**Meme Generator Documentation:**

1. Creating an app using create-react-app toolkit
   1. npx create-react-app meme-generator
2. Now, Delete all the unnecessary files which we wont be using
   1. **Graphical user interface, text, application, chat or text message

      Description automatically generated**
3. Inorder to use the Routes we make use of react-router-dom.
4. To install react-router-dom library we make use of following command
   1. npm install react-router-dom
5. Now, Inside index.js import {BrowserRouter} from ‘react-router-dom’ and Wrap the <App> Component with <BrowserRouter>
   1. **Text

      Description automatically generated**
6. InOrder to Route through the pages we create a folder called Pages inside src
7. Inside pages Folder we create a Home.js

**A screenshot of a computer

Description automatically generated with medium confidence**

1. Now, we will Create a Route for the Home page inside App.js
2. To create a Route we need to import {Routes,Route} from ‘react-router-dom’
3. Also import Home from pages
4. Inside Route we will give the path and element as Home
5. Wrap the Route with the Routes.

**Graphical user interface, text

Description automatically generated**

1. Now we will install react-bootstrap by using the following command
   1. npm install react-bootstrap
2. Inside src we create a folder called Components
3. Inside components we create a MemeCard component

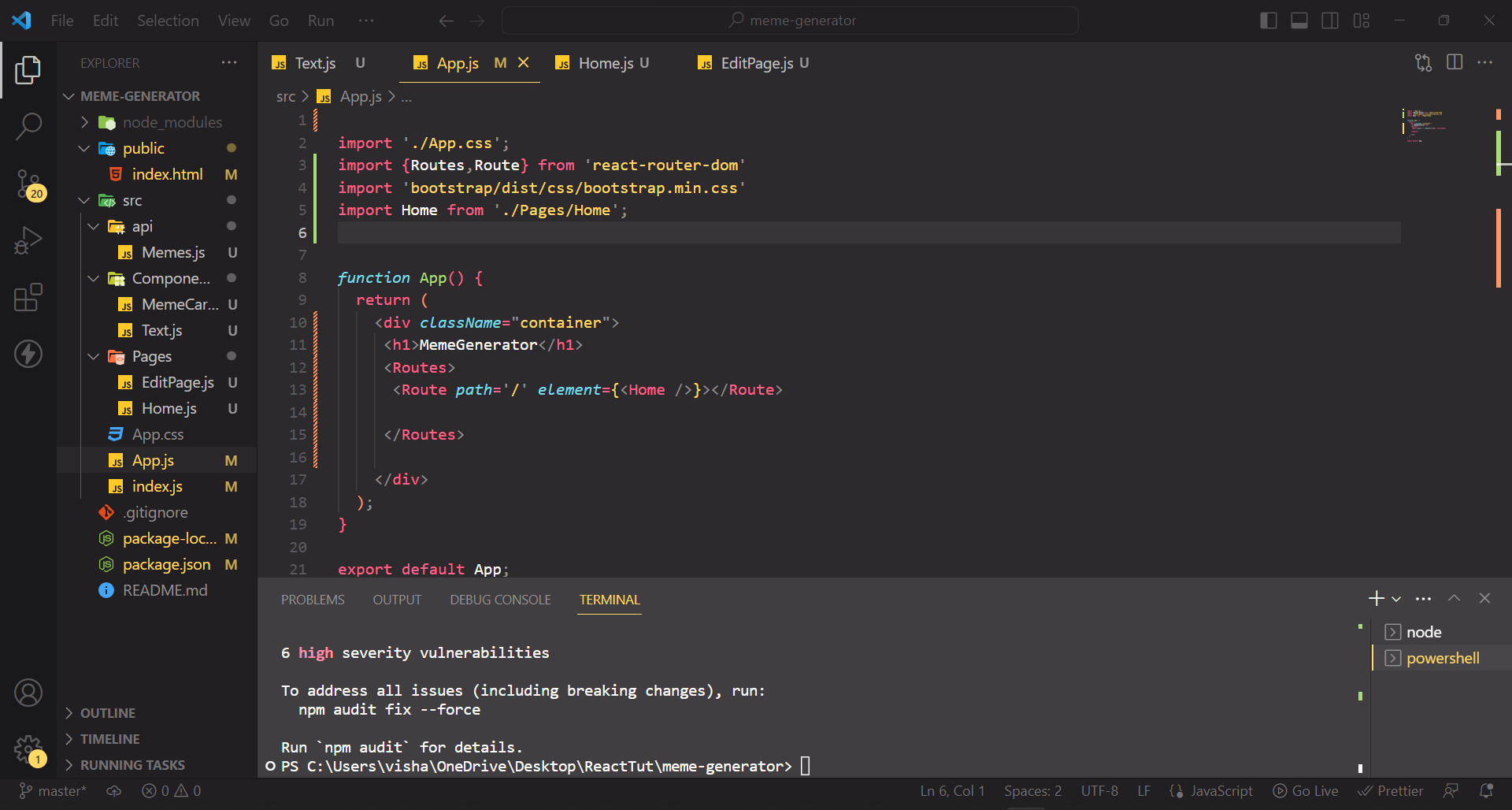
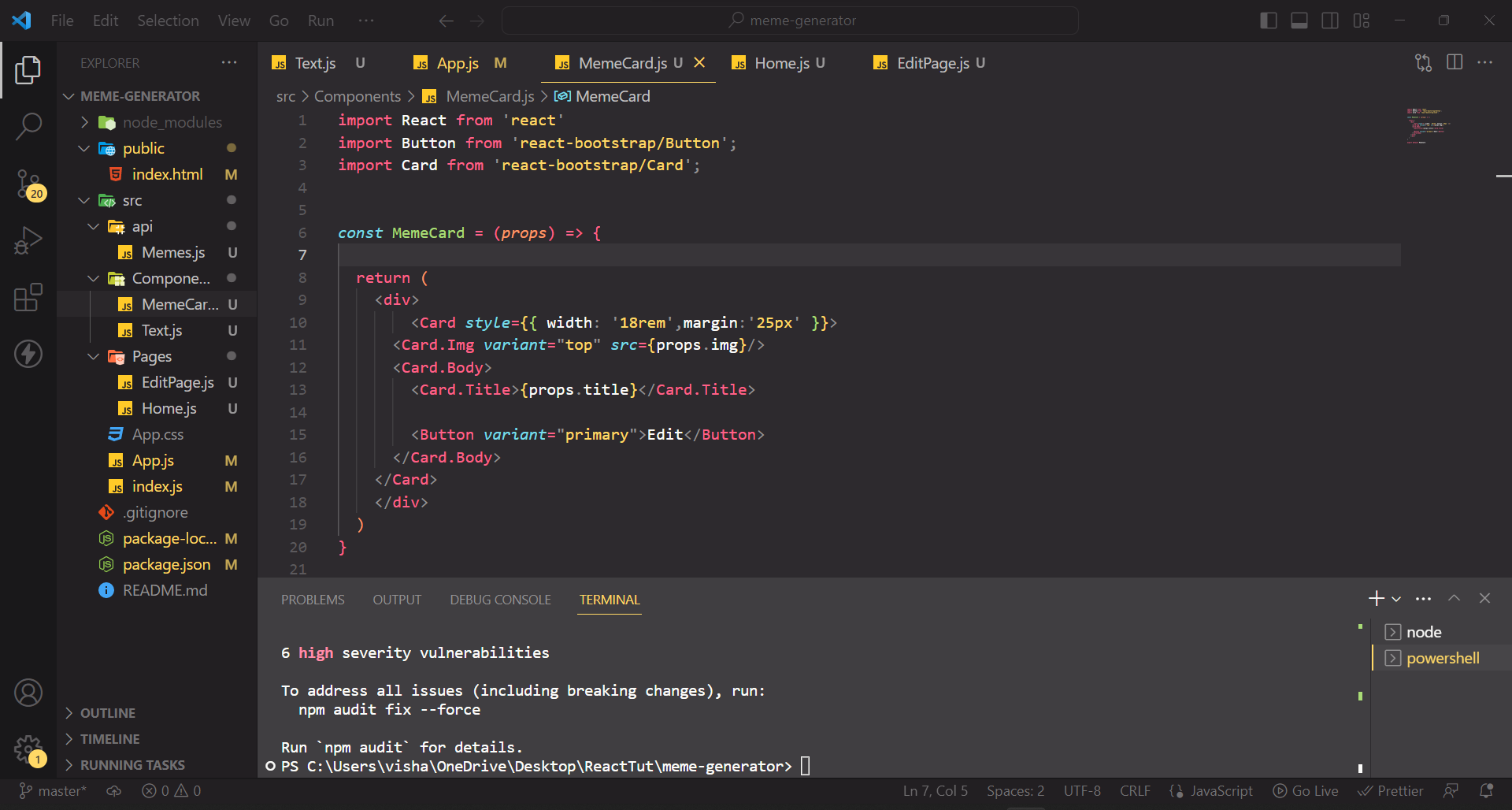
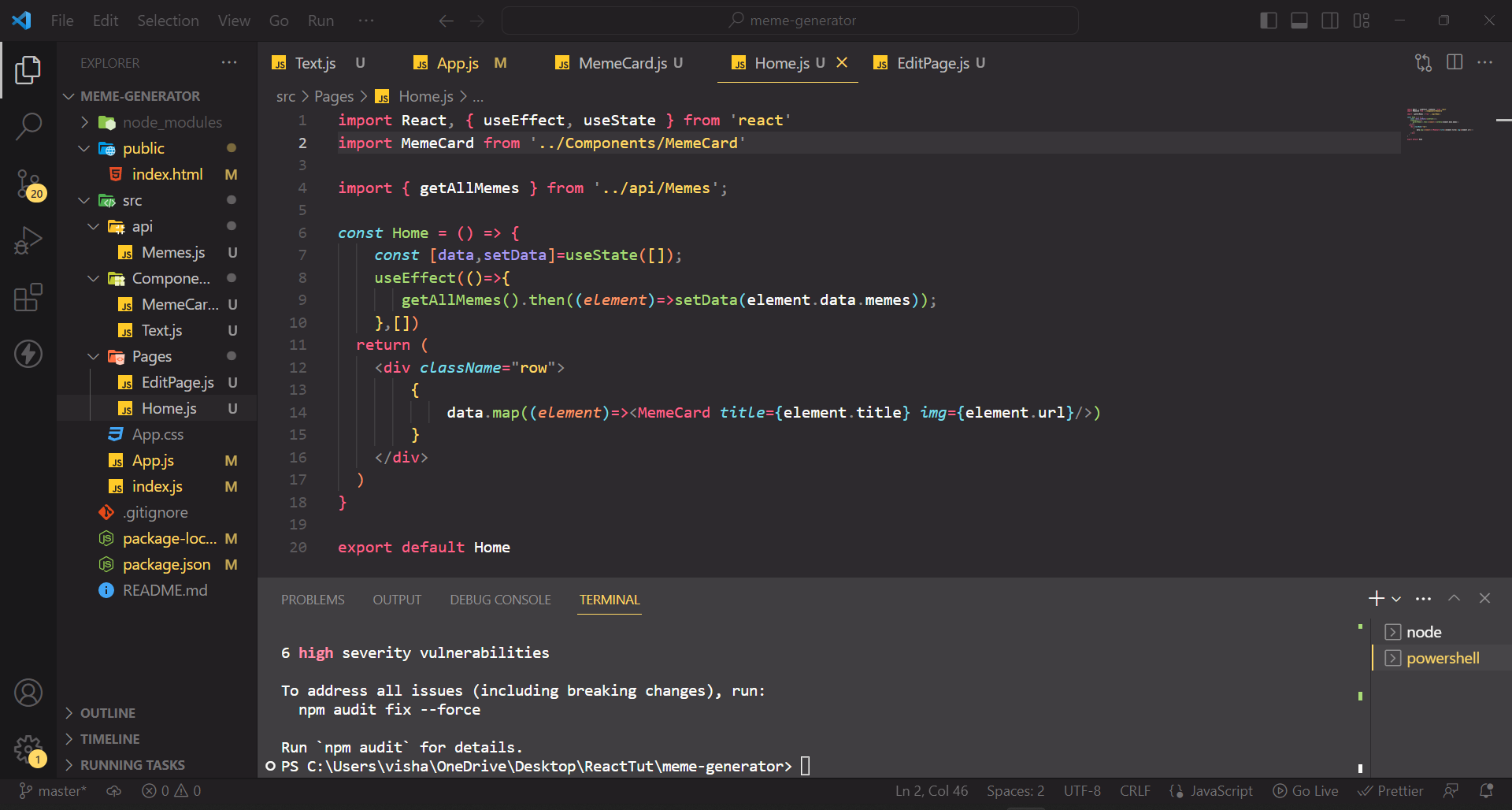
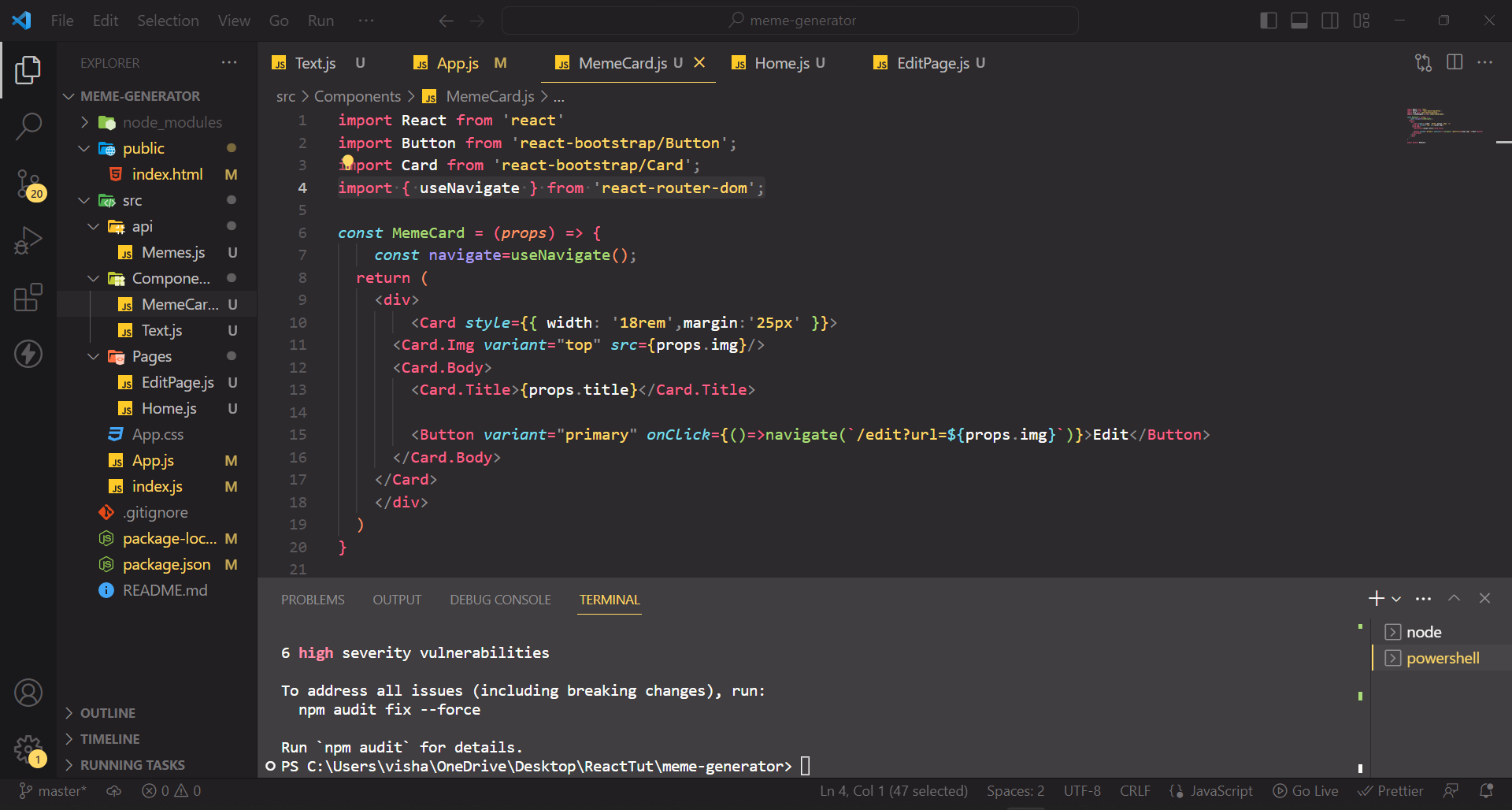
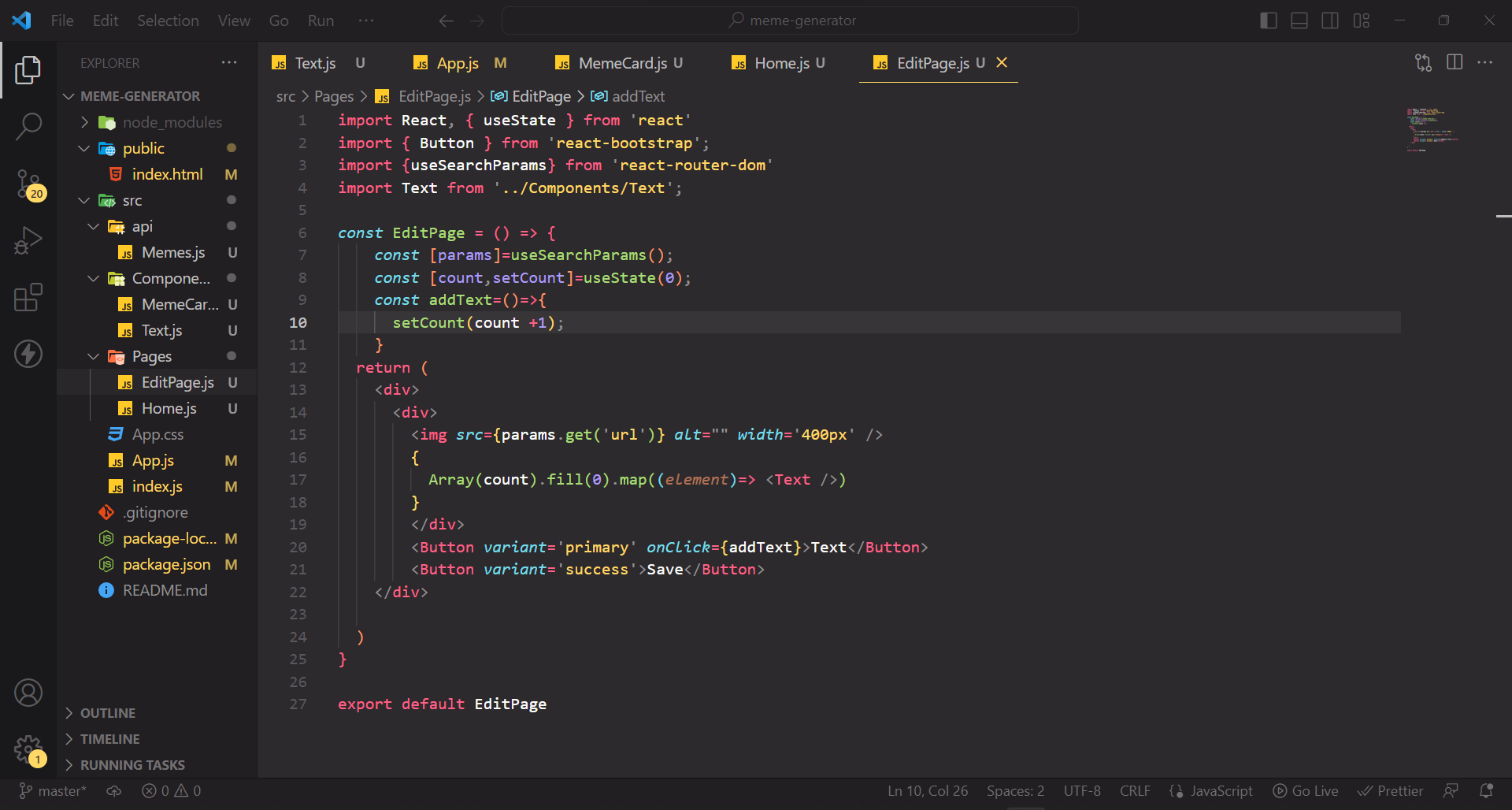
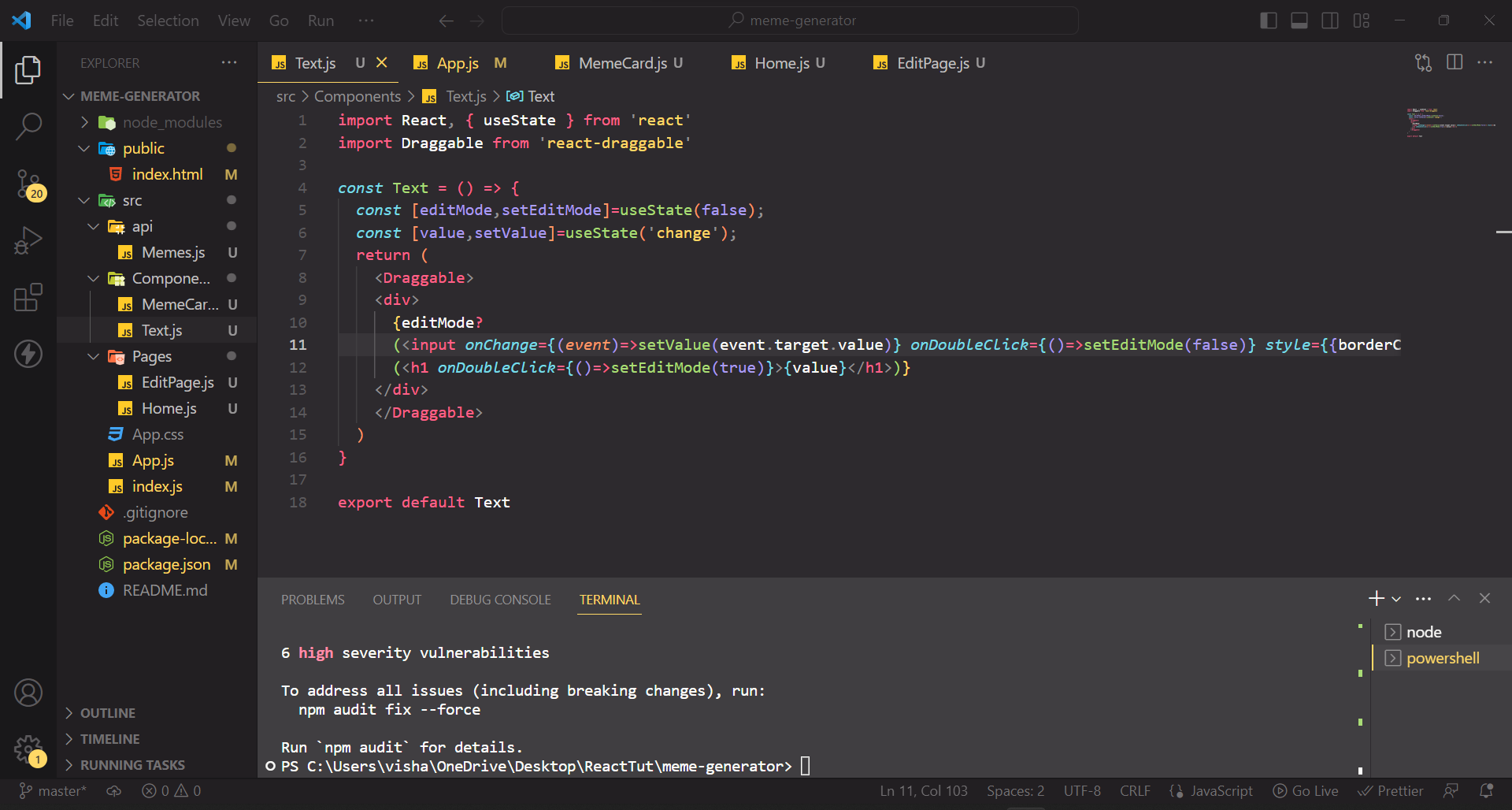
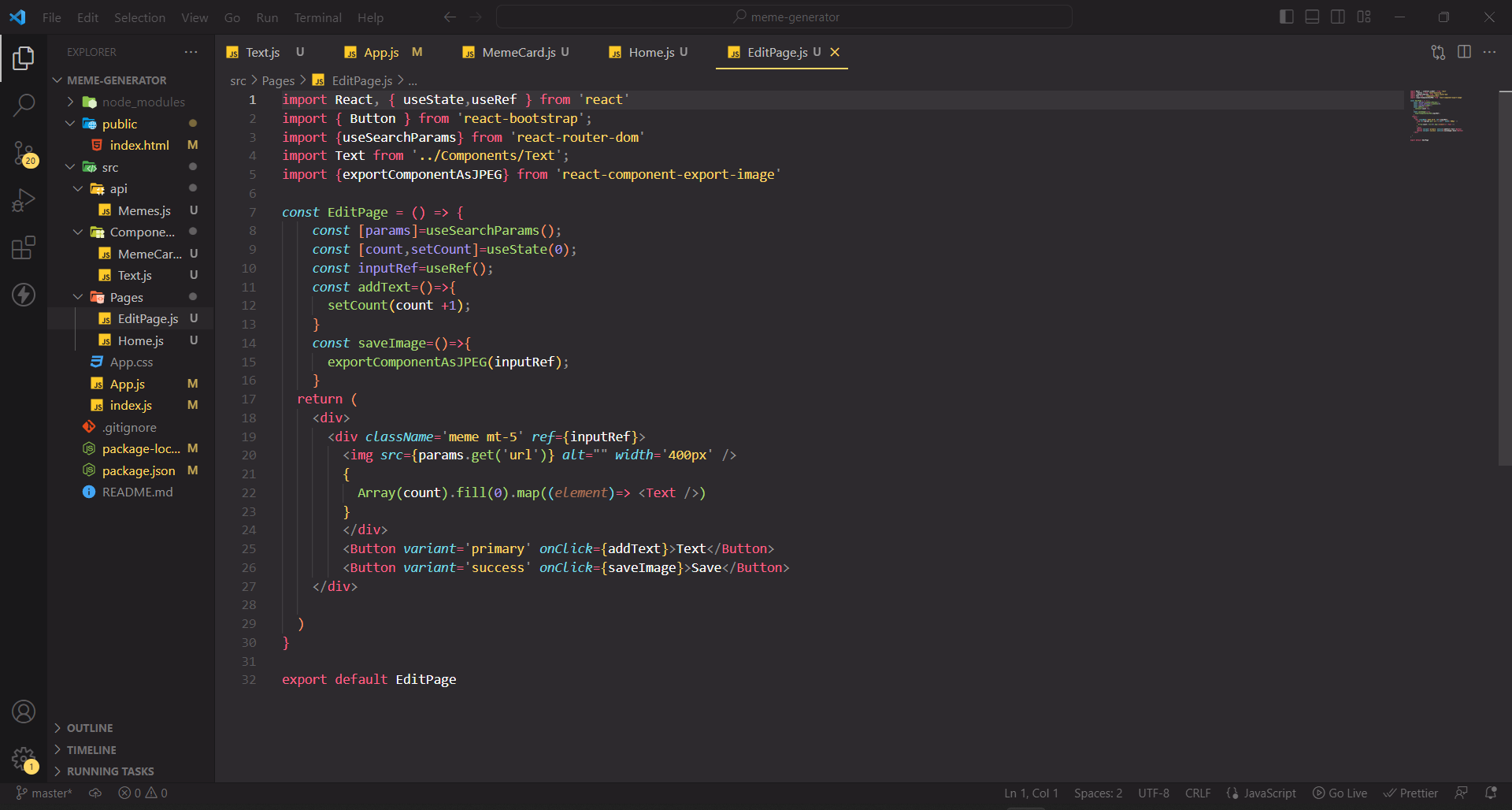
**A screenshot of a computer

Description automatically generated with medium confidence**

1. Now, we will create a folder api.
2. Inside api we create a file named Memes.js
3. Inside Memes.js we create an async function called getAllMemes();
4. By making use of fetch method we will call the url.
5. And return the response.json()

**Graphical user interface, text

Description automatically generated**

1. Now, we will install react-bootstrap to include the bootstrap in our App.
2. After installing, we will include the file in to our App.js file.
3. ****
4. Now, inside MemeCard we will copy paste the card component from react-bootstrap.
5. Memecard will accept props and then will replace the title and image.
6. ****
7. Inside Home.js, we will call useEffect().
8. Inside useEffetc(), we will call getAllMemes() from api.
9. getAllMemes() will return promise to resolve we make use of .then()
10. memes is present inside the data.
11. We will make use of useState([]) hook to set the data returned from getAllMemes().
12. After that, we will map the elements of data to component in the following way.
13. ****
14. Now, To edit the image we will create a new page named EditPage.js
15. We will create a route for EditPage inside app.js
16. Inside Memecard component, we will make use of useNavigate() hook to navigate through the url.
17. Edit button will pass the url of the image using navigate().
18. ****
19. Inside EdiPage.js we will import {useSearchParams} from react=router-dom
20. We will create a const [params] to call useSearchParams() functions.
21. Inside <img> tag src will have params.get(‘url’).
22. We will create a component named Text.js .
23. Inside EdiPage, we will create a button called Add Text.
24. When ever Button is clicked, <Text /> component is called.
25. We will create a useState() hook for count variable.
26. When ever edit button is clicked, count will be incremented by one.
27. Now, we will create an Array of count size and through each element.
28. ****
29. Now, we will install react-dragabble inorder to make draggabletect.
30. Wrap the div tag inside Text.js using <Draggable>.
31. Now, Inorder to dynamically change the text, i.e when ever user double click on the text it should be changed in to editable format.
32. we will import useState() to track whether it is in edit mode or not.
33. Another useState() to save the value
34. Now inside div we will create a jsx component.
35. Using Ternary Operator,\ we will check for the EditMode, if it is true we will create an input tag and using onchange handler we save the value and onDoubleClick listener we will set the edit mode to false. if it is false, we will display the value and onDoubleClick we will change the edit mode to true
36. ****
37. Inorder to download the Meme, we will install react-component-export-image.
38. Inside editPage, we will import useRef, and we will create const and call the useRef()
39. When the save button is clicked we want to save the meme.so we will ref attribute and pass the const which is created inside the div.
40. we will import {exportComponentAsJPEG} from ‘react-component-export-image’
41. onClick() save button we will create a function and pass to it.
42. Inside the function we will call exportComponentAsJPEG() and pass the const ref.
43. ****