



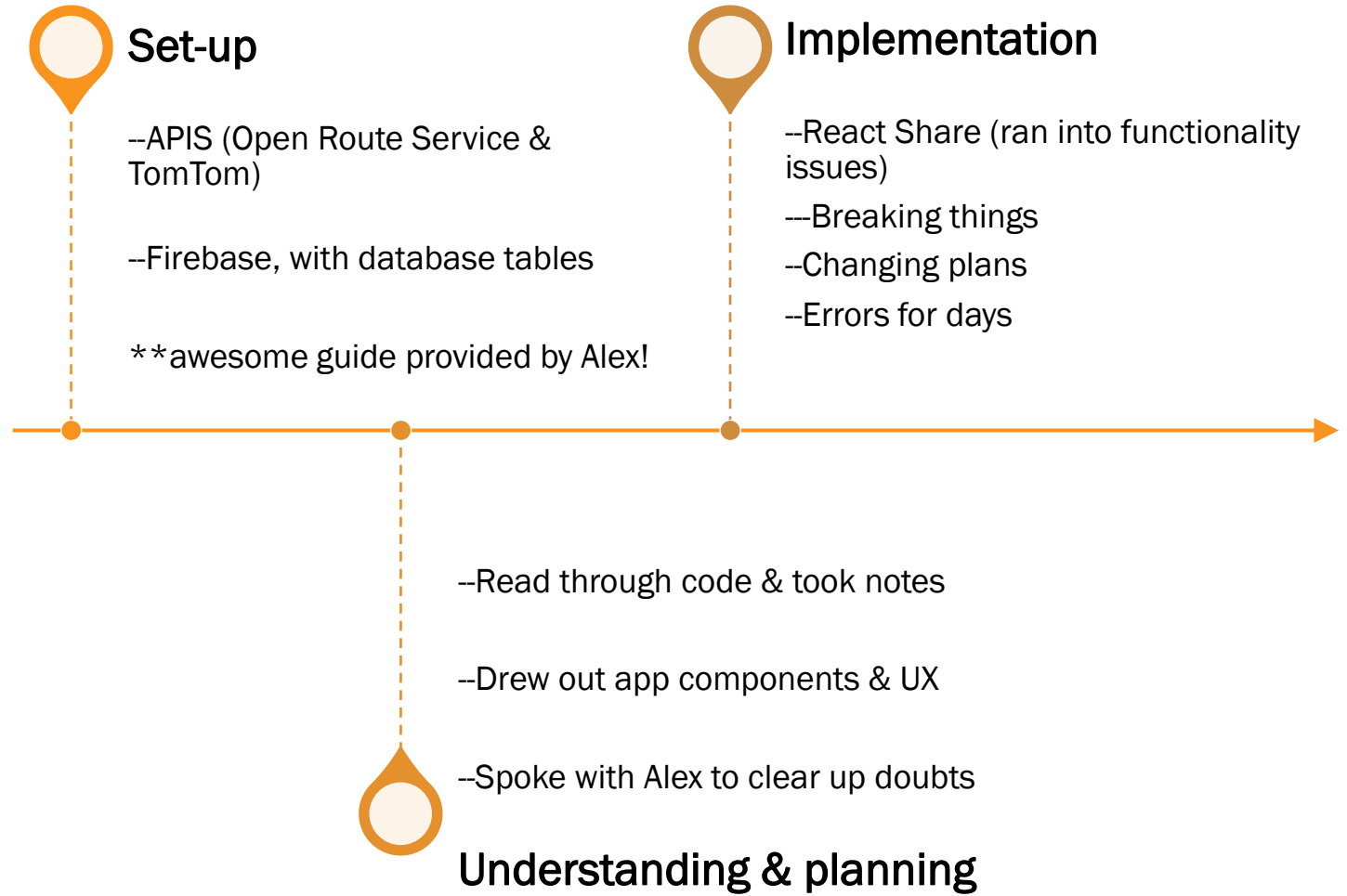
Meet in the Middle: Feature Extension

SEONAIID MCNABB


Why I wanted to work on this project...

- To get a firsthand look at how Alex styled the project
- To learn more about using external APIs & Firebase
- Because, although I have a good sense of direction, I'm terrible at reading maps—so I thought that in some way this might be extra challenging / educational

STEPS



READING PROCESS

The background features a complex geometric design. It consists of several large, overlapping triangles and polygons in two shades of orange. Interspersed among these solid shapes are white triangular and quadrilateral areas. A series of thin, dark grey parallel lines are drawn across the composition, following the angles of the geometric shapes, creating a sense of depth and movement.

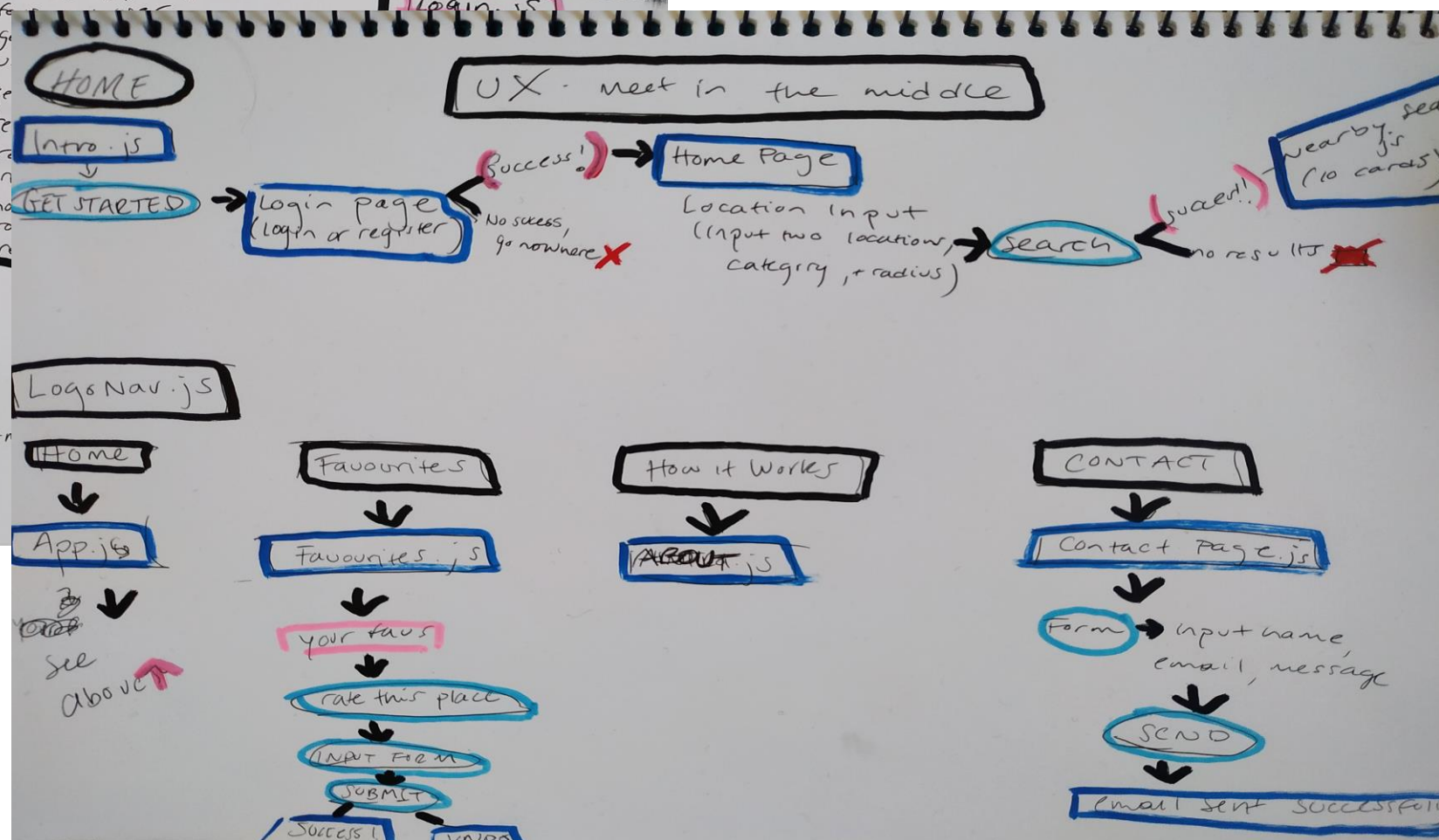
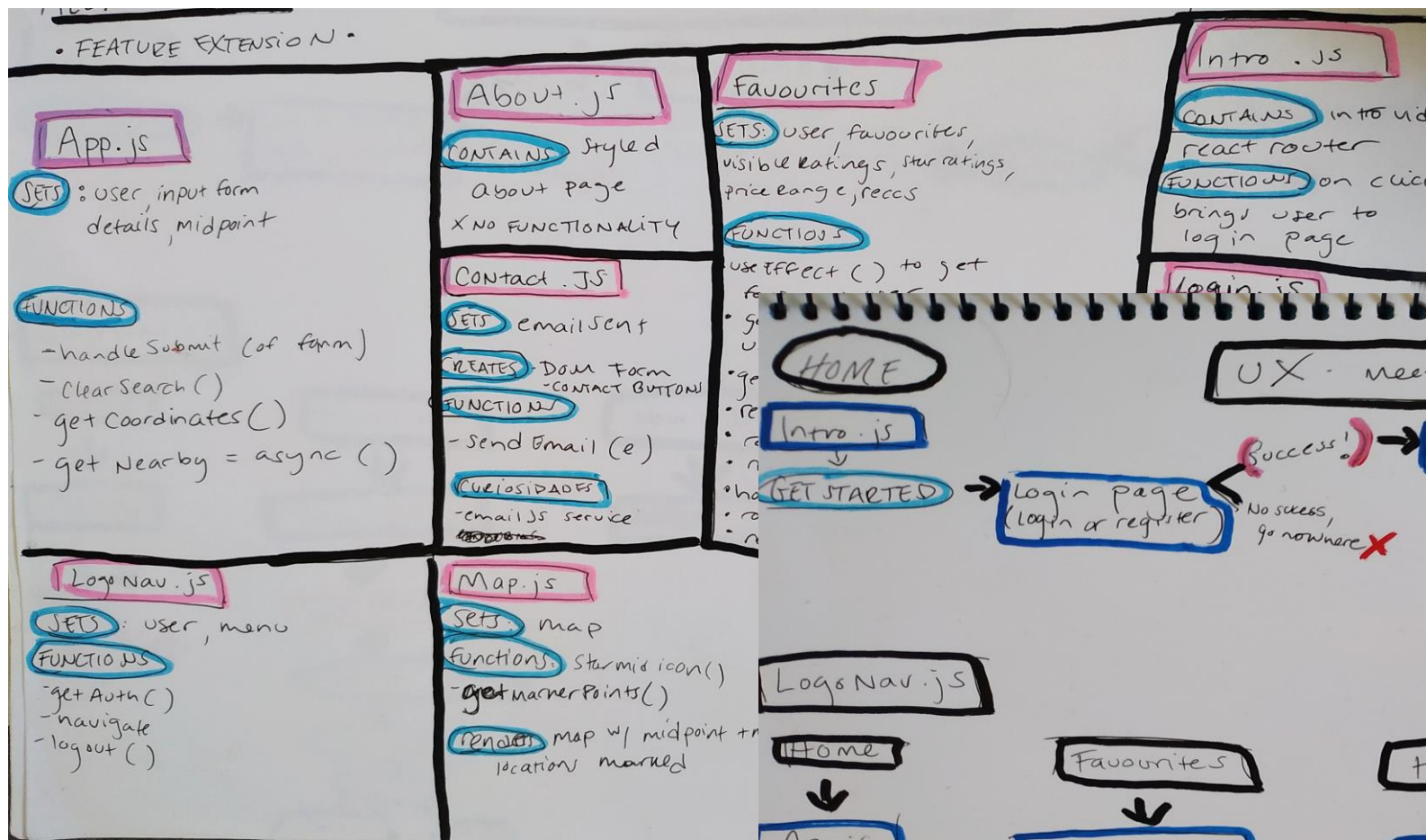
In-Code Notes

```
//this app function:
//contains all logic for determing and setting user
//determing and setting midpoint and associated nearby POI
//^^sends detailed data to nearby search page
function App() {
  let [loading, setLoading] = useState(false);
  //loading state
  let [user, setUser] = useState({});
  //set the user

  //here & below set the states that the user has input to meet in the middle
  let [input, setInput] = useState({ inputOne: "", inputTwo: "" });
  let [points, setPoints] = useState({ pointOne: "", pointTwo: "" });
  let [category, setCategory] = useState({ category: "" });
  let [radius, setRadius] = useState({ radius: "" });
  let [nearby, setNearby] = useState({
    id: "",
    name: "",
    address: "",
    url: "",
    favourite: false,
```

I went through Alex's code line-by-line and tried to add notes to the parts I was unsure about or where I needed to walk myself through the logic & talked to her about some parts I didn't understand

```
for (let place of top10) {
  // for the purpose of marking places on the map, the final places and their details are pushed to
  //an array and assigned to the markers state
  //for each of the 10 results captured in top10, these details are pushed to nearbyLatLng array
  nearbyLatLng.push({
    name: place.name, //name
    position: [place.lat, place.lng], //location
    address: place.address, //address
  });
  setMarkers(nearbyLatLng); //markers set to the nearbyLatLng
}
```

Component sketching

FEATURE IMPLEMENTATION PROCESS

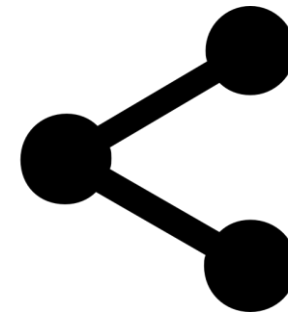
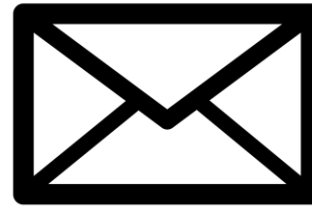
The background features a dynamic composition of overlapping geometric shapes. Large, angular blocks of a vibrant orange color are interspersed with white space. Thin, parallel lines in a dark grey or blue hue are drawn across the composition, following the angles of the orange shapes, which creates a sense of depth and movement. The overall aesthetic is modern and minimalist.

Proposal: Share Buttons

PROCESS & CHALLENGES

- Looked for out-of-the-box program to add share icons
- Installed “react-share” package
- Implemented e-mail share button as a first feature but ran into issues as was unable to include inline html elements in the e-mail body and so sharing the search result was not very useful

--**Learned:** importing and using icons from *fontAwesome ... and that I still need to learn more about adding sharing functionality*



Small “Fixes”

1

- **Elements**: favorite heart button, login link, add rating link, remove from favourites link
- **Issue**: on hover over these links, text edit cursor was displayed
- **Fix**: changed CSS of element classes onHover to cursor: pointer

2

- **Elements**: ratings form, chevron icon
- **Issue**: on click, the rating form displayed, but did not close on clicking again
- **Fix-ish**: added conditional rendering to form & icon elements (but it's buggy, sometimes it opens multiple forms at once)

Filtering Results Feature, aka: the moment I gave up

CHALLENGES

- This was the point of the project at which I felt quite lost. The idea was to implement a search filter so that users could sort through favorited places by user recommendations
- I like the idea. I get the theory. I even “got” most of the code thanks to Alex’s great set up instructions and answering my questions along the way... but I didn’t feel confident about where to actually begin implementing the feature.



CHANGE OF PLANS

Catastrophic failure (0x8000FFFF)

Please try again.

OK

Changing Focus

PROPOSAL

Because I didn't feel confident about adding a new feature that would actually be valuable / work, I proposed adding tests (with the idea being that I could test existing features to make current & future debugging easier, write suggested tests for additional features, etc.)

Also considered working on accessibility in the code and implementing a dark theme option, which sounded like an interesting idea.

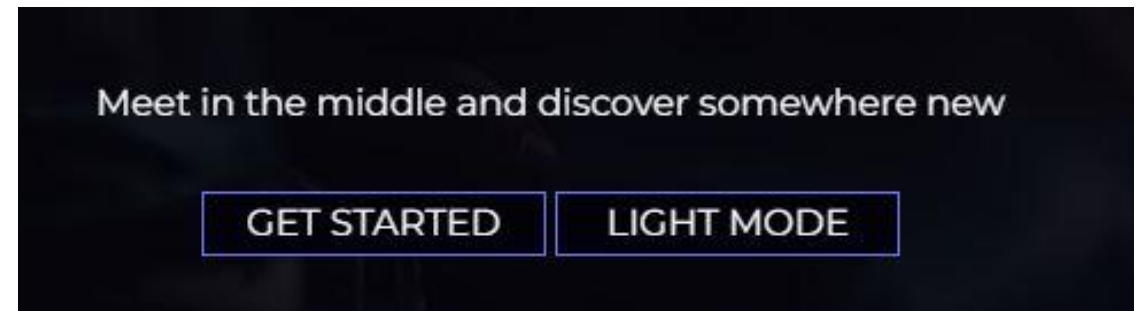
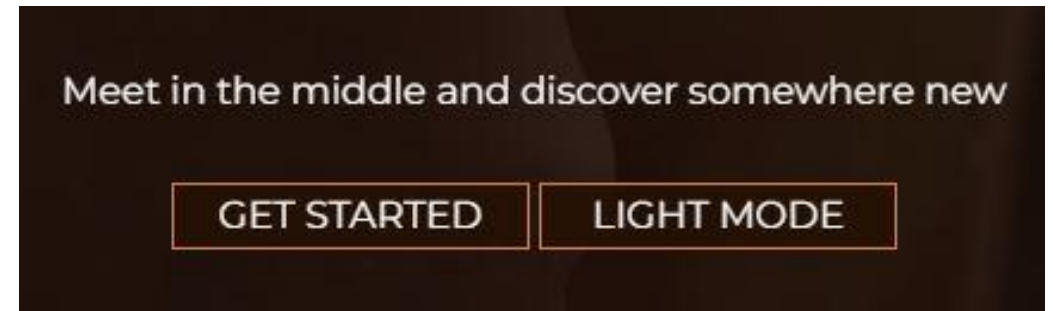


The mantra of Test-Driven Development (TDD) is "red, green, refactor."

Accessibility Proposal: Light and Dark Modes

PROCESS & CHALLENGES

- Read tutorial
- Went through Alex's App.css and wrote notes on which elements each CSS class or element was manipulating
- Started on the Intro component, implementing local storage for storing dark / light stage, and toggle button
- Styled light and dark theme elements for the intro page
- Successively styled each page



Light & Dark Modes Code Process

```
<div className="login-body" data-theme={theme}>
  <div className="login-container">
    <img src={Logo} className="login-
  ) : (
    <img src={DarkLogo} className="lo
  )}

  <h4>Register</h4>
  { /* REGISTRATION FORM */ }
  <form className="login-form">
    <div className="row">
      <div className="col-lg-6 col-xs-
```

- Kept component I was styling, index.css and app.css open at the same time to help the process / locate which elements needed to be styled

```
1 import React from "react";
2 import DesktopIntro from "../videos/meet
3 import MobileIntro from "../videos/meet-
4 import { useNavigate } from "react-route
5 import useLocalStorage from "use-local-s
6
7 export default function Intro() {
8   const navigate = useNavigate();
9
10  const routeChange = () => {
11    navigate("/login"); // using React R
12  };
13
14  let defaultDark = window.matchMedia("(
15    //^^checks to see if user has set a th
16
17  let [theme, setTheme] = useLocalStorag
18    "theme",
19    defaultDark ? "dark" : "light"
20  );
21
22  //^^theme variable tied to local stora
23  //^^defaults to light unless defaultDa
24
25  const switchTheme = () => {
26    const newTheme = theme === "light" ?
27    setTheme(newTheme);
28  };
29
30  return (
31    <div>
```

```
299 h5 {
300   color: #77c29e;
301 }
302
303 /*adjusts size appearance of stars*/
304 .star-rating {
305   width: 50%;
306 }
307
308 /* INTRO PAGE STYLING */
309
310 /*styles the size and positioning of the
311 .intro-video {
312   width: 100%;
313   height: 100%;
314   object-fit: cover;
315   position: fixed;
316   left: 0;
317   right: 0;
318   top: 0;
319   bottom: 0;
320   z-index: -1;
321 }
322
323 /*styles the text that says "meet in the
324 .intro-text {
325   background-color: var(--intro-message-
326   position: absolute;
327   left: 0;
328   right: 0;
329   bottom: 0;
330   margin-left: 40px;
```

```
28
29
30 1 (
31  />
32  <div className="intro-container">
33    <video loop autoPlay muted className="in
34    <source src={DesktopIntro} type="video
35    </video>
36    <video loop autoPlay muted className="mc
37    <source src={MobileIntro} type="video/
38    </video>
39  </div>
40
41  <div className="intro-text" data-theme={th
42    <p>Meet in the middle and discover some
43    <button className="intro-button" onClick
44      Get started
45    </button>
46    {"\n"}
47
48    <button className="intro-button intro-co
49    {theme === "light" ? "Dark" : "Light"}
50    </button>
51  </div>
52  </div>
53
54
55
```


Proposal: Testing

PROCESS & CHALLENGES

- Added Jest and Enzyme to the project, watched testing tutorial
 - Implemented test on About Component (passed!)
- Tried implementing tests on other components and ran into a billion errors—uninstalled and reinstalled packages, deleted, undeleted, broke everything, reverted to old code
 - With help, managed to resolve the first error and get a new one
- **Conclusion:** didn't get to implement any more complex tests, but made progress. Learned the joy of getting a NEW error.



```

11 | export default function Contact() {
12 |   let [emailSent, setEmailSent] = useState("");
13 |
14 |   const form = useRef(); //useRef allows you to directly
15 |

```

FAIL src/components/Favourites.test.js (5.543 s)

- Testing the Favourites Component › should have paragraph elements

Invalid hook call. Hooks can only be called inside of the body of a function component. This could happen for one of the following reasons:

1. You might have mismatching versions of React and the renderer (such as React DOM)
2. You might be breaking the Rules of Hooks
3. You might have more than one copy of React in the same app

See <https://reactjs.org/link/invalid-hook-call> for tips about how to debug and fix this problem.

```

at resolveDispatcher (node_modules/react/cjs/react.development
at useState (node_modules/react/cjs/react.development.js:1507
at Contact (src/components/Contact.js:12:35)
at ReactShallowRenderer.render (../node_modules/react-test-re

```

```

2)
at renderElement (../node_modules/enzyme-adapter-react-16/src/ReactSixteenAdapter.js:643:26)
at fn (../node_modules/enzyme-adapter-react-16/src/ReactSixteenAdapter.js:707:46)
at withSetStateAllowed (../node_modules/enzyme-adapter-react-16/src/ReactSixteenAdapter.js:707:46)
at Object.render (../node_modules/enzyme-adapter-react-16/src/ReactSixteenAdapter.js:707:46)
at new ShallowWrapper (../node_modules/enzyme-adapter-react-16/src/ReactSixteenAdapter.js:707:46)
at shallow (../node_modules/enzyme-adapter-react-16/src/ReactSixteenAdapter.js:707:46)
at Object.<anonymous> (src/components/Contact.js:12:35)

```

Compiled with problems:

ERROR in ./node_modules/leaflet/dist/leaflet-src.js

Module build failed (from ./node_modules/source-map-loader/dist/cjs.js):

Error: ENOENT: no such file or directory, open 'C:\Users\Seonaid\Desktop\CodeOp\meet-in-the-middle\client\node_modules\leaflet\dist\leaflet-src.js'

ERROR in ./node_modules/leaflet/dist/leaflet.css (./node_modules/css-loader/dist/cjs.js??

ruleSet[1].rules[1].oneOf[5].use[1]!./node_modules/postcss-loader/dist/cjs.js??

ruleSet[1].rules[1].oneOf[5].use[2]!./node_modules/source-map-loader/dist/cjs.js!./node_modules/leaflet/dist/leaflet.css)

Module build failed (from ./node_modules/source-map-loader/dist/cjs.js):

Error: ENOENT: no such file or directory, open 'C:\Users\Seonaid\Desktop\CodeOp\meet-in-the-middle\client\node_modules\leaflet\dist\leaflet.css'

```

├─ react@16.14.0 deduped
├─ @react-google-maps/api@2.7.0
├─ react@16.14.0 deduped
├─ enzyme-adapter-react-16@1.15.6
├─ enzyme-adapter-utils@1.14.0
├─ airbnb-prop-types@2.16.0
├─ react@16.14.0 deduped invalid: "react@16.14.0" is not a valid semver version
├─ react@16.14.0 deduped invalid: "react@16.14.0" is not a valid semver version
└─ react-test-renderer@16.14.0

```


REFLECTIONS

The background features a complex geometric design. It consists of several overlapping triangular and polygonal shapes in two shades of orange. Interspersed among these solid-colored shapes are white areas containing multiple sets of parallel, dark grey lines. These lines are arranged in a way that suggests depth and reflection, with some lines appearing to converge or diverge at various angles, creating a dynamic and modern aesthetic.

Tech Added

- Jest
- Enzyme
- React-Share (not used, but could be useful on future projects)
- `useLocalStorage` hook (to store user's theme preference)

Resources

- [Dark Mode Tutorial](#)
- A [Very Exciting Video](#) about TDD
- [Article about testing](#) with Jest/Enzyme, shared by Alex
- [Accessibility Checklist](#) (just for learning, didn't get to apply it)
- [FireBase](#) tutorial from Alex

What worked, what didn't

GREAT:

- Alex's set-up instructions + she was very communicative with any questions and making icons for dark mode

NOT SO GREAT:

- In hindsight, I would have started with testing
- *Styling*: tried to intuitively style but think a bit of planning would have been helpful

Most Difficult

- Understanding lat & long/midpoint process
- Understanding external API calls & setup process
- Firebase, Axios, EmailJS
- Jest & Enzyme Errors
- Writing favorites with a U

Learned

- How to set up Firebase
- More about CSS / how elements relate to one another and are styled
- Jest and Enzyme testing for React Applications (beginning steps)
- Accessibility conventions

Proudest Moments

- Finding an alternative to my original plan
- When I finally got all the components styled in dark mode & none of them were broken and light-mode still worked!

Issue	Component
Dark mode: rating stars color still fills as brown, didn't have time to address this issue	NearbySearch
Dark mode: Gifs in "how it works" section were made on light mode, so don't correspond to dark view	About
Toggling: toggling the ratings form has a bug – sometimes opens the correct form, sometimes opens several at once on click	Favourites
E-mail share function: icon is rendered and onClick opens mail client, but the parameters are currently unedited--- only subject is filled in. I tried to get the body to display a clickable directions link but ran into issues	NearbySearch
Review display: not sure if I caused this error myself, but on ratings page, my recent reviews does not display the reason I recommended the place (error handled and see the code has access to the review, so not sure why it doesn't work)	Favourites
Dark Mode: currently you can only choose this on the get-started page, it would be nice to add an accessible button throughout the app where people can change that preference	Throughout