LABORATORIO 2 RIA 2020

16 de junio de 2016

Federico Bentancor CI: 4.735.761-6

Facundo Laborde CI: 4.405.478-4

Tecnólogo Informático

Contenido

Manual de configuración y uso

1. Descripción de las tecnologías:

a. Github

Usamos Github para el versionado de nuestra aplicación. El código está público en https://github.com/fedeben11/rias2020

b. REACT

Elegimos REACT JS para desarrollar nuestra aplicación web. Nos permite tener una aplicación dinámica y con una capa de abstracción sobre JavaScript, lo cuál es una ventaja ya que nos simplifica de sobremanera el desarrollo del sitio, la curva de aprendizaje y el tiempo utilizado para el desarrollo.

i. Funcionalidades utilizadas de REACT en el desarrollo:

- Componentes
- Render
- Funciones
- Propiedades
- Estados

c. BOOSTRAP

Usamos boostrap para el layout del sitio. Boostrap nos permitió tener un diseño elaborado de manera fácil importanto un theme.

Además boostrap nos permite tener con unos pocos cambios un sitio web responsive.

d. HTML

HTML como base del sitio web, y sobre esto los estilos, las clases de boostrap y el código REACT

e. API REST

REST es cualquier interfaz entre sistemas que use HTTP para obtener datos o generar operaciones sobre esos datos en todos los formatos posibles, como XML y JSON

En nuestro caso recibimos un JSON desde https://rapidapi.com/theapiguy/api/free-nba

Tiene 2 métodos GET publicados en su endpoint.

i. Documentación oficial de la API:

ii. Jugadores:

Método Players

Get All Players

Este endpoint devuelve todos los jugadores de todas las temporadas.

HTTP Request

GET /api/v1/players

Parámetros de consulta:

page – Número de página, usado para la paginación.

per_page – El número de resultados retornados por llamado. Usado para la paginación.

search – Usado para filtrar jugadores basado en el nombre. Por ejemplo ?search=davis retornará jugadores que contengan 'davis' en su nombre o su apellido.

Respuesta ejemplo:


```
{
 "data":[
  {
   "id":237,
   "first_name":"LeBron",
   "last_name":"James",
```

```
    "position":"F",
    "team":{
      "id":14,
      "abbreviation":"LAL",
      "city":"Los Angeles",
      "conference":"West",
      "division":"Pacific",
      "full_name":"Los Angeles Lakers",
      "name":"Lakers"
    }
  }
  ...
],
"meta": {
  "total_pages": 50,
  "current_page": 1,
  "next_page": 2,
  "per_page": 25,
  "total_count": 9999
 }
}
```

Get a Specific Player

Este endpoint devuelve un jugador específico

HTTP Request

GET /players/<ID>

Parámetros de consulta:

ID – El ID del jugador a devolver

curl /players/237

El anterior comando retorna un archivo JSON estructurado como el siguiente:


```
{
 {
  "id":237,
  "first_name":"LeBron",
  "last_name":"James",
  "position":"F",
  "team":{
   "id":14,
   "abbreviation":"LAL",
   "city":"Los Angeles",
   "conference":"West",
   "division":"Pacific",
   "full_name":"Los Angeles Lakers",
   "name":"Lakers"
  }
 }
```

```
}
```

iii. Equipos

Get All Teams

'curl "/teams"'

El anterior comando retorna un archivo JSON estructurado como el siguiente:

```
{
 "data": [
   {
     "id":14,
     "abbreviation":"LAL",
     "city":"Los Angeles",
     "conference":"West",
     "division":"Pacific",
     "full_name":"Los Angeles Lakers",
     "name":"Lakers"
   },
   ...
 ],
 "meta": {
   "total_pages": 1,
   "current_page": 1,
   "next_page": null,
   "per_page": 30,
   "total_count": 30
 }
}
```

Este endpoint devuelve todos los equipos de la actual temporada.

HTTP Request

GET teams

Parámetros de consulta:

page – Número de páginas. Usado para paginación.

per_page - El número de resultados retornados por llamado. Usado para la paginación.

Get a Specific Team

curl "/teams/14"
El siguiente comando retorna un archivo JSON como el siguiente:


```
{
 "id":14,
 "abbreviation":"LAL",
 "city":"Los Angeles",
 "conference":"West",
 "division":"Pacific",
 "full_name":"Los Angeles Lakers",
 "name":"Lakers"
}
```


Este endpoint devuelve un equipo específico

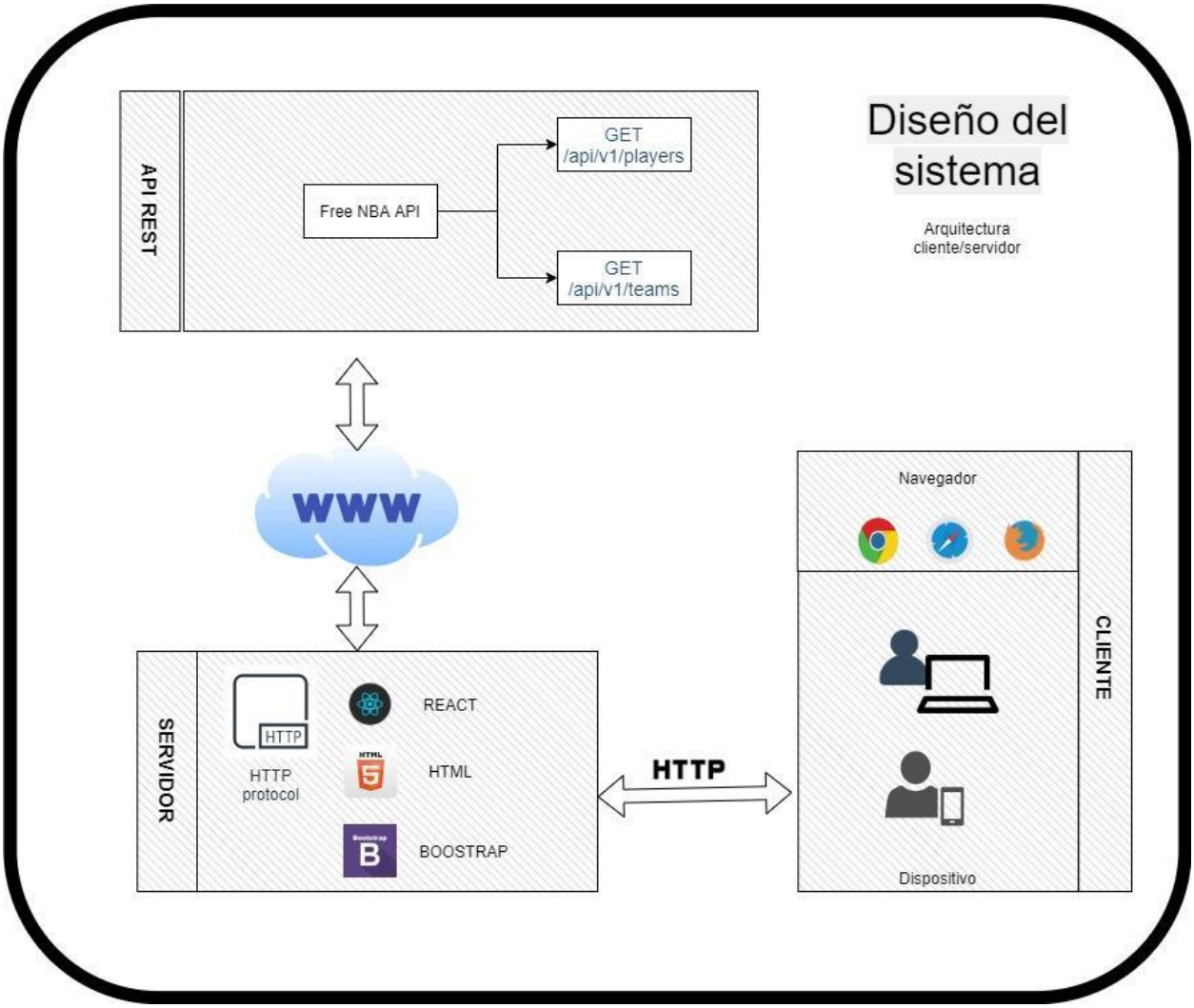
HTTP Request

GET /teams/<ID>

URL Parameters:

ID – El ID del equipo a devolver

2. Diseño del sistema:



3. Manual del usuario

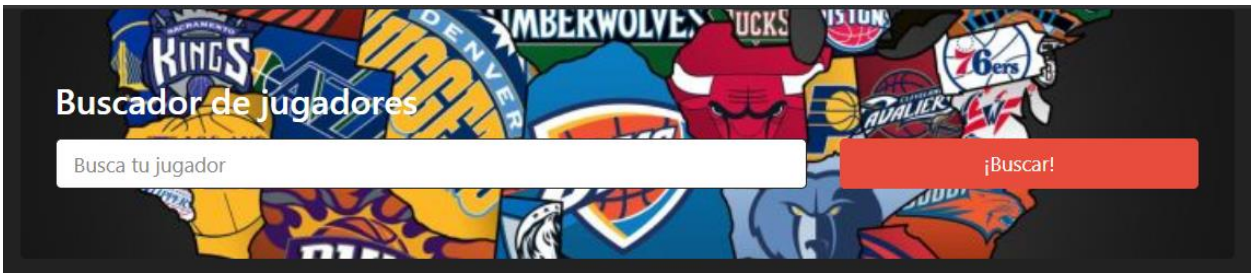
La aplicación cuenta de una sola página donde se encuentra el buscador.

Esta decisión fue tomada para poder explotar las funcionalidades de react y desplegar toda la información desde una única pantalla.

Como veremos más adelante la información consumida de de la API Rest es desplegada en primera instancia en la página principal y luego en Modal de boostrap.

Desplegar todos los jugadores

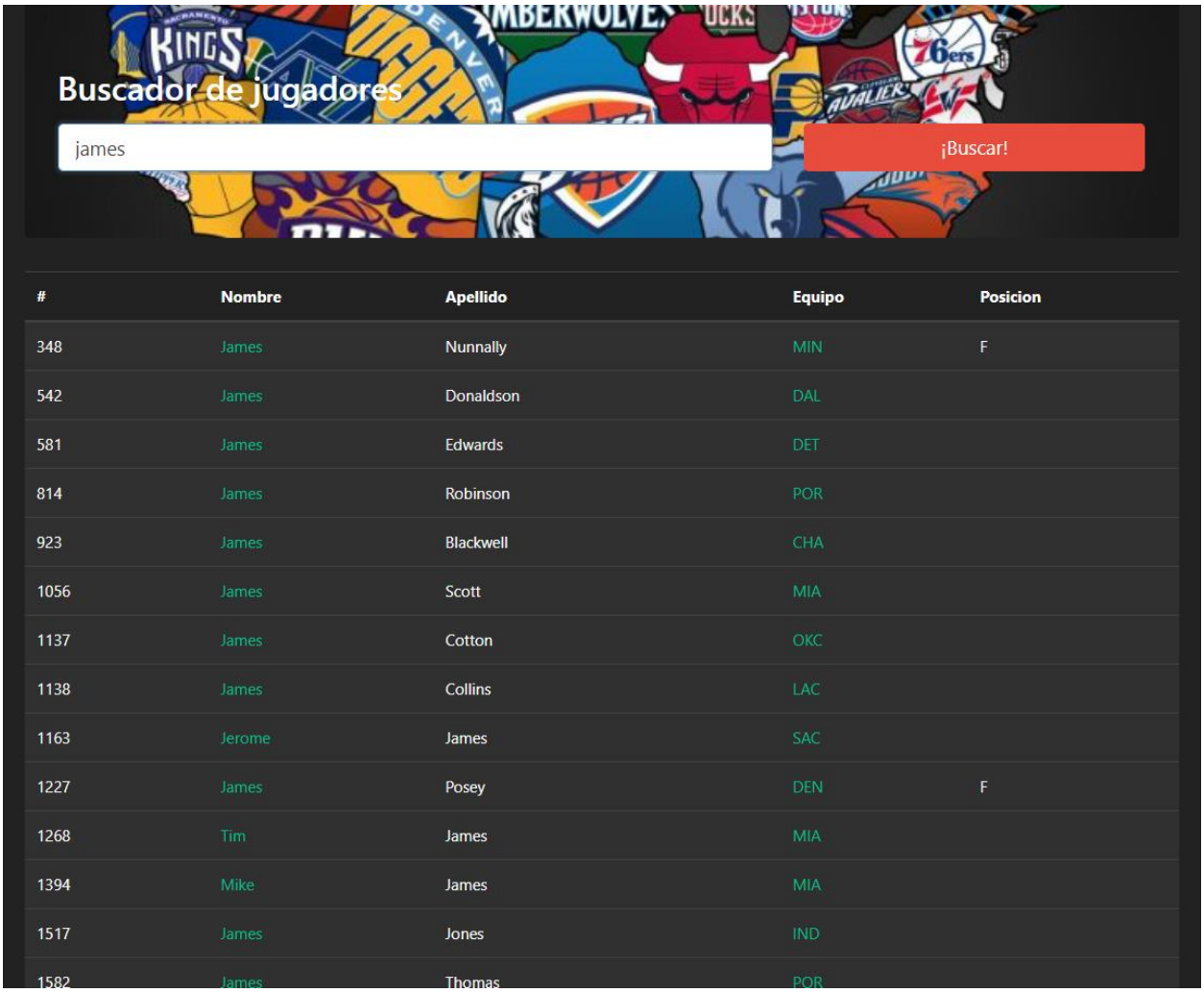
Hacer click sobre el botón “¡Buscar ahora!” y se desplegarán todos los jugadores. Por cada jugador desplegado puedo hacer click sobre su nombre para obtener detalles del jugador o hacer click sobre el nombre del equipo abreviado (por ejemplo Indiana Parcers es IND) y se desplegará información sobre el equipo.



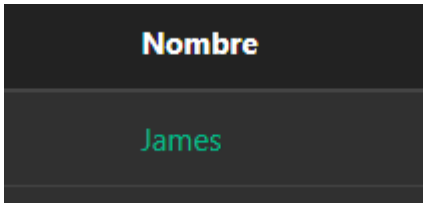
Luego de ingresar el valor por el cual queremos buscar y clickear “¡Buscar!”, se cargaran los resultados en forma de tabla.

Buscar por una condición de nombre

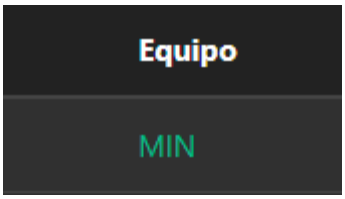
Para buscar por un jugador en particular podemos usar la búsqueda por string. Consiste en digitar en el buscador nombre o apellido del jugador solicitado y el sistema desplegará todos los que cumplan con la condición.



Dentro de tabla tenemos dos secciones para obtener más información, si clickeamos sobre el nombre del jugador



o si clickeamos sobre la sigla del equipo.



En cada caso se desplegará un Modal con la información correspondiente, el Modal es reutilizado según la opción seleccionada.