Implement DS/Mobile on Nav Bar

Assignee	Ⓧ Xavier
Tags	Dynamic Sizing Implement Pages
Priority	Medium
Sprint	Sprint 4
Project	Frontend
Status	Done
Task ID	46

Nav Bar: Implement Dynamic Sizing / Mobile sizing

** Read everything before starting

This is what it currently looks like at ~800px width screen:



There is code that attempts to have a drop down in place of it

- Please find/take a look at it to see if it is useful enough to continue building

- If not, delete this code (make sure its only the code for the mobile nav bar and that nothing else, including CSS/styling, is not affected)

If this current code doesn't work:

- Create a classic drop down nav bar button (mobile design can be found in Figma ; left bar in "Mobile View")
    - Something that looks like so:



- Should look something like this but with the proper font and size



- Make sure each button goes to the correct pages and that the exit button (the x) works
- Also make sure that the original nav bar (the one that shows up for desktop) doesn't show up for this size

Refer to below (this was made mainly for those sizing a page but most of it holds true to your ticket)

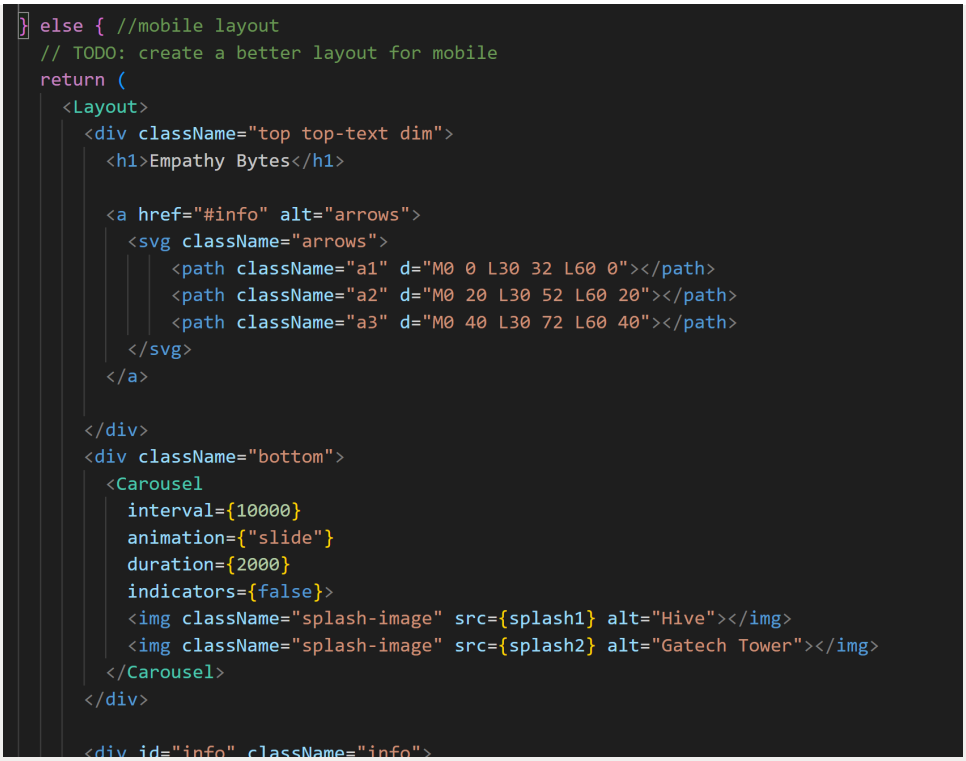
🚨 **How you must achieve this:**

- You must **ONLY** use the CSS files to achieve this
  - i.e. Utilize classes, IDs, etc. and change their styling within the CSS file (continue reading for how)
  - **DO NOT** copy/paste or edit .js code unless it's to add the above mentioned or if there is no other way to get the dynamic sizing right with the current format of the page
    - (which probably will happen)
- **DO NOT** use px for sizing (it's the whole reason why we had Sprint 2 tickets)
- **DO NOT** touch the current CSS (they by default, affect everything on the page so as of rn, both mobile and desktop)
- We are basically overhauling the entire/current mobile sizing format since it's outdated
- Read **"Getting started on editing these CSS files"** below
- All text must be legible and easy to follow (aka not squished up, font size is just right (1 rem), etc)

**If, in the .js file you are working in, you see an if/else statement encasing the return, please delete the content within the else statement that is (or should be) labelled for mobile

- run `npm run develop` to make sure the desktop wasn't affected
- Please also delete the if/else statement itself (but not the content for desktop)
- Here's some pics of what that looks like:
  (this is from index.js)





```
} else { //mobile layout
  // TODO: create a better layout for mobile
  return (
    <Layout>
      <div className="top top-text dim">
        <h1>Empathy Bytes</h1>

        <a href="#info" alt="arrows">
          <svg className="arrows">
            <path className="a1" d="M0 0 L30 32 L60 0"></path>
            <path className="a2" d="M0 20 L30 52 L60 20"></path>
            <path className="a3" d="M0 40 L30 72 L60 40"></path>
          </svg>
        </a>

      </div>
      <div className="bottom">
        <Carousel
          interval={10000}
          animation={"slide"}
          duration={2000}
          indicators={false}>
          <img className="splash-image" src={splash1} alt="Hive"></img>
          <img className="splash-image" src={splash2} alt="Gatech Tower"></img>
        </Carousel>
      </div>

      <div id="info" className="info">
```

Getting started on editing these CSS files:

**Note: DO NOT edit global variables/elements (aka body, div, section, etc.). It will not be fun to fix.

**Also, learn the basics of CSS before trying (onboarding), so we can make the least amount of commented out stuff, clutter, etc

1. So how can we tell CSS to change style / sizing based on if it's a mobile/small screen?
    - Read up on Media Queries (please)
    - For our project, we will use **800px** as our minimum (one of the only times we are going to use px)
    - So within these queries, we can edit the styling of each class/id/etc (see #2 bullet)

    - these won't affect the sizing of any screens above 800px

2. Notice how in the pics above (about the if/else statements), each or most elements have a `className` (or ID, etc)?
    - You can add multiple class names to it BUT be careful since they can conflict with each other
        - So you might want to reduce how many there are (like `<div className="top top-text dim">` where `top top-text dim` can easily be put into a CSS file for a class)
    - Inside the CSS file that is used for the .js file, you can edit the styling of that class/id/etc. to fit the targeted design (like from row to col, wider or narrower, etc.)

3. `COMMENT` your code (like what does this block of code represent or say "ayo this is the mobile sizing for ____ page")
    - I want to take some burden off the future members who need to deal with cleaning the code (which is already a lot of work that needs to be done)

4. Dealing with current format of your current .js file:

- Unfortunately, you may run into the problem where the current .js file just doesn't work well with this way of dynamic sizing (mostly since it was largely centered around desktop and px sizes)
- We cannot just rebuild the whole mobile design on the .js file (see if/else statement) since it would be a nightmare to implement any new designs for the website
- It'll be kinda hard, but you may reorganize/reformat the code to where you can properly do the #2 bullet
    - aka try to rebuild it while KEEPING the same design
    - you may not redesign the page (therefore no visual changes to the desktop screen, only organizational/cleanup changes)

- Try to do this as **a last resort** unless you know it needs it for you to move on (def consult Team Leads if you want consultation / have us look at it for a decision)

I know this is a lot, but these instructions will help everyone be more consistent and practice good coding.

Please be mindful of those in the future who will have to **see/edit your code** so the future people don't have to deal with something like this all over again.

When you're finished

(This was added 02/17/2025)

- Lmk when you're done so I can give feedback for you to fix
  - aka put in that pull request
- Once all feedback is fixed and you get a thumbs up from me, take screenshots of the mobile version (less than 800px width when you do inspect on the preview) of your pages
  - try to get everything on the page (you may take multiple)
- Make a frame in the Figma (Designs - 2025 and in Mobile View from the left nav bar) so you can paste them
  - Follow the way others have made it (Kelly's is a good one)
  - Paste all your screenshots there and properly name the frame / pics

Lmk if you have any questions/comments/concerns!