

Ubicaciones

2

Nombre: Testicle Monster Dimension Tipo: Dimension Dimension: Testicle Monster Dimension	Nombre: Signus 5 Expanse Tipo: unknown Dimension: Cromulon Dimension	Nombre: Earth (C-500A) Tipo: Planet Dimension: Dimension C-500A	Nombre: Rick's Battery Microverse Tipo: Microverse Dimension: Replacement Dimension	Nombre: The Menagerie Tipo: Menagerie Dimension: unknown
Nombre: Earth (K-83) Tipo: Planet Dimension: Dimension K-83	Nombre: Hideout Planet Tipo: Planet Dimension: unknown	Nombre: Unity's Planet Tipo: Planet Dimension: Replacement Dimension	Nombre: Dorian 5 Tipo: Planet Dimension: Replacement Dimension	Nombre: Earth (Unknown dimension) Tipo: Planet Dimension: unknown
Nombre: Earth (J1977) Tipo: Planet Dimension: Dimension J1977	Nombre: Roy: A Life Well Lived Tipo: Game Dimension: Replacement Dimension	Nombre: Eric Stoltz Mask Earth Tipo: Planet Dimension: Eric Stoltz Mask Dimension	Nombre: Earth (Evil Rick's Target Dimension) Tipo: Planet Dimension: Evil Rick's Target Dimension	Nombre: Planet Squanch Tipo: Planet Dimension: Replacement Dimension



Proyecto final Backend II

Nombre: Jonathan De La O González

Ubicaciones

Nombre: Testicle Monster Dimension Tipo: Dimension Dimension: Testicle Monster Dimension	Nombre: Signus 5 Expanse Tipo: unknown Dimension: Cromulon Dimension	Nombre: Earth (C-500A) Tipo: Planet Dimension: Dimension C-500A	Nombre: Rick's Battery Microverse Tipo: Microverse Dimension: Replacement Dimension	Nombre: The Menagerie Tipo: Menagerie Dimension: unknown
Nombre: Earth (K-83) Tipo: Planet Dimension: Dimension K-83	Nombre: Hideout Planet Tipo: Planet Dimension: unknown	Nombre: Unity's Planet Tipo: Planet Dimension: Replacement Dimension	Nombre: Dorian 5 Tipo: Planet Dimension: Replacement Dimension	Nombre: Earth (Unknown dimension) Tipo: Planet Dimension: unknown
Nombre: Earth (J19z7) Tipo: Planet Dimension: Dimension J19z7	Nombre: Roy: A Life Well Lived Tipo: Game Dimension: Replacement Dimension	Nombre: Eric Stoltz Mask Earth Tipo: Planet Dimension: Eric Stoltz Mask Dimension	Nombre: Earth (Evil Rick's Target Dimension) Tipo: Planet Dimension: Evil Rick's Target Dimension	Nombre: Planet Squanch Tipo: Planet Dimension: Replacement Dimension

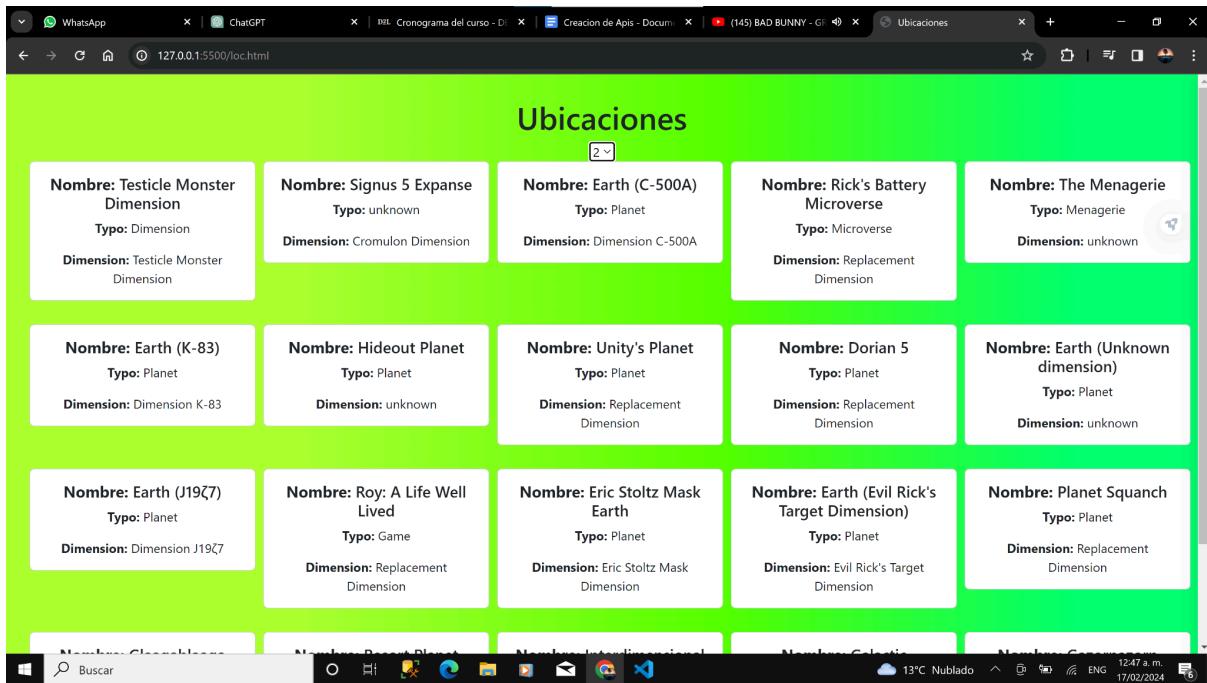
Matricula: 21030432

Modalidad: Ejecutiva

Asignatura: Desarrollo de Software Backend II

Fecha: 16/02/2024

Introducción



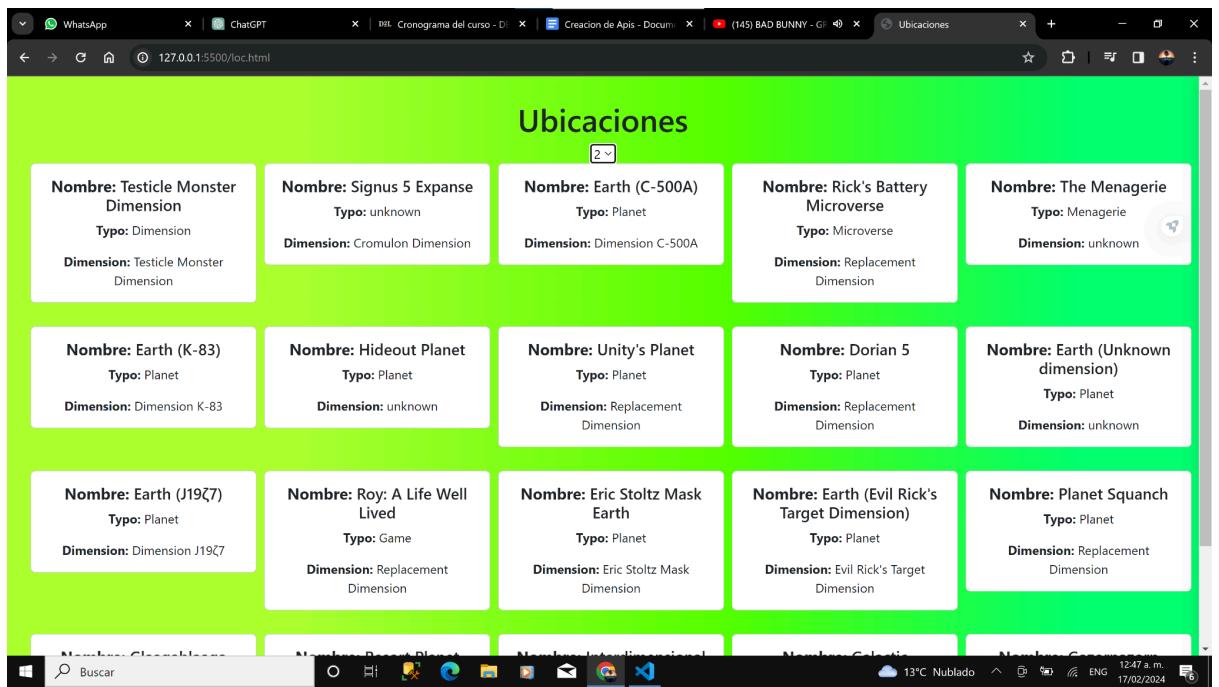
En este conjunto de códigos HTML y JavaScript, se ha creado una interfaz web para interactuar con la API de Rick and Morty. Cada página, dedicada a personajes, ubicaciones y episodios, utiliza JavaScript para realizar solicitudes a la API y mostrar dinámicamente la información en tarjetas dentro de la página. Además, se han aplicado estilos CSS para mejorar la apariencia visual de la interfaz.

Explicación 1 index.html

```

index.html X loc.html epi.html epis.js rick.js loc.js # style.css
<index.html> <html> <body> <nav.navbar.navbar-expand-lg.bg-body-tertiary> <div.container-fluid> <div.navbarNav.collapse.navbar-collapse> <ul.navbar-nav>
1   <!DOCTYPE html>
2   <html lang="en">
3     <head>
4       <meta charset="UTF-8">
5       <meta name="viewport" content="width=device-width, initial-scale=1.0">
6       <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-T3c6CoIi6uLrA9TneNNeo7RxnatzjcDSCmGIMxxSR1GAsx" type="text/css">
7       <title>Rick y Morty API</title>
8     </head>
9     <body>
10       <nav class="navbar navbar-expand-lg bg-body-tertiary">
11         <div class="container-fluid">
12           <h1 class="navbar-brand">Rick y Morty API</h1>
13           <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label="T...
14             <span class="navbar-toggler-icon"></span>
15           </button>
16           <div class="collapse navbar-collapse" id="navbarNav">
17             <ul class="navbar-nav">
18               <li class="nav-item">
19                 <a class="nav-link active" aria-current="page" href="index.html">Personajes</a>
20               </li>
21               <li class="nav-item">
22                 <a class="nav-link" href="epi.html">Episodios</a>
23               </li>
24               <li class="nav-item">
25                 <a class="nav-link" href="loc.html">Ubicaciones</a>
26               </li>
27             </ul>
28           </div>
29         </div>
30       </nav>
31       <h2>PAGINA</h2>
32       <select name="pagina" id="pagina" onchange="apiRick(this.value)">
33         <script>
34           for (let i = 1; i <= 42; i++){
35             document.write(`<option value="${i}">${i}</option>`);
36         }
37       </script>

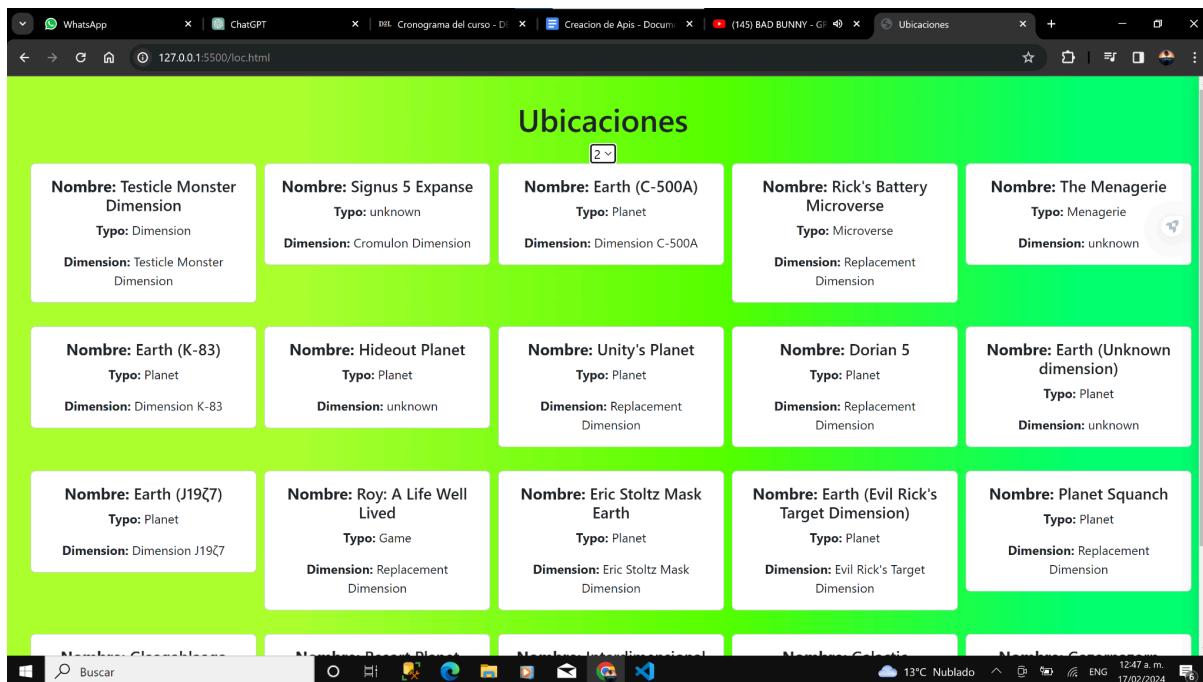
```



```
index.html x loc.html epi.html epis.js rick.js loc.js # style.css
index.html > body > nav.navbar.navbar-expand-lg.bg-body-tertiary > div.container-fluid > div#navbarNav.collapse.navbar-collapse > ul.navbar-nav
30   </div>
31   </nav>
32   <h2>PAGINA
33   <select name="pagina" id="pagina" onchange="apiRick(this.value)">
34     <script>
35       for (let i = 1; i <= 42; i++) {
36         document.write('<option value="' + i + '">' + i + '</option>');
37       }
38     </script>
39   </select></h2>
40
41   <div id="resultado">
42
43   </div>
44
45   <script src="rick.js"></script>
46
47
48
49
50 </body>
51 </html>
```

Esta primera parte del código es una estructura básica de una página web que utiliza Bootstrap y se conecta a la API de Rick y Morty para mostrar información dinámica. Aquí hay una explicación detallada del código:

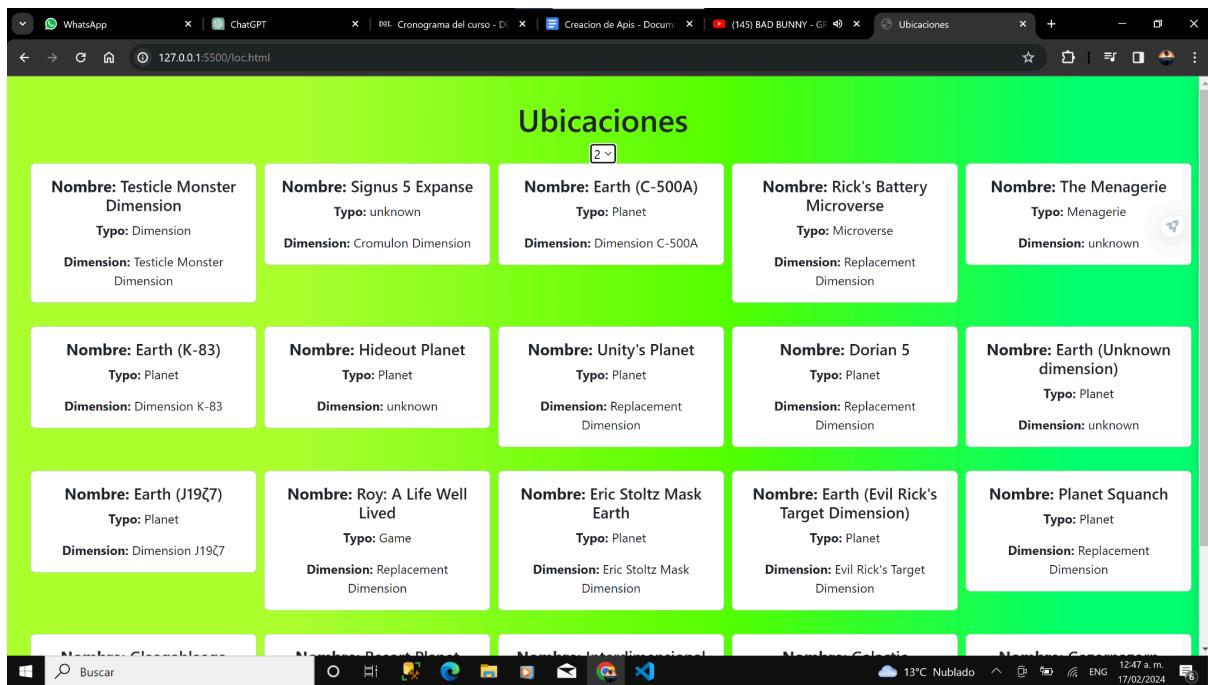
1. **<!DOCTYPE html>**: Declaración que indica que el documento sigue la especificación HTML5.
2. **<html lang="en">**: Elemento raíz del documento HTML con el atributo "lang" que establece el idioma en inglés.
3. **<head>**: Contiene metadatos y enlaces a recursos externos.
 - **<meta charset="UTF-8">**: Define la codificación de caracteres como UTF-8.



- **<meta name="viewport" content="width=device-width, initial-scale=1.0">**: Configura la vista para dispositivos móviles.
- Enlaces a hojas de estilo:
- **<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-">**: Enlace a la hoja de estilo de Bootstrap para el diseño y los estilos.
- **<link rel="stylesheet" href="style.css">**: Enlace a una hoja de estilo local llamada "style.css".
- **<title>Rick y Morty API</title>**: Define el título de la página que aparecerá en la pestaña del navegador.

4. <body>: Contiene el contenido principal de la página.

- Barra de navegación (`<nav>`):
- Se utiliza el componente de navegación de Bootstrap.
- Enlaces a "index.html" (Personajes), "epi.html" (Episodios), y "loc.html" (Ubicaciones).
- Sección para seleccionar la página (`<h2>` y `<select>`):
- Muestra el título "PAGINA".
- Un menú desplegable (`<select>`) con opciones generadas dinámicamente mediante JavaScript para las páginas del 1 al 42.
- Contenedor para los resultados (`<div id="resultado">`): Se utilizará para mostrar la información de la API después de seleccionar una página.
- Se incluye el script "rick.js" al final del body (`<script src="rick.js"></script>`): Contiene funciones JavaScript para interactuar con la API y manipular la página dinámicamente.
- Un script dentro del `<select>`:



- Utiliza un bucle for para generar opciones del 1 al 42 dinámicamente y las agrega al menú desplegable.

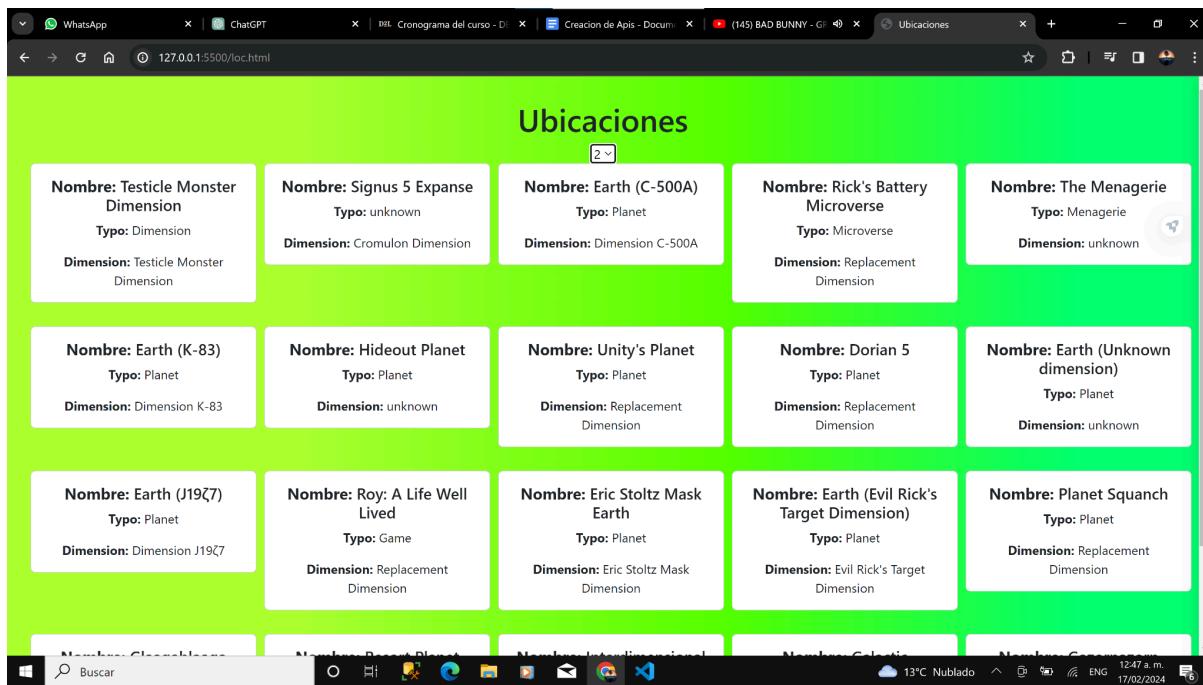
Explicación 2 loc.html

```

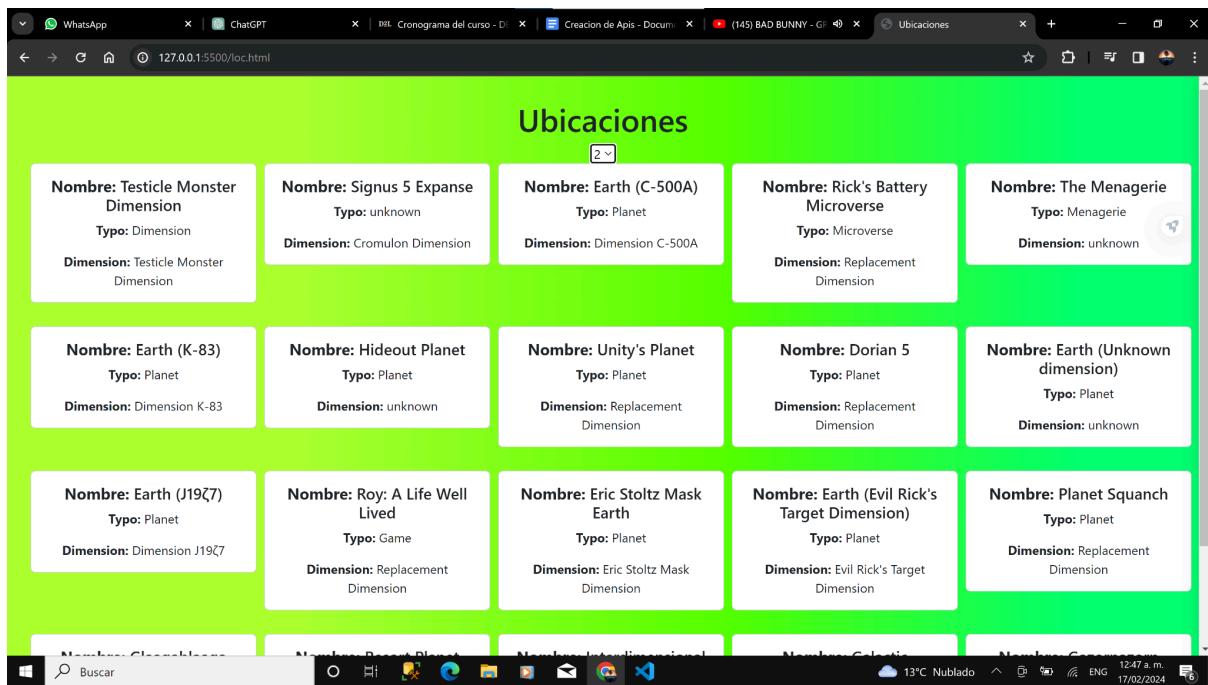
index.html  loc.html  epi.html  epis.js  rick.js  loc.js  style.css
  loc.html > html > body
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-T3c6CoIi6uLrA9TneNEoa7RxnatzcjDSCmGIMXxSR1GAsX" href="style.css">
7      <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js" integrity="sha384-KnHt8GjUdM8Q9v4AAQ6F0Oo034RTfRqOBXKuZPZL9BfZKJ" href="script.js">' + i + '</option>');
21
22             }
23         </script>
24     </select></h2>
25
26     <div id="resultado">
27
28     </div>
29
30     <script src="loc.js"></script>
31
32 </body>
33 </html>

```

En esta segunda parte del código HTML también representa una estructura básica de una página web, más bien se utilizó la misma base pero está más simplificado en comparación con el código anterior. Aquí está la explicación:



1. **<!DOCTYPE html>**: Declaración que indica que el documento sigue la especificación HTML5.
2. **<html lang="en">**: Elemento raíz del documento HTML con el atributo "lang" que establece el idioma en inglés.
3. **<head>**: Contiene metadatos y enlaces a recursos externos.
 - **<meta charset="UTF-8">**: Define la codificación de caracteres como UTF-8.
 - **<meta name="viewport" content="width=device-width, initial-scale=1.0">**: Configura la vista para dispositivos móviles.
 - Enlaces a hojas de estilo:
 - **<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-...">**: Enlace a la hoja de estilo de Bootstrap para el diseño y los estilos.
 - **<link rel="stylesheet" href="style.css">**: Enlace a una hoja de estilo local llamada "style.css".
 - **<title>Ubicaciones</title>**: Define el título de la página que aparecerá en la pestaña del navegador, que en este caso es "Ubicaciones".
4. **<body>**: Contiene el contenido principal de la página.
 - Título principal (**<h1>**): Muestra el título "Ubicaciones".
 - Sección para seleccionar una ubicación (**<select>**):
 - Un menú desplegable (**<select>**) con opciones generadas dinámicamente mediante JavaScript para las ubicaciones del 1 al 7.



- Contenedor para los resultados (<div id="resultado">): Se utilizará para mostrar la información después de seleccionar una ubicación.
- Se incluye el script "loc.js" al final del body (<script src="loc.js"></script>): Contiene funciones JavaScript para interactuar con la API de Rick y Morty específicamente para ubicaciones y manipular la página dinámicamente.

Explicación Epi.html

```

<div id="resultado"></div>
<script src="epis.js"></script>
<script src="loc.js"></script>
<script src="rick.js"></script>
<script src="style.css"></script>

```

```

<html>
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-T3c6CoIi6uLrA9TneNEoa7RxnatzcjDSCmG1MXxSR1GAsX" crossorigin="anonymous">
    <title>Episodios</title>
  </head>
  <body>
    <h1>Episodios</h1>

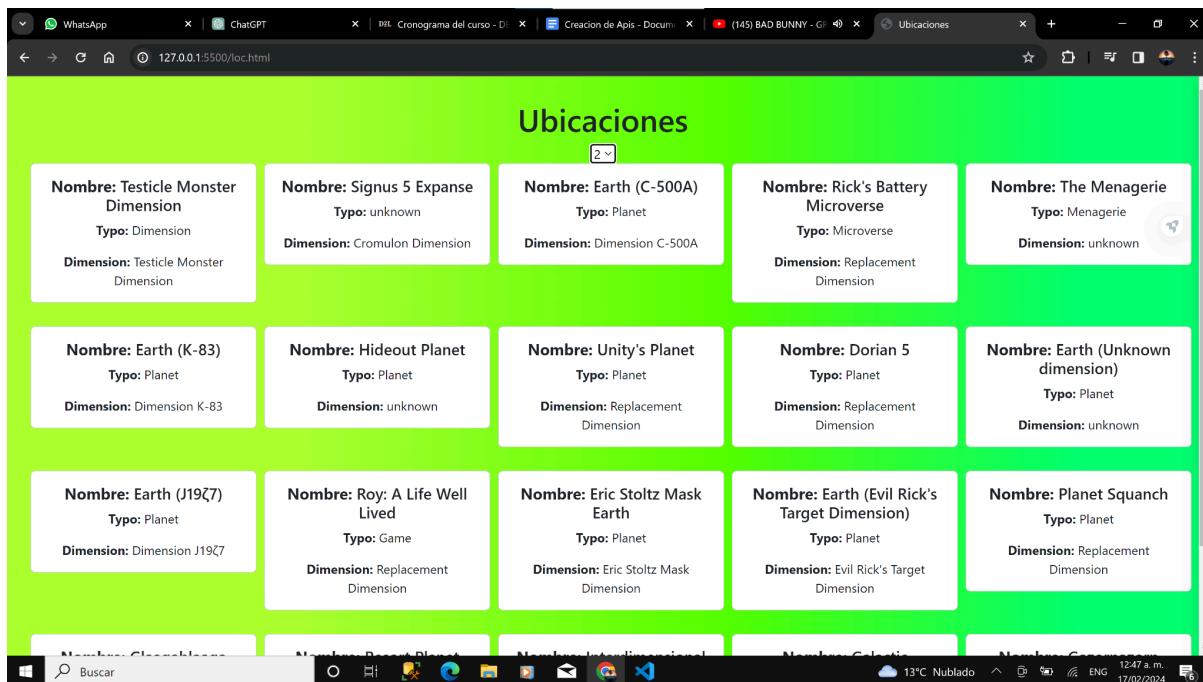
    <select name="episodios" id="epi" onchange="epiRick(this.value)">
      <script>
        for (let i = 1; i <= 7; i++){
          document.write('<option value="' + i + '">' + i + '</option>');
        }
      </script>
    </select></h1>

    <div id="resultado"></div>

    <script src="epis.js"></script>
  </body>
</html>

```

Este último código HTML también representa una estructura básica de una página web, similar a los ejemplos anteriores. Aquí está la explicación detallada:



- <!DOCTYPE html>**: Declaración que indica que el documento sigue la especificación HTML5.
- <html lang="en">**: Elemento raíz del documento HTML con el atributo "lang" que establece el idioma en inglés.
- <head>**: Contiene metadatos y enlaces a recursos externos.
 - <meta charset="UTF-8">**: Define la codificación de caracteres como UTF-8.
 - <meta name="viewport" content="width=device-width, initial-scale=1.0">**: Configura la vista para dispositivos móviles.
 - Enlaces a hojas de estilo:

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-...>: Enlace a la hoja de estilo de Bootstrap para el diseño y los estilos.

- <link rel="stylesheet" href="style.css">**: Enlace a una hoja de estilo local llamada "style.css".
- <title>Episodios</title>**: Define el título de la página que aparecerá en la pestaña del navegador, que en este caso es "Episodios".

- <body>**: Contiene el contenido principal de la página.
 - Título principal (**<h1>**): Muestra el título "Episodios".
 - Sección para seleccionar un episodio (**<select>**):

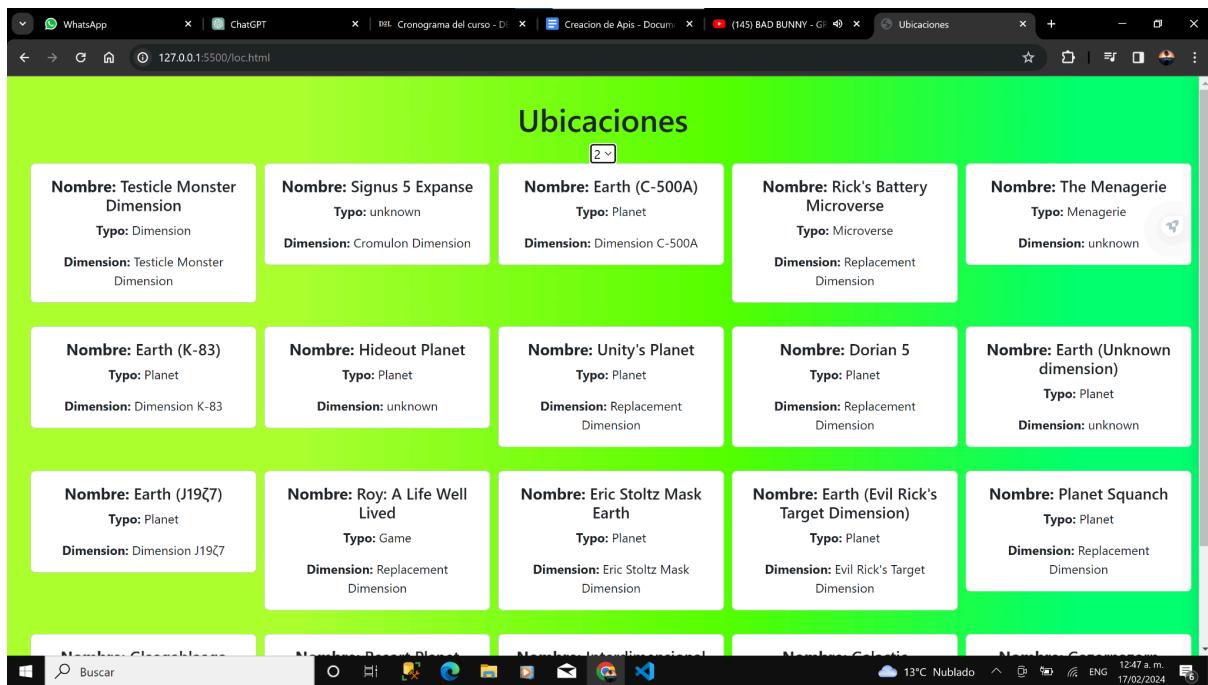
The screenshot shows a web application titled "Ubicaciones" (Locations) displaying a grid of 15 location cards. Each card contains the following information:

Nombre	Tipo	Dimension
Testicle Monster Dimension	Dimension	Testicle Monster Dimension
Signus 5 Expanse	unknown	Cromulon Dimension
Earth (C-500A)	Planet	Dimension C-500A
Rick's Battery	Microverse	Replacement Dimension
The Menagerie	Menagerie	unknown
Earth (K-83)	Planet	Dimension K-83
Hideout Planet	Planet	unknown
Unity's Planet	Planet	Replacement Dimension
Dorian 5	Planet	Replacement Dimension
Earth (Unknown dimension)	Planet	unknown
Earth (J19 ζ 7)	Planet	Dimension J19 ζ 7
Roy: A Life Well Lived	Game	Replacement Dimension
Eric Stoltz Mask Earth	Planet	Eric Stoltz Mask Dimension
Earth (Evil Rick's Target Dimension)	Planet	Evil Rick's Target Dimension
Planet Squanch	Planet	Replacement Dimension

The browser interface includes a search bar, a taskbar at the bottom with various pinned icons, and a system tray showing the date and time.

- Un menú desplegable (<select>) con opciones generadas dinámicamente mediante JavaScript para los episodios del 1 al 7.
- Contenedor para los resultados (<div id="resultado">): Se utilizará para mostrar la información después de seleccionar un episodio.
- Se incluye el script "epis.js" al final del body (`<script src="epis.js"></script>`): Contiene funciones JavaScript para interactuar con la API de Rick y Morty específicamente para episodios y manipular la página dinámicamente.

Explicación Style.css

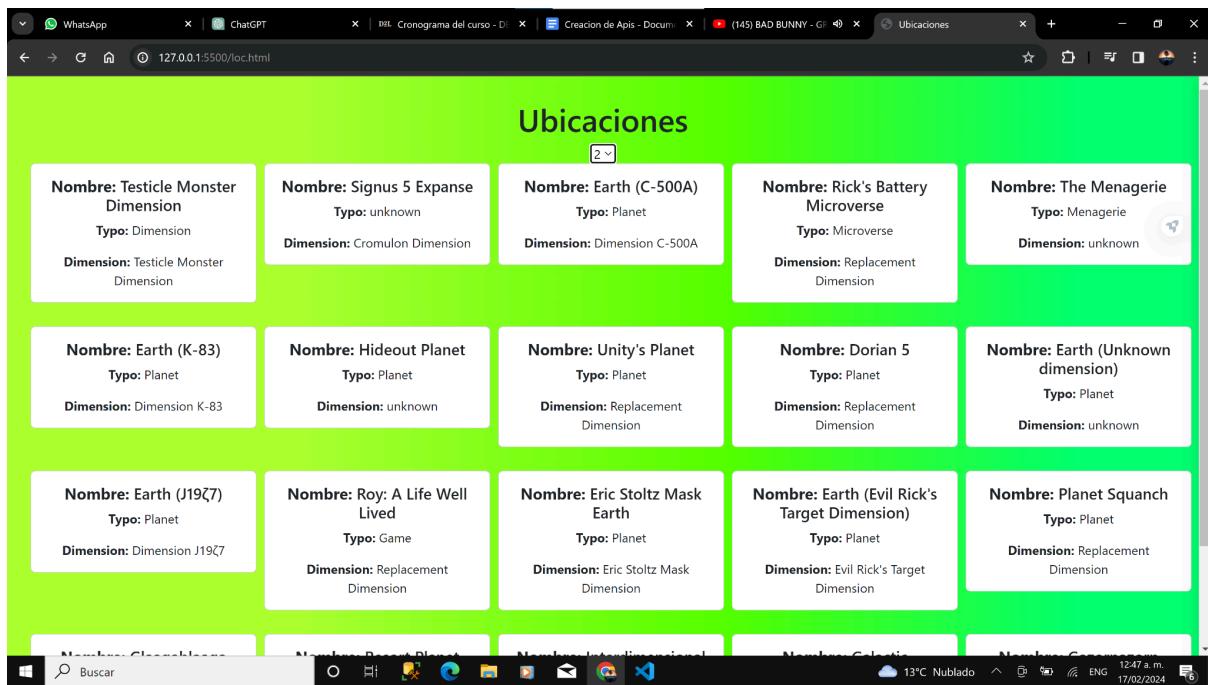


```
# style.css > ...
1  *{
2      margin: 0%;
3      padding: 0%;
4      box-sizing: border-box;
5  }
6  body{
7      text-align: center;
8      margin: 30px;
9      background: linear-gradient(90deg, #rgb(171,255,49));
10     background: linear-gradient(90deg, #rgba(171,255,49,1) 25%, #rgba(86,255,0,1) 60%, #rgba(0,255,111,1) 90%);
11  }
12
13 #resultado{
14     display: grid;
15     grid-template-columns: repeat(5, 1fr);
16     grid-template-rows: auto;
17     gap: 30px 10px;
18 }
19
```

Este código es un conjunto de reglas de estilo CSS que se aplicarán a los elementos HTML de la página web Rick y Morty. Aquí se dará una explicación detallada de las reglas:

1. * { margin: 0%; padding: 0%; box-sizing: border-box; }:

- *: Selecciona todos los elementos de la página.
- margin: 0%; padding: 0%;: Establece el margen y el relleno de todos los elementos en 0%. Esto elimina los márgenes y rellenos predeterminados del navegador, brindando un punto de partida consistente para los estilos.
- box-sizing: border-box;: Hace que el modelo de caja incluya el tamaño del borde y el relleno, en lugar de agregarlos al ancho y alto. Esto facilita el control del tamaño total de un elemento.

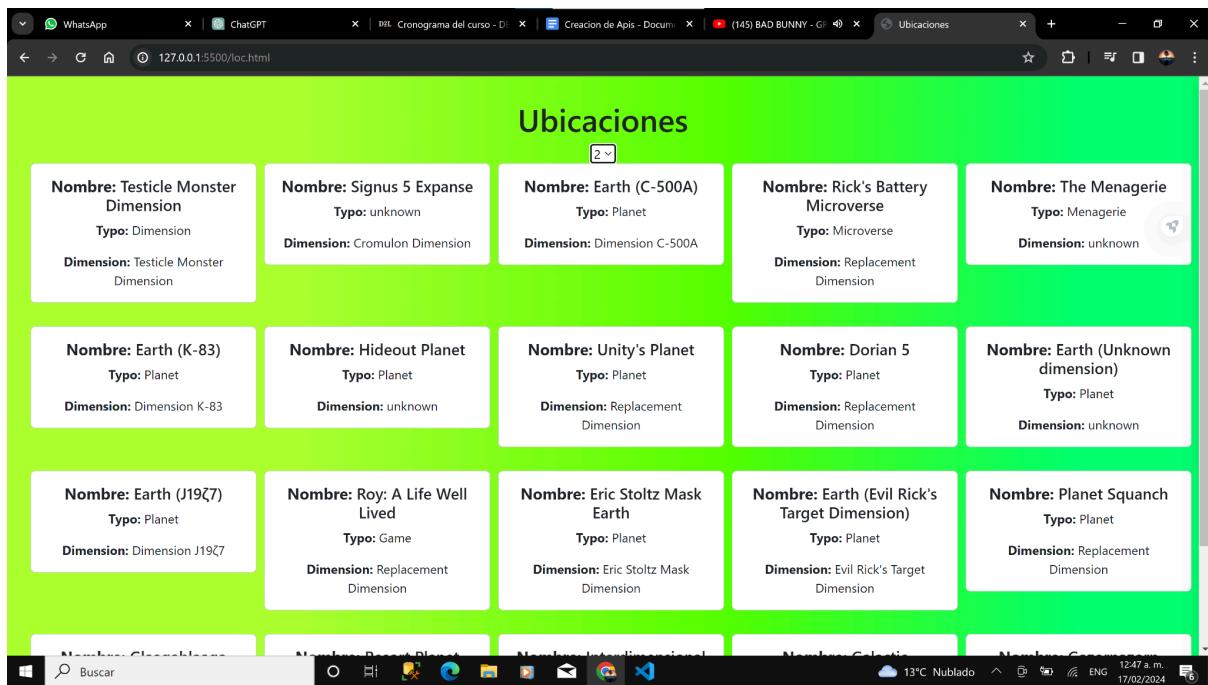


2. body { text-align: center; margin: 30px; background: ... }:

- body: Selecciona el elemento `<body>` de la página.
- text-align: center;: Centra el texto dentro del cuerpo de la página.
- margin: 30px;: Establece un margen de 30 píxeles alrededor del cuerpo de la página.
- background: ...: Establece un fondo para el cuerpo utilizando un degradado lineal. El degradado se compone de tres colores diferentes:
 - Color 1: rgba(171, 255, 49, 1) con una posición del 25%.
 - Color 2: rgba(86, 255, 0, 1) con una posición del 60%.
 - Color 3: rgba(0, 255, 111, 1) con una posición del 90%.
 - Este fondo crea un degradado de colores en la dirección horizontal.

3. #resultado { display: grid; grid-template-columns: repeat(5, 1fr); grid-template-rows: auto; gap: 30px 10px; }:

- #resultado: Selecciona el elemento con el ID "resultado".
- display: grid;: Hace que el contenedor utilice el modelo de diseño de cuadrícula.
- grid-template-columns: repeat(5, 1fr): Divide el contenedor en 5 columnas de tamaño igual ('1fr' significa una fracción del espacio disponible).
- grid-template-rows: auto: Las filas se ajustarán automáticamente según su contenido.
- gap: 30px 10px: Establece espacios entre las filas y las columnas de 30 píxeles y 10 píxeles respectivamente.



Explicación Rick.js

```

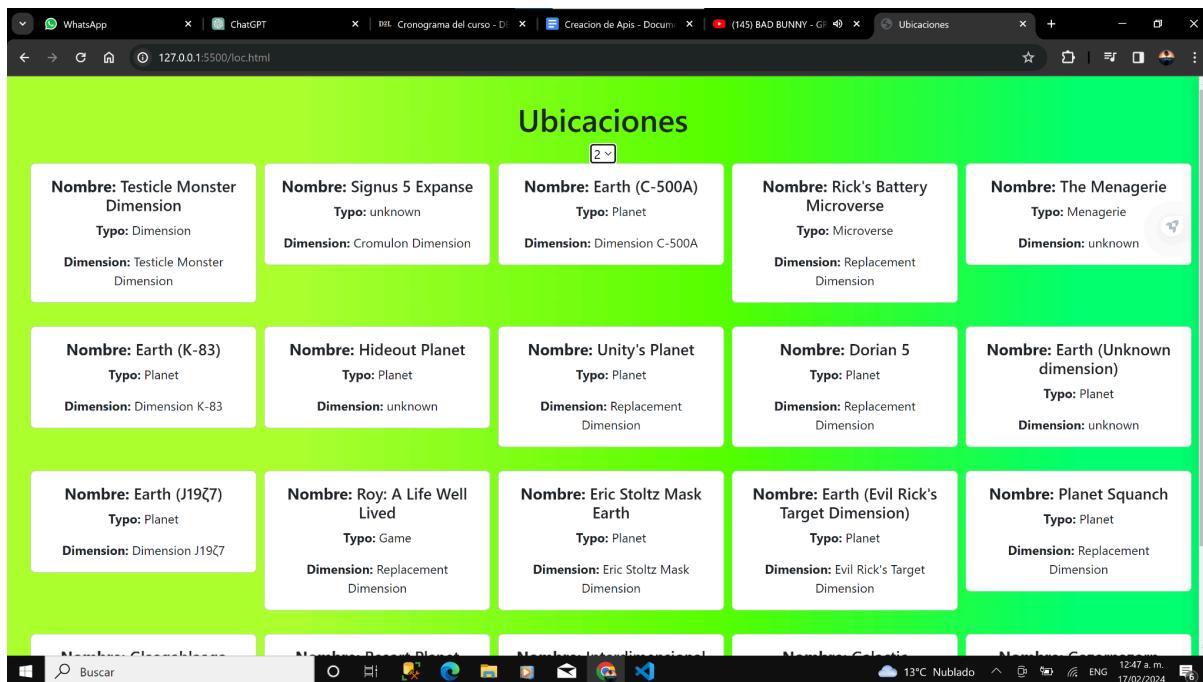
index.html | loc.html | epi.html | epis.js | rick.js | loc.js | style.css

js rick.js > [e] apiRick > ⚡ data.results.map() callback
1 const apiRick=async(pagina)=>{
2     let url="https://rickandmortyapi.com/api/character/?page="+pagina;
3     const api= await fetch(url);
4     const data= await api.json();
5     console.log(data);
6     divRes=document.querySelector("#resultado");
7     divRes.innerHTML="";
8     data.results.map(item=>[
9         divItem=document.createElement('div')
10        divItem.innerHTML=`

11            
12            <div class="card-body">
13                <h5 class="card-title"><b>${item.name}</b></h5>
14                <p class="card-text"><b>Estatus:</b> ${item.status}</p>
15                <p class="card-text"><b>Especie:</b> ${item.species}</p>
16                <p class="card-text"><b>Género:</b> ${item.gender}</p>
17
18
19
20                <a href="http://127.0.0.1:5500/loc.html" class="btn btn-primary">Ubicaciones</a>
21            </div>
22        </div>
23
24        divRes.appendChild(divItem);
25    ])
26 }
27 apiRick(1)
28
29
30


```

Este código JavaScript realiza una solicitud a la API de Rick and Morty para obtener información sobre personajes y luego muestra esa información en la página web:



1. const apiRick = async (pagina) => {:

- Declara una función llamada `apiRick` que acepta un parámetro página.

2. let url = "https://rickandmortyapi.com/api/character/?page=" + pagina;

- Construye la URL de la API concatenando la página proporcionada al final.

3. const api = await fetch(url);

- Utiliza la función fetch para realizar una solicitud a la API utilizando la URL construida.
- La palabra clave await se utiliza porque fetch devuelve una promesa, y se espera a que esa promesa se resuelva antes de continuar.

4. const data = await api.json();

- Convierte la respuesta de la API en formato JSON.
- Nuevamente, se utiliza await porque json() también devuelve una promesa.

5. console.log(data);

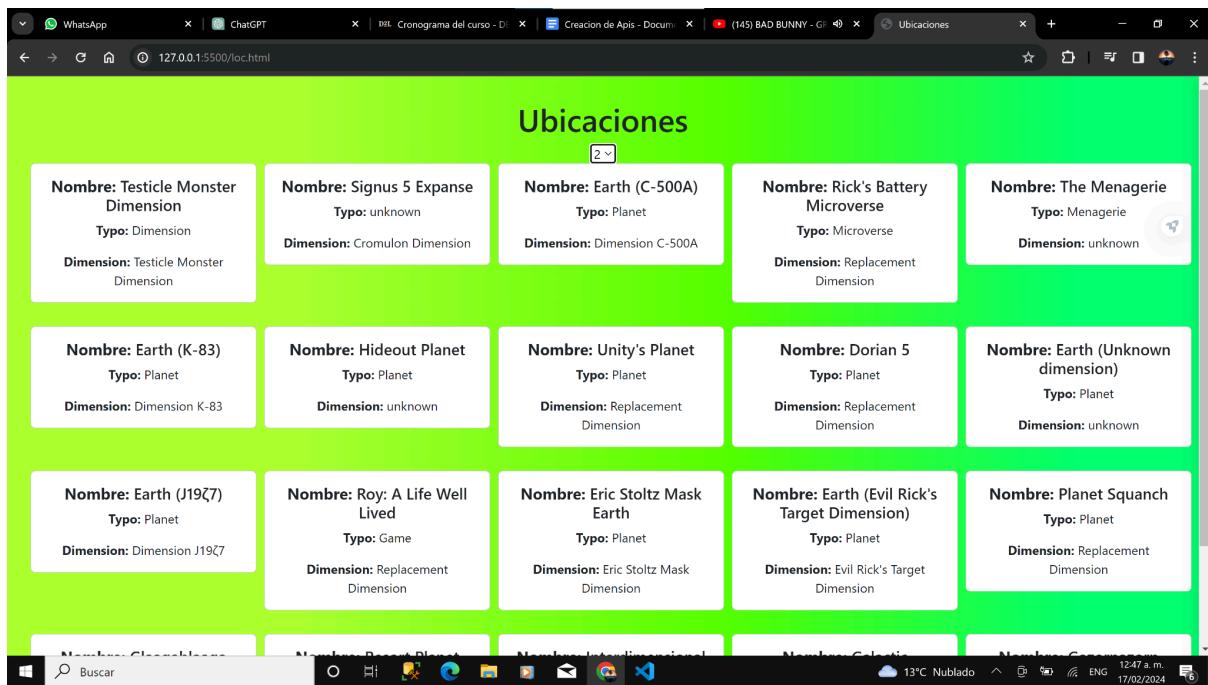
- Imprime los datos obtenidos de la API en la consola del navegador.

6. divRes = document.querySelector("#resultado");

- Selecciona el elemento HTML con el ID "resultado" donde se mostrará la información.

7. divRes.innerHTML = "":

- Limpia el contenido actual del elemento con ID "resultado".



8. `data.results.map(item => {`:

- Utiliza el método map para iterar sobre los resultados de la API.

9. `divItem = document.createElement('div')`:

- Crea un nuevo elemento `div` para cada personaje.

10. `divItem.innerHTML = " "`:

- Define el contenido HTML del elemento div creado con información específica de cada personaje, como la imagen, nombre, estatus, especie y género.

11. `divRes.appendChild(divItem)`:

- Agrega el elemento `div` con la información del personaje al contenedor con ID "resultado".

12. `apiRick(1)`:

- Llama a la función apiRick con el argumento 1, lo que inicializa la carga de información de personajes en la página con la primera página de la API.

Explicación loc.js

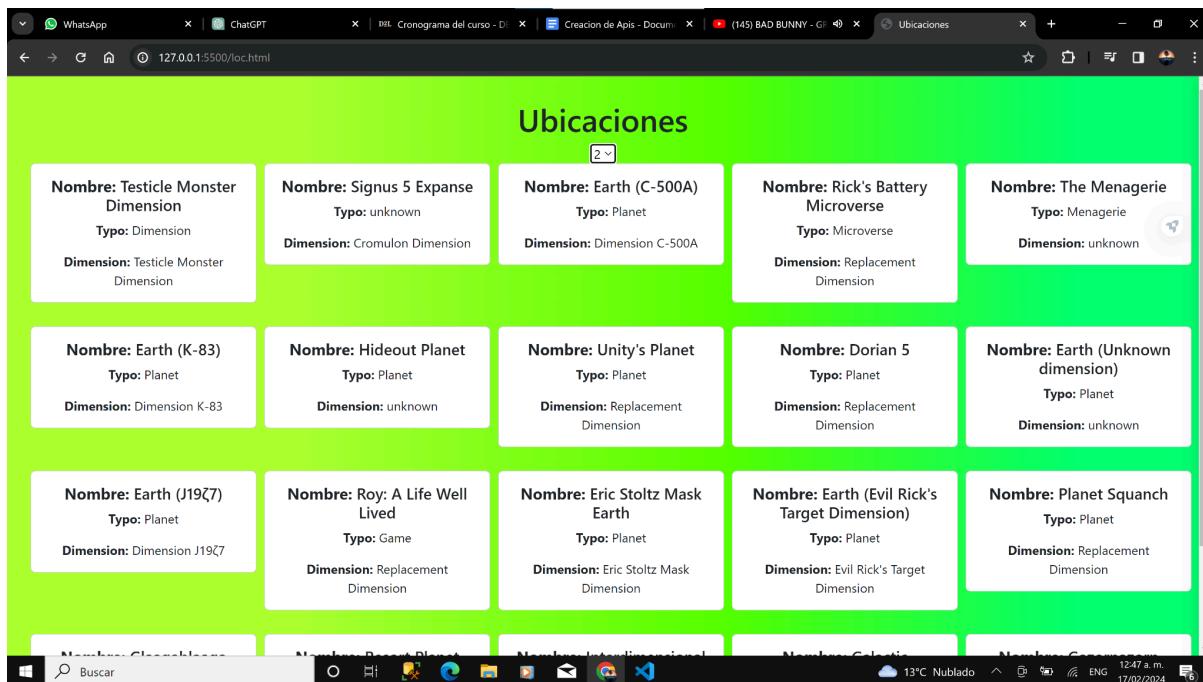


```
index.html  loc.html  epi.html  epis.js  rick.js  loc.js  # style.css

JS loc.js > locRick > data.results.map() callback
1 const locRick = async (loc) => {
2   let url = "https://rickandmortyapi.com/api/location?page=" + loc;
3   const api = await fetch(url);
4   const data = await api.json();
5   console.log(data);
6   divRes=document.querySelector("#resultado");
7   divRes.innerHTML="";
8   data.results.map(item=>{
9     divItem=document.createElement('div')
10    divItem.innerHTML=
11      <div class="card" style="width: 18rem;">
12        <div class="card-body">
13          <h5 class="card-title"><b>Nombre:</b> ${item.name}</h5>
14          <p class="card-text"><b>Tipo:</b> ${item.type}</p>
15          <p class="card-text"><b>Dimension:</b> ${item.dimension}</p>
16        </div>
17      </div>
18    '
19    divRes.appendChild(divItem);
20  })
21}
22 locRick(1)
23
```

Este código realiza una solicitud a la API de Rick and Morty para obtener información sobre ubicaciones y luego muestra esa información en la página web:

- 1. const locRick = async (loc) => {:**
 - Declara una función llamada `locRick` que acepta un parámetro `loc`.
- 2. let url = "https://rickandmortyapi.com/api/location?page=" + loc;:**
 - Construye la URL de la API concatenando la página proporcionada al final.



3. const api = await fetch(url):

- Utiliza la función fetch para realizar una solicitud a la API utilizando la URL construida.
- La palabra clave await se utiliza porque `fetch` devuelve una promesa, y se espera a que esa promesa se resuelva antes de continuar.

4. const data = await api.json():

- Convierte la respuesta de la API en formato JSON.
- Nuevamente, se utiliza `await` porque `json()` también devuelve una promesa.

5. console.log(data):

- Imprime los datos obtenidos de la API en la consola del navegador.

6. divRes = document.querySelector("#resultado"):

- Selecciona el elemento HTML con el ID "resultado" donde se mostrará la información.

7. divRes.innerHTML = "":

- Limpia el contenido actual del elemento con ID "resultado".

8. data.results.map(item => {

- Utiliza el método map para iterar sobre los resultados de la API.

9. divItem = document.createElement('div'):

- Crea un nuevo elemento `div` para cada ubicación.

10. divItem.innerHTML = “”:

The screenshot shows a web page titled "Ubicaciones" (Locations) with a green header. Below the header is a grid of 10 cards, each containing information about a specific location. The cards are arranged in three rows: Row 1 has five cards; Row 2 has five cards; and Row 3 has five cards. Each card has a title, type, and dimension.

Nombre: Testicle Monster Dimension	Nombre: Signus 5 Expanse Type: unknown Dimension: Cromulon Dimension	Nombre: Earth (C-500A) Type: Planet Dimension: Dimension C-500A	Nombre: Rick's Battery Microverse Type: Microverse Dimension: Replacement Dimension	Nombre: The Menagerie Type: Menagerie Dimension: unknown
Nombre: Earth (K-83) Type: Planet Dimension: Dimension K-83	Nombre: Hideout Planet Type: Planet Dimension: unknown	Nombre: Unity's Planet Type: Planet Dimension: Replacement Dimension	Nombre: Dorian 5 Type: Planet Dimension: Replacement Dimension	Nombre: Earth (Unknown dimension) Type: Planet Dimension: unknown
Nombre: Earth (J19z7) Type: Planet Dimension: Dimension J19z7	Nombre: Roy: A Life Well Lived Type: Game Dimension: Replacement Dimension	Nombre: Eric Stoltz Mask Earth Type: Planet Dimension: Eric Stoltz Mask Dimension	Nombre: Earth (Evil Rick's Target Dimension) Type: Planet Dimension: Evil Rick's Target Dimension	Nombre: Planet Squanch Type: Planet Dimension: Replacement Dimension

- Define el contenido HTML del elemento `div` creado con información específica de cada ubicación, como nombre, tipo y dimensión.

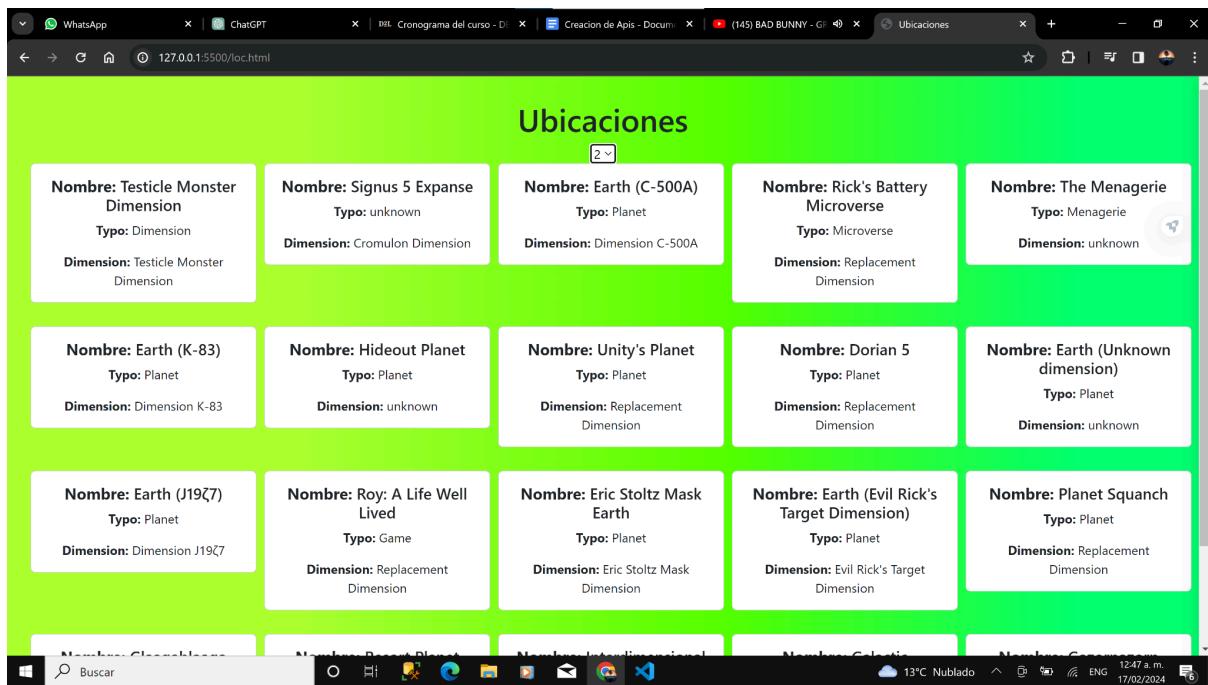
11. `divRes.appendChild(divItem):`

- Agrega el elemento div con la información de la ubicación al contenedor con ID "resultado".

12. `locRick(1):`

- Llama a la función locRick con el argumento 1, lo que inicializa la carga de información de ubicaciones en la página con la primera página de la API.

Explicación epis.js



```
index.html | loc.html | epi.html | JS epis.js | X JS rick.js | JS loc.js | # style.css

JS epis.js > [e] epiRick > ⚡ data.results.map() callback
1 const epiRick=async(epi)=>{
2     let url="https://rickandmortyapi.com/api/episode?page="+epi;
3     const api= await fetch(url);
4     const data= await api.json();
5     console.log(data);
6     divRes=document.querySelector("#resultado");
7     divRes.innerHTML="";
8     data.results.map(item=>[
9         divItem=document.createElement('div')
10        divItem.innerHTML=`

11            

12                

##### <b>ID: </b>${item.id}</h5>


13                <p class="card-text"><b>Nombre: </b>${item.name}</p>
14                <p class="card-text"><b>Fecha de episodio: </b>${item.air_date}</p>
15                <p class="card-text"><b>Episodio: </b>${item.episode}</p>
16


17


18    `)
19    divRes.appendChild(divItem);
20 }
21 epiRick(1)
22
23
24
25
26
27
28
29
30
31
```

Este último código realiza una solicitud a la API de Rick and Morty para obtener información sobre episodios:

Ubicaciones				
Nombre: Testicle Monster Dimension Tipo: Dimension Dimension: Testicle Monster Dimension	Nombre: Signus 5 Expanse Tipo: unknown Dimension: Cromulon Dimension	Nombre: Earth (C-500A) Tipo: Planet Dimension: Dimension C-500A	Nombre: Rick's Battery Microverse Tipo: Microverse Dimension: Replacement Dimension	Nombre: The Menagerie Tipo: Menagerie Dimension: unknown
Nombre: Earth (K-83) Tipo: Planet Dimension: Dimension K-83	Nombre: Hideout Planet Tipo: Planet Dimension: unknown	Nombre: Unity's Planet Tipo: Planet Dimension: Replacement Dimension	Nombre: Dorian 5 Tipo: Planet Dimension: Replacement Dimension	Nombre: Earth (Unknown dimension) Tipo: Planet Dimension: unknown
Nombre: Earth (J19z7) Tipo: Planet Dimension: Dimension J19z7	Nombre: Roy: A Life Well Lived Tipo: Game Dimension: Replacement Dimension	Nombre: Eric Stoltz Mask Earth Tipo: Planet Dimension: Eric Stoltz Mask Dimension	Nombre: Earth (Evil Rick's Target Dimension) Tipo: Planet Dimension: Evil Rick's Target Dimension	Nombre: Planet Squanch Tipo: Planet Dimension: Replacement Dimension

1. const epiRick = async (epi) => {:

- Declara una función llamada `epiRick` que acepta un parámetro `epi`.

2. let url = "https://rickandmortyapi.com/api/episode?page=" + epi;

- Construye la URL de la API concatenando la página proporcionada al final.

3. const api = await fetch(url);

- Utiliza la función fetch para realizar una solicitud a la API utilizando la URL construida.
- La palabra clave await se utiliza porque fetch devuelve una promesa, y se espera a que esa promesa se resuelva antes de continuar.

4. const data = await api.json();

- Convierte la respuesta de la API en formato JSON.
- Nuevamente, se utiliza await porque json() también devuelve una promesa.

5. console.log(data);

- Imprime los datos obtenidos de la API en la consola del navegador.

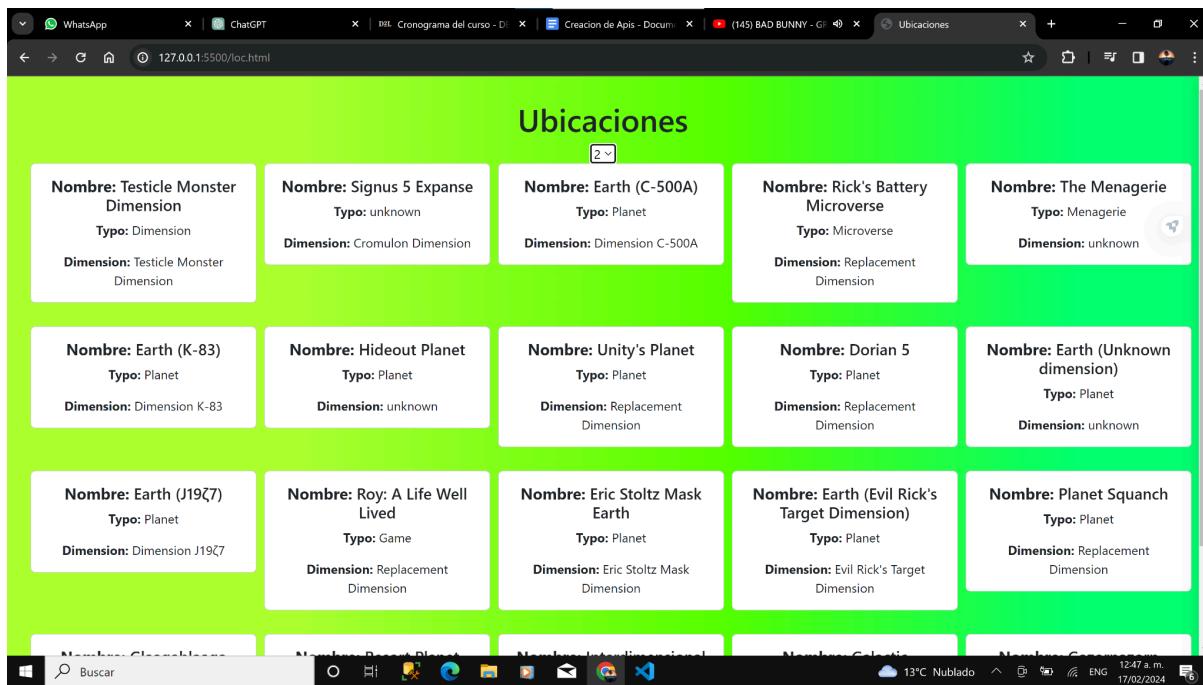
6. divRes = document.querySelector("#resultado");

- Selecciona el elemento HTML con el ID "resultado" donde se mostrará la información.

7. divRes.innerHTML = "":

- Limpia el contenido actual del elemento con ID "resultado".

8. data.results.map(item => {:



- Utiliza el método map para iterar sobre los resultados de la API.

9. **divItem = document.createElement('div'):**

- Crea un nuevo elemento `div` para cada episodio.

10. **divItem.innerHTML = “”:**

- Define el contenido HTML del elemento div creado con información específica de cada episodio, como ID, nombre, fecha del episodio y número de episodio.

11. **divRes.appendChild(divItem):**

- Agrega el elemento div con la información del episodio al contenedor con ID "resultado".

12. **epiRick(1):**

- Llama a la función `epiRick` con el argumento 1, lo que inicializa la carga de información de episodios en la página con la primera página de la API.

Pantalla inicio

Como se puede observar en la página se puede apreciar imágenes de los personajes que han estado en la serie de rick y morty, abajo de la imagen del personaje contiene información del personaje como por ejemplo el nombre, el

Ubicaciones

2 ▾

Nombre: Testicle Monster Dimension Tipo: Dimension Dimension: Testicle Monster Dimension	Nombre: Signus 5 Expanse Tipo: unknown Dimension: Cromulon Dimension	Nombre: Earth (C-500A) Tipo: Planet Dimension: Dimension C-500A	Nombre: Rick's Battery Microverse Tipo: Microverse Dimension: Replacement Dimension	Nombre: The Menagerie Tipo: Menagerie Dimension: unknown
Nombre: Earth (K-83) Tipo: Planet Dimension: Dimension K-83	Nombre: Hideout Planet Tipo: Planet Dimension: unknown	Nombre: Unity's Planet Tipo: Planet Dimension: Replacement Dimension	Nombre: Dorian 5 Tipo: Planet Dimension: Replacement Dimension	Nombre: Earth (Unknown dimension) Tipo: Planet Dimension: unknown
Nombre: Earth (J19 ζ 7) Tipo: Planet Dimension: Dimension J19 ζ 7	Nombre: Roy: A Life Well Lived Tipo: Game Dimension: Replacement Dimension	Nombre: Eric Stoltz Mask Earth Tipo: Planet Dimension: Eric Stoltz Mask Dimension	Nombre: Earth (Evil Rick's Target Dimension) Tipo: Planet Dimension: Evil Rick's Target Dimension	Nombre: Planet Squanch Tipo: Planet Dimension: Replacement Dimension

estado si esta vivo o muerto, el tipo de especie, por ejemplo si es humano o alien y por ultimo si es de género masculino o femenino

Rick y Morty API

Personajes Episodios Ubicaciones

PAGINA 1 ▾

 Rick Sanchez Estatus: Alive Especie: Human Género: Male Ubicaciones	 Morty Smith Estatus: Alive Especie: Human Género: Male Ubicaciones	 Summer Smith Estatus: Alive Especie: Human Género: Female Ubicaciones	 Beth Smith Estatus: Alive Especie: Human Género: Female Ubicaciones	 Jerry Smith Estatus: Alive Especie: Human Género: Male Ubicaciones
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Ubicaciones

2

Nombre: Testicle Monster Dimension Tipo: Dimension Dimension: Testicle Monster Dimension	Nombre: Signus 5 Expanse Tipo: unknown Dimension: Cromulon Dimension	Nombre: Earth (C-500A) Tipo: Planet Dimension: Dimension C-500A	Nombre: Rick's Battery Microverse Tipo: Microverse Dimension: Replacement Dimension	Nombre: The Menagerie Tipo: Menagerie Dimension: unknown
Nombre: Earth (K-83) Tipo: Planet Dimension: Dimension K-83	Nombre: Hideout Planet Tipo: Planet Dimension: unknown	Nombre: Unity's Planet Tipo: Planet Dimension: Replacement Dimension	Nombre: Dorian 5 Tipo: Planet Dimension: Replacement Dimension	Nombre: Earth (Unknown dimension) Tipo: Planet Dimension: unknown
Nombre: Earth (J19 ζ 7) Tipo: Planet Dimension: Dimension J19 ζ 7	Nombre: Roy: A Life Well Lived Tipo: Game Dimension: Replacement Dimension	Nombre: Eric Stoltz Mask Earth Tipo: Planet Dimension: Eric Stoltz Mask Dimension	Nombre: Earth (Evil Rick's Target Dimension) Tipo: Planet Dimension: Evil Rick's Target Dimension	Nombre: Planet Squanch Tipo: Planet Dimension: Replacement Dimension

Rick y Morty API Personajes Episodios Ubicaciones

PAGINA 2

Aqua Morty Estatus: unknown Especie: Humanoid Género: Male Ubicaciones	Aqua Rick Estatus: unknown Especie: Humanoid Género: Male Ubicaciones	Arcade Alien Estatus: unknown Especie: Alien Género: Male Ubicaciones	Armagheadon Estatus: Alive Especie: Alien Género: Male Ubicaciones	Armothy Estatus: Dead Especie: unknown Género: Male Ubicaciones

Pantalla ubicaciones

Como se puede notar no solo te dara el nombre y estatus de los personajes la pagina tambien incluira páginas exclusivas para los episodios y las ubicaciones(Planetas)

Ubicaciones

Nombre: Testicle Monster Dimension Tipo: Dimension Dimension: Testicle Monster Dimension	Nombre: Signus 5 Expanse Tipo: unknown Dimension: Cromulon Dimension	Nombre: Earth (C-500A) Tipo: Planet Dimension: Dimension C-500A	Nombre: Rick's Battery Microverse Tipo: Microverse Dimension: Replacement Dimension	Nombre: The Menagerie Tipo: Menagerie Dimension: unknown
Nombre: Earth (K-83) Tipo: Planet Dimension: Dimension K-83	Nombre: Hideout Planet Tipo: Planet Dimension: unknown	Nombre: Unity's Planet Tipo: Planet Dimension: Replacement Dimension	Nombre: Dorian 5 Tipo: Planet Dimension: Replacement Dimension	Nombre: Earth (Unknown dimension) Tipo: Planet Dimension: unknown
Nombre: Earth (J19z7) Tipo: Planet Dimension: Dimension J19z7	Nombre: Roy: A Life Well Lived Tipo: Game Dimension: Replacement Dimension	Nombre: Eric Stoltz Mask Earth Tipo: Planet Dimension: Eric Stoltz Mask Dimension	Nombre: Earth (Evil Rick's Target Dimension) Tipo: Planet Dimension: Evil Rick's Target Dimension	Nombre: Planet Squanch Tipo: Planet Dimension: Replacement Dimension

Rick y Morty API Personajes Episodios Ubicaciones

La única diferencia de los personajes son las imágenes ya que no se conoce con exactitud como es en sí el planeta, solo tendrá el nombre del planeta, dimensión en la que se encuentra este planeta, etc.

Ubicaciones

Nombre: Earth (C-137) Tipo: Planet Dimension: Dimension C-137	Nombre: Abadango Tipo: Cluster Dimension: unknown	Nombre: Citadel of Ricks Tipo: Space station Dimension: unknown	Nombre: Worldender's lair Tipo: Planet Dimension: unknown	Nombre: Anatomy Park Tipo: Microverse Dimension: Dimension C-137
Nombre: Interdimensional Cable Tipo: TV Dimension: unknown	Nombre: Immortality Field Resort Tipo: Resort Dimension: unknown	Nombre: Post-Apocalyptic Earth Tipo: Planet Dimension: Post-Apocalyptic Dimension	Nombre: Purge Planet Tipo: Planet Dimension: Replacement Dimension	Nombre: Venzenulon 7 Tipo: Planet Dimension: unknown
Nombre: Bepis 9 Tipo: Planet Dimension: unknown	Nombre: Cronenberg Earth Tipo: Planet Dimension: Cronenberg Dimension	Nombre: Nuptia 4 Tipo: Planet Dimension: unknown	Nombre: Giant's Town Tipo: Fantasy town Dimension: Fantasy Dimension	Nombre: Bird World Tipo: Planet Dimension: unknown
Nombre: St. Gloopy Noops Hospital Tipo: Space station	Nombre: Earth (5-126) Tipo: Planet	Nombre: Mr. Goldenfold's dream Tipo: Dream	Nombre: Gromflom Prime Tipo: Planet	Nombre: Earth (Replacement Dimension) Tipo: Planet

Pantallas episodios

The screenshot shows a web browser window with a green header bar containing the title "Ubicaciones". Below the header is a grid of 15 cards, each representing a location from the Rick and Morty universe. The cards are arranged in three rows of five. Each card contains the following information:

Nombre	Tipo	Dimension
Testicle Monster Dimension	Dimension	Testicle Monster Dimension
Signus 5 Expanse	unknown	Cromulon Dimension
Earth (C-500A)	Planet	Dimension C-500A
Rick's Battery	Microverse	Replacement Dimension
The Menagerie	Menagerie	unknown
Earth (K-83)	Planet	Dimension K-83
Hideout Planet	Planet	unknown
Unity's Planet	Planet	Replacement Dimension
Dorian 5	Planet	Replacement Dimension
Earth (Unknown dimension)	Planet	unknown
Earth (J19ζ7)	Planet	Dimension J19ζ7
Roy: A Life Well Lived	Game	Replacement Dimension
Eric Stoltz Mask Earth	Planet	Eric Stoltz Mask Dimension
Earth (Evil Rick's Target Dimension)	Planet	Evil Rick's Target Dimension
Planet Squanch	Planet	Replacement Dimension

Aquí encontrarás el nombre del episodio, su fecha de lanzamiento y el número de episodio. es similar a la de los planetas, esta página contiene todos los episodios de la serie.

The screenshot shows a web browser window with a green header bar containing the title "Episodios". Below the header is a grid of 15 cards, each representing an episode from the first season of Rick and Morty. The cards are arranged in three rows of five. Each card contains the following information:

ID	Nombre	Fecha de episodio	Episodio
1	Pilot	December 2, 2013	S01E01
2	Lawnmower Dog	December 9, 2013	S01E02
3	Anatomy Park	December 16, 2013	S01E03
4	M. Night Shaym-Aliens!	January 13, 2014	S01E04
5	Meeseeks and Destroy	January 20, 2014	S01E05
6	Rick Potion #9	January 27, 2014	S01E06
7	Raising Gazorpazorp	March 10, 2014	S01E07
8	Rixty Minutes	March 17, 2014	S01E08
9	Something Ricked This Way Comes	March 24, 2014	S01E09
10	Close Rick-counters of the Rick Kind	April 7, 2014	S01E10
11	Ricksy Business	April 14, 2014	S01E11
12	A Rickle in Time	July 26, 2015	S02E01
13	Mortynight Run	August 2, 2015	S02E02
14	Auto Erotic Assimilation	August 9, 2015	S02E03
15	Total Rickall	August 16, 2015	S02E04

Conclusión

En este proyecto, hemos explorado la integración de la API de Rick and Morty en una interfaz web utilizando tecnologías web estándar como HTML, CSS y JavaScript. Cada página dedicada a personajes, ubicaciones y episodios presenta la

The screenshot shows a web browser window with multiple tabs open at the top. The active tab is titled "Ubicaciones". The main content area displays a grid of 15 cards, each representing a location with the following details:

Nombre	Tipo	Dimension
Testicle Monster Dimension	Dimension	Testicle Monster Dimension
Signus 5 Expanse	unknown	Cromulon Dimension
Earth (C-500A)	Planet	Dimension C-500A
Rick's Battery	Microverse	Replacement Dimension
The Menagerie	Menagerie	unknown
Earth (K-83)	Planet	Dimension K-83
Hideout Planet	Planet	unknown
Unity's Planet	Planet	Replacement Dimension
Dorian 5	Planet	Replacement Dimension
Earth (Unknown dimension)	Planet	unknown
Earth (J19z7)	Planet	Dimension J19z7
Roy: A Life Well Lived	Game	Replacement Dimension
Eric Stoltz Mask Earth	Planet	Eric Stoltz Mask Dimension
Earth (Evil Rick's Target Dimension)	Planet	Evil Rick's Target Dimension
Planet Squanch	Planet	Replacement Dimension

The browser interface includes a search bar, pinned icons, and system status indicators like battery level and date/time.

información de manera estructurada y atractiva, permitiendo a los usuarios navegar y explorar los datos de la serie de manera fácil. El uso de funciones asíncronas y el manejo de promesas en JavaScript ha sido esencial para realizar solicitudes a la API de manera eficiente. Este proyecto sirve como ejemplo práctico de cómo aprovechar las APIs en el desarrollo web para crear experiencias interactivas y dinámicas para los usuarios.