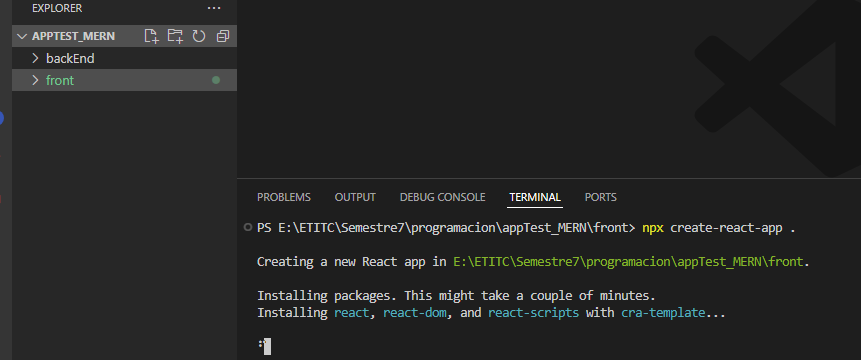
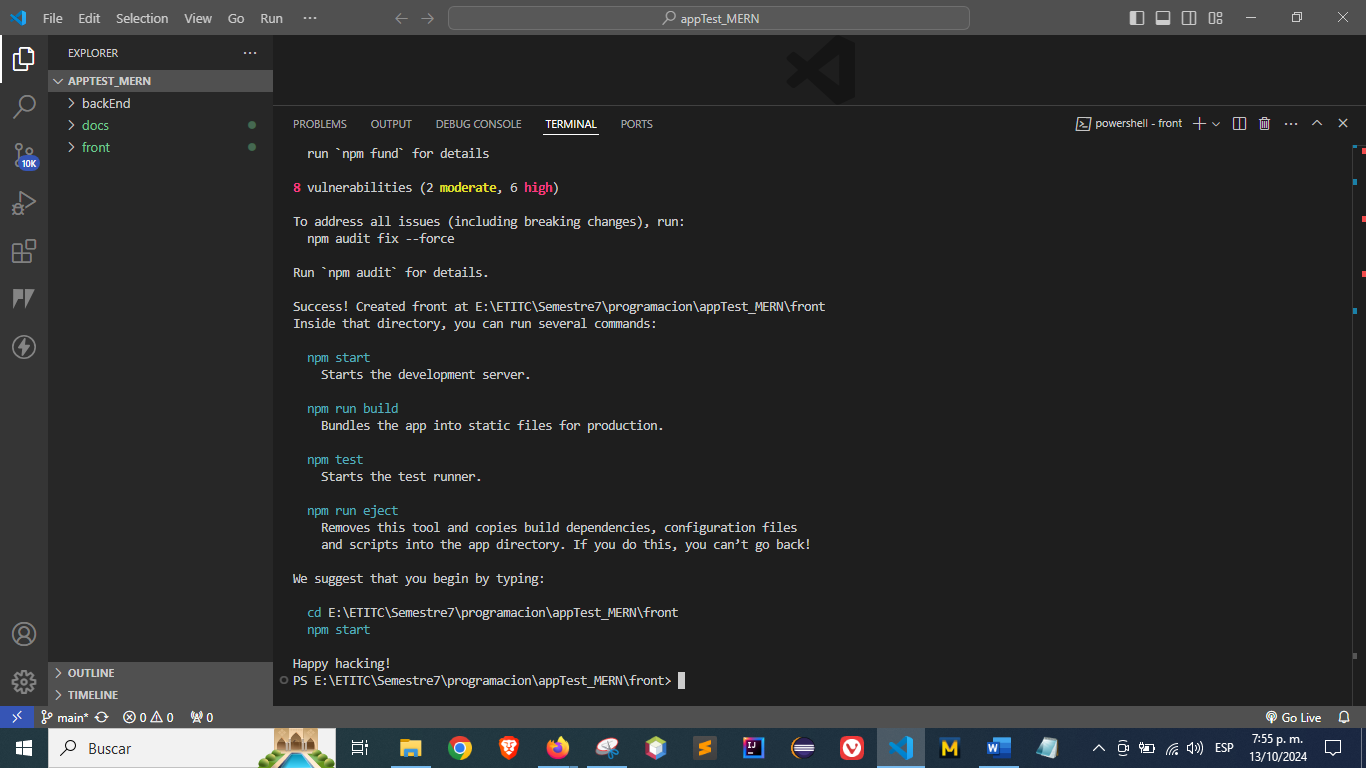
# Desarrollo de aplicación para pruebas o test en línea con la pila MERN

Creamos dos carpetas, una para el FrontEnd y otra para el BackEnd.

Empezamos ubicándonos en la terminal en la carpeta frontEnd y creamos la aplicación:

npx create-react-app .





Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

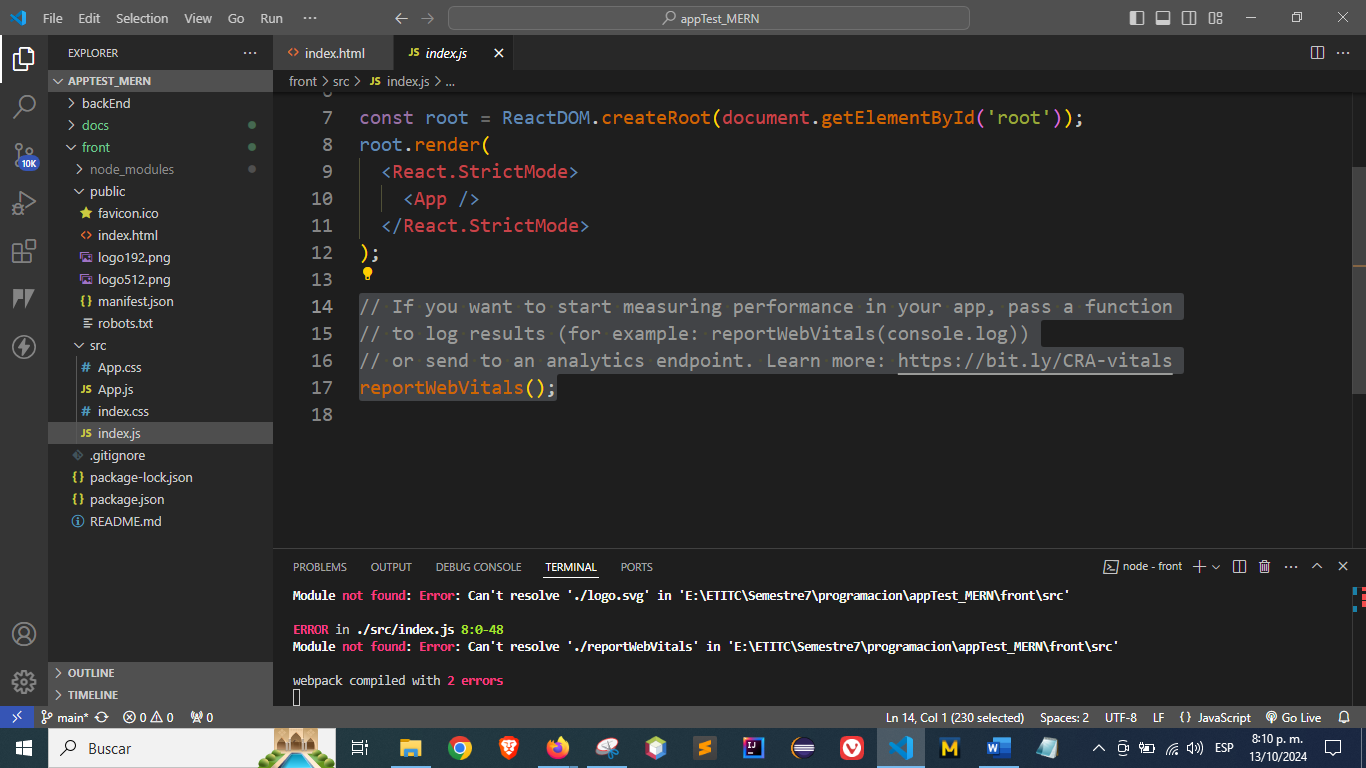
Captura de pantalla de computadora

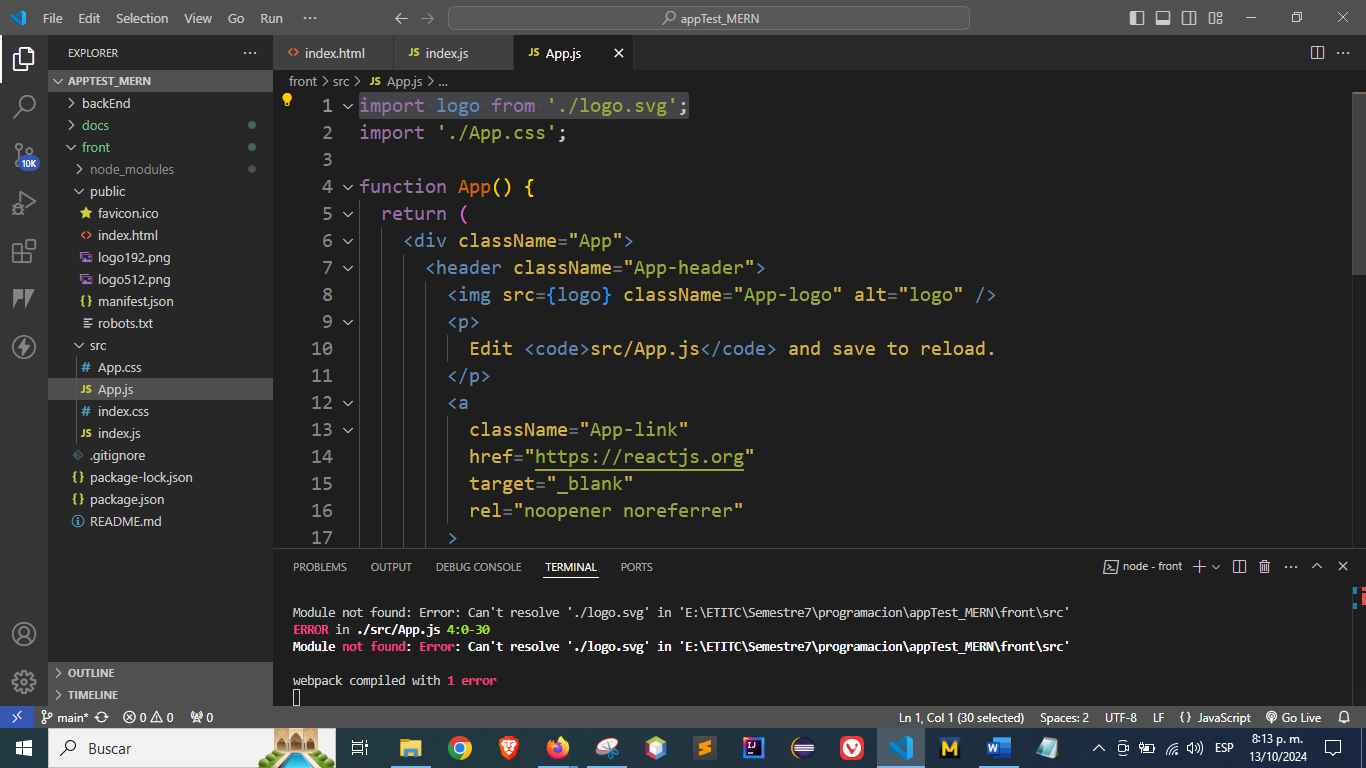
Descripción generada automáticamente

Eliminamos los archivos:

App.test.js setup.Test.js reportWebVitals logo.svg

En el archivo index eliminamos lo innecesario





Así tenemos nuestra aplicación en blanco

Interfaz de usuario gráfica, Texto, Aplicación, Word

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

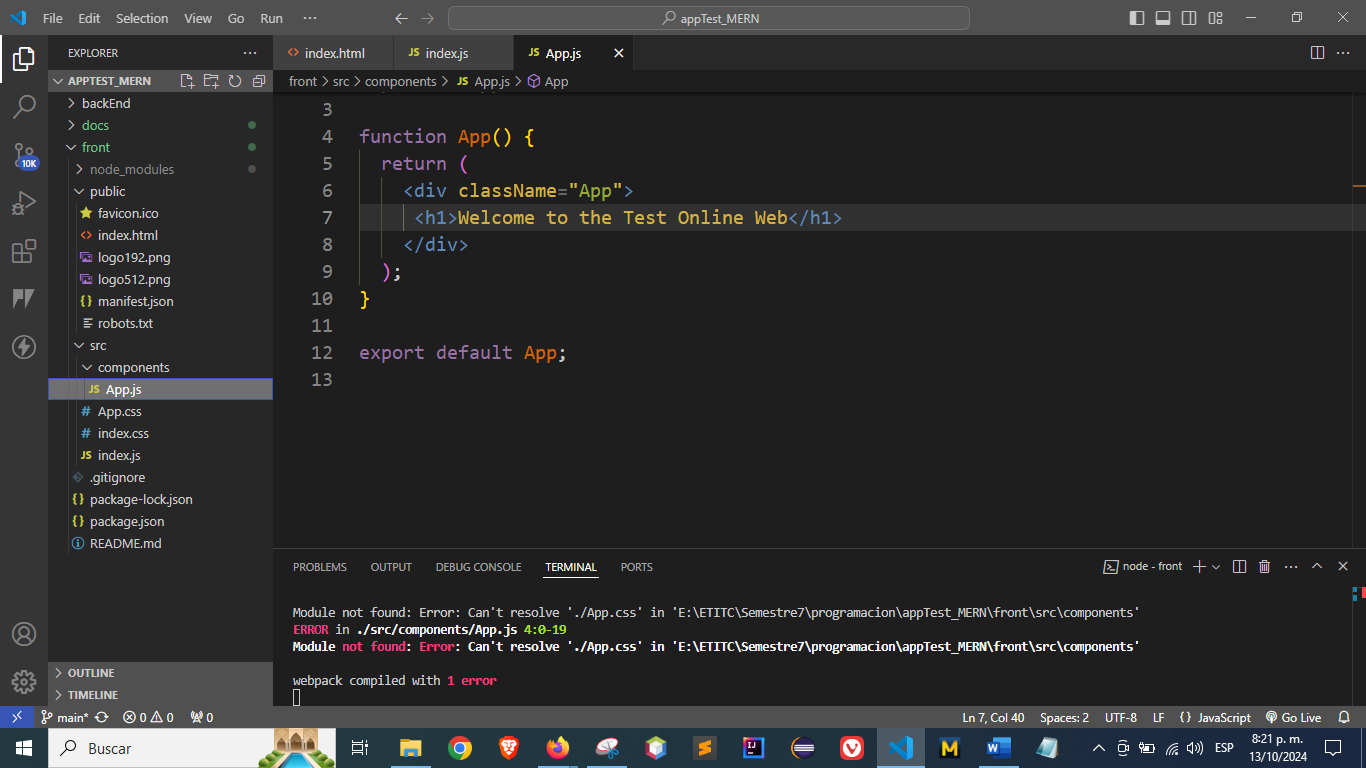
Descripción generada automáticamente

Creamos la carpeta components en la carpeta src

Captura de pantalla de computadora

Descripción generada automáticamente

Movemos el archivo app.js a la carpeta components

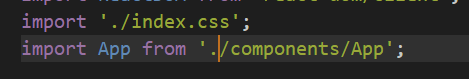


Creamos la carpeta styles y movemos los archivos css

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Ahora hay que modificar las rutas de los archivos css agregando la ruta de la carpeta en los archivos index.js y App.js

 Interfaz de usuario gráfica, Sitio web

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Instalando las librerías:

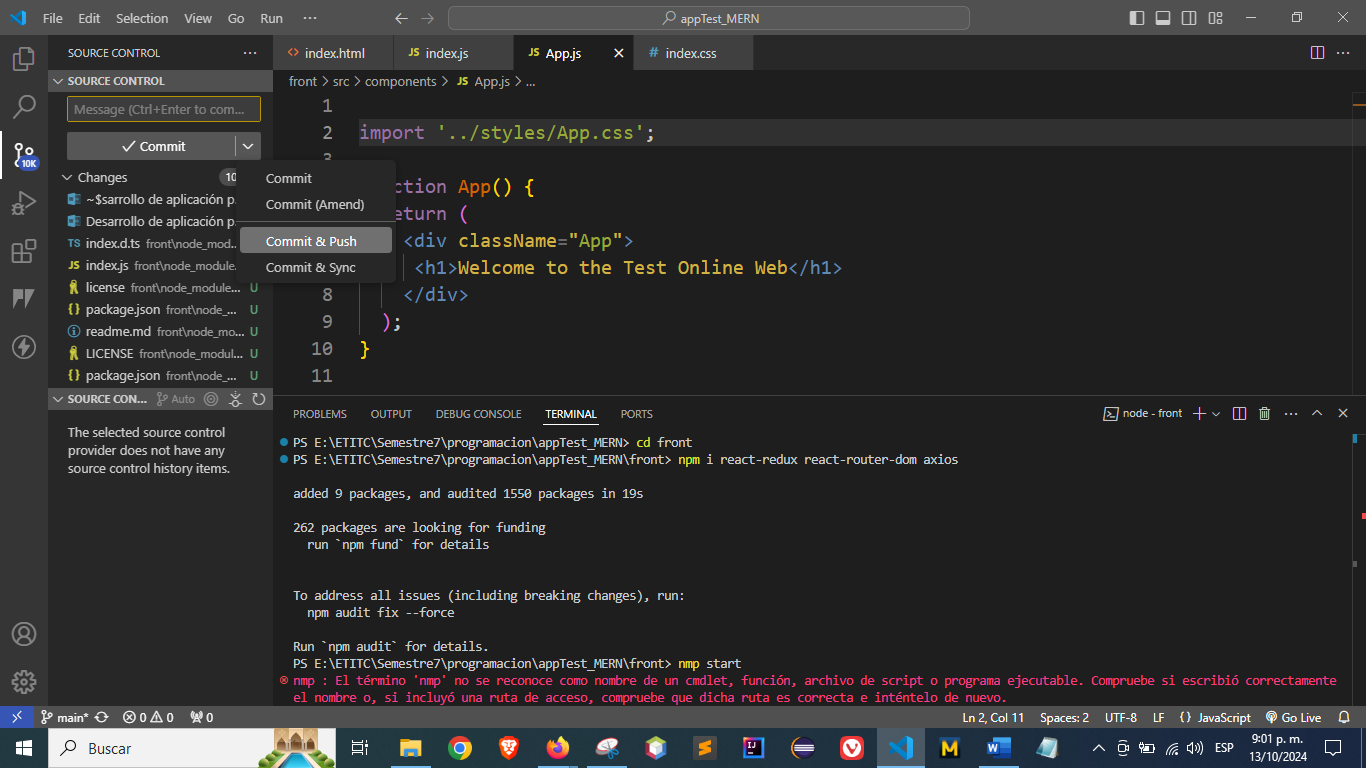


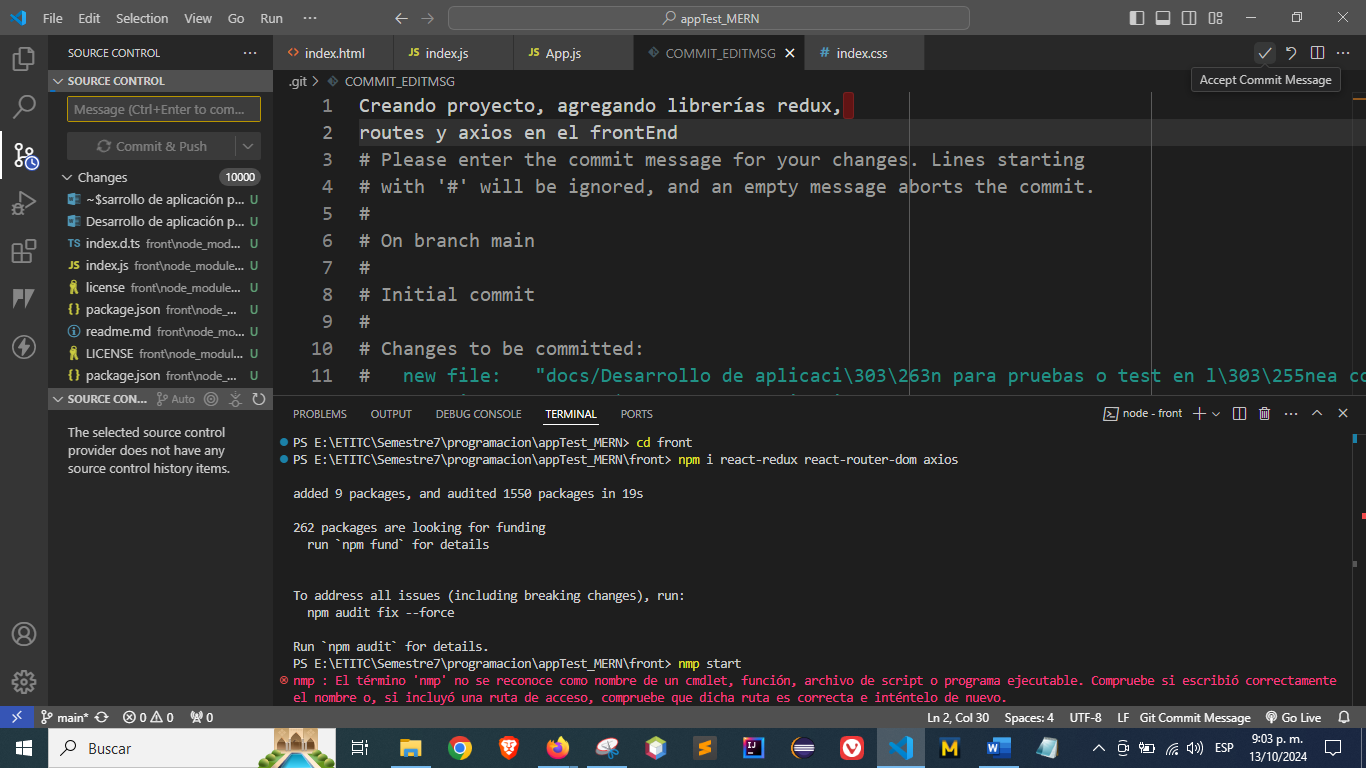
npm i react-redux react-router-dom axios

Texto

Descripción generada automáticamente

Haciendo primer commit desde local a repositorio





## Creando las rutas

import {createBrowserRouter,RouterProvider} from "react-router-dom"

const router = createBrowserRouter([

{

path:'/',

element: <div>Root Element</div>

},

{

path:'/quiz',

element: <div>Quiz Component</div>

},

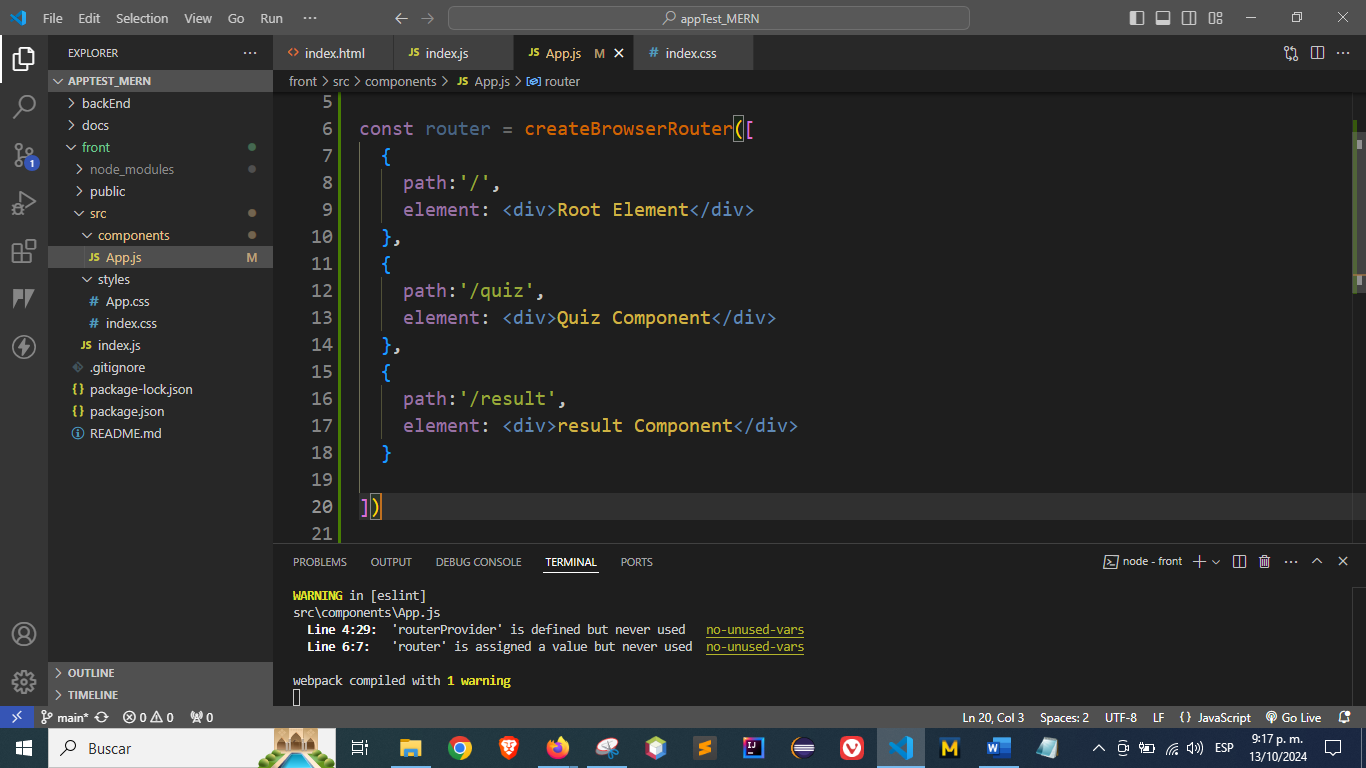
{

path:'/result',

element: <div>result Component</div>

}

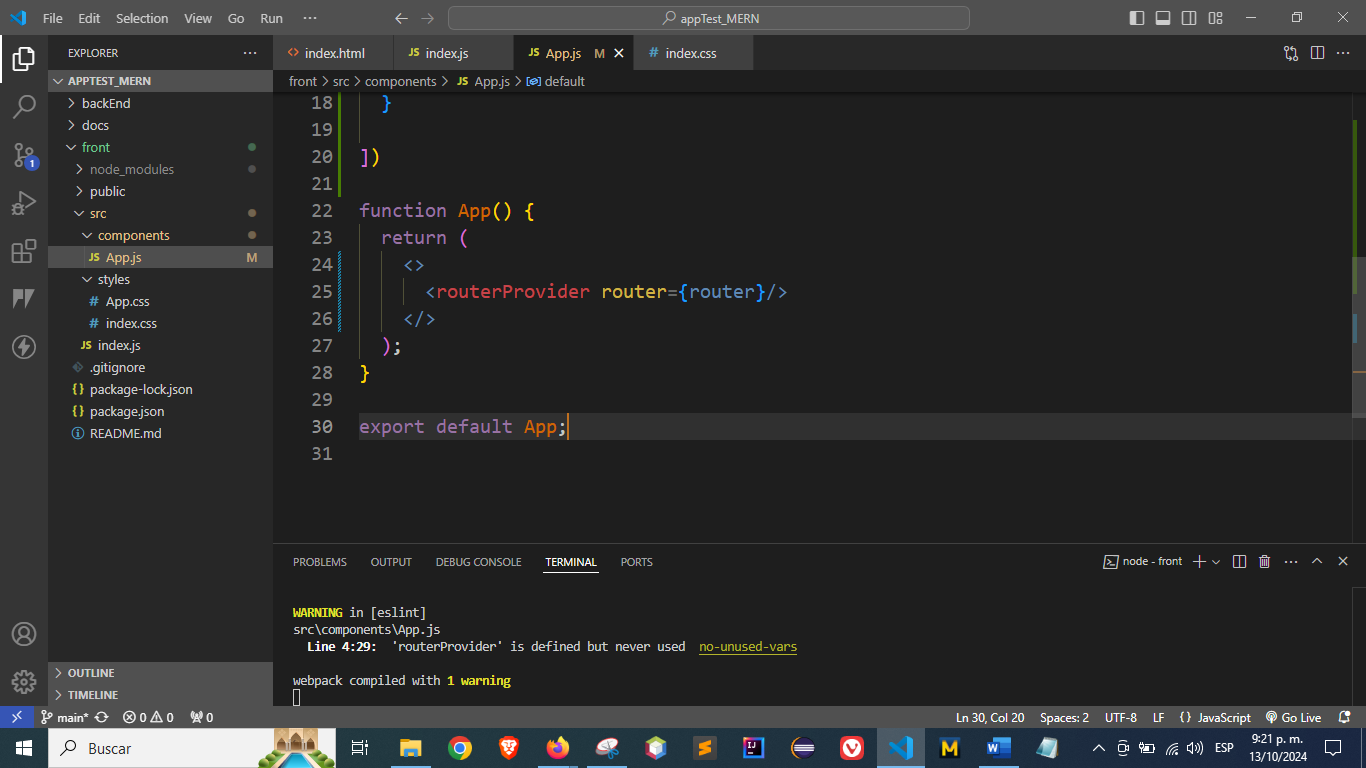
])



## Agregando las rutas a la función principal

Texto

Descripción generada automáticamente



Ahora esta es la página principal

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Vista del quiz

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

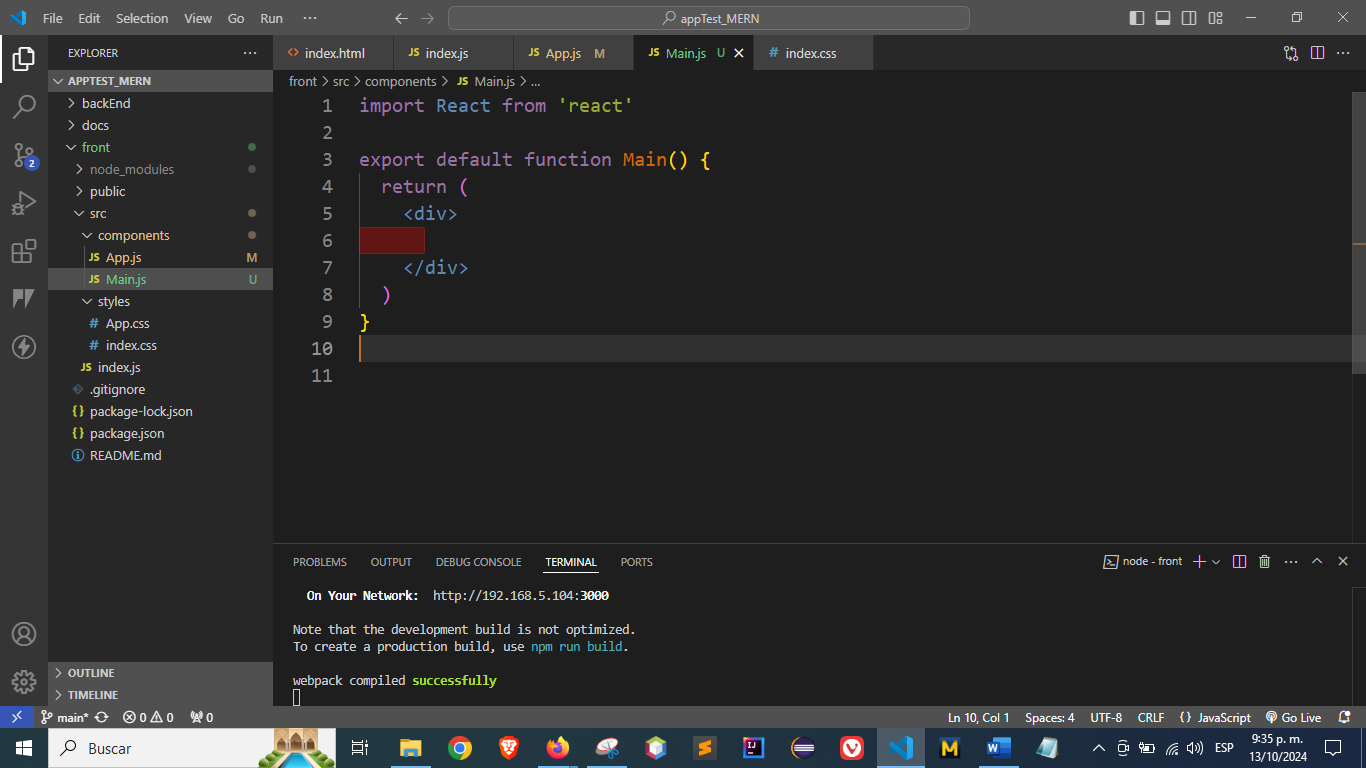
Vista del resultado

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

En la carpeta components creamos los siguientes archivos

### Main.js



En el archivo App.js importamos

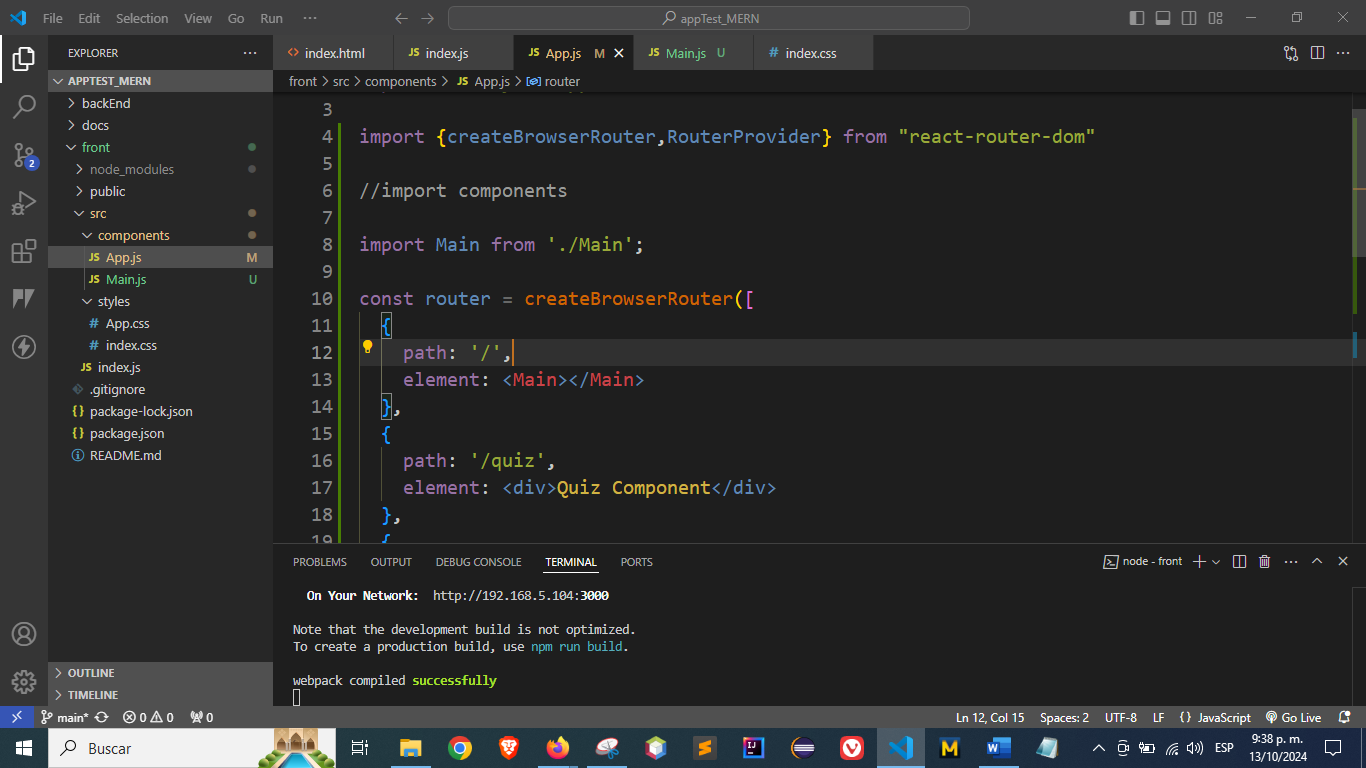
import Main from './Main';

{

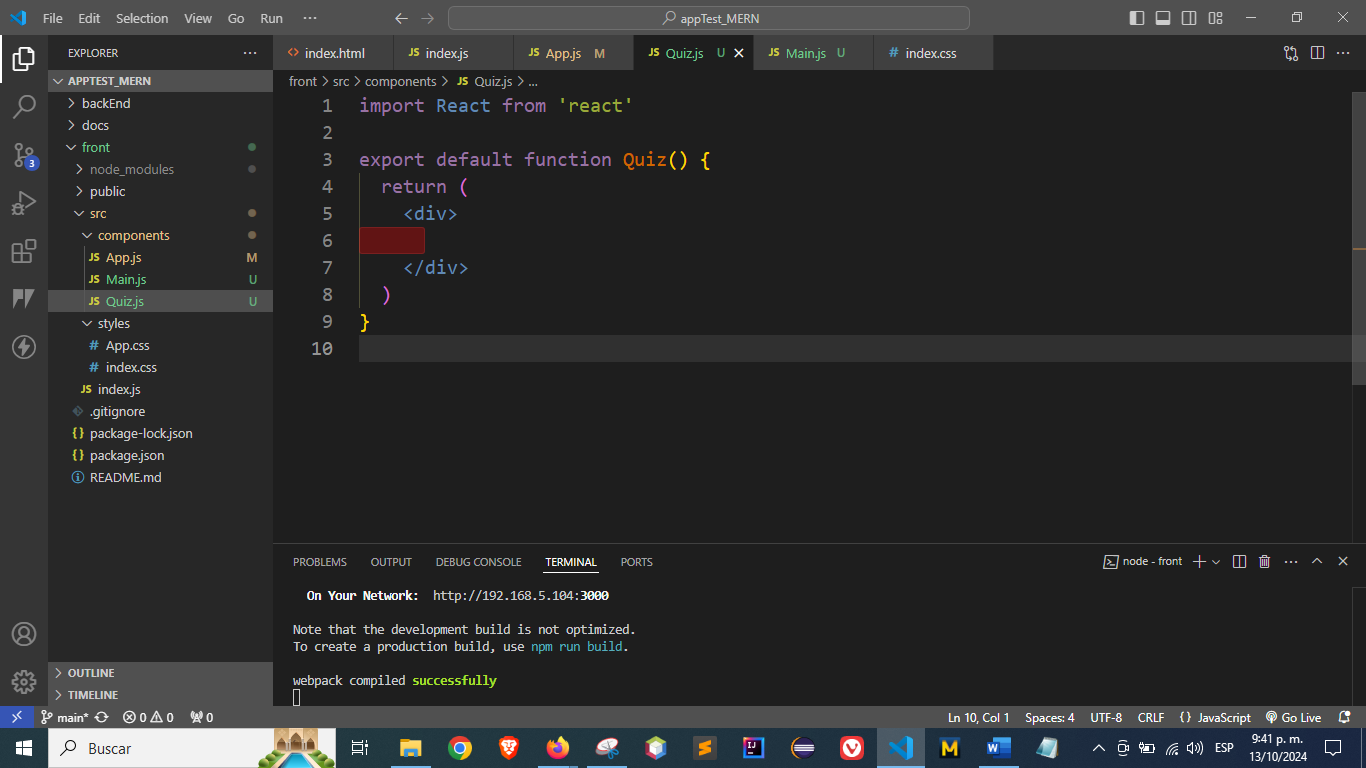
    path: '/',

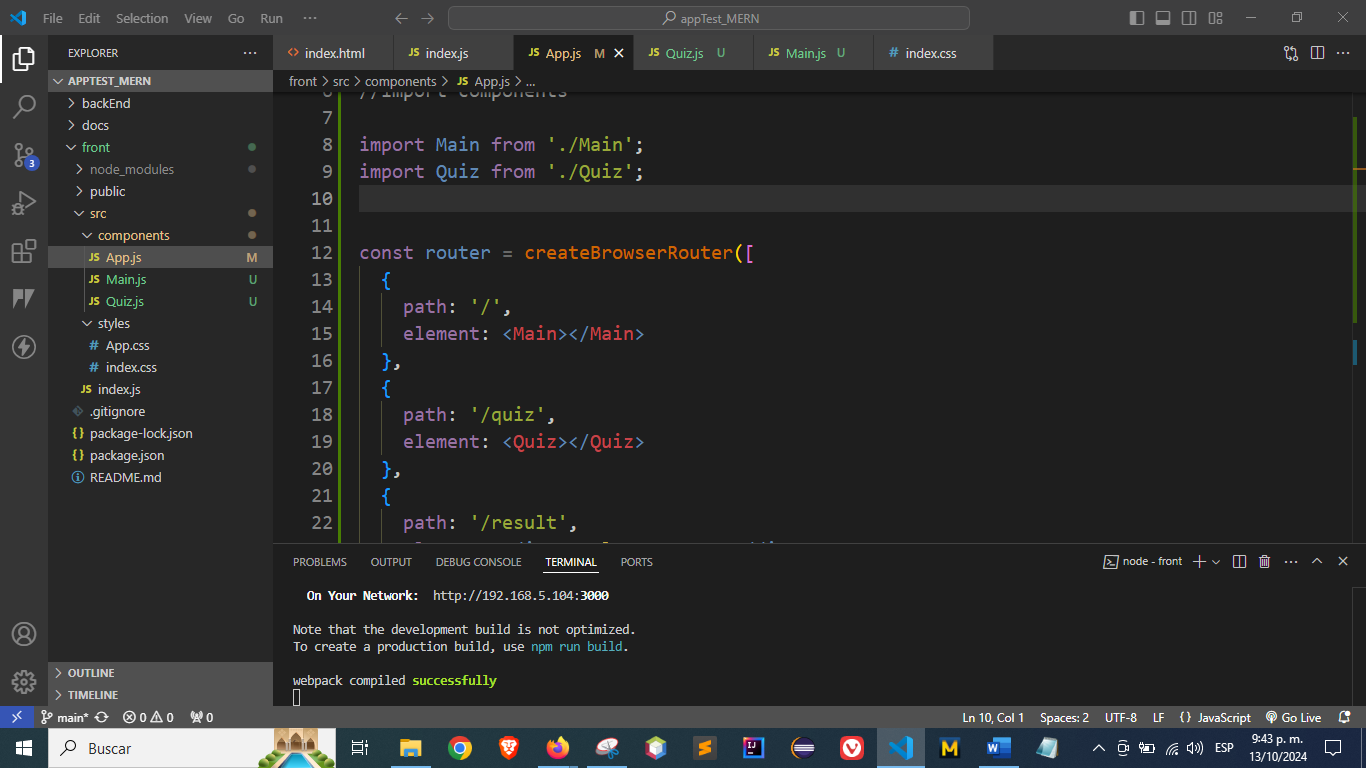
    element: <Main></Main>

  },

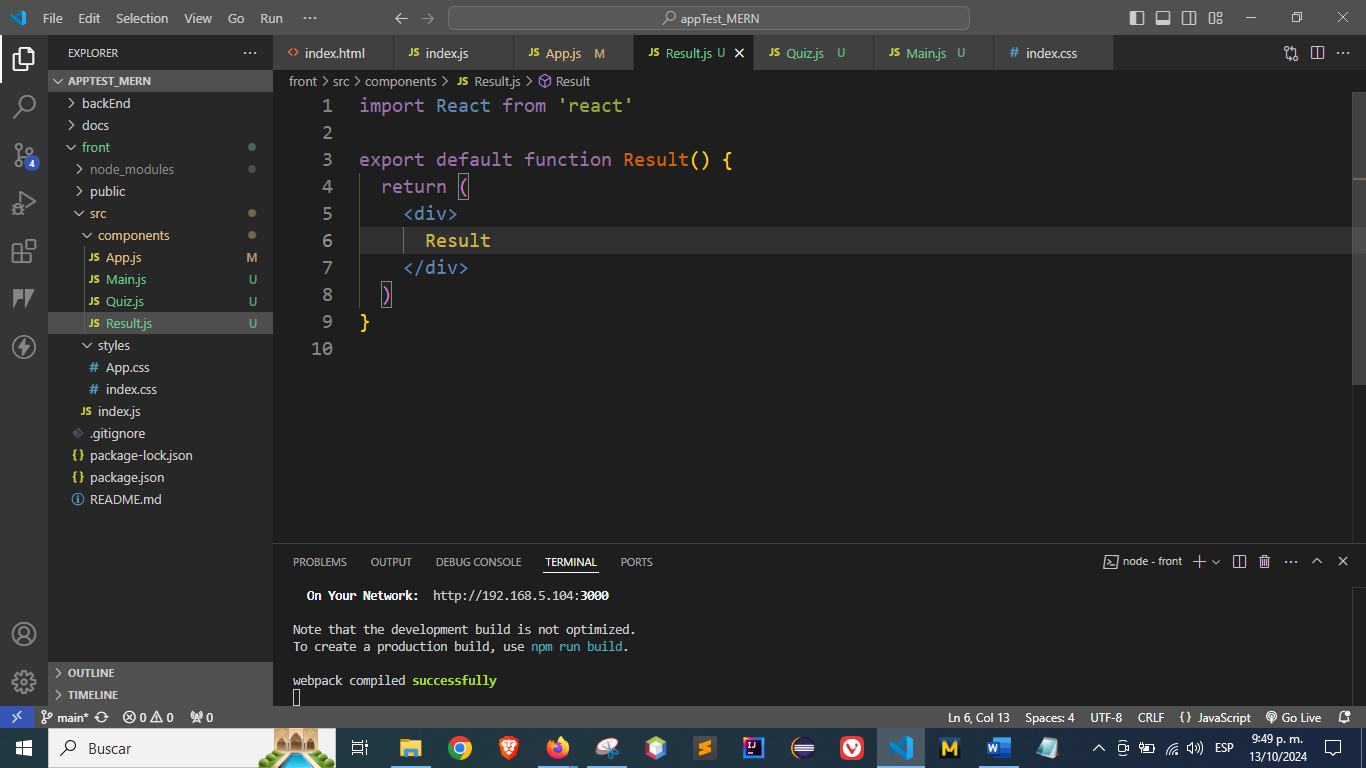


### Quiz.js





### Result



## Editando el componente Main.js

Captura de pantalla de computadora

Descripción generada automáticamente

import { Link } from 'react-router-dom'

 <div className='start'>

            <Link className='btn' to={'quiz'}></Link>

        </div>

<li>

<input type='radio'

value={true}

name='options'

id='question1'

onChange={onSelect}/>

<label className='' htmlFor='question1'>Option</label>

<div className='check checked'></div>

</li>

Texto, Escala de tiempo

Descripción generada automáticamente

Resultado en el navegador antes de dar estilos

Interfaz de usuario gráfica, Texto, Aplicación, Word

Descripción generada automáticamente

#### Main.css

\*{

margin: 0;

padding: 0;

box-sizing: border-box;

text-decoration: none;

}

body{

background-image: url("../images/fondo.JPG");

background-repeat: no-repeat;

background-size: cover;

}

.App-header {

margin: 0;

background-color: #282c34;

height: 10vh;

display: flex;

align-items: center;

justify-content: space-evenly;

font-size: calc(10px + 2vmin);

color: white;

}

nav{

width: 40%;

display: flex;

align-items: center;

justify-content: space-evenly;

}

.container-content{

width: 80%;

height: 70vh;

margin: 5%;

display: flex;

flex-direction: column;

align-items: center;

justify-content: space-evenly;

color: rgb(1, 16, 56);

font-size: 1.5rem;

}

.container-content h1{

color: brown;

font-size: 4rem;

}

.container-content h2{

color: chocolate;

font-size: 3rem;

}

.container-content input{

height: 40px;

width: 100%;

text-align: center;

font-size: 25px;

color: chocolate;

border-radius: 10px;

}

.btn {

color: #bb6b10;

height: 40px;

font-size: 35px;

background-color: rgb(243, 23, 7);

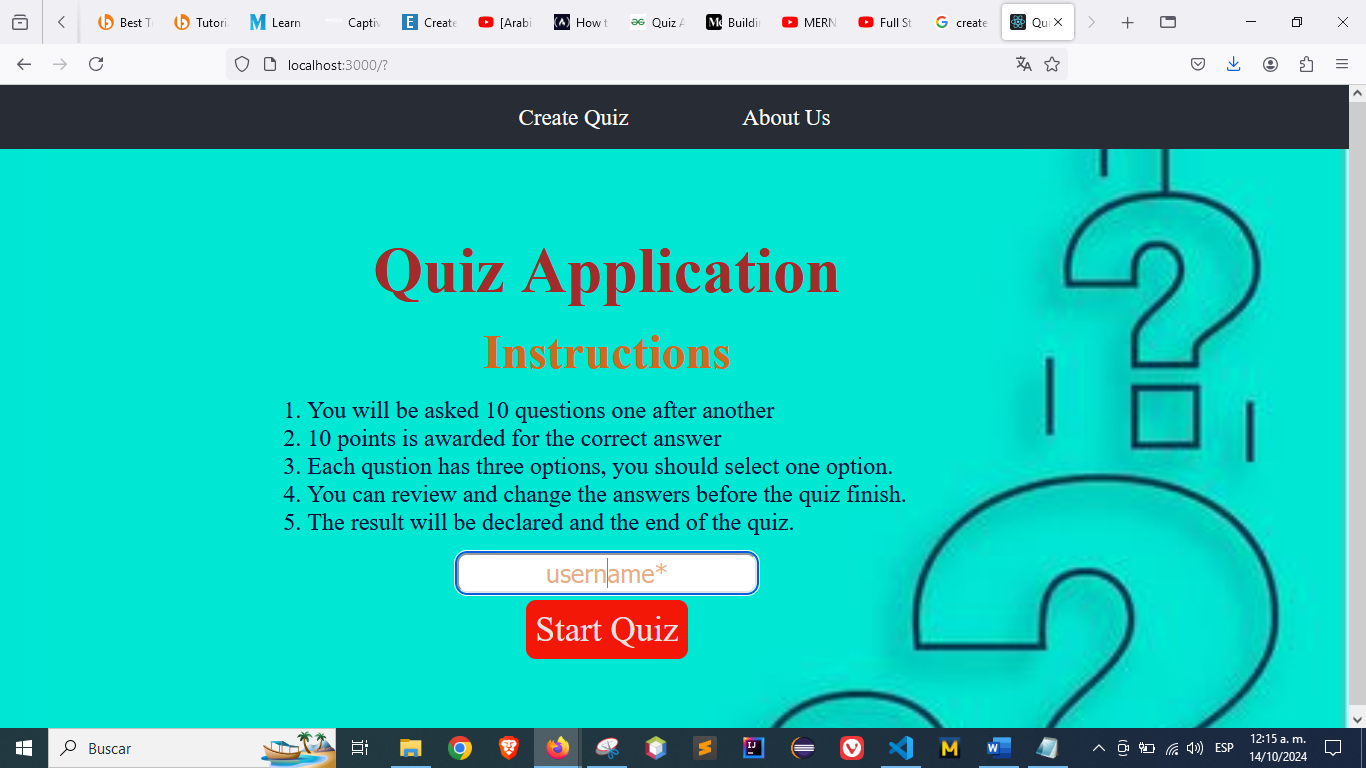
color: rgb(235, 226, 226);

padding: 10px;

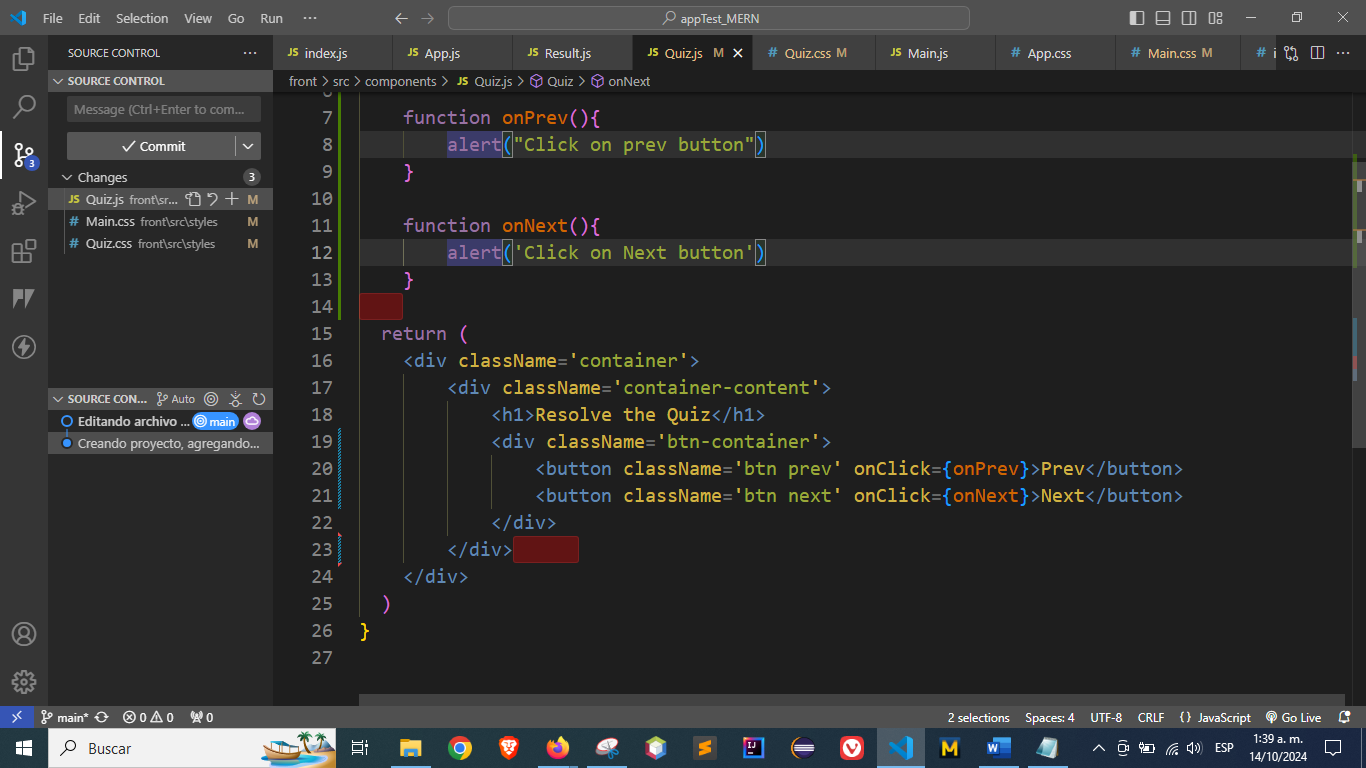
margin-top: 20px;

border-radius: 10px;

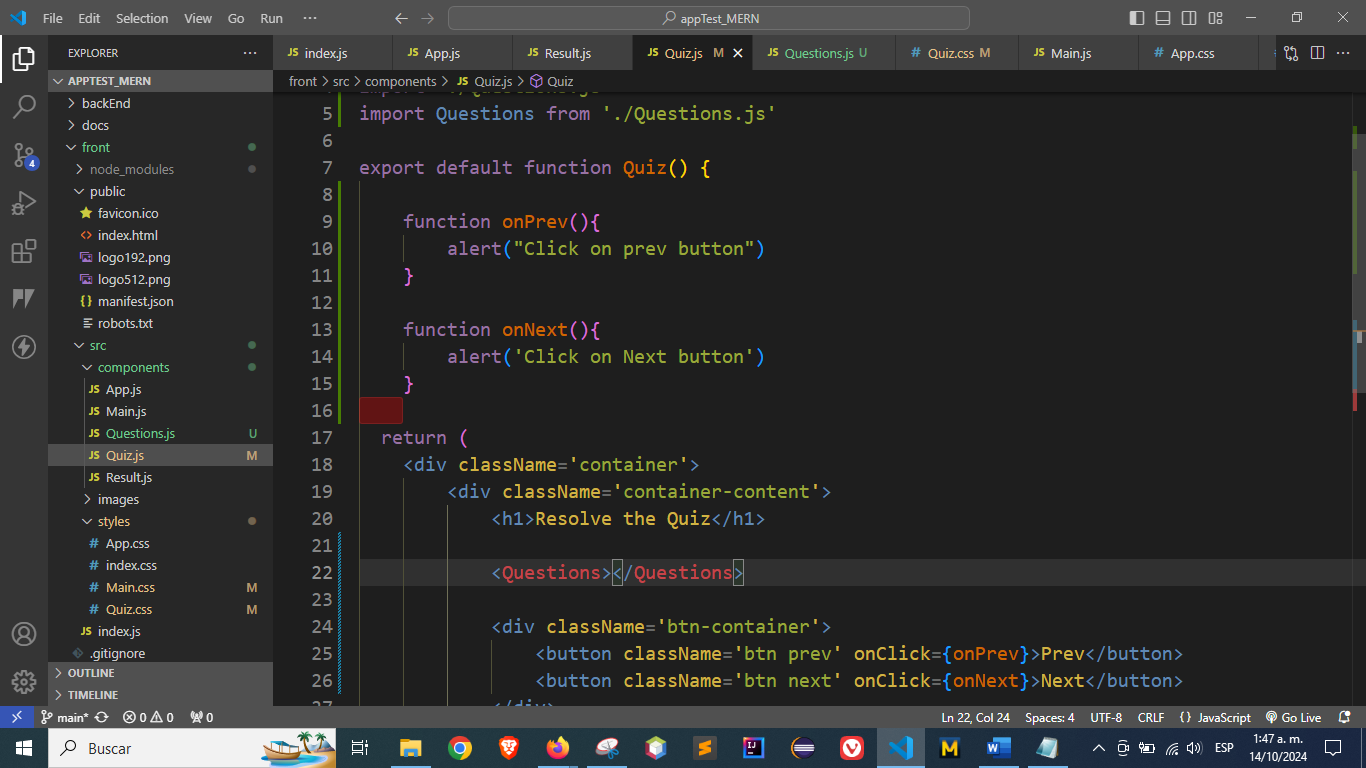
}

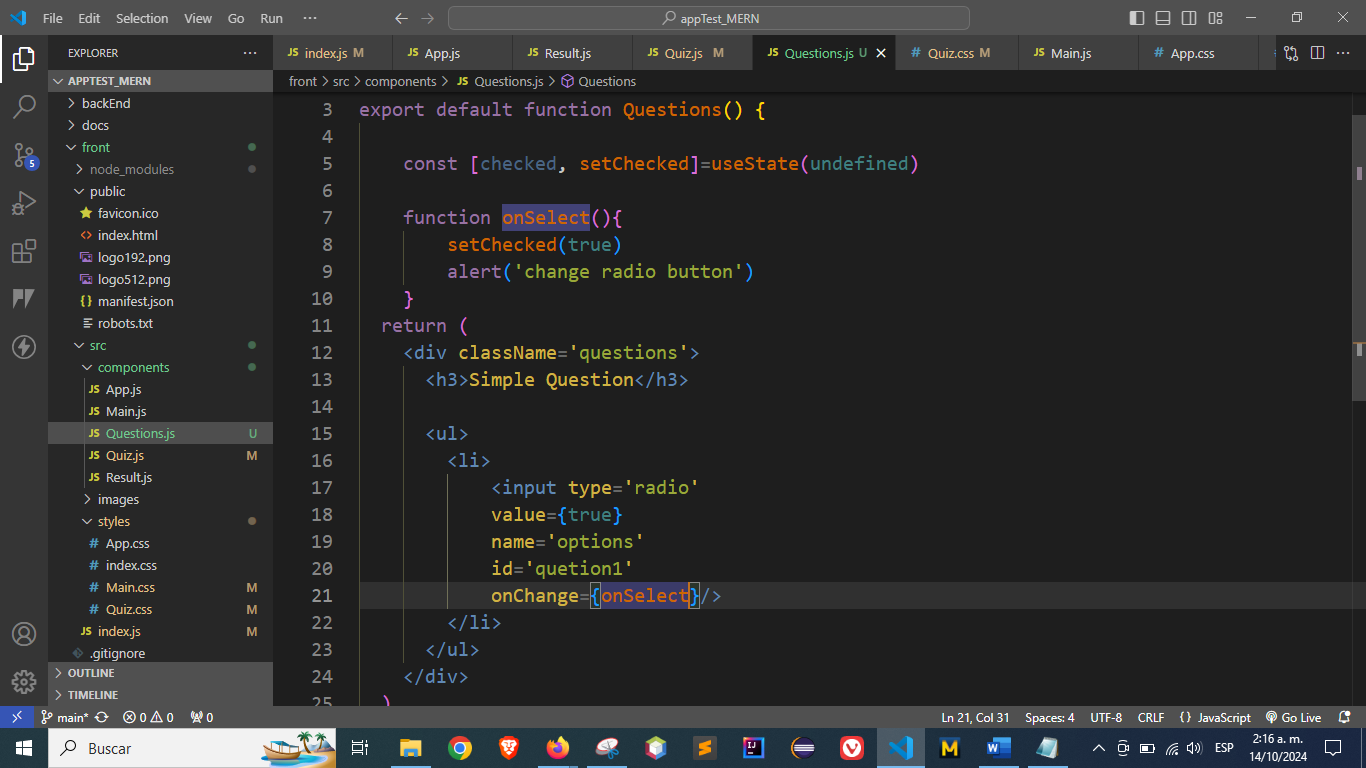


## Editando Archivo Quiz.js



## Preguntas

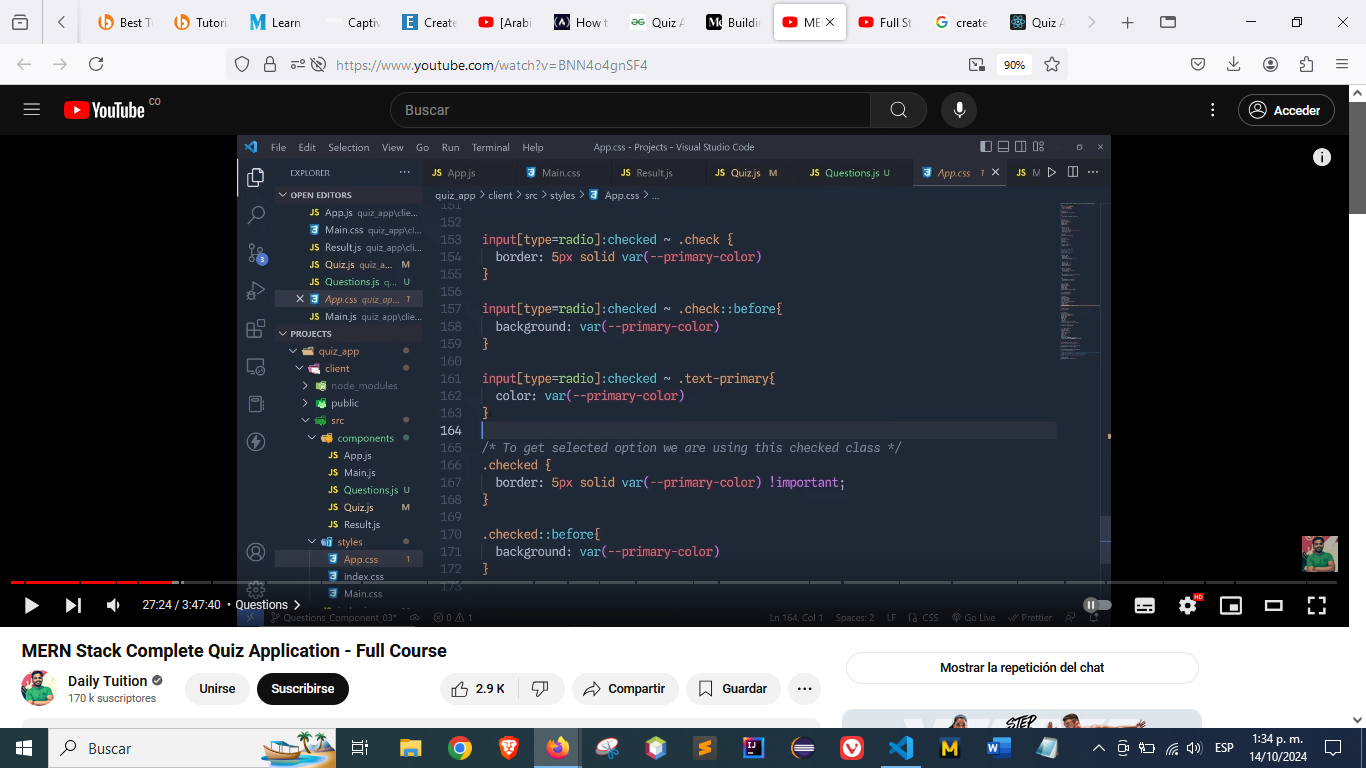




Captura de pantalla de computadora

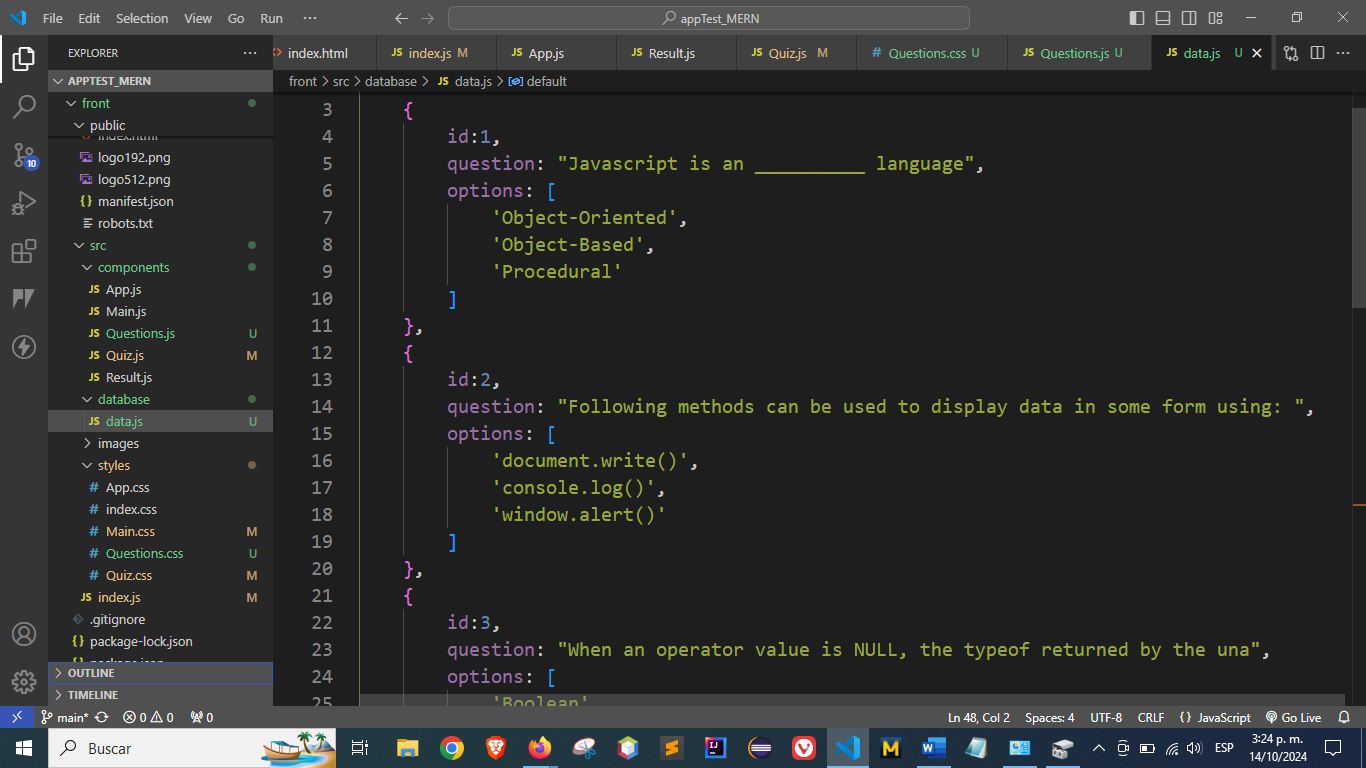
Descripción generada automáticamente

Estilos app.css para inputs



## Database

Crear la carpeta database y un archivo para manejo de los datos, data.js



export default[

{

id:1,

question: "Javascript is an \_\_\_\_\_\_\_\_\_\_ language",

options: [

'Object-Oriented',

'Object-Based',

'Procedural'

]

},

{

id:2,

question: "Following methods can be used to display data in some form using: ",

options: [

'document.write()',

'console.log()',

'window.alert()'

]

},

{

id:3,

question: "When an operator value is NULL, the typeof returned by the una",

options: [

'Boolean',

'Undefined',

'Object'

]

},

{

id:4,

question: "What does the toString() method return?",

options:[

'Return Object',

'Return String',

'Return Integer'

]

},

{

id:5,

question: "Wich function is used to serialize an object into a JSON string?",

options: [

'stringify()',

'parse()',

'convert()'

]

}

]

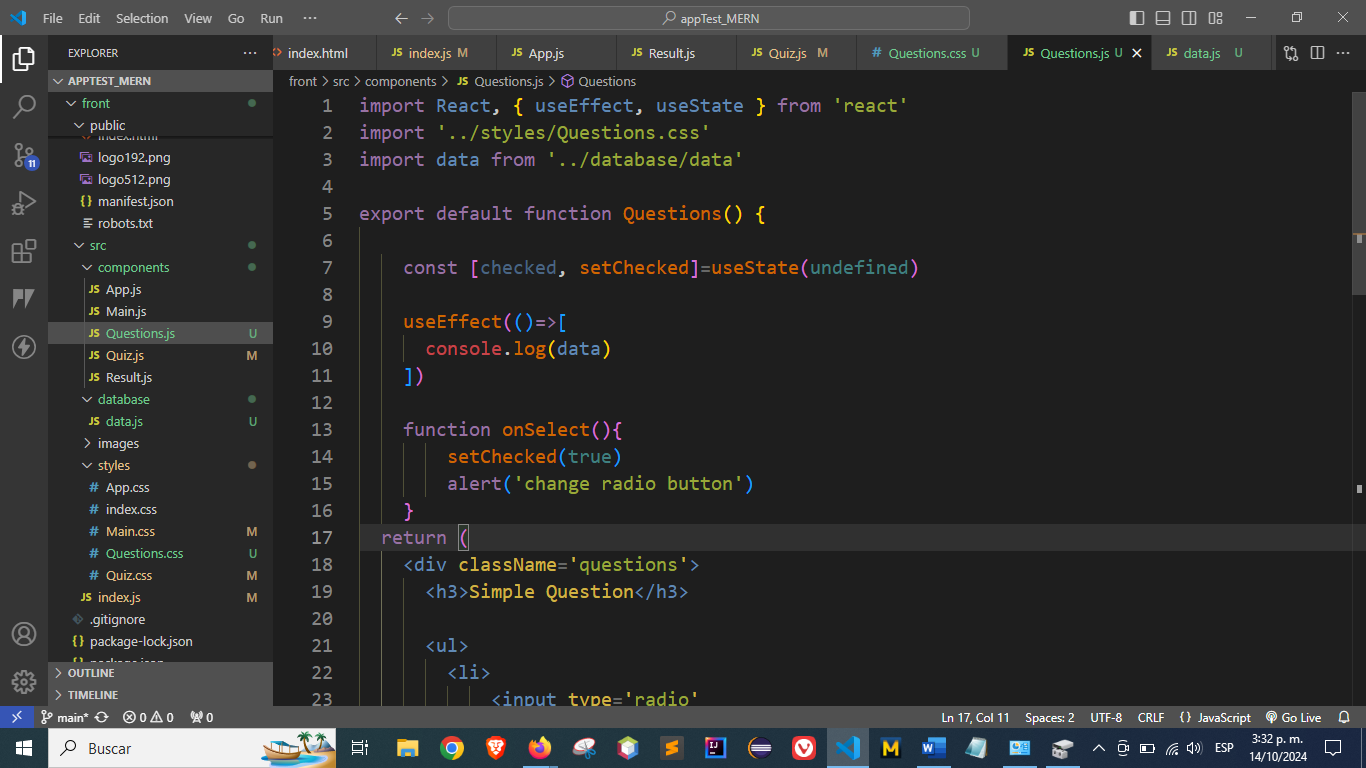
Importando al archivo Questions.js

import data from '../database/data'

useEffect(()=>[

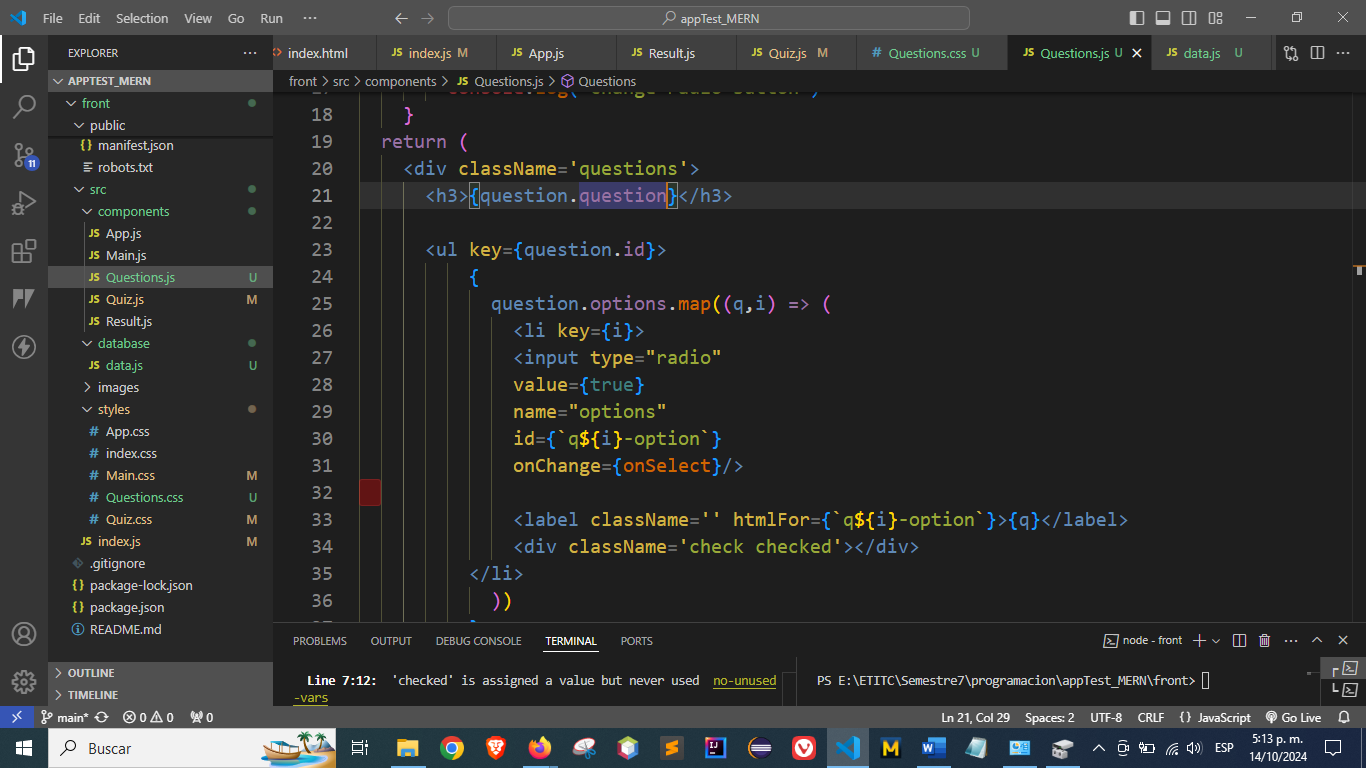
      console.log(data)

    ])

 Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Para mostrar en la página web



Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

## Editando Result.js

Captura de pantalla de computadora

Descripción generada automáticamente

import React from 'react'

import '../styles/Main.css'

import {Link} from 'react-router-dom'

export default function Result() {

function restart(){

console.log("On restart")

}

return (

<div className='container-content'>

<h1>Quiz's Result</h1>

<div className='contaniner-score'>

<span>Username </span>

<span>William Sandoval Rosas</span>

</div>

<div className='contaniner-score'>

<span>Total Quiz Points </span>

<span>50</span>

</div>

<div className='contaniner-score'>

<span>Total Questions </span>

<span>5</span>

</div>

<div className='contaniner-score'>

<span>Total Attempts: </span>

<span>3</span>

</div>

<div className='contaniner-score'>

<span>Total Earn Points </span>

<span>30</span>

</div>

<div className='contaniner-score'>

<span>Quiz Result </span>

<span>Passed</span>

</div>

<div className='start'>

<Link className='btn' to={'/'} onClick={restart}>Restart Quiz</Link>

</div>

</div>

)

}

