. (PUT from /consultation/:id/edit) to update status (end consult) or details (update diagnosis, prescription, price and chatbox) of the consult. 2. (PUT from /consultation/:id) to update status of consult (accept or start a consult). (only show accept button when status is req). (only show start btn when status is accepted).

/switch-profile: (GET) to switch from doctor to patient profile, vice versa. Redirect to /profile route and render ejs based on request.cookie.identity.

patients:

/patient-dashboard: (GET) to show dashboard. 1. New consult btn (/clinics). 2. Upcoming consultations (/consultations?status=upcoming and /consultation/:id). 3. history btn (/consultations?status=ended and /consultation/:id). 4. Ongoing consultations (/consultations?status=ongoing and /consultation/:id/edit). 5. Requested consultations (/consultations?status=requested and /consultation/:id). 6. Profile btn (/profile/:id).

/clinics (GET) to show a list of clinics. 1. Select clinic btn (GET) (/clinics/:id).

/clinics/:id (GET) to show a list of doctors. 1. Create appt btn (form) for each doctor. Value of each btn is the doctor's user id. (GET) (/consultation/new?doctor-id=<some-id>)

/consultation/new?doctor-id=<some-id> (GET) to render a form to create a consultation. Use the request.param to get doctor id. Form must contain doctor's name and id. 1. Submit btn (POST) (/consultation): to send a post request to create the consultation.

/consultation: (POST) to receive post request to create consultation. Redirect to (/consultation/:id)

/consultation/:id (GET) render details of consultation. 1. Cancel consult btn (DELETE) (Do app.delete(/consultation/:id). Only show this btn when consultation status is upcoming or requested). 2. Render a form to add a message if status is ongoing (PUT to /consultation/:id).

/consultation/:id (PUT) to accept request to add a message. Redirect to /consultation/:id.

Both:

/consultations?status=requested: (GET) to render list of consult requests.

/consultations?status=upcoming: (GET) to render list of accepted consults.

/consultations?status=ongoing: (GET) to render list of ongoing consults.

/consultations?status=ended: (GET) to render list of past consults.

/profile: (GET) render doctor/patient info (either ejs file for doctor or patient) based on the id found in the database. 1. Edit profile btn (/profile/edit). 2. Logout btn (DELETE) (/logout).

/profile/edit: (GET) render form for doctor/patient info. 1. Update btn(/profile/edit) to put updated info.

/profile: (PUT) update the user info.