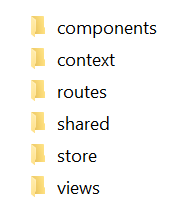
## Organización

Como se ha mencionado anteriormente, el lado **cliente** de la aplicación ha sido desarrollado con el framework **React.js**. El código de este queda organizado de la siguiente forma:

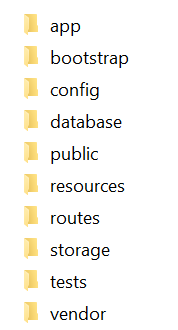


*Imagen 1*

* **Components:** Dentro están todos los componentes de la aplicación. Se entiende como componente un fragmento de código que se va a repetir en varias vistas, como por ejemplo un botón con un estilo predeterminado o una ventana modal con una funcionalidad implementada que tendrá el mismo funcionamiento en distintas vistas.
* **Views:** Aquí se encuentran todas las vistas de la aplicación. Cada una contiene dos archivos: el que contiene la estructura y las funcionalidades; y el otro que contiene los estilos.
* **Routes:** Donde se reúnen todos los archivos relacionados con las rutas de la aplicación, tanto objetos para crear las rutas como para configurar los menús.
* **Shared:** Aquí se encuentran estilos, variables y funciones que se van a utilizar a lo largo de todas las vistas de la aplicación.
* **Context y store:** En estos directorios se monta una estructura para almacenar las peticiones que se van a realizar al servidor, los datos que se reciben; y tener estos datos organizados en secciones.

Una vez desarrollado todo el código cliente, se compila se generará un fichero el cual se deberá añadir al servidor.

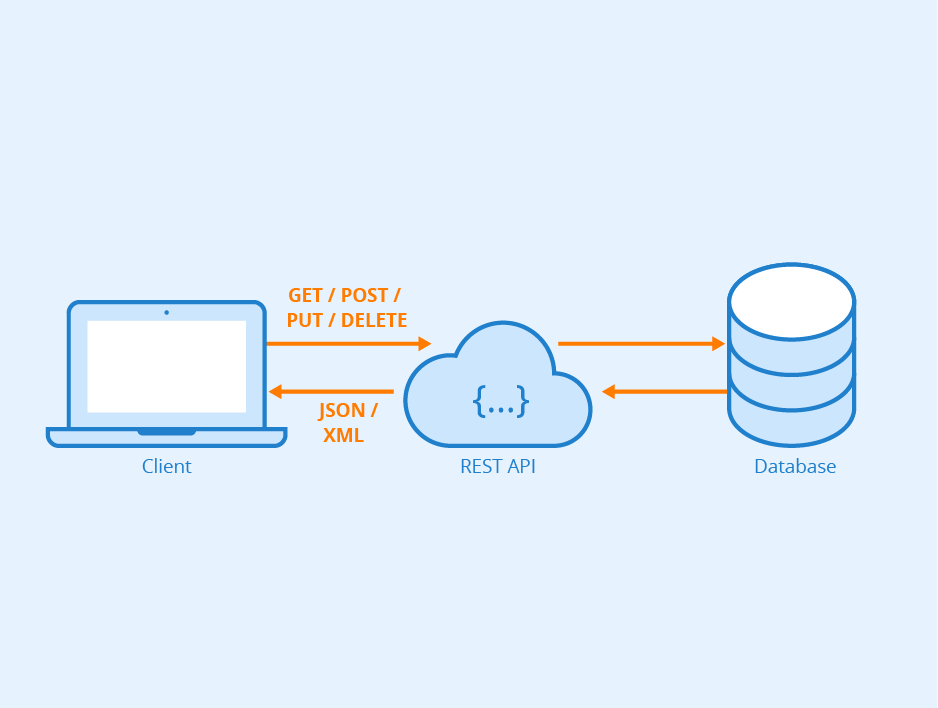
Luego, la organización del servidor es la siguiente:



*Imagen 2*

* **App:** Dentro están los controladores con los métodos pertenecientes a cada ruta.
* **Database:** Aquí se almacenan todas las migraciones que se ejecutarán para crear las tablas de la base de datos.
* **Resources:** Contiene el fichero generado una vez compilado el cliente que muestra la vista de la aplicación.
* **Routes:** Contiene un fichero con todas las rutas a las cuáles se harán las peticiones.
* **Storage:** Dentro se almacenan todas las imágenes empleadas en la aplicación, tanto estáticas como las que se subirán al servidor.

Para la obtención de los datos en el lado cliente, desde este se envían las **peticiones http** (con parámetros o no) a las **url** programas en el servidor. En él se realiza la conexión con la base de datos mediante **MySQL** para realizar las consultas y buscar los datos ahí; y una vez obtenidos estos, el servidor los envía al cliente en un **JSON**, funcionando así como una **API**.



## Funcionalidades

En la aplicación han sido implementadas las siguientes funcionalidades:

* Validación de formularios.
* Pantalla de carga mientras se realizan las peticiones.
* Registro de usuarios almacenando la contraseña hasheada.
* Login de usuarios con un sistema de token identificando al usuario logueado.
* Sistema de roles y permisos.
* Control de las vistas para usuarios logueados o administradores.
* Creación del club del usuario.
* Implementación de un mensaje de notificación para informar al usuario de errores o de acciones con éxito.
* Middlewares para controlar si el usuario tiene los permisos necesarios para ejecutar una acción.
* Menú lateral desplegable.
* Migas de pan para volver a secciones anteriores con más facilidad.
* Botones para volver a la sección anterior.
* Paginación.
* Cambiar el nombre y el escudo del club.
* Borrar el club.
* Insertar, eliminar y modificar permisos.
* Insertar, eliminar y modificar jugadores.
* Insertar, eliminar y modificar países.
* Insertar, eliminar y modificar ligas.
* Insertar, eliminar y modificar equipos.
* Insertar, eliminar y modificar tipos de cartas.
* Insertar, eliminar y modificar cartas.
* Insertar, eliminar y modificar sobres.
* Filtrar jugadores.
* Vender un jugador a cambio de monedas.
* Abrir sobres en los cuales salen una cantidad de jugadores, programado con un algoritmo el cual a mayor media, mayor será la probabilidad que salga. El jugador tendrá que pagar una cantidad de monedas para abrirlo, y los jugadores que salgan los podrá guardar en su club.
* Crear plantillas con los jugadores del club.
* Modificar el nombre de una plantilla.
* Borrar una plantilla.
* Colocar jugadores en la plantilla.
* Quitar jugadores de la plantilla.
* Intercambiar jugadores de una posición a otra.
* Jugar al draft para ganar monedas.