



Instituto Nacional de México

Instituto Tecnológico de Tijuana

Departamento de sistemas y computación

Ingeniería informática

Práctica 1:

Actividad de React

Materia:

Desarrollo de aplicaciones

Periodo: Verano de Ago-Dic 2020-2

Unidad: 2

Facilitador:

Dra. Daniela A. Sánchez Vizcarra

Alumno:

Luis Eduardo Fierro López

Grupo:

1

Fecha:

Tijuana Baja California a 26 de Octubre del 2020

Problemática

Desarrollar una página web con node js para replicar el ejemplo proporcionado en clase utilizando el lenguaje JavaScript y VS Code para manejar dicho lenguaje y cumplir con los distintos requisitos como lo son :

Componente: Banner

Props:

- texto="Desarrollo ..."

Componente: Header

Props:

- ninguno

Componente: Banner

Props:

- texto ="26/Oct/2020"

Componente: Body

Props:

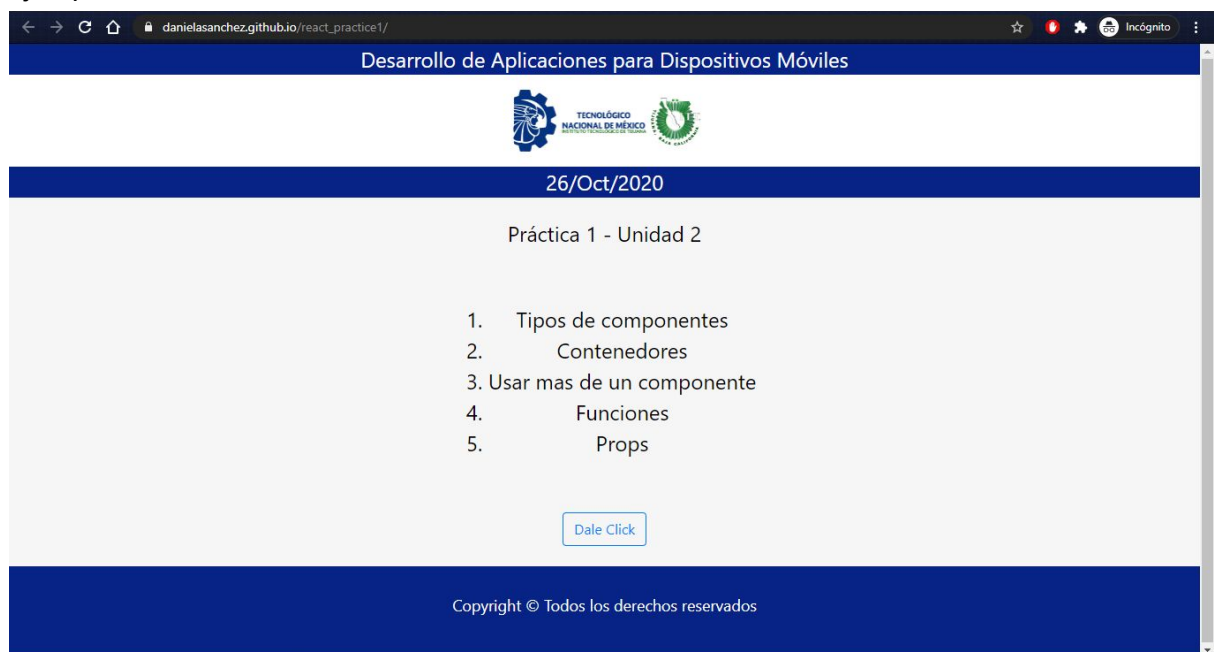
- practica="Práctica 1 - Unidad 2"
- temas={temas}

Componente: Footer

Props:

- pie={<p>Copyright © Todos los derechos reservados </p>}

Ejemplo:



Herramientas

Para la realización de esta práctica se utilizaron diversas herramientas para manejar el lenguaje JavaScript y alojar la página en un servidor web como lo es GH Pages.

- Node js
- Visual Studio Code
- Lenguaje JavaScript
- GitHub
- GitHub Pages
- Desarrollo

Desarrollo

En este apartado se presentarán los pasos que fueron necesarios para realizar esta actividad

Como requisito haber instalado Node js y posteriormente crear una carpeta en la cual se descargar los archivos con los que trabajaremos pero como no tome SS y la instalación tarda bastante omitiremos este paso .

Carpeta en el escritorio:



Carpeta del proyecto:



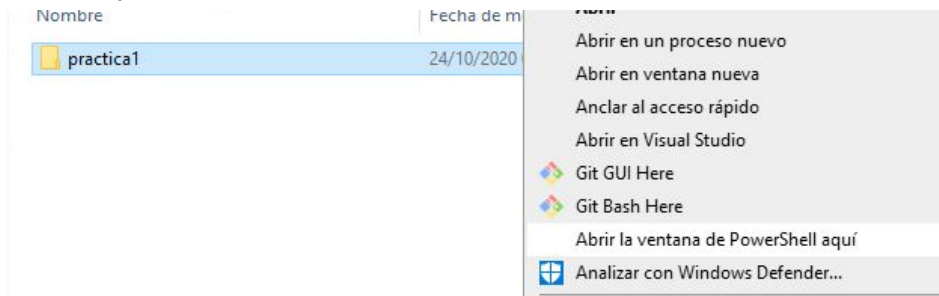
Contenido:

node_modules	22/10/2020 10:11 p. m.	Carpeta de archivos	
public	22/10/2020 08:40 p. m.	Carpeta de archivos	
src	24/10/2020 09:15 p. m.	Carpeta de archivos	
.gitignore	09/10/2020 06:29 p. m.	Documento de te...	1 KB
package.json	22/10/2020 10:11 p. m.	Archivo JSON	1 KB
package-lock.json	22/10/2020 10:11 p. m.	Archivo JSON	564 KB
README.md	22/10/2020 08:40 p. m.	Archivo MD	3 KB
yarn.lock	22/10/2020 08:42 p. m.	Archivo LOCK	462 KB

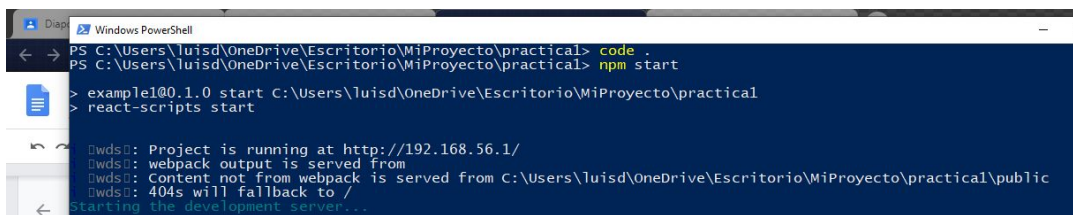
Abrir proyecto

Para abrir el proyecto en vs code y habilitar los puertos para visualizarlos necesitamos abrir una ventana de powershell e introducir un par de comandos.

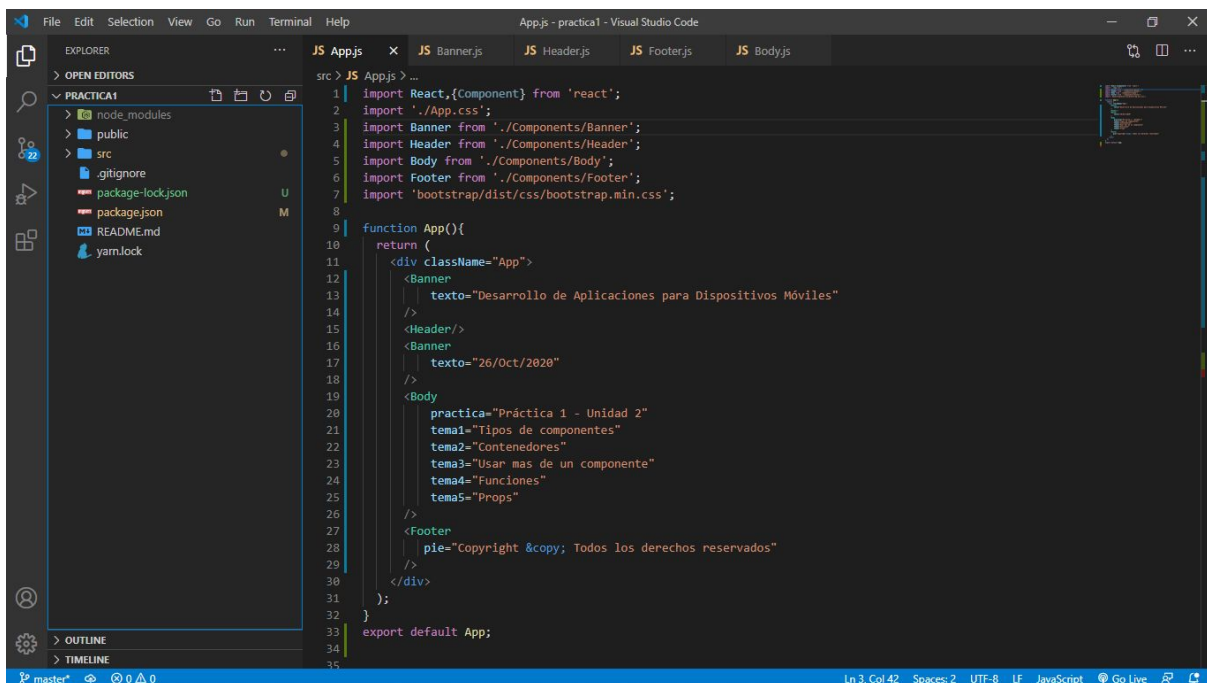
Para abrir la ventana de powershell necesitamos seleccionar la carpeta con shift + click derecho y seleccionar la opción de “Abrir la ventana de PowerShell aquí”



En la ventana de powershell agregamos el comando de “code .” para abrir la carpeta en vs code y el comando “npm start” para abrir el servidor local, a veces puede tardar una eternidad así que toca esperar.



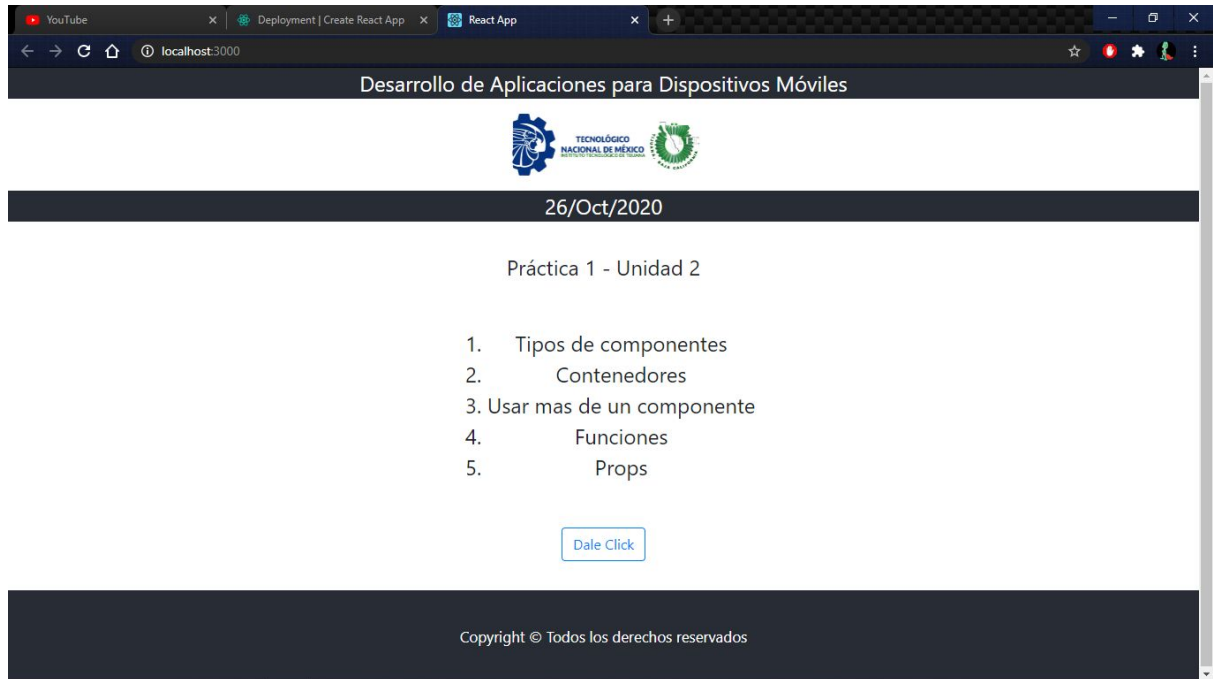
VS Code:



App

pantalla principal de la actividad finalizada de manera Local

Localhost: 3000:



Código de la App

```
8
9 | function App(){
10 |   return (
11 |     <div className="App">
12 |       <Banner
13 |         texto="Desarrollo de Aplicaciones para Dispositivos Móviles"
14 |       />
15 |       <Header/>
16 |       <Banner
17 |         texto="26/Oct/2020"
18 |       />
19 |       <Body
20 |         practica="Práctica 1 - Unidad 2"
21 |         tema1="Tipos de componentes"
22 |         tema2="Contenedores"
23 |         tema3="Usar mas de un componente"
24 |         tema4="Funciones"
25 |         tema5="Props"
26 |       />
27 |       <Footer
28 |         pie="Copyright &copy; Todos los derechos reservados"
29 |       />
30 |     </div>
31 |   );
32 | }
33 | export default App;
34
```

Componentes

Estos son los componentes utilizados para clase app.

Como se puede observar, están los cuatro principales que se piden para desarrollar la actividad los cuales son “Banner, Header, Body y Footer”.

```
import React,{Component} from 'react';
import './App.css';
import Banner from './Components/Banner';
import Header from './Components/Header';
import Body from './Components/Body';
import Footer from './Components/Footer';
import 'bootstrap/dist/css/bootstrap.min.css';
```

Props

Props utilizados en la actividad, el componente Banner se reutiliza pero cambiando las props

El componente Header no utiliza ningún props.

```
<div className="App">
  <Banner
    | texto="Desarrollo de Aplicaciones para Dispositivos Móviles"
  />
  <Header/>
  <Banner
    | texto="26/Oct/2020"
  />
  <Body
    | practica="Práctica 1 - Unidad 2"
    | tema1="Tipos de componentes"
    | tema2="Contenedores"
    | tema3="Usar mas de un componente"
    | tema4="Funciones"
    | tema5="Props"
  />
  <Footer
    | pie="Copyright &copy; Todos los derechos reservados"
  />
</div>
```

Banner

Este componente tiene un props de texto en el cual muestra el texto indicado desde la App.

```
<div className="App">
  <Banner
    | texto="Desarrollo de Aplicaciones para Dispositivos Móviles"
  />
  <Header/>
```



Desarrollo de Aplicaciones para Dispositivos Móviles

Código del componente y estilo CSS que utiliza:

JS App.js

JS Banner.js

Banner.css

src > Components > JS Banner.js > default

```

1  import React from 'react';
2  import './Banner.css';
3
4  function Banner({texto}){
5      return(
6          <div className="Banner">
7              {texto}
8          </div>
9      );
10 }
11
12 export default Banner;

```

App.js

JS Banner.js

Banner.css

JS

c > Components > Banner.css > .Banner

```

1  .Banner {
2      background-color: #282c34;
3      display: flex;
4      flex-direction: column;
5      align-items: center;
6      justify-content: center;
7      color: #fff;
8      min-height: 5vh;
9      font-size: calc(10px + 2vmin);
10 }

```

Header

El componente Header consiste en una clase la cual muestra el logo que utilizará en su espacio asignado cuando se manda a llamar.

```

/>
<Header />

```



Código del componente y Estilo CSS que utiliza.

JS App.js

JS Banner.js

JS Header.js

Header.css

JS Footer.js

JS

src > Components > JS Header.js > ...

```

1  import React, {Component} from 'react';
2  import './Header.css';
3  import logotec from '../logotec.jpg';
4
5  class Header extends Component{
6      render(){
7          return(
8              <div className="Header">
9                  <img src={logotec} className="App-logo" alt="logotec" />
10             </div>
11         );
12     }
13 }
14
15 export default Header;

```

JS App.js

JS Banner.js

JS Header.js

src > Components > Header.css > .Header

```

1  .Header {
2
3      background-color: #fff;
4
5      min-height: 15vh;
6      display: flex;
7      flex-direction: column;
8      align-items: center;
9      justify-content: center;
10     font-size: calc(5px + 2vmin);
11     color: #fff;
12 }

```

Banner

Reutilización del componente Banner pero cambiando el texto que muestra en el mismo props siendo en esta ocasión, la fecha.

```
<Banner
  |   texto="26/Oct/2020"
/>
```

26/Oct/2020

Código del componente y estilo CSS que utiliza:

JS App.js

JS Banner.js

Banner.css

src > Components > JS Banner.js > [default]

```

1  import React from 'react';
2  import './Banner.css';
3
4  function Banner({texto}){
5    return(
6      <div className="Banner">
7        {texto}
8      </div>
9    );
10 }
11
12 export default Banner;
```

JS App.js

JS Banner.js

Banner.css

c > Components > Banner.css > .Banner

```

1  .Banner {
2    background-color: #282c34;
3    display: flex;
4    flex-direction: column;
5    align-items: center;
6    justify-content: center;
7    color: #fff;
8    min-height: 5vh;
9    font-size: calc(10px + 2vmin);
10
11 }
```

Body

Cuerpo de la página, es una función la cual muestra el contenido principal de la página. Al utilizar muchos props se generalizan en la función como "props" y se mandan a llamar individualmente de la siguiente manera "props.práctica" y así con todos los demás

```
<Body
  |   practica="Práctica 1 - Unidad 2"
  |   tema1="Tipos de componentes"
  |   tema2="Contenedores"
  |   tema3="Usar mas de un componente"
  |   tema4="Funciones"
  |   tema5="Props"
/>
```


Práctica 1 - Unidad 2

1. Tipos de componentes
2. Contenedores
3. Usar mas de un componente
4. Funciones
5. Props

[Dale Click](#)

Código del componente y estilo CSS que utiliza:

```
return(  
  <div className="Body">  
    <p>  
      {props.practica}  
    </p>  
    <br/>  
    <ol>  
      <li>{props.tema1}</li>  
      <li>{props.tema2}</li>  
      <li>{props.tema3}</li>  
      <li>{props.tema4}</li>  
      <li>{props.tema5}</li>  
    </ol>  
    <br/>  
    <button type="button" class=  
  </div>  
)  
}  
  
export default Body;
```

```
1 .Body {  
2   min-height: 60vh;  
3   display: flex;  
4   flex-direction: column;  
5   align-items: center;  
6   justify-content: center;  
7   font-size: calc(10px + 2vmin);  
8 }
```

Footer

Componente que utiliza una función para colocar un pie de página.

Dicho componente contiene un props el cual indica el texto que se muestra en pantalla.

```
<Footer  
  pie="Copyright &copy; Todos los derechos reservados"  
/>  
</div>
```

Copyright © Todos los derechos reservados

Código del componente y estilo CSS utilizado:

```
src > Components > JS Footer.js > ...
1  import React from 'react';
2  import './Footer.css';
3
4  function Footer({pie}){
5    return(
6      <div className="Footer">
7        {pie}
8      </div>
9    );
10 }
11
12 export default Footer;
```

```
src > Components > Footer.css > .Footer
1  .Footer {
2    background-color: #282c34;
3    min-height: 15vh;
4    display: flex;
5    flex-direction: row;
6    align-items: center;
7    justify-content: center;
8    font-size: calc(3px + 2vmin);
9    color: white;
10 }
11 }
```

Botón Hacer clic

No sirve y no hace clic porque me falló el script pero mejor ni le muevo porque ya funciona la página.

Dale Click

Página subida a Gh-Pages

Olvidé tomar capturar pero sale siguiendo estos pasos de GitHub Pages #

<https://create-react-app.dev/docs/deployment/#github-pages>

The screenshot shows the 'Deployment' section of the Create React App documentation. The left sidebar lists various sections, with 'Deployment' and 'Deployment' (sub-section) highlighted. The main content area is titled 'GitHub Pages #' and includes a note about the required version of react-scripts. It then outlines 'Step 1: Add homepage to package.json', emphasizing its importance and providing a code example for the homepage field in package.json. The right sidebar lists other deployment options like Static Server, Other Solutions, and various cloud providers.

Project Console: <https://console.firebase.google.com/project/example-app-fd690/overview>
Hosting URL: <https://example-app-fd690.firebaseio.com>

For more information see [Firebase Hosting](#).

GitHub Pages

Note: this feature is available with `react-scripts@0.2.0` and higher.

Step 1: Add `homepage` to `package.json`

The step below is important!

If you skip it, your app will not deploy correctly.

Open your `package.json` and add a `homepage` field for your project:

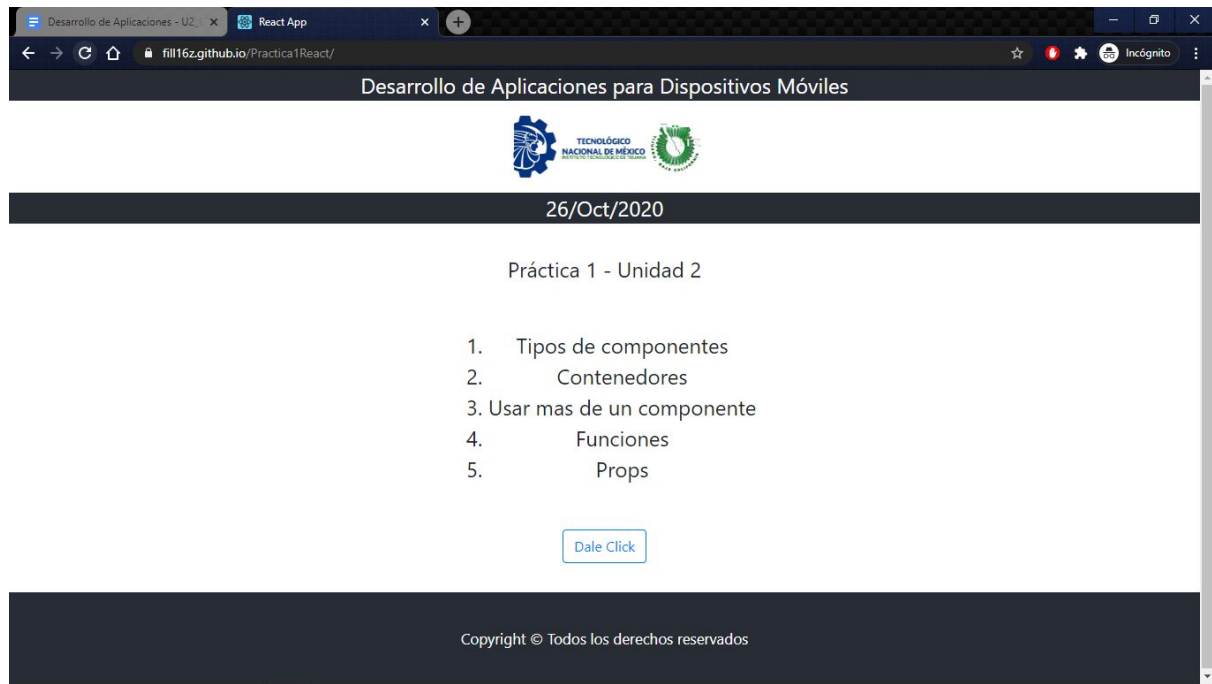
```
"homepage": "https://myusername.github.io/my-app",
```

or for a GitHub user page:

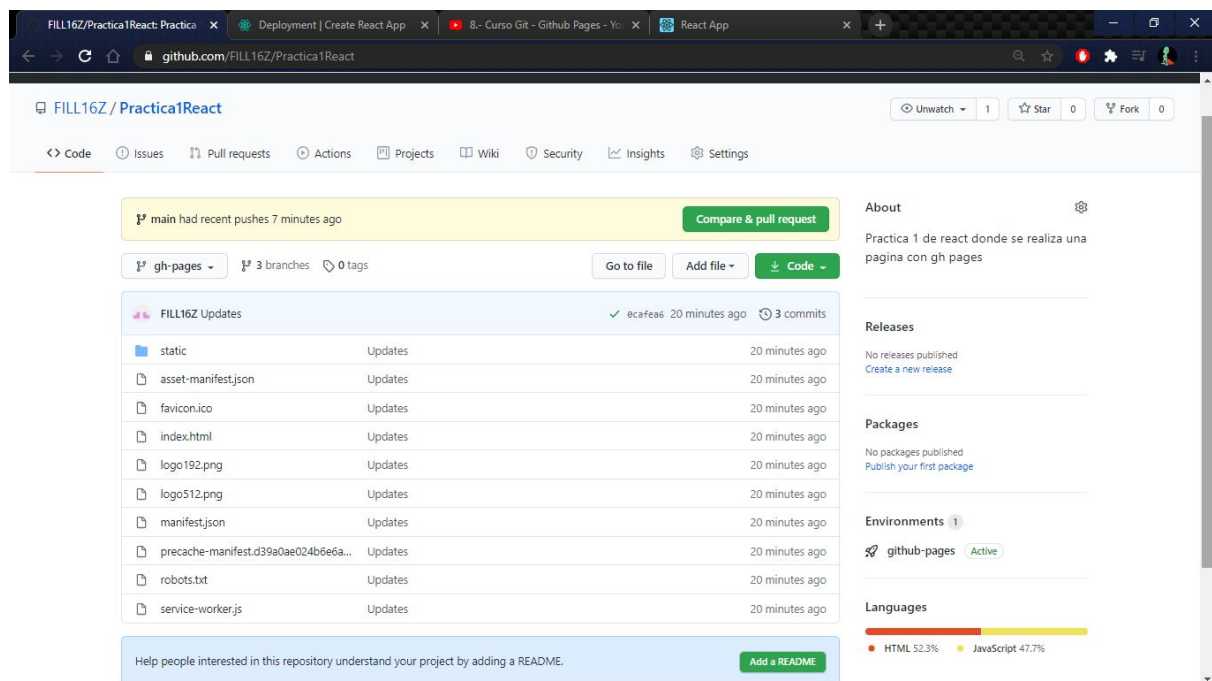
Static Server
Other Solutions
Serving Apps with Client-Side Routing
Building for Relative Paths
Serving the Same Build from Different Paths
Customizing Environment Variables for Arbitrary Build Environments
AWS Amplify
Azure
Firebase
GitHub Pages
Step 1: Add `homepage` to `package.json`
Step 2: Install `gh-pages` and add `deploy` to `scripts` in `package.json`
Step 3: Deploy the site by running `npm run deploy`
Step 4: For a project page,

Resultados

Página de GitHub funcionando



Repositorio Chipeado que aparentemente no debe de servir pero funciona en su totalidad



Links (Hipervínculos)

Página de la aplicación

<https://fill16z.github.io/Practica1React/>

Directorio normal en la rama main

<https://github.com/FILL16Z/Practica1React/tree/main>

Directorio chipeado en la rama gh-pages

<https://github.com/FILL16Z/Practica1React>

Directorio de GitHub principal con las demás tareas y PDF's

<https://github.com/FILL16Z/Desarrollo-de-Aplicaciones>

Ahora si a mimir

