

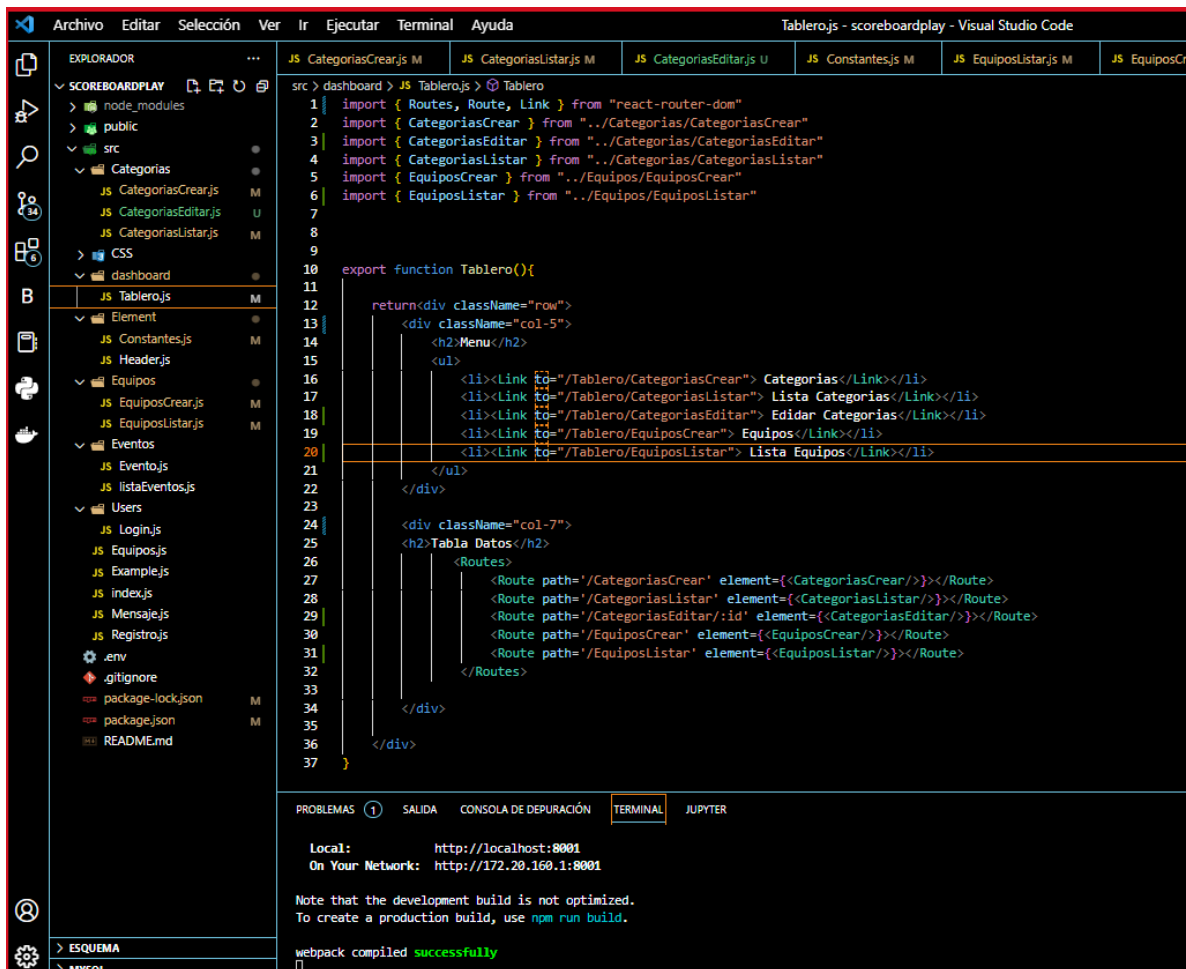


### SPRINT 3: Desarrollo del Frontend

Identificación Proyecto	
Nombre Proyecto:	ScoreBoardPlay
Número Equipo:	Grupo 1
Integrantes del equipo	
Rol (Líder-Desarrollador – Cliente)	Nombre
Scrum master	Luisa Juliana Carrillo Cacua
Product owner	Brayan Duvan Bernal Sarmiento
development team	Lady Viviana Fandiño Angel
development team	Daniel Francisco Basto Arenas
development team	José Luis Castillo Camacho

### Evidencia construcción del Frontend

Como evidencia de la construcción del Frontend, se debe presentar capturas de pantalla donde se visualice el proceso de construcción del Frontend.





Archivo

Editar

Selección

Ver

Ir

Ejecutar

Terminal

Ayuda

EquiposCrearjs - scoreboardplay - Visual Studio Code

JS CategoríasCrearjs M

JS CategoríasListarjs M

JS CategoríasEditarjs U

JS Constantesjs M

JS EquiposListarjs M

JS EquiposCrearjs M X

JS Tablerojs M

EXPLORADOR

SCOREBOARDPLAY

node\_modules

public

src

Categorías

JS CategoríasCrearjs M

JS CategoríasEditarjs U

JS CategoríasListarjs M

CSS

dashboard

JS Tablerojs M

Element

JS Constantesjs M

JS Headerjs

Equipos

JS EquiposCrearjs M

JS EquiposListarjs M

Eventos

JS Eventojs

JS listaEventosjs

Users

JS Loginjs

JS Equiposjs

JS Examplejs

JS indexjs

JS Mensajejs

JS Registrojs

env

gignore

package-lock.json M

package.json M

README.md

ESQUEMA

```

src > Equipos > JS EquiposCrearjs > EquiposCrear
1 import {url} from '../Element/Constantes'
2 import React,{useRef} from 'react'
3
4 export function EquiposCrear(){
5   const refNombre=useRef(null)
6   const HandleSubmit=(ev)=>{
7     ev.preventDefault()
8     console.log('enviando datos')
9     const requestOptions={
10       method:'POST',
11       headers: {'Content-Type':'application/json'},
12       body:JSON.stringify({nombre:refNombre.current.value})
13     }
14     fetch(url+`/Equipos/`,requestOptions)
15     .then(response => response.json())
16     .then(data => console.log(data))
17     .catch(error => console.log('el error'+error))
18   }
19
20   return <div>
21     <h2>Crear Equipo</h2>
22     <div className="container-fluid">
23       <div className="row">
24         <div className="col-md-4 offset-4">
25           <form onSubmit=HandleSubmit>
26             <div className="mb-3">
27               <label for="exampleInputname" className="form-label">Nombre Equipo</label>
28               <input type="text" className="form-control" id="exampleInputname" placeholder="Ingrese Equipo" aria-describedby="nombre Categoría" ref={refNombre} />
29             </div>
30           </form>
31           <button type="submit" className="btn btn-primary">Submit</button>
32         </div>
33       </div>
34     </div>
35   </div>
36 </div>
37
38 </div>

```

PROBLEMAS 1 SALIDA CONSOLA DE DEPURACIÓN TERMINAL JUPYTER

Local: http://localhost:8001

On Your Network: http://172.20.160.1:8001

Note that the development build is not optimized.

To create a production build, use `npm run build`.

webpack compiled **successfully**



Archivo

Editar

Selección

Ver

Ir

Ejecutar

Terminal

Ayuda

Presentando...

EXPLORADOR

SCOREBOARDPLAY

node\_modules

public

src

Categorías

JS CategoríasCrear.js M

JS CategoríasEditar.js U

JS CategoríasListar.js M

CSS

dashboard

JS Tablero.js M

Element

JS Constantes.js M

JS Header.js

Equipos

JS EquiposCrear.js M

JS EquiposListar.js M

Eventos

JS Evento.js

JS listaEventos.js

Users

JS Login.js

JS Equipos.js

JS Example.js

JS index.js

JS Mensaje.js

JS Registro.js

.env

.gitignore

package-lock.json M

package.json M

README.md

JS CategoríasCrear.js M

JS CategoríasListar.js M

JS CategoríasEditar.js U

src > Eventos > JS Eventos.js > ...

1

2

3 export function Evento(props){

4

5 return <div className="row">

6 <h1>2022/11/21 Equipo1 vs Equipo2</h1>

7 <div className="col"><h5>Equipo1</h5></div>

8 <div className="col"><h4>4</h4></div>

9 <div className="col"><h5>-</h5></div>

10 <div className="col"><h5>Equipo2</h5></div>

11

12 </div>

13

14 }

PROBLEMAS 1

SALIDA

CONSOLA DE DEPURACIÓN

TERMINAL

JUPYTER

Local: http://localhost:8001

On Your Network: http://172.20.160.1:8001

Note that the development build is not optimized.

To create a production build, use npm run build.

webpack compiled successfully



Visual Studio Code interface showing the development of a frontend application. The Explorer panel on the left displays the project structure, including the 'src' directory with files like 'Categorias.js', 'CategoriasListar.js', and 'Equipos.js'. The main editor displays the code for 'CategoriasListar.js', which includes a React component for listing categories. The code uses the 'fetch' API to retrieve data from a REST API and displays it in a table. The Terminal panel at the bottom shows the output of the development build, indicating that the application is running successfully on localhost:8001.

```

src > Categorias > JS CategoriasListar.js > CategoriasListar > cargar
1 import {url} from '../Element/Constantes'
2 import React, {useState,useEffect} from 'react'
3 import { Link } from 'react-router-dom'
4
5 export function CategoriasListar(){
6   const [categorias, setCategorias]=useState([])
7
8   function cargar(){
9     const requestOptions={
10       method:'GET',
11       headers:{'Content-Type':'application/json'}
12     }
13
14     fetch(url+"/categorias/",requestOptions)
15       .then(response => response.json())
16       .then(data => {console.log(data)
17         setCategories(data)
18       })
19       .catch(error => console.log('el error'+error))
20   }
21
22   useEffect(()=>{cargar(),[]})
23
24   return <div>
25     <div>
26       <Link to="/Tablero/CategoriasCrear" className='btn btn-outline-success'>Crear Categoria</Link>
27     </div>
28     <table className='table'>
29       <thead>
30         <tr> <th>Lista Categorias</th> </tr>
31       </thead>
32       <tbody>
33         {
34           categorias.map(dato=>{
35             <tr>
36               <td>
37                 {dato.nombre}
38               </td>

```

Local: http://localhost:8001  
On Your Network: http://172.20.160.1:8001

Note that the development build is not optimized.  
To create a production build, use `npm run build`.

webpack compiled **successfully**

## Evidencias del Frontend en funcionamiento

Como evidencia del funcionamiento del Frontend, se debe presentar capturas de pantalla donde se visualice el consumo de recursos del API REST, desde el Frontend.



The screenshot displays a web application interface and its underlying code. On the left, a dark blue sidebar contains a 'Menu' with links to 'Categorías', 'Lista Categorías', 'Editar', 'Equipo', and 'Listado'. The main content area has a header 'Bienvenidos a la copa mundo' and a 'Tablero' button. Below the header, there's a 'Tabla Datos' section with a 'Crear Categorías' button. A modal titled 'Categoria Guardada' with a green checkmark icon and the text 'Categoria almacenada con éxito' is shown, with an 'OK' button. The background shows a code editor with HTML and CSS for a table and a console log showing an array of category objects.

**HTML Code (Table Structure):**

```
<div class="col-5"></div>
<div class="col-7">
  <h2>Tabla Datos</h2>
</div>
</div>
```

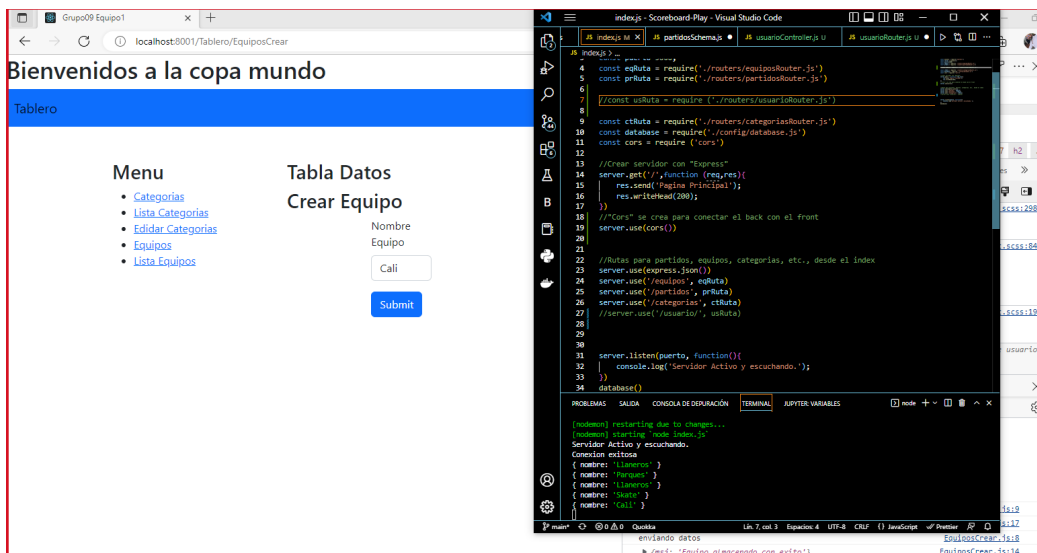
**CSS Code (Table Styling):**

```
.h2, h2 {
  font-size: calc(1.325rem + .9vw);
}
.h1, .h2, .h3, .h4, .h5, .h6, h1, h2, h3, h4, h5, h6 {
  margin-top: 0;
  margin-bottom: .5rem;
  font-weight: 500;
  line-height: 1.2;
}
*, ::after, ::before {
  box-sizing: border-box;
}
.h2 {
  display: block;
}
```

**Console Log (Array of Objects):**

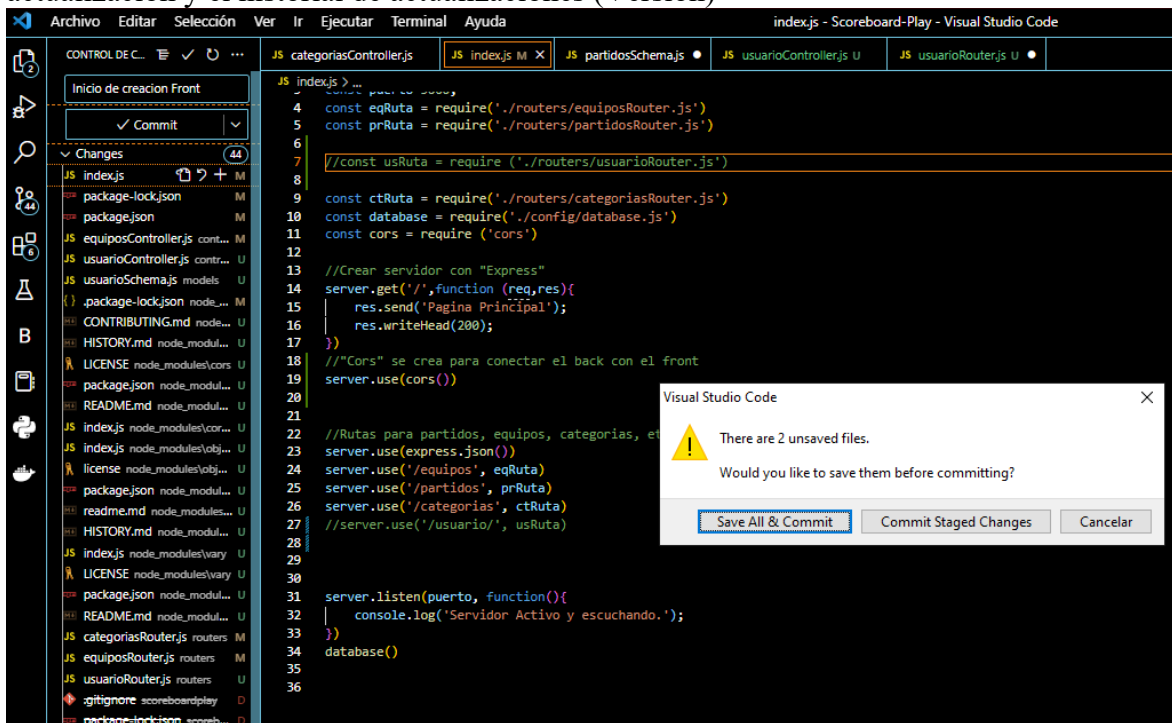
```
[
  {
    _id: '63742d1f5ef4a3738f8a94ae', nombre: 'Aguilas Doradas', __v: 0
  },
  {
    _id: '63742d3e5ef4a3738f8a94ab', nombre: 'Deportivo Pasto', __v: 0
  },
  {
    _id: '63742d4e5ef4a3738f8a94ac', nombre: 'Medellin', __v: 0
  },
  {
    _id: '6385f11d532d38a95f82a87f', nombre: 'Cucuta', __v: 0
  },
  {
    _id: '6386a49003f92a4e75352d08', nombre: 'Cucuta', __v: 0
  },
  {
    _id: '6386a756b56b846c156e0b322', nombre: 'Patriotas', __v: 0
  },
  {
    _id: '6386a8b7fd7138aa99f1cccd', nombre: 'Jaguaires', __v: 0
  },
  {
    _id: '6386914686f7457959f82808', nombre: 'Llaneros', __v: 0
  }
]
```

[illegible]



## Evidencia GitLab o GitHub

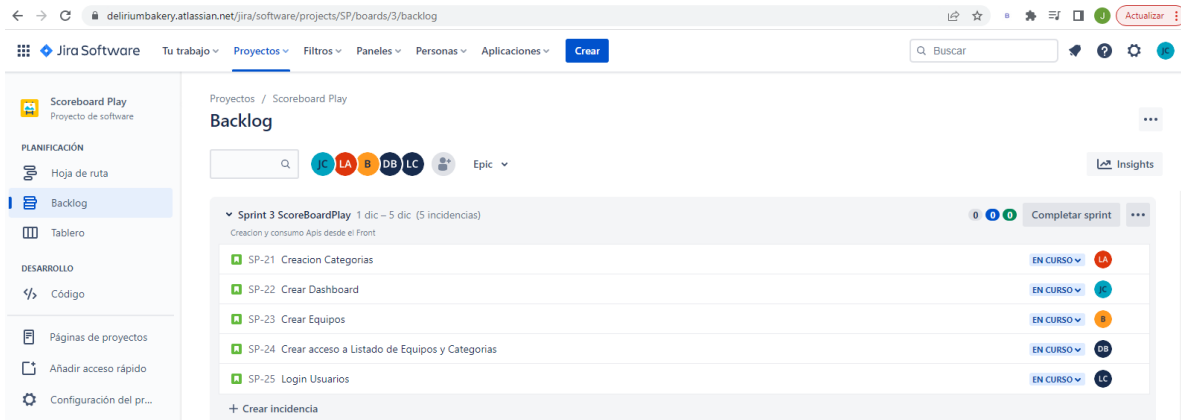
Evidencia de la realización de alguna actualización (commit), donde se visualice la actualización y el historial de actualizaciones (Versión)





## Evidencia JIRA (Seguimiento del proyecto)

Como evidencia del seguimiento del proyecto con la metodología ágil SCRUM, utilizando el software JIRA, se debe presentar capturas de pantalla donde se visualice la ejecución de los Sprints con las historias de usuario relacionadas con el desarrollo del Frontend.



## Evidencias de las Reuniones de Equipo

Como evidencia de las reuniones que efectúa el equipo del proyecto, presentar capturas de pantalla de las reuniones efectuadas y si lo consideran pertinente algunas actas de las reuniones.

