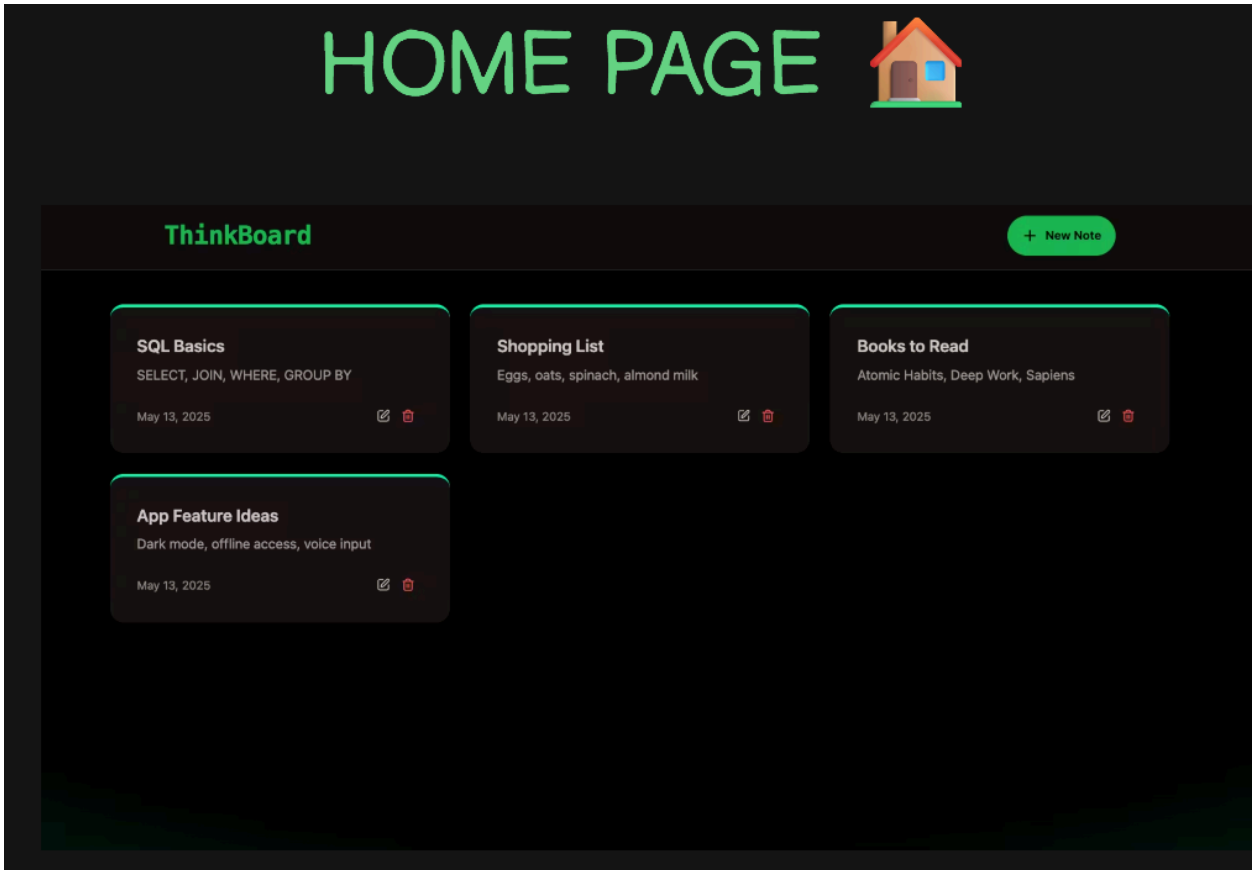
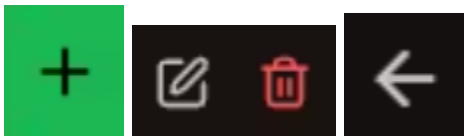


Where nav bar component, is:

Logo**(ThinkBoard)** on the left, link**(+ New Note)** on the right hand side.
Once we click on link**(+ New Note),** Takes to the create page.
We also have different **icons:**



For these icons we use the package, install under the frontend

Install the packages:

1. Stop the frontend server
2. Install: **npm i lucide-react**

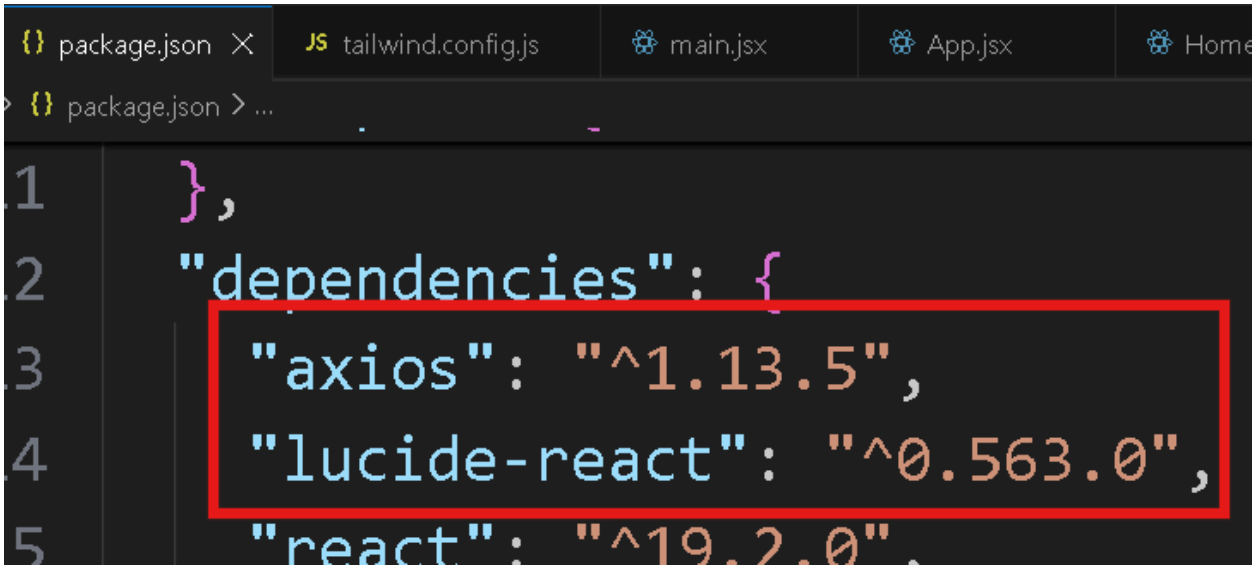
```
\mern-thinkboard\frontend> npm i lucide-react
```

3. Install: **npm i axios**

Replaces the fetch APIs with the axios

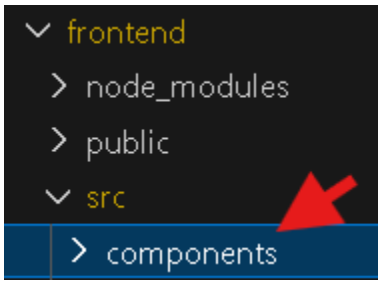
```
\mern-thinkboard\frontend> npm i axios
```

In package.json:

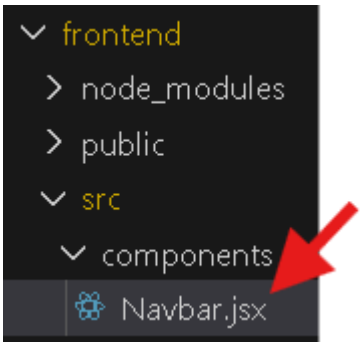


In terminal run the server in frontend: npm run dev

Create components folder in src:



Create Navbar.jsx, within



Type: rafce in **Navbar.jsx** file, then click enter to get the code below:

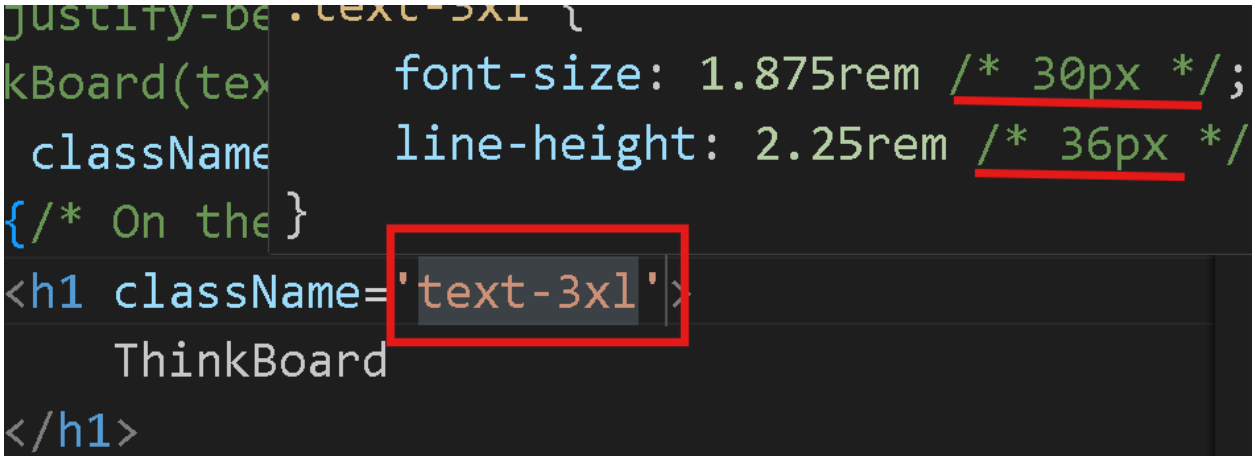
```
import React from 'react'

const Navbar = () => {
  return (
    <div>

    </div>
  )
}

export default Navbar
```

Code in HomePage.jsx:

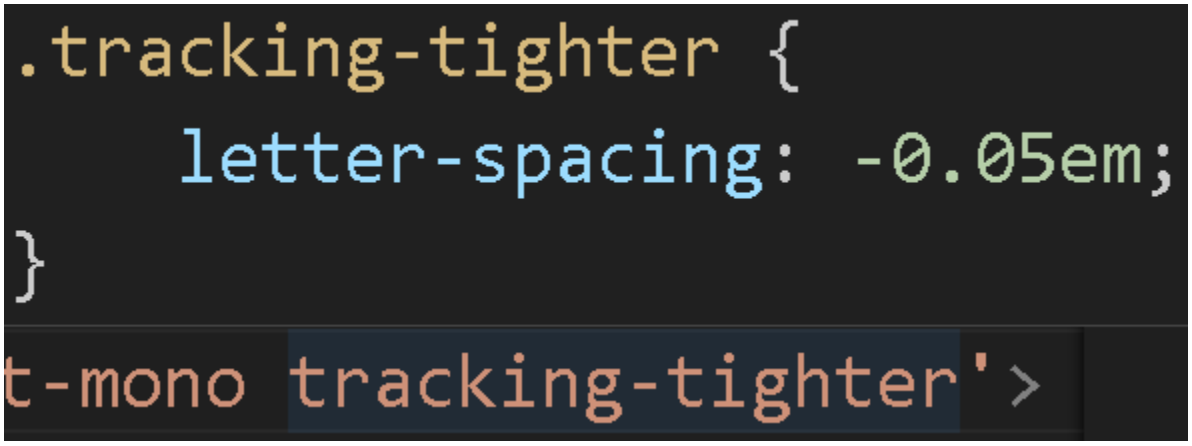
```
import React from 'react'
import Navbar from '../components/Navbar'
```

```
const HomePage = () => {
  return (
    // min-h-screen = minimum height will be screen
    // takes entire screen
    <div class="min-h-screen">
      {/* First it will have navbar component */}
      <Navbar />
    </div>
  )
}

export default HomePage
```

To get the actual size, hover over it:



Tracking-tighter for letter spacing:



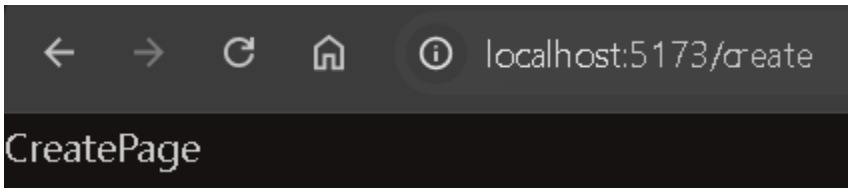
Code in Navbar.jsx:

```
import { PlusIcon } from 'lucide-react'
import {Link} from "react-router"
```

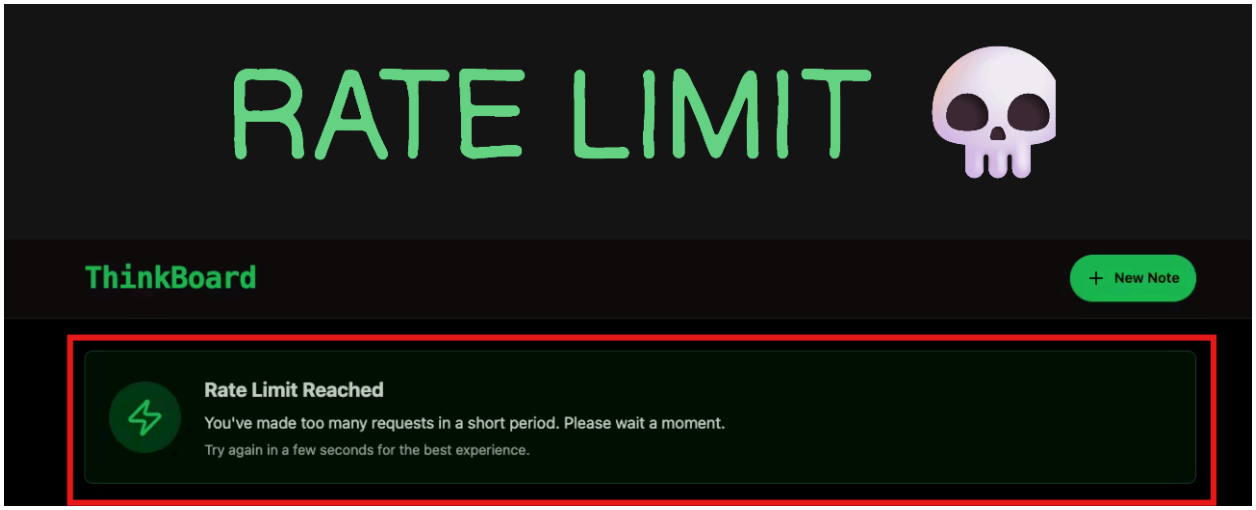
```
import React from 'react'

const Navbar = () => {
  return (
    <header className="bg-base-300 border-b border-base-content/10">
      <div className="mx-auto max-w-6xl p-4">
        <div className='flex items-center justify-between'>
          <h1 className='text-3xl font-bold text-primary font-mono tracking-tight'>
            ThinkBoard
          </h1>
          <div className='flex items-center gap-4'>
            <Link to={"/create"} className="btn btn-primary">
              <PlusIcon className='size-5' />
              <span>New Note</span>
            </Link>
          </div>
        </div>
      </div>
    </header>
  )
}

export default Navbar
```

In browser:

Click on new Note to open the create page:

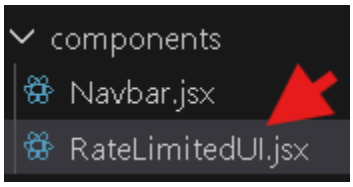


In homepage, if we send lots of requests, or rate limited



We need to get above kind of UI.

Create a file for ratelimiter:



Code in RateLimitedUI.jsx:

```
import { ZapIcon } from "lucide-react";
// Code that does not have any logic at all
const RateLimitedUI = () => {
  return (
    <div className="max-w-6xl mx-auto px-4 py-8">
      <div className="bg-primary/10 border border-primary/30 rounded-lg shadow-md">
        <div className="flex flex-col md:flex-row items-center p-6">
          <div className="flex-shrink-0 bg-primary/20 p-4 rounded-full mb-4 md:mb-0 md:mr-6">
            <ZapIcon className="size-10 text-primary" />
          </div>
          <div className="flex-1 text-center md:text-left">
            <h3 className="text-xl font-bold mb-2">Rate Limit Reached</h3>
            <p className="text-base-content mb-1">
              You've made too many requests in a short period. Please wait a moment.
            </p>
            <p className="text-sm text-base-content/70">
              Try again in a few seconds for the best experience.
            </p>
          </div>
        </div>
      </div>
```

```
    </div>
  );
};

export default RateLimitedUI;
```

Code in HomePage.jsx:

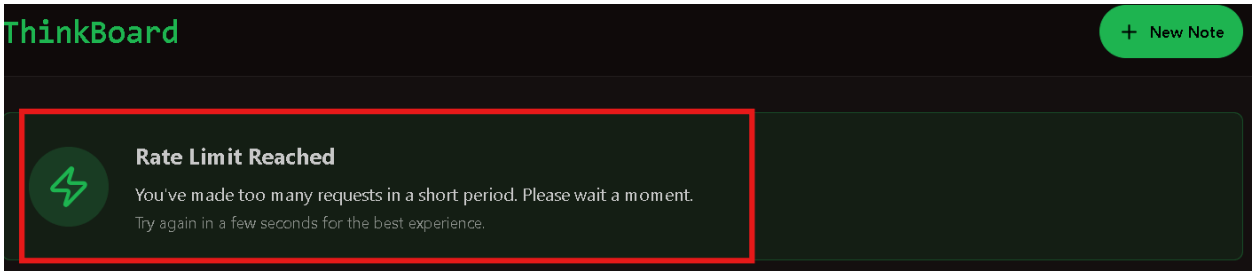
```
import React from 'react'
import { useState } from 'react';
import Navbar from '../components/Navbar'
import RateLimitedUI from '../components/RateLimitedUI';

const HomePage = () => {
  const [isRateLimited, setIsRateLimited] = useState(false);
  return (
    // min-h-screen = minimum height will be screen
    // takes entire screen
    <div className="min-h-screen">
      {/* First it will have navbar component */}
      <Navbar />
      {isRateLimited && <RateLimitedUI />}
    </div>
  );
};

export default HomePage;
```

In browser:

If code in **HomePage.jsx**, set usestate to **true**:

```
const [isRateLimited, setIsRateLimited] = useState(true);
```

In browser:

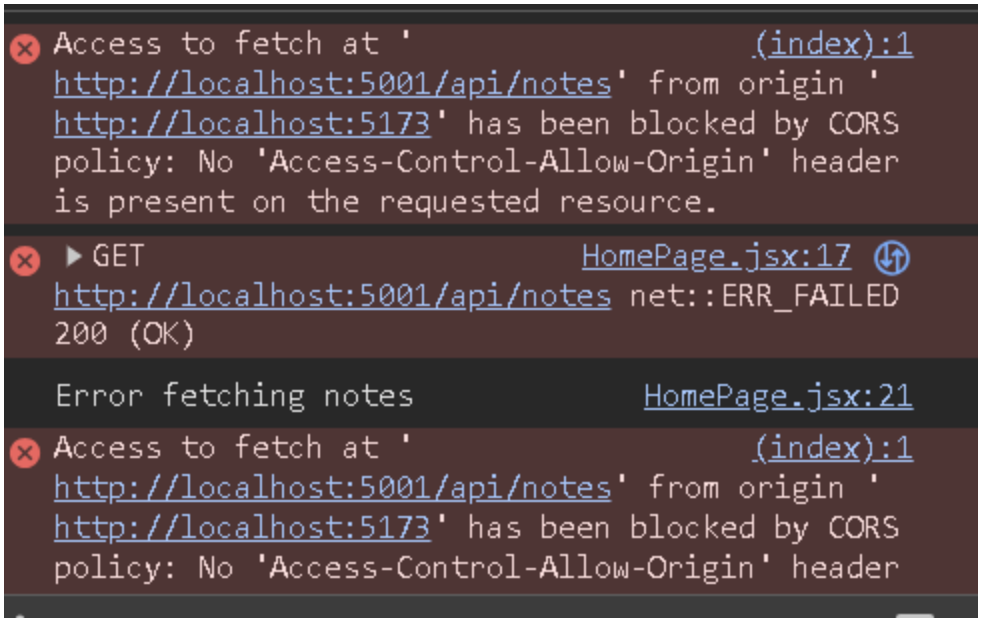


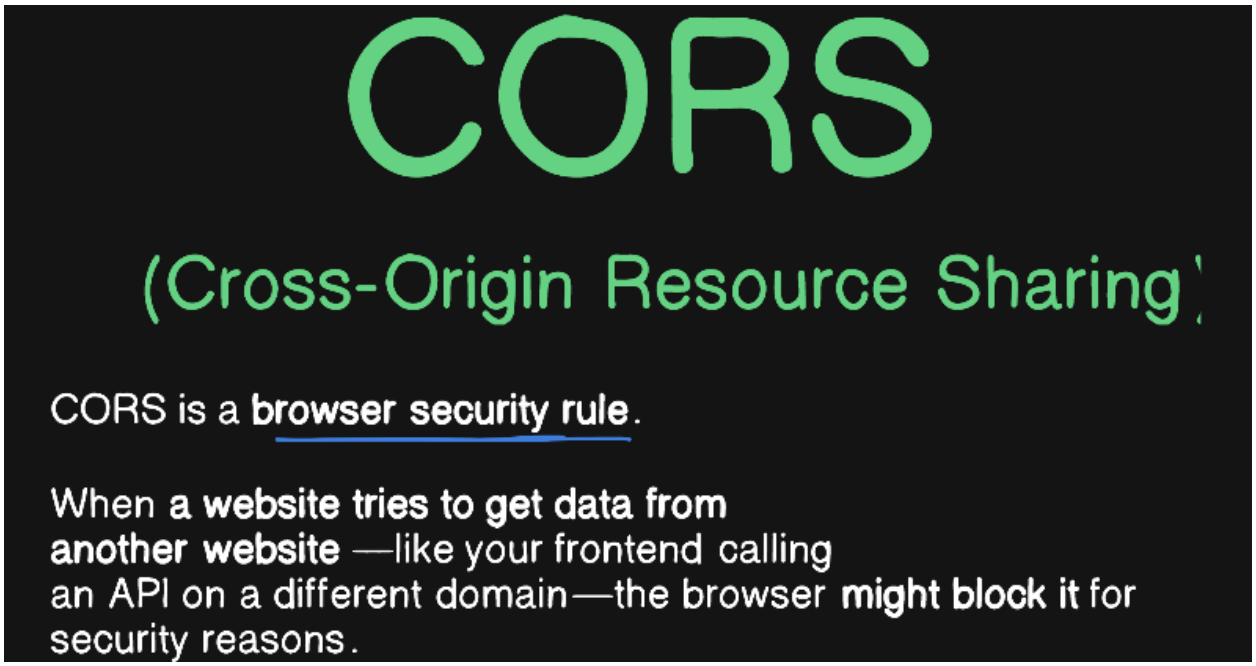
Code in HomePage.jsx: *Using fetching*

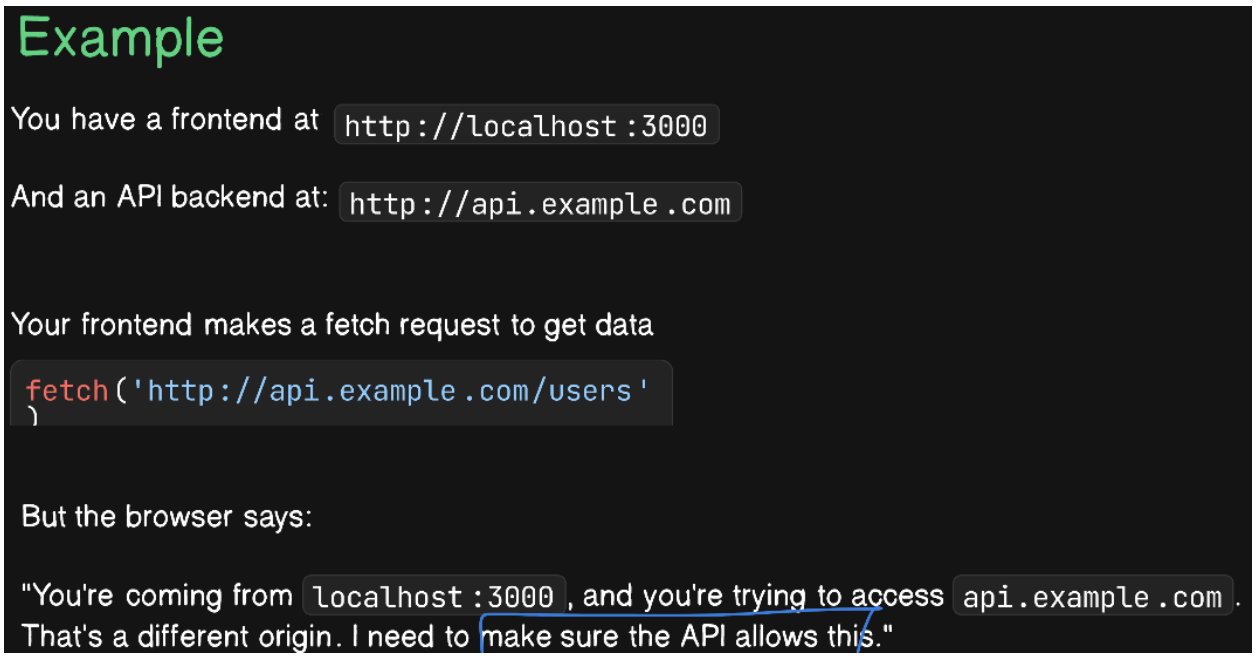
```
import React, { useEffect } from 'react'
import { useState } from 'react';
import Navbar from '../components/Navbar'
import RateLimitedUI from '../components/RateLimitedUI';

const HomePage = () => {
  const [isRateLimited, setIsRateLimited] = useState(true);
  // fetching the notes
  const [notes, setNotes] = useState([])
  // to keep track of loading state, set to true, soon
  // as we load the page, we need to fetch the notes
  const [loading, setLoading] = useState(true)
  // To able to fetch that
  useEffect(() => {
    const fetchNotes = async () => {
      try {
        const res = await fetch("http://localhost:5001/api/notes")
        const data = await res.json()
        console.log(data)
      } catch (error) {
        console.log("Error fetching notes")
      }
    }

    fetchNotes()
  }, [])
  return (
    // min-h-screen = minimum height will be screen
    // takes entire screen
    <div className="min-h-screen">
      {/* First it will have navbar component */}
      <Navbar />
      {isRateLimited && <RateLimitedUI />}
    </div>
  );
};

export default HomePage;
```

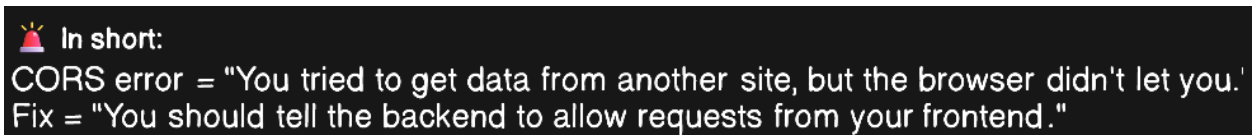
In browser, we get CORS error when inspecting it:

This is nothing related to fetch, but different error


Access to fetch at ' (index):1
http://localhost:5001/api/notes' from origin '
http://localhost:5173' has been blocked by CORS
policy: No 'Access-Control-Allow-Origin' header
is present on the requested resource.
▶ GET HomePage.jsx:17
http://localhost:5001/api/notes net::ERR_FAILED
200 (OK)
Error fetching notes HomePage.jsx:21
Access to fetch at ' (index):1
http://localhost:5001/api/notes' from origin '
http://localhost:5173' has been blocked by CORS
policy: No 'Access-Control-Allow-Origin' header

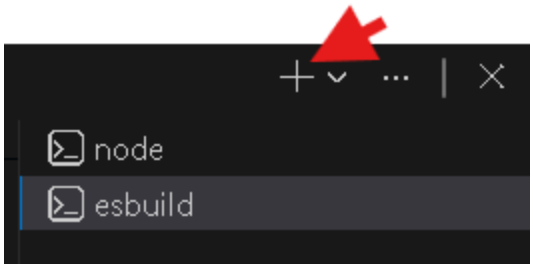






To make sure the API allows this.

Create new terminal:



Visit backend folder.

Install the package:

In terminal: **npm i cors@2.8.5**

```
PS C:\Users\kiran\OneDrive\Desktop\mern-thinkboard\backend> npm i cors@2.8.5

changed 1 package, and audited 130 packages in 2s

18 packages are looking for funding
  run `npm fund` for details

9 vulnerabilities (3 low, 1 moderate, 4 high, 1 critical)

To address all issues, run:
  npm audit fix

Run `npm audit` for details.
```

In package.json:

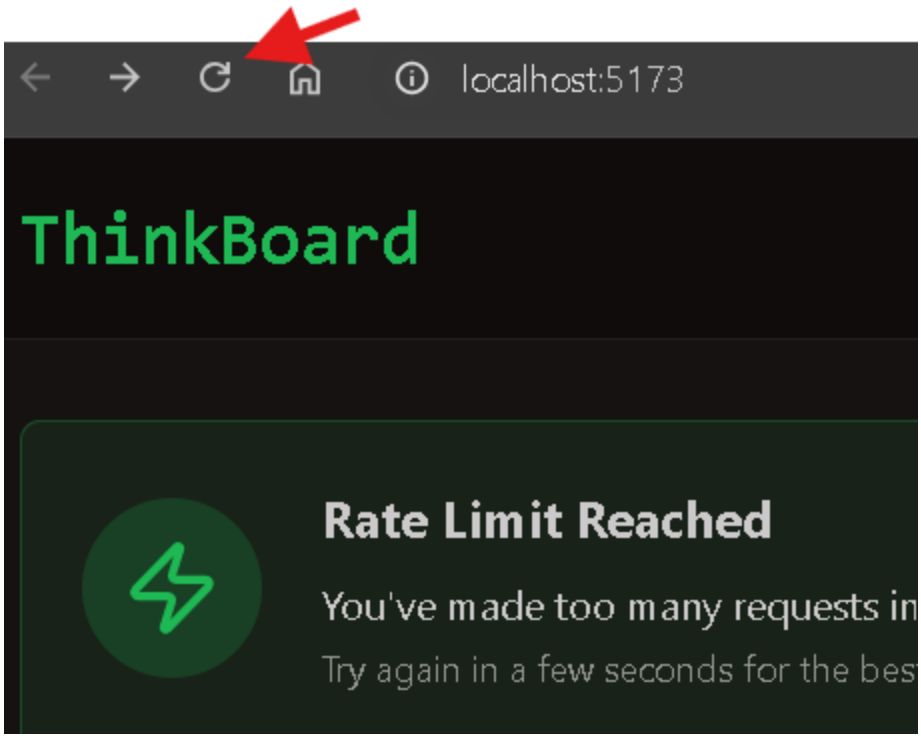
Code in server.js: available in backend/src

import express from "express"

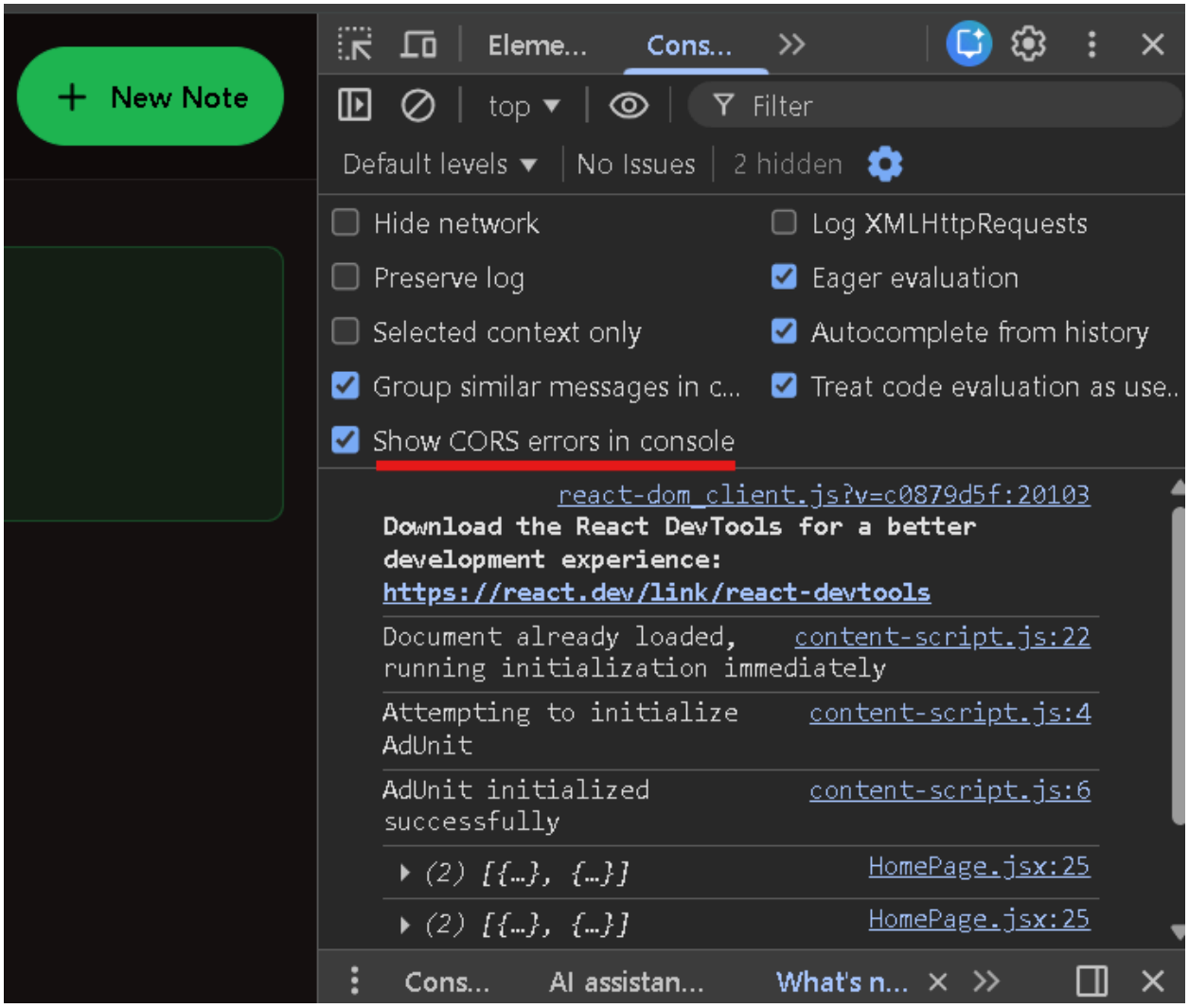
```
import dotenv from "dotenv"
import cors from "cors"

import notesRoutes from "./routes/notesRoutes.js"
import { connectDB } from "./config/db.js"
import rateLimiter from "./middleware/rateLimiter.js"

dotenv.config()

//console.log(process.env.MONGO_URI)

const app = express()
// if process.env.PORT is undefined then PORT = 5001(by default value)
const PORT = process.env.PORT || 5001

//connectDB()

// First try to remove the cors error, then use middleware
app.use(cors({
   // removing error in home page url
   origin: "http://localhost:5173"
}))
// middleware
app.use(express.json()) // this middleware will parse JSON bodies: req.body
app.use(rateLimiter)

// Our simple custom middleware
// app.use((req, res, next) => {
//     console.log(`Req method: ${req.method}.\nReq URL: ${req.url}.`)
//     next()
// })

app.use("/api/notes", notesRoutes)

connectDB().then(() =>{
   app.listen(PORT, () => {
      console.log("Server started on PORT:", PORT)
   })
})
```

Note: bold letters are added.

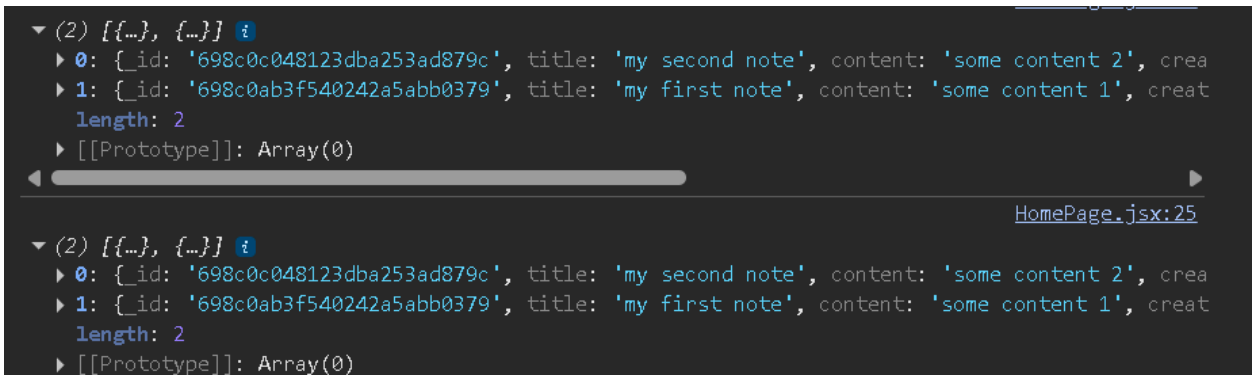
Then refresh the home page:



Then we don’t get CORS errors:



Notes coming from database:



Code in HomePage.jsx:

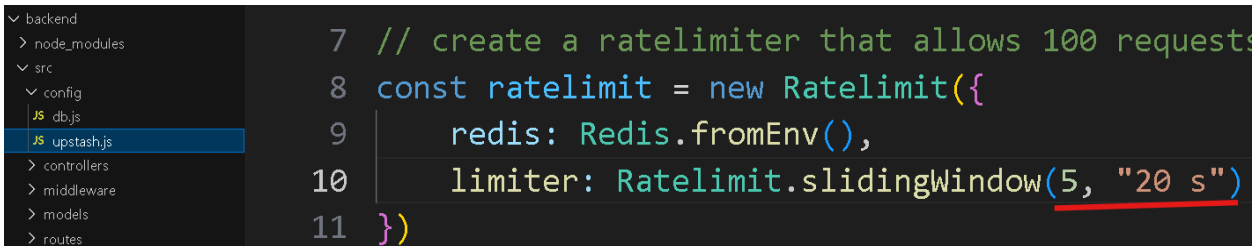
```
import React, { useEffect } from 'react'
import { useState } from 'react';
import Navbar from '../components/Navbar'
```

```
import RateLimitedUI from '../components/RateLimitedUI';
import axios from 'axios';
import toast from 'react-hot-toast';

const HomePage = () => {
  const [isRateLimited, setIsRateLimited] = useState(false);
  // fetching the notes
  const [notes, setNotes] = useState([])
  // to keep track of loading state, set to true, soon
  // as we load the page, we need to fetch the notes
  const [loading, setLoading] = useState(true)
  // To able to fetch that
  useEffect(() => {
    const fetchNotes = async () => {
      try {
        // For get method
        // const res = await axios.get("http://localhost:5001/api/notes")
        // For post method
        // const res = await axios.post("http://localhost:5001/api/notes")
        const res = await axios.get("http://localhost:5001/api/notes")
        // const data = await res.json()
        // Which is lot more comfortable then fetch API
        // console.log(res.data)// instead of console logging the data
        // We want to update the notes state
        setNotes(res.data)
        // If we can get the data, ratelimited to false
        setIsRateLimited(false)
      } catch (error) {
        console.log("Error fetching notes")
        // We want to check for the state in:
        // const [isRateLimited, setIsRateLimited] = useState(false);
        // 429 for rateLimited
        if(error.response.status === 429){
          setIsRateLimited(true)
        }else{
          toast.error("Failed to load Notes!")
        }
      } finally{
        setLoading(false)
      }
    }

    fetchNotes()
  }, [])
```

```
  return (
    // min-h-screen = minimum height will be screen
    // takes entire screen
    <div className="min-h-screen">
      {/* First it will have navbar component */}
      <Navbar />
      {isRateLimited && <RateLimitedUI />}
    </div>
  );
};

export default HomePage;
```

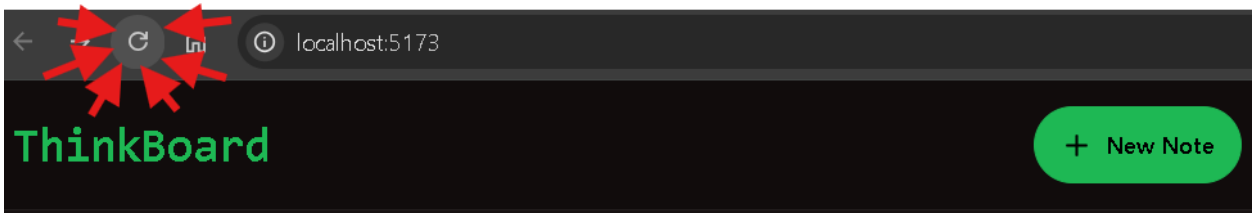
Changing the ratelimit, for testing:

backend>src>config>upstash.js:

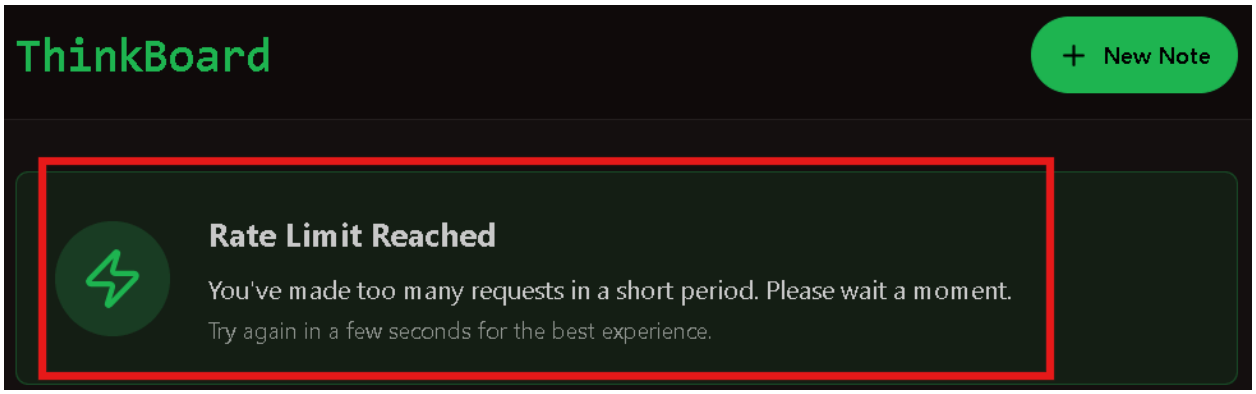


5 request per 20 seconds.

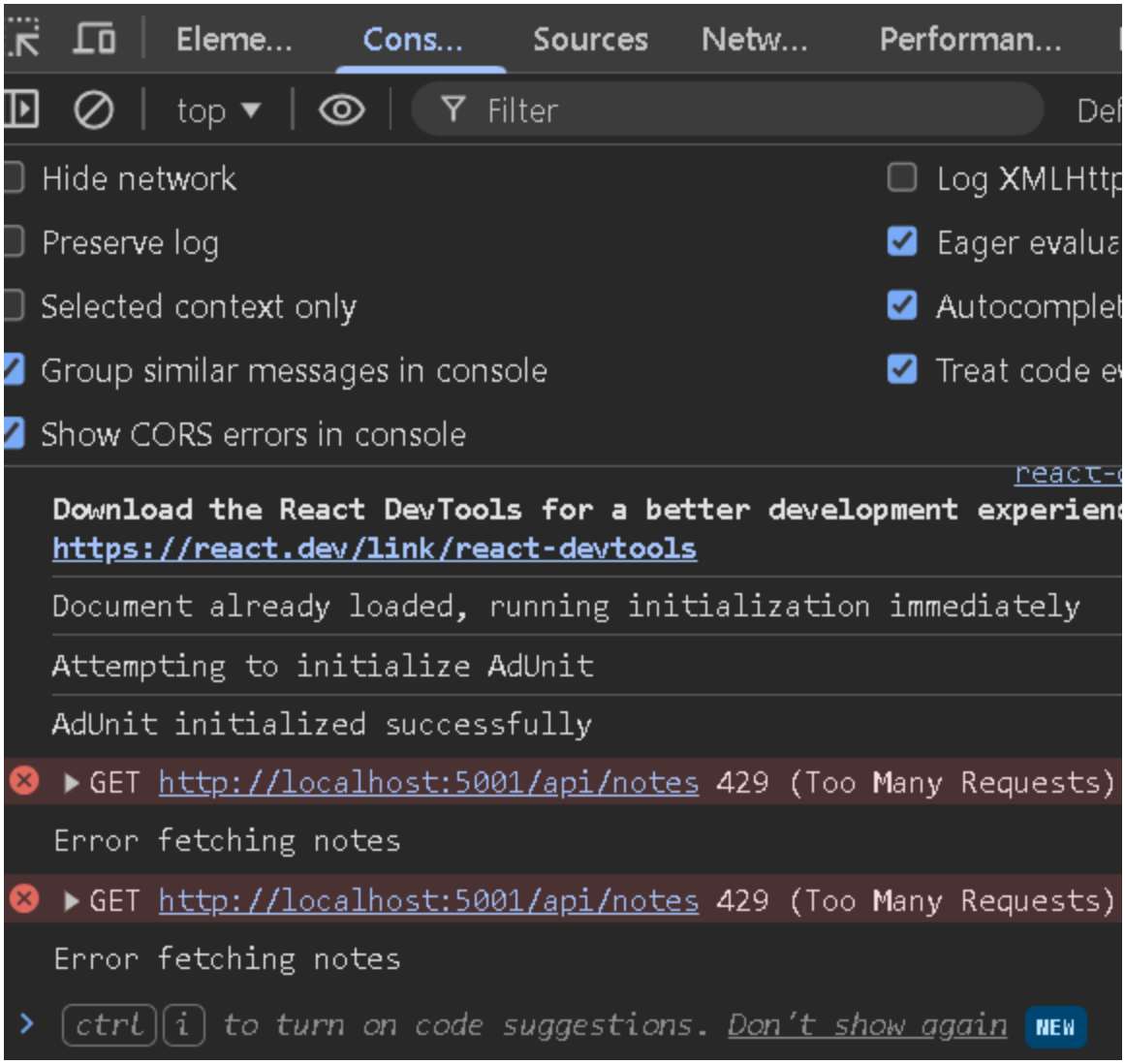
Then try to refresh more than 5 times within 20 seconds:



Then UI:



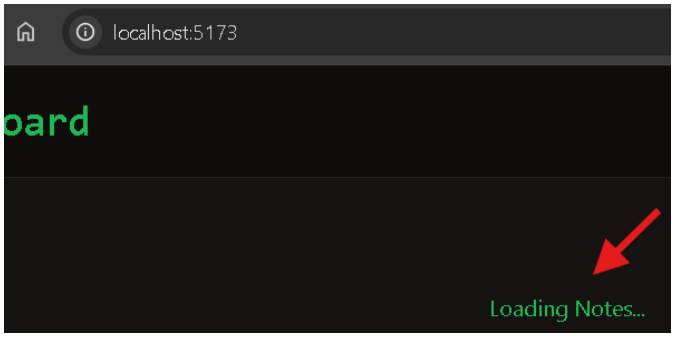
In console:



Some code in HomePage.jsx: if true is set

```
<div className='max-w-7xl mx-auto p-4 mt-6'>
        {true && <div className='text-center text-primary py-10'>Loading Notes...</div>}
    </div>
```

In browser:

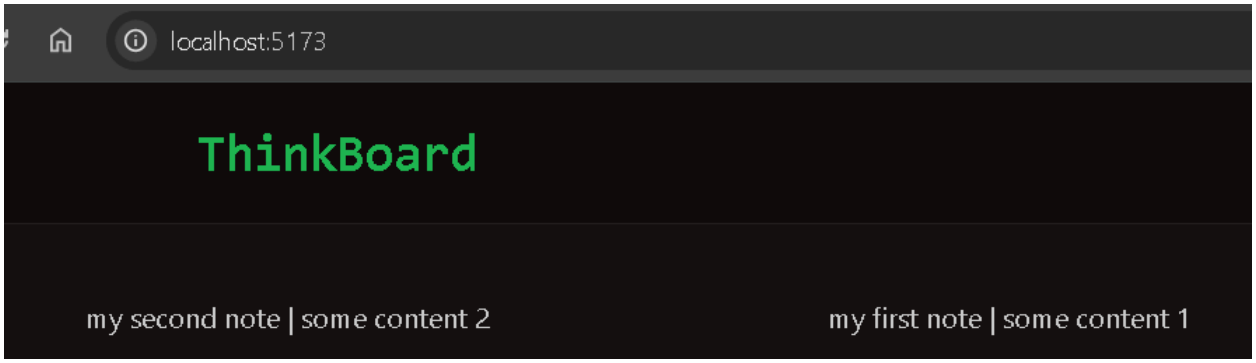


Code in HomePage.jsx:

```
import React, { useEffect } from 'react'
import { useState } from 'react';
import Navbar from '../components/Navbar'
import RateLimitedUI from '../components/RateLimitedUI';
import axios from 'axios';
import toast from "react-hot-toast";

const HomePage = () => {
  const [isRateLimited, setIsRateLimited] = useState(false);
  // fetching the notes
  const [notes, setNotes] = useState([])
  // to keep track of loading state, set to true, soon
  // as we load the page, we need to fetch the notes
  const [loading, setLoading] = useState(true)
  // To able to fetch that
  useEffect(() => {
    const fetchNotes = async () => {
      try {
        // For get method
        // const res = await axios.get("http://localhost:5001/api/notes")
        // For post method
        // const res = await axios.post("http://localhost:5001/api/notes")
        const res = await axios.get("http://localhost:5001/api/notes")
        // const data = await res.json()
        // Which is lot more comfortable then fetch API
        // console.log(res.data)// instead of console logging the data
        // We want to update the notes state
        setNotes(res.data)
        // If we can get the data, ratelimited to false
        setIsRateLimited(false)
      } catch (error) {
        console.log("Error fetching notes")
        console.log(error.response)
        // We want to check for the state in:
        // const [isRateLimited, setIsRateLimited] = useState(false);
        // 429 for rateLimited
        if(error.response?.status === 429){
          setIsRateLimited(true)
        }else{
          toast.error("Failed to load Notes")
        }
      } finally{
```

```
        setLoading(false)
      }
    }

    fetchNotes()
  }, [])
  return (
    // min-h-screen = minimum height will be screen
    // takes entire screen
    <div className="min-h-screen">
      {/* First it will have navbar component */}
      <Navbar />
      {isRateLimited && <RateLimitedUI />}

      <div className="max-w-7xl mx-auto p-4 mt-6">
        {loading && <div className="text-center text-primary py-10">Loading notes...</div>}
        {notes.length > 0 && !isRateLimited && (
          <div className="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-3 gap-6">
            {notes.map(note => (
              <div>
                {note.title} | {note.content}
              </div>
            ))}
          </div>
        )}
      </div>
    </div>
  );
};

export default HomePage;
```

In browser, refresh it:

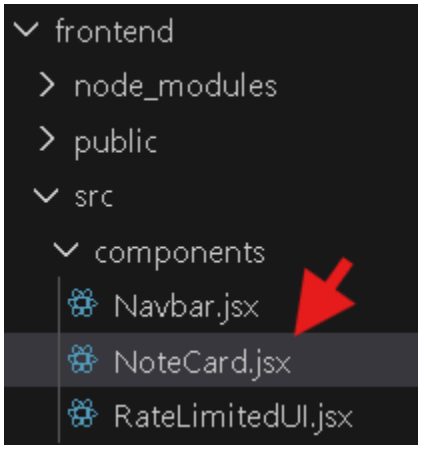


From database:

```
_id: ObjectId('698c0ab3f540242a5abb0379')
title : "my first note"
content : "some content 1"
createdAt : 2026-02-11T04:50:59.240+00:00
updatedAt : 2026-02-11T04:50:59.240+00:00
__v : 0
```

```
_id: ObjectId('698c0c048123dba253ad879c')
title : "my second note"
content : "some content 2"
createdAt : 2026-02-11T04:56:36.167+00:00
updatedAt : 2026-02-11T04:56:36.167+00:00
__v : 0
```

Replace bold letters in HomePage.jsx:
<NoteCard key={note._id} note={note} />
Then create NoteCard.jsx:



Code in HomePage.jsx:

```
import React, { useEffect } from 'react'
import { useState } from 'react';
import Navbar from '../components/Navbar'
import RateLimitedUI from '../components/RateLimitedUI';
import axios from 'axios';
import toast from "react-hot-toast";
import NoteCard from "../components/NoteCard"

const HomePage = () => {
```

```
const [isRateLimited, setIsRateLimited] = useState(false);
// fetching the notes
const [notes, setNotes] = useState([])
// to keep track of loading state, set to true, soon
// as we load the page, we need to fetch the notes
const [loading, setLoading] = useState(true)
// To able to fetch that
useEffect(() => {
  const fetchNotes = async () => {
    try {
      // For get method
      // const res = await axios.get("http://localhost:5001/api/notes")
      // For post method
      // const res = await axios.post("http://localhost:5001/api/notes")
      const res = await axios.get("http://localhost:5001/api/notes")
      // const data = await res.json()
      // Which is lot more comfortable then fetch API
      // console.log(res.data)// instead of console logging the data
      // We want to update the notes state
      setNotes(res.data)
      // If we can get the data, ratelimited to false
      setIsRateLimited(false)
    } catch (error) {
      console.log("Error fetching notes")
      console.log(error.response)
      // We want to check for the state in:
      // const [isRateLimited, setIsRateLimited] = useState(false);
      // 429 for rateLimited
      if(error.response?.status === 429){
        setIsRateLimited(true)
      }else{
        toast.error("Failed to load Notes")
      }
    } finally{
      setLoading(false)
    }
  }

  fetchNotes()
}, [])
return (
  // min-h-screen = minimum height will be screen
  // takes entire screen
  <div className="min-h-screen">
```

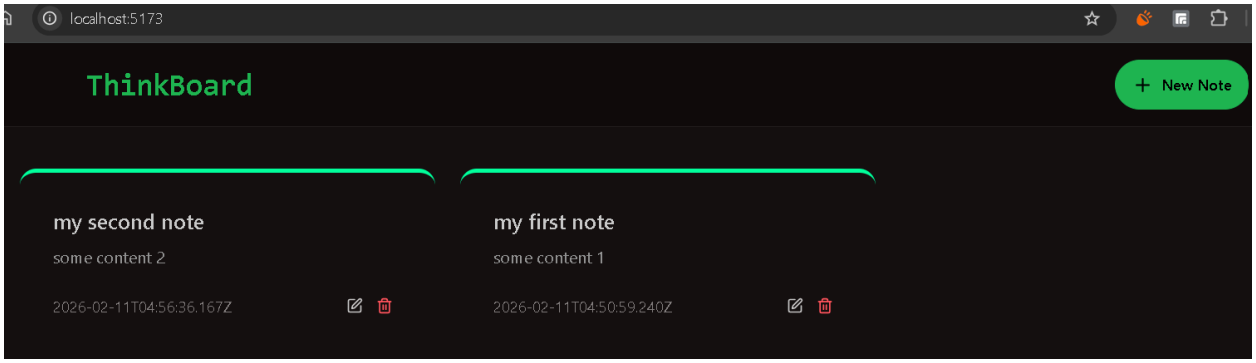
```
      {/* First it will have navbar component */}
      <Navbar />
      {isRateLimited && <RateLimitedUI />}

      <div className="max-w-7xl mx-auto p-4 mt-6">
        {loading && <div className="text-center text-primary py-10">Loading notes...</div>}
        {notes.length > 0 && !isRateLimited && (
          <div className="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-3 gap-6">
            {notes.map(note => (
              <NoteCard key={note._id} note={note} />
            ))}
          </div>
        )}
      </div>
    </div>
  );
};

export default HomePage;
```

Code in NoteCard.jsx:

```
import { PenSquareIcon, Trash2Icon } from "lucide-react";
import { Link } from "react-router";

const NoteCard = ({note}) => {
  return (
    <Link
      to={`/note/${note._id}`}
      className="card bg-base-100 hover:shadow-lg transition-all duration-200
      border-t-4 border-solid border-[#00FF9D]"
    >
      <div className="card-body">
        <h3 className="card-title text-base-content">{note.title}</h3>
        <p className="text-base-content/70 line-clamp-3">{note.content}</p>
        <div className="card-actions justify-between items-center mt-4">
          <span className="text-sm text-base-content/60">
            {note.createdAt}
          </span>
          <div className="flex items-center gap-1">
            <PenSquareIcon className="size-4" />
            <button
              className="btn btn-ghost btn-xs text-error"
            >
```

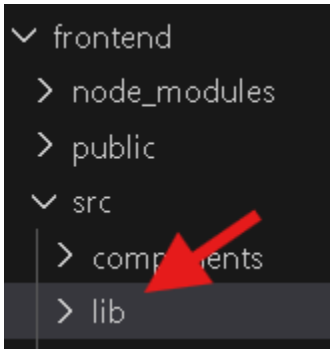
```
            <Trash2Icon className="size-4" />
          </button>
        </div>
      </div>
    </div>
  </Link>
 );
}

export default NoteCard
```

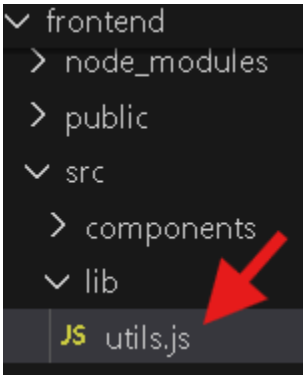
In browser:



But date is not formatted well.

Create a lib folder:





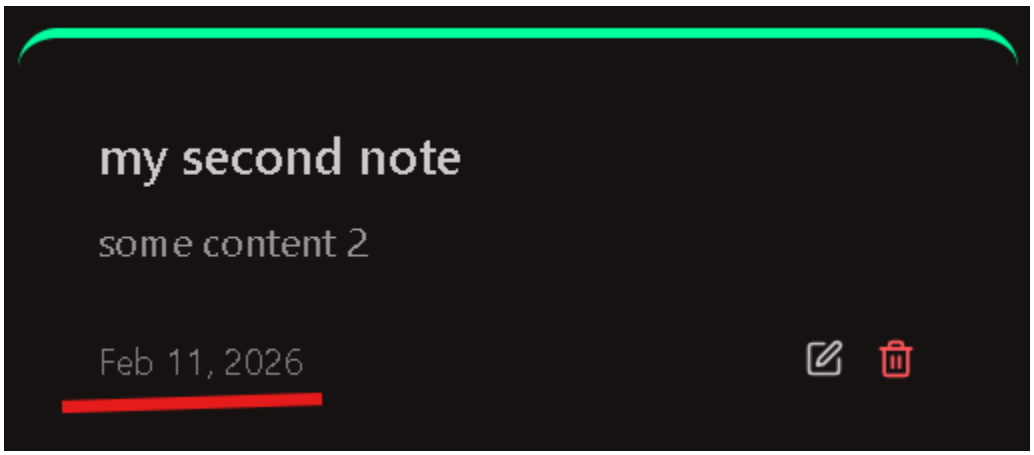
https://youtu.be/F9gB5b4jgOI?t=8931

Code in utils.js:

```
export function formatDate(date){
  return date.toLocaleDateString("en-US", {
    month: "short",
    day: "numeric",
    year: "numeric"
  })
}
```

Sample Code in NoteCard.jsx:

In browser:



Code in App.jsx:

```
import React from 'react'
import { Route, Routes } from 'react-router'
import HomePage from './pages/HomePage'
import CreatePage from './pages/CreatePage'
import NoteDetailPage from './pages/NoteDetailPage'
// import toast
import toast from "react-hot-toast"

const App = () => {
  return (
    // Relative height=full, width=full
    <div className="relative h-full w-full">
      <div className="absolute inset-0 -z-10 h-full w-full items-center px-5 py-24
[background:radial-gradient(125%_125%_at_50%_10%,#000_60%,#00FF9D40_100%)]" />
```

```
    <Routes>
      <Route path="/" element={<HomePage />} />
      <Route path="/create" element={<CreatePage />} />
      {/* :id <- is dynamic */}
      <Route path="/note/:id" element={<NoteDetailPage />} />
    </Routes>

    </div>
  )
}

export default App
```

In browser: