
This is the **published version** of the bachelor thesis:

Estupiñan Caicedo, Jhon Freddy; Roca i Marvà, Francesc Xavier, dir. Tikitapp.
2022. (958 Enginyeria Informàtica)

This version is available at <https://ddd.uab.cat/record/264177>

under the terms of the  license

TikitApp

Jhon Freddy Estupiñan Caicedo

27 de junio de 2022

Resum– En el fútbol base participan jóvenes aspirantes a profesionales de entre 6 y 18 años. En esta etapa y por diversas razones, es muy común que los jugadores quieran cambiar de club hasta encontrar comodidad y estabilidad en un equipo. A diferencia de las ligas profesionales, los traspasos de jugadores en las ligas modestas se producen a menor escala debido al elevado coste que supone la contratación de ojeadores, encargados de detectar nuevos talentos para los equipos, y de agentes, encargados de representar a los jugadores, negociar los contratos y actuar como mediadores entre sus clientes y los clubes.

En este contexto, entra en juego el proyecto TikitApp, que pretende resolver este problema. Consiste en una aplicación web donde los nuevos talentos y los nuevos clubes pueden encontrarse y comunicarse entre sí en caso de interés mutuo.

Paraules clau– fútbol, fútbol base, entrenadores, clubs, jugadores, comunicación.

Abstract– Grassroots football involves young aspiring professionals between the ages of 6 and 18. At this stage and for various reasons, it is very common for players to want to move between clubs until they find comfort and stability within a team. In contrast to the professional leagues, player transfers in lower leagues occur on a smaller scale due to the high cost of hiring scouts, who are responsible for scouting new talents for the teams, and agents, who are responsible for representing players, negotiating contracts and acting as the mediator between their clients and the clubs.

In this context, the TikitApp project comes into play, which aims to solve this problem. It consists of a web application where new talents and new clubs can be found and communicate with each other in case of mutual interest.

Keywords– football, grassroots football, scouts, clubs, football players, communication.



1 INTRODUCCIÓN

EN la actualidad, el fútbol es uno de los deportes más populares en España y en la mayoría de los países del mundo, con lo cual, es normal que los niños desde temprana edad quieran formarse en este deporte con el objetivo de llegar a ser profesional.

Originalmente en el fútbol solamente existían competiciones compuestas por jugadores adultos, después de la época de los 50 los clubs empezaron a incorporar un equipo filial a su organización, el filial era una plantilla alternativa en el cual se formaban los futuros profesionales para club.

Al pasar de los años los equipos fueron incrementando las plantillas alternativas hasta formar una escuela de fútbol, donde sí incluían jugadores de edades menores, a día de hoy, las escuelas de fútbol que comprende jugadores entre los rangos de edades de los 6 y 18 años se conoce como fútbol base, etapa donde los integrantes mejoran la técnica,

la táctica, desarrollan al máximo sus habilidades y sobre todo los valores como futbolista. En los clubs modernos es una pieza fundamental, los equipos más pudientes invierten en personal altamente cualificado e infraestructuras avanzadas para gestionar de manera óptima sus diferentes plantillas, sin embargo hay otros que no tienen tantos recursos [15].

El fútbol base además reparte su estructura en diferentes categorías, divididas por la edad de los jugadores, y cada categoría cuenta con diferentes divisiones de ligas competitivas, en las que se posicionan según el rendimiento grupal del equipo durante las temporadas.

Hay muchas maneras de dirigir una plantilla, influyen las habilidades e ideología del cuerpo técnico que esté a cargo, con lo que los entrenadores de los equipos, tienen un estilo de juego concreto en los que saben qué tipo de jugadores necesitan en su plantilla, buscan cualidades específicas en los jugadores para las diferentes posiciones del campo. En el ámbito de fútbol base en España, al contrario que en el fútbol profesional, completar una plantilla con nuevos talentos puede resultar una tarea difícil para la mayoría de equipos, ya que no existe ningún sistema asequible en el que se puedan encontrar y contactar con jugadores dispuestos a ‘cambiar de aires’.

En este documento se muestra el desarrollo de un proyecto

• E-mail de contacto: jhonfreddy.uab@gmail.com
 • Mención realizada: Ingeniería del Software
 • Trabajo tutorizado por: Xavier Roca (departamento de Ciencias de la Computación)
 • Curso 2021/22

que pretende solucionar el problema comentado, se trata de una aplicación web que servirá como repositorio de jugadores de fútbol base, en la cual se podrán registrar entrenadores de fútbol y jugadores, y ambos tendrán la posibilidad de comunicarse entre ellos en caso de que estén interesados mutuamente. En primer lugar, se explican los objetivos del proyecto, se describe qué resultados que se esperan al finalizar el plan. El estado del arte, donde se investiga el mercado actual tecnológico en el ámbito del fútbol. Seguidamente, la metodología usada en el proyecto y el proceso de desarrollo. Por último, se explicarán los resultados obtenidos, las pruebas realizadas y se resaltarán las conclusiones.

2 OBJETIVOS

2.1. Objetivo general

El objetivo general de este proyecto es mostrar el desarrollo de una aplicación web, tendrá la posibilidad de registro de dos tipos de usuarios, jugador y club, los cuales, una vez dados de alta en la aplicación, en su sesión, puedan consultar otros jugadores o clubes que hayan dados de alta, según el rol de su cuenta, éstos inicialmente mostraran solo datos públicos que servirán de orientación para saber si mostrar interés o no, en caso de interés mutuo poder abrir vías de comunicación. En la aplicación se desarrollará una lógica específica para determinar ese interés de ambas partes. La aplicación también deberá disponer de un espacio donde los usuarios puedan ver y modificar sus datos como el nombre, foto de perfil, etc. Y por último un espacio buscador donde los usuarios de perfil 'club' puedan filtrar la lista de jugadores por diferentes entradas.

2.2. Objetivos específicos

Configurar una base de datos correctamente creada con los atributos necesarios y que se logren guardar de manera que cumplan los tres principios de la seguridad, la integridad, confidencialidad y consistencia de los datos.

Una interfaz intuitiva en la cual los usuarios no experimentados tengan la facilidad de navegar.

Implementar la aplicación de manera que se pueda conseguir un buen rendimiento en cuanto a la fluidez de la aplicación y tiempos de respuestas rápidos en la interacción con el usuario.

2.3. Motivaciones

El principal motivo de la elección de este proyecto es poder aportar y ayudar en el sector del fútbol, le tengo especial interés ya que a lo largo de mi etapa adolescente participé en ligas formativas de fútbol base, y a día de hoy mi entorno personal está involucrado en él, tengo familiares y amistades que trabajan en ese sector, y se de primera mano los problemas de estancamiento de jugadores por diversas razones, y por parte de los entrenadores la dificultad de conseguir nuevos jugadores para mejorar sus plantillas. Son problemas que existían en mi periodo de jugador de escuela y siguen existiendo.

3 ESTADO DEL ARTE

La estabilidad, comodidad y otros factores psicológicos de un jugador dentro de un equipo, es fundamental para el desarrollo de sus habilidades, también para las demandas del entorno de este deporte, sin esta condición psicológica es fácil que el jugador pierda motivación y por lo tanto, no rinda de manera óptima en las competencias[16].

Los equipos de fútbol profesional participan en competiciones donde se juegan la reputación, el descenso a una liga inferior y en consecuencia la estabilidad económica, con lo cual tienen muy en cuenta estos factores anímicos de su plantilla, ya que necesitan el máximo rendimiento de sus jugadores para competir. Los cuerpos técnicos están constantemente trabajando, para optimizar su equipo, buscando nuevos talentos para reforzar su plantilla, y no les resulta tan difícil ya que se invierten parte de los fondos generados en profesionales que se alinean juntamente con la directiva del equipo para llevar a cabo esta actividad de optimización constante. En el caso de los futbolistas profesionales, también invierten parte de sus fondos en contratar a mediadores que ayudan a representarlos antes otros clubs, y buscan continuamente cubrir las necesidades anímicas del jugador en cuantos a estabilidad y confort.

En el fútbol base, las necesidades no son diferentes, por parte de los entrenadores también necesitan mantener su plantilla optimizada, y los jugadores necesitan un buen estado anímico. En la mayoría de los equipos, estas actividades pueden ser no viables, tienen altos costes y en sus divisiones participan en competiciones donde no se producen los ingresos necesarios para la inversión de los profesionales adecuados. Los traspasos de jugadores, en las ligas más modestas de fútbol base se produce a menor escala, puesto que los jugadores no tienen la facilidad de comunicarse con otros clubs, y estos no saben de la existencia de futbolistas dispuestos a un 'cambio de aires'. Actualmente las prácticas que realizan los entrenadores y jugadores para cubrir sus necesidades no son tan efectivas, uno de estos métodos es recurrir a la búsqueda de conocidos propios o conocidos de amigos, esta actividad no es óptima ya que el alcance es un muy cercano, existen muchos jugadores y muchos clubs, y sería más conveniente contar con un sistema que permita ampliar más el rango de posibilidades.

En el ámbito de aplicaciones enfocadas al mundo del fútbol, y que las ideas son similares, existe una aplicación, llamada Transfer Room, el entorno de fútbol base está totalmente al margen debido a su alcance ya que solamente aplica al ámbito del fútbol profesional, y su alto coste de suscripción, esta aplicación está dirigida a cuerpos técnicos de alto nivel y les permite entre otras cosas "comparar jugadores, observar sus estadísticas, analizar su trayectoria, mostrar interés, realizar ofertas o poner a la venta jugadores." [11]. Existe otra aplicación, AiScoutApp, es una aplicación con una base de datos de futbolistas de medio y alto nivel, y mediante una serie de inputs por el usuario, combinado con servicios de inteligencia artificial, la aplicación intenta buscar jugadores perfectos según los filtros de entrada[12].

Actualmente no hay un sistema que cubra las necesidades que se plantean en este proyecto, lo cual lo hace una propuesta atractiva.

4 MÉTODO DE DESARROLLO

Para llevar a cabo el proyecto correctamente es necesario aplicar una metodología que se adecúe al tipo de proyecto, aparte de una buena planificación de trabajo, en el cual queden definidas y desglosadas las tareas a realizar, con su respectiva previsión de tiempo. El desarrollo completo del proyecto se puede dividir en tres partes principales:

- La documentación continua del proyecto y el diseño de la aplicación.
- La creación de la base de datos y Back End, la interfaz de usuario Front End, la lógica de los dos tipos de usuarios y la integración de todas las partes anteriores.
- Las pruebas y las mejoras continuas que sean necesarias en la aplicación ya construida.

4.1. Metodología

La metodología es, por definición, el conjunto de métodos, técnica y herramientas que se usan para llevar a cabo un proyecto, en el ámbito del desarrollo de software se puede entender como la guía definida a seguir desde que nace el proyecto hasta que se consiguen los resultados esperados, todo en un marco de tiempo acordado. Se ha considerado, por experiencia, usar una mezcla entre una metodología ágil incremental iterativa y Scrum, ya que es la que se adapta con facilidad a proyectos relacionados con el desarrollo de aplicaciones de este estilo.

Una metodología ágil es la que nos permite realizar proyectos de manera iterativa en sprints, en los cuales se definen una serie de tareas a realizar y se llevan a cabo con periodicidad definida, normalmente cada sprint dura una semana; también nos permite ver el crecimiento del proyecto poco a poco y soporta variaciones en pequeñas características del sistema respecto la definición inicial, aunque recomendable que sean mínimas para evitar elevar el riesgo de fracaso del proyecto.

En la figura 1, se ilustra las partes específicas de esta metodología Scrum.

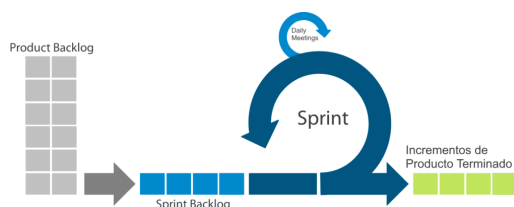


Fig. 1: Scrum

- **Product Backlog:** Es una lista de las funcionalidades que se han definido del producto demandado, y ordenada según la prioridad.
- **Sprint Backlog:** Se seleccionan algunas de las funcionalidades del Product Backlog y se marca el tiempo en que van a ser desarrolladas.

- **Daily Meeting:** Es la reunión diaria, y se realiza para ver el estatus del proyecto, y sirve para tomar decisiones rápidas sin tener que perder demasiado tiempo.
- **Sprint:** Es el desarrollo en sí de lo que se ha decidido en el Sprint Backlog
- **Incremento del producto:** Es el avance esperado funcional, o no, del producto en desarrollo.

Hay un concepto que no está reflejado en la figura 1, pero es igual de importante que el resto. El **Sprint Retrospective**, Reunión para analizar los objetivos cumplidos, ver si existen fallos, y analizarlos para no cometer los mismos errores en el futuro. Este punto también sirve para ver la implementación de posibles mejoras y/o cambios respecto a lo definido en los objetivos[1].

5 DESARROLLO DEL PROYECTO

La idea del proyecto, como se menciona en el apartado 2.1, es crear una aplicación web, que cuente con una base de datos que soporte los tipos de usuarios jugador y club, que puedan darse de alta para mostrarse, y puedan contactar entre ellos en caso de interés mutuo. La aplicación será del tipo cliente-servidor, constará de un login para registrar los datos de los usuarios, diferentes tipos de perfil (club, jugador), y una base de datos donde se puedan almacenar los datos de los usuarios.

5.1. Descripción

El recorrido lógico que describe el concepto principal de la aplicación creada es el siguiente:

Al abrir la aplicación en un dispositivo, se debe iniciar sesión con usuario y contraseña, habrá la opción de registro en caso de que el usuario no se haya registrado en la aplicación antes; a la hora de darse de alta, habrá que elegir con qué tipo de rol se desea registrar, jugador o club, si se decide un alta con rol jugador, aparecen los campos necesarios para crear un perfil jugador, y si se decide un rol club, aparecerán los campos a rellenar propios de un club. Una vez dado de alta, si el usuario es del rol club, se mostrará en la vista principal, una lista de jugadores aleatorios y la posibilidad de filtrarlos por diferentes métodos de entrada, como nombre, o posición, los campos que tendrán los tipos de usuarios en la BBDD se comentan en el apartado 5.4. Si el usuario es del rol jugador, se mostrará pues una lista de clubes aleatorios en los cuales se pueda interesar. El mapa de estados de la aplicación se ilustra en el apartado A1 del apéndice.

Finalmente, el punto clave de la aplicación, si hay fijación en alguna perfil de la lista, se puede entrar a ver el perfil detallado y ver algunos de los datos, los públicos, y en caso de interés mutuo, es decir, si de casualidad ambos se "siguen clicando un botón", "siguiendo al usuario", se podrán ver el resto de los datos, los datos de contacto, como número de teléfono, ya que no es público a primeras, Se podrán contactar por las vías de comunicación que el usuario tenga disponible (whatsapp, Gmail, etc.) .el

esquema lógico del concepto interés mutuo se detalla en la figura 2.

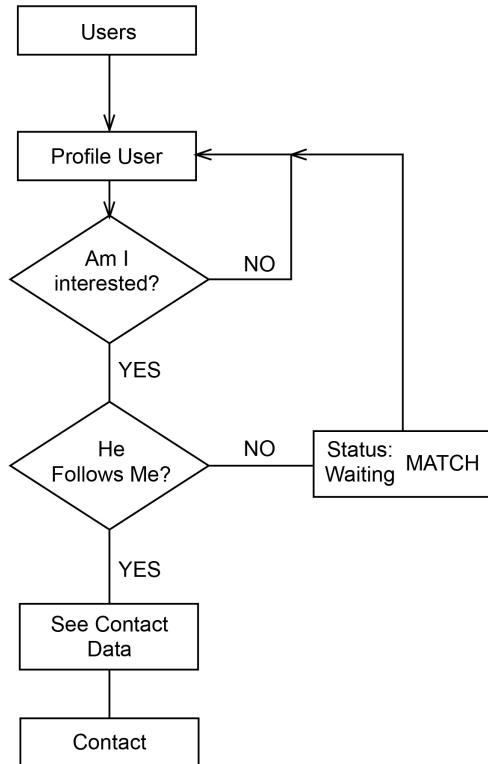


Fig. 2: Lógica follow

Independientemente del modo que se elija, se podrán modificar datos personales, como la posición de juego, ciudad de domicilio, etc; La aplicación contará con un apartado de "Mis intereses", donde se mostrará la lista de usuarios en los cuales ha habido interés. También dispondrá de un apartado de búsqueda para filtrar el total de lista según varios aspectos, como el nombre, edad, posición de juego, solamente para los perfiles con rol 'club', esto es porque la relación número de jugadores - clubes tiene una proporción media de 25- 1, se calcula con el número de jugadores que compone una plantilla normalmente.

5.2. Casos de uso

Los casos de uso definen los escenarios en los que la aplicación ejecuta su funcionalidad, la manera en las que se representan en este proyecto es a través de un diagrama de casos de uso, donde se muestra el actor (en este caso jugador/club), las acciones que puede realizar, y cómo se relacionan entre ellas. Se puede ver en la figura 3

5.2.1. Diagrama casos de uso

En el diagrama se usan las relaciones **include**, muestra que un caso de uso estrictamente necesita que se ejecute otro. por ejemplo: si no estoy logueado, no puedo consultar la lista de jugadores que existan. Y la relación **extends**, muestra que un caso de uso no necesita estrictamente que se ejecute otro. por ejemplo: puedo o no "seguir" a un jugador el cual estoy interesado en su perfil.

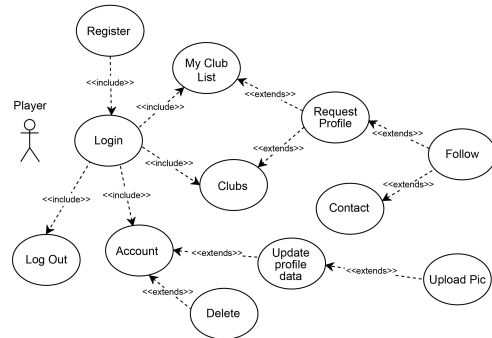


Fig. 3: Diagrama casos de uso

5.3. Arquitectura de la aplicación

La arquitectura del proyecto sigue el paradigma de aplicación web moderna, Full Stack. Compuesta por un Back End conectado a una BBDD de Mongo, y hace el papel de servidor, se encarga de recibir consultas, tratar datos y servirlos; la otra parte es el Front End que es la parte que se ejecuta en un navegador web, y es la encargada de mostrar los datos al cliente y ejecutar las acciones que estén implementadas. Una ilustración de ésta arquitectura se puede ver en la figura 4.

5.3.1. Back End

Para el Servidor del Back End, se ha creado una aplicación API REST en Nest JS. Nest es un marco para crear aplicaciones de servidor, y aunque permite el desarrollo en JavaScript, tiene todo el soporte principal para el desarrollo en TypeScript[5].

Para facilitar tanto el escalamiento como la unión con el Front End, se ha respetado la arquitectura estándar de Nest JS, está compuesta por:

- **Modulo:** Es donde se define el decorador @Module de Nest y es la raíz de la aplicación, sirve para montar la estructura del proyecto.
- **Controlador:** Es donde se definen las rutas de la aplicación, cada ruta es un end point. Es el responsable de manejar las solicitudes entrantes http request y devolver la respuesta al cliente.
- **Servicio:** La idea principal de un servicio es que se puede inyectar como una dependencia, es donde están creadas las funciones para tratar los datos a devolver si es necesario.

Se ha elegido Mongo por su agilidad y versatilidad, cualidades que pueden llegar a ser notables para cumplir

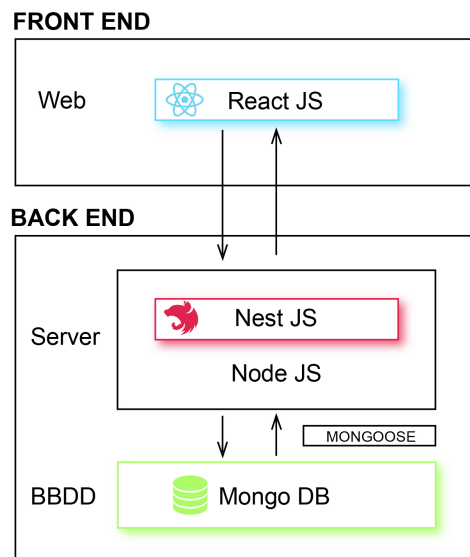


Fig. 4: Arquitectura del proyecto

uno de los objetivos del proyecto, la velocidad y la fluidez de la aplicación; También Mongo, en comparación con otras bases de datos, es mucho más escalable y flexible, lo cual permite la inclusión de nuevas funcionalidades y pequeños cambios al proyecto.

En cuanto a la creación de la base de datos, el diseño fue simple ya que la parte fuerte se centra en la comunicación, listado y filtrado de usuarios, por lo tanto se decidió configurar una colección donde guardar los datos de los usuarios en MongoDB.

Se decidió crear una tabla en MongoDB, en la cual se guardarán los dos tipos de usuarios, se usó una única tabla para aprovechar la flexibilidad de Mongo, ya que en una misma colección se pueden guardar diferentes tipos de registros, y se pueden manejar como objetos. En este caso, los usuarios tipo jugador y club son parecidos, el hecho de que les diferencia, entre otras, es el rol. Por ejemplo, si se desea obtener la lista de usuarios con el rol player, se puede crear un servicio en el Back End, y recuperar una lista de usuarios con esa condición tratándola como una lista de objetos.

Los campos que se usan tanto de jugador como clubes, se pueden ver en las figuras 5, 6 y 7.

5.3.2. Front End

Se ha decidido realizar el proyecto con el Front End de React JS. React JS es una de las librerías más utilizadas según sus estadísticas en GitHub, ya que es un framework muy flexible, a la vez que escalable. Otra de las ventajas es que si se quisiera migrar una aplicación web a una aplicación nativa, se podría reutilizar todo, o gran parte del código. El crecimiento de esta tecnología es casi inevitable ya que respecto a las competencias tiene mucho que

User Schema		C	P
K	_Id		
	Username	x	x
	User Last Name		x
	Role	x	x
	Email	x	x
	Password	x	x
	Born Date		x
	Profile Image	x	x
	Phone Number Player		x
	Height (cm)		x
	Weight (Kg)		x
	Player City		x
	Actual Club		x
	Position		x
	Following []	x	x
	Profile Image []	x	x
	Goals Number		x
	Assists		x
	Played Matches		x
	Actual Club Since		x
	Foundation Date	x	
	Phone Number Club	x	
	Address	x	
	Club City	x	
	Titles Number	x	
	Category Team	x	
	Division League	x	
	Players Team Number	x	
	Trainer Name	x	

*K = Key
*C = Club
*P = Player

Fig. 5: Campos en la colección MongoDB

```

{
  "_id": ObjectId("62b8a20ab37efeb365727ddd"),
  "userName": "Martorell FC",
  "role": "Club",
  "email": "martorell18fc.com",
  "password": "52b5105v8q8.2zi6icawfA.ndtX1eeB00R4mJ7nKRiNdfrdruXR21D0paeQqe",
  "following": Array,
  "0": "62b89be7c2c27ade282284fe",
  "foundationDate": 2010-03-28T22:00:00.000+00:00,
  "profileImage": Array,
  "0": Object,
  "fieldName": "file",
  "originalName": "favicon.ico",
  "encoding": "7bit",
  "mimeType": "image/x-icon",
  "buffer": "AABAAQAEBAAAAEAIAADjAGAAAgAAABgYAAABACAAADAMAAKCAAAAgIAAAAGAgAdkDAAA1BQ..",
  "size": 3870,
  "phoneNumberClub": 622622622,
  "address": "943 Merritt Knolls",
  "clubCity": "Martorell",
  "titlesNumber": 19,
  "categoryTeam": "Infantil",
  "divisionLeague": "Segunda división Catalana",
  "playersTeamNumber": 15,
  "trainerName": "Manuel",
  "__v": 0
}

```

Fig. 6: Usuario club en BBDD

ofrecer, y día a día van mejorando funcionalidades.

Dentro de la aplicación del Front End se han usado diferentes librerías, tanto de React como externas para conseguir el total de las funcionalidades que se necesitan.

- **Axios:** es la librería que permite hacer consultas a una API, ejecutándose desde el navegador, y específicándole una ruta hacia un end point.

```

_id: ObjectId('62b89be7c2c27ade28284fe')
userName: "jose"
userLastName: "Smith"
role: "player"
email: "john@smith.com"
password: "5b103yyEzbTfJlTIPoc2mw9Cime3b6mo58eYNhfgHwobIb6imy8Noot2Ie"
following: Array
phoneNumberPlayer: 600077219
playerCity: "Barcelona"
goalsNumber: 7
assists: 50
playedMatches: 27
actualClub: "Terrassa FC"
actualClubSince: 2020-03-17T23:00:00.000+00:00
bornDate: 1995-12-17T23:00:00.000+00:00
position: "Delantero"
profileImage: Array
  0: Object
    fieldName: "file"
    originalName: "favicon.ico"
    encoding: "7bit"
    mimeType: "image/x-icon"
    buffer: "AAABAAQAEBAABAAEATADJAQAARgAAABgYAAABACAAADAAACAAACAAAgIAAAAGAgADKAAA1BQ..."
    size: 3870
  v: 0
height: "170"
weight: "85"

```

Fig. 7: Usuario player en BBDD

- **Componentes:** Son los elementos que se pueden crear en una aplicación React, se pueden reutilizar y se le puede dar comportamiento y estilo según las necesidades del entorno. Aunque React permite la creación de aplicaciones *“one page web”*, en este proyecto se ha decidido hacer uso de los componentes para aprovechar al máximo las ventajas que el framework ofrece en cuanto a la reutilización y limpieza de código.
- **Hooks:** Los hooks en React son una característica que han incluido últimamente y ayudan a solucionar problemas comunes dentro de los componentes, por ejemplo, si se desean modificar datos de un componente en tiempo de ejecución, se puede hacer gracias al hook estado, que permite guardar información temporal. Se ha hecho uso de los hooks en casi todas las partes de la aplicación.
- **Auth Provider:** es una librería de React JS que sirve para darle seguridad a la aplicación, en este proyecto se ha utilizado como punto centralizado de manejo de los datos, se ha implementado una función componente de rutas protegidas que pregunta al *'Auth Provider'* si el usuario está dentro de una sesión que le permita navegar a través de la aplicación.
- **React Router:** Se ha utilizado esta librería para definir las rutas de la aplicación de esta manera es mas ordenado aplicar la seguridad a la aplicación, ya que se puede gestionar desde aquí las rutas protegidas y las que no, aparte de propinar datos a solamente componentes específicos.

5.4. Paper prototyping

Se ha realizado la técnica del paper prototyping, es una técnica de baja fidelidad, pero con mucho potencial, ya que es hacer un diseño prototipo de la aplicación, en papel, lo cual no lo hace muy fiable, pero es una manera de obtener un feedback de un público (quien tú decidas) de manera rápida sobre un prototipo de aplicación[3].

En este caso, se realizó la encuesta a 7 personas, en 3 iteraciones se consiguieron las vistas de la aplicación que figuran en el apéndice, debido a varios problemas y un cambio de concepto, hicieron que no fueran las finales en la parte del menú y el contacto, y aunque no fueron las definitivas, el

paper prototyping es una técnica que ha servido como guía para las vistas actuales. El resultado de éste método utilizado se pueden ver en el apartado A2 del apéndice. Para las vistas de la aplicación finalmente se siguieron patrones de diseño actuales de una aplicación React, se pueden ver algunas en el siguiente apartado.

6 RESULTADOS

6.1. Funcionalidades

El resultado es el propio desarrollo de la aplicación y se ha conseguido implementar una versión funcional que se puede ejecutar a nivel local. A continuación se muestran las diferentes funcionalidades de la aplicación y su demostración gráfica.

Login/logout y Sesión: Los usuarios pueden loggarse, y desloggarse simultáneamente, siempre y cuando se hayan registrado previamente. La sesión se consigue con el renderizado condicional de los componentes, y esto permite que, según estés *logueado*, o no, se puedan ver ciertas partes de la aplicación. Por seguridad, se ha implementado también una protección de acceso por ruta, donde interviene el navegador, al iniciar sesión se guarda un token que permite navegar por la aplicación, el token expira en 4 horas, o si por lo contrario el usuario decide cerrar sesión, en ese caso la aplicación se redirige a la pantalla de Login/Home.

Fig. 8: Home

Registrar, y modificar usuarios: Desde la aplicación web un usuario puede registrarse, y modificar campos de su cuenta, sea rol jugador o club.

Consultar lista de usuarios y consultar un perfil: Un usuario dado de alta en la aplicación puede ver la lista de usuario, según el rol de su cuenta, y también puede consultar un perfil específico del listado dando clic en el nombre.

Concepto follow: El concepto follow que se comenta en el apartado 5.1, está también implementado. Cuando un usuario visita un perfil por primera vez, puede ver la información general pública, entonces puede mostrar interés haciendo clic en el botón *“follow”* pueden ocurrir dos cosas. Que el usuario consultado no se haya interesado por el

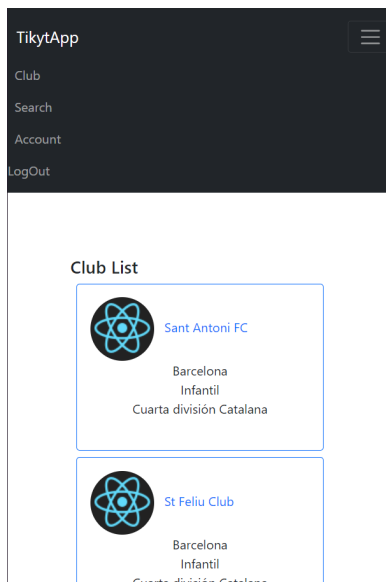


Fig. 9: Menú logueado

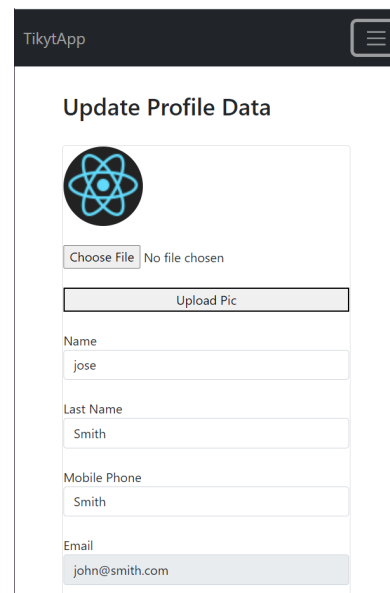


Fig. 11: Modificación rol jugador

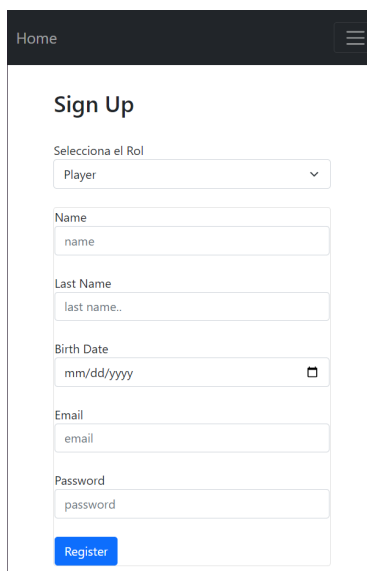


Fig. 10: Registro jugador

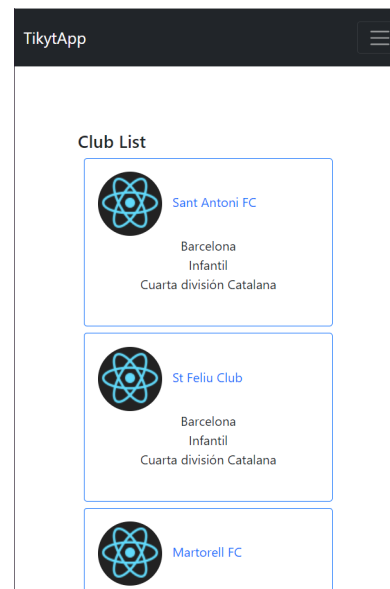


Fig. 12: Lista de usuarios club

usuario consultante, y en ese caso no se puede ver la información de contacto, además el perfil se guarda en el apartado del menú "Mis intereses". O que el usuario consultado previamente también haya dado a "seguir.^{en} tu perfil, en ese caso el sistema permite ver el resto de información de un usuario, la información privada de contacto. Se muestran en las figuras 14 y 15.

6.2. Pruebas

6.2.1. Back End

En el Back End se definieron métodos de consulta denominados end points, para poder así utilizarlos y devolver la información a la parte de Front End, se pueden ver en la figura. 16.

Para probar las funciones de la API REST, se han hecho pruebas de caja negra, utilizando la aplicación PostMan, que ha servido como simulador de un cliente quien pide datos al Back End, y se comprueba que se devuelven los datos

en la manera correcta. Teniendo en cuenta la tabla de end points se han probado todas las funciones, y el resultado es correcto. se muestran en las figuras desde del apartado A3 del apéndice.

6.2.2. Front End

A partir del diagrama de casos de uso del apartado 5.2.1 se ha extraído una lista de acciones a realizar en la aplicación, y se han probado una por una haciendo el recorrido completo de la funcionalidades creadas, se pretende obtener feedback del trabajo, y ha sido bueno, se cumple con la expectativa de la UI intuitiva. Esta prueba la han realizado varios usuarios ajenos al proyecto y se constata el correcto funcionamiento del trabajo elaborado. En el apartado A4 del apéndice, se muestra gráficamente el resultado de la prueba hecha por uno de los usuarios.

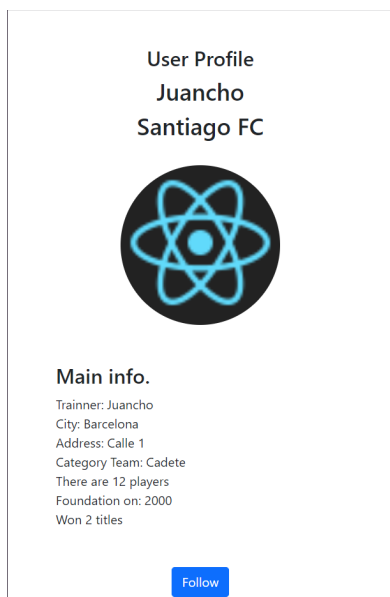


Fig. 13: Perfil Club

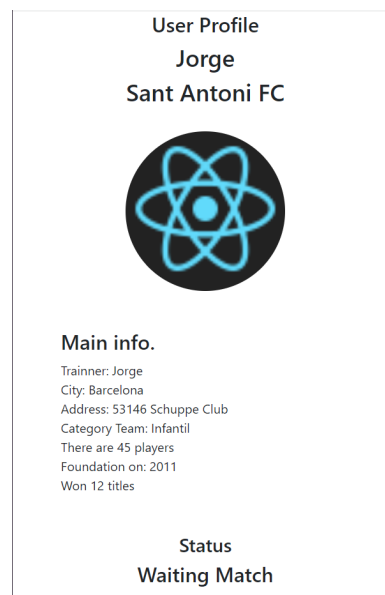


Fig. 15: Siguiendo pero no me sigue

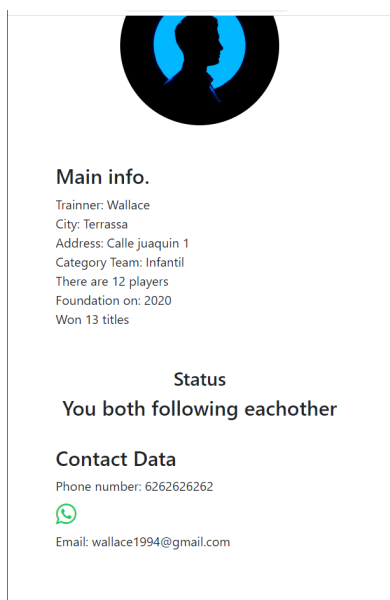


Fig. 14: Nos seguimos

Funciones BBDD En orden de prioridad					
Funcion	Descripcion	TODO / DONE	Parametros Entrada	Formato salida	
1	signUpPlayer	Alta a un nuevo jugador en BBDD	DONE	Todos los campos de Jugador	OK
2	signUpClub	Alta a un nuevo club en BBDD	DONE	Todos los campos de club	OK
3	getPlayersList	Listar todos los jugadores	DONE		Lista
4	getClubsList	Listar todos los clubes	DONE		Lista
5	editUser	Modificar los datos de un jugador	DONE	Campos a cambiar	OK, Jugador modificado
6	signInUser	Iniciar sesion	DONE	User, password	usuario, token
7	getUserX	Obtener usuario X	DONE	ID user	usuario
8	deleteUser	Eliminar usuario	DONE	ID user	usuario

Fig. 16: Métodos de consulta

se usan más los estándares de aplicaciones web existentes, y aunque se finalmente no son iguales, la técnica ayudó como guía y se parecen en su mayoría.

También hace falta destacar que React es un framework muy popular y que está en auge, y dentro del framework hay varios cambios que conllevan a actualizaciones de módulos dentro del mismo lenguaje y de sus dependencias, por lo que algunas veces afecta al desarrollo. Para este proyecto alguna vez se han usado dependencias que a lo largo del proyecto ya están obsoletas, por lo tanto obliga a mantenerse actualizado durante la elaboración de un proyecto. React Js dentro de lo que cabe es un lenguaje con una lógica de programación bastante cómoda de usar y de integrar con otros servicios, y confío que en un futuro lo consigan mejorar casi del todo.

7 CONCLUSIONES

7.1. Problemas y soluciones

Se destaca que hubo un pequeño cambio en la metodología, se ha fusionado una metodología ágil con una metodología iterativa con 'resultados funcionales', esto quiere decir que cambia un poco el concepto de la metodología que se había presentado. Se hace especial énfasis en la idea de 'resultado funcional', es decir que el crecimiento de la aplicación conlleva la fusión de la vista con la funcionalidad, dejando así para lo último (y no menos importante) el estilo del diseño. La aplicación crece en su conjunto con las partes ya fusionadas trabajando a la vez.

Otro de los "problemas" que ha habido a lo largo del proyecto, es que para la técnica del paper prototyping se definieron unas vistas que difieren un poco del resultado final, la razón es porque esas vistas se pensaron más orientado a una aplicación móvil nativa, y en la realidad,

7.2. Futuro de la aplicación

La aplicación que se ha desarrollado es funcional en una versión básica, como se ha comentado antes se podrían mejorar varios puntos, destaco los siguientes:

- Mejorar a nivel general las vistas de la aplicación, aplicando estilos avanzados y animados.
- Añadir un apartado de notificaciones push.
- Añadir más información dinámica a los perfiles, por ejemplo, un mapa donde se muestren ubicaciones.
- Permitir a los usuarios publicar más contenido digital, como fotos y vídeos en su perfil.

7.3. Qué hemos logrado

Se puede decir que a nivel de proyecto se han conseguido los objetivos previstos, la lógica principal de la aplicación está muy conseguida y funcional en su versión de entrega.

A nivel académico, se han puesto a prueba muchos de los conocimiento adquiridos, sobre todo en el ámbito de la organización, y aplicación de técnicas de desarrollo de problemas.

A nivel personal, ha sido todo un reto, ya que las tecnologías usadas en el proyecto han sido todas nuevas para mí, lo cual conlleva una gran dedicación de aprendizaje a lo largo del proyecto, ha sido satisfactorio porque pienso que he sabido aplicar los conocimiento y combinarlo para llevar a cabo la propuesta.

A nivel global, considero que es un proyecto interesante ya que se propone una solución a un problema real en el ámbito del fútbol base, y aunque tiene mucho margen de mejora, lo básico se ha conseguido y además se ha desarrollado de manera que se pueda escalar y mejorar fácilmente.

8 AGRADECIMIENTOS

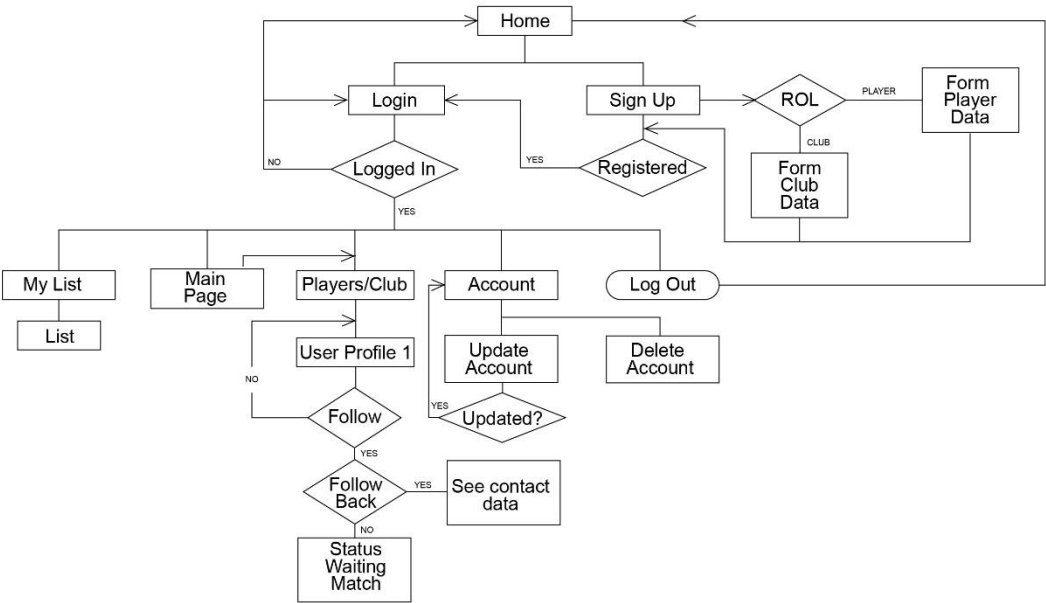
Hago especial mención a mi tutor del proyecto, ya que en todo momento me ha dado la oportunidad de avanzar en el trabajo, y darme comentarios a mejorar en los momentos decisivos, también agradecer a mi pareja, compañeros y familiares que han participado en de alguna manera en el proyecto, y han dado feedback que me ha servido para incluir y mejorar la aplicación.

REFERENCIAS

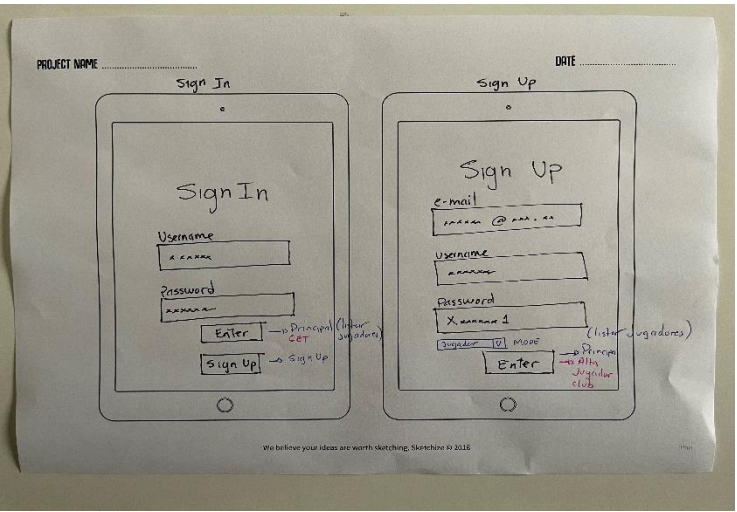
- [1] Scrum <https://www.iebschool.com/blog/metodologia-scrum-agile-scrum/>
- [2] Monday <https://monday.com/?utm-source=adwordsbrandutm/>
- [3] Prototipos de papel <http://mpiua.invid.udl.cat/prototipos-de-papel-paper-prototyping/> [Consulta: 12-04-2019]
- [4] Visual Studio Code <https://code.visualstudio.com/>
- [5] NestJs <https://docs.nestjs.com/>
- [6] Advanced REST client <https://chrome.google.com/webstore/detail/advanced-rest-client>
- [7] Why MongoDB <https://platzi.com/blog/7-razones-mongodb/?utm-source/>
- [8] MongoDB API <https://www.mongodb.com/>
- [9] Postman Client Side <https://www.postman.com/>
- [10] React JS <https://es.reactjs.org/>
- [11] TransferRoom <https://www.transferroom.com/>
- [12] Authentication in React <https://www.codingdeft.com/posts/react-authentication-mern-node-passport-express-mongo/implementing-the-front-end>
- [13] Rutas Privadas ReactJS <https://www.robinwieruch.de/react-router-private-routes/>
- [14] Autenticación JWT Node JS <https://bluuweb.github.io/node/07-jwt/consumir-api>
- [15] El fútbol base <https://www.youcoach.es/articulo/el-futbol-base>
- [16] La condición psicológica del futbolista <https://www.clubcolegiosalgui.es/psicologia/tecnicos-y-jugadores/item/723-condicion-psicologica-futbolista>

Apéndice

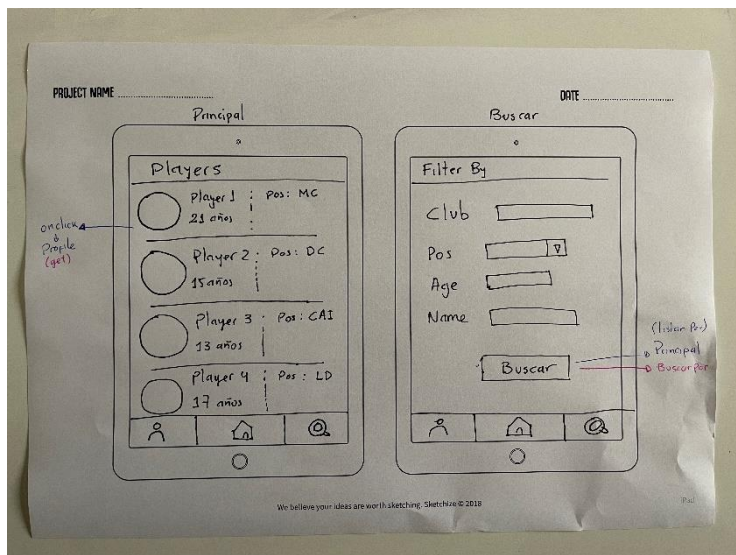
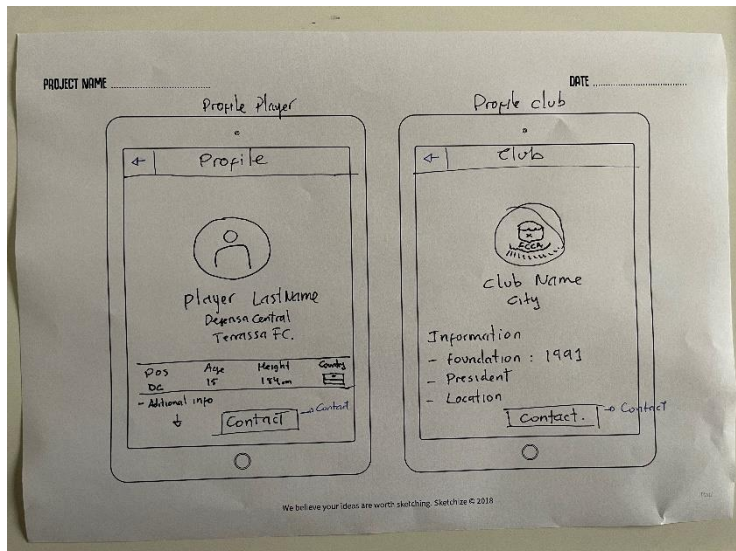
A1 – Mapa de estados de la aplicación



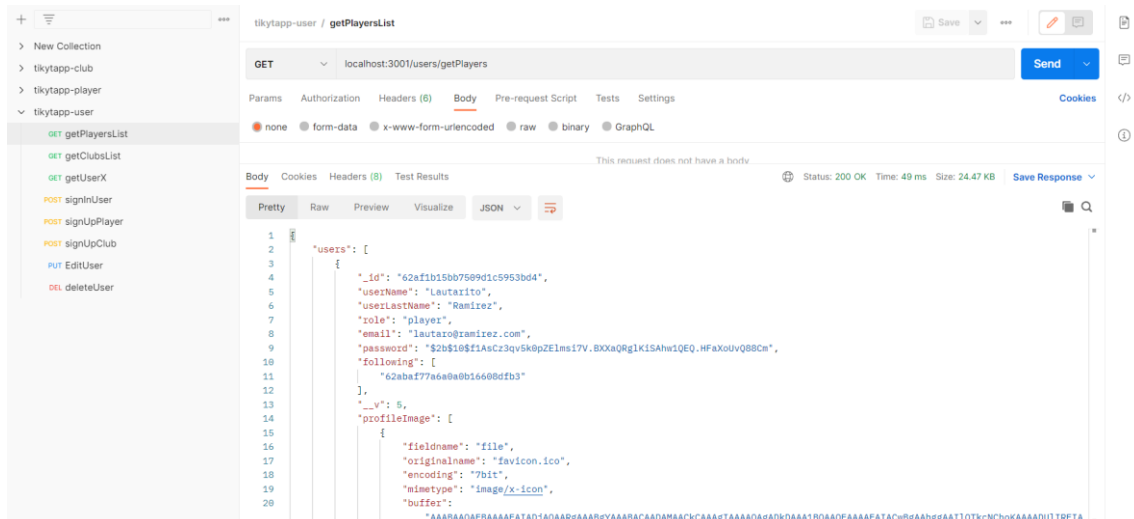
A2 – Técnica paper prototyping



Jhon Freddy Estupiñan Caicedo: Tikitapp



A3 – Pruebas de Back End en PostMan



Jhon Freddy Estupiñan Caicedo: Tikitapp

[illegible]

The screenshot shows the Postman REST client interface. The URL bar displays the endpoint: `localhost:3001/users/get/62af1b15bb7509d1c5953bd4`. The method is set to **GET**. The response is a JSON object:

```

{
  "user": {
    "id": "62af1b15bb7509d1c5953bd4",
    "username": "Lautarito",
    "userLastName": "Ramirez",
    "role": "player",
    "email": "lautaro@ramirez.com",
    "password": "$2b$10$fiAsc23qv5k8p2Flms17V.BXxaQgIKi5Ahws1QE0.HFaXouVq08Cn",
    "following": [
      "62abaf77a6a0e0b16608dfb3"
    ]
  },
  "vv": 5,
  "profileImage": [
    {
      "fieldName": "file",
      "originalName": "favicon.ico",
    }
  ]
}

```

The screenshot shows the Swagger UI for the 'tikyapp-user' API. The 'POST /localhost:3001/users/signin' endpoint is selected. The request body is shown in the 'Body' tab, displaying a JSON object with user details like email, password, and role.

Jhon Freddy Estupiñan Caicedo: Tikitapp

+

≡

> New Collection

> tikitapp-club

> tikitapp-player

> tikitapp-user

> GET getPlayersList

> GET getClubsList

> GET getUserX

> POST signInUser

> POST signUpPlayer

> POST signUpClub

> PUT EditUser

> DEL deleteUser

tikitapp-user / signUpPlayer

POST localhost:3001/users/signup

Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1 {
2   "userName": "www",
3   "userLastName": "xxxx",
4   "role": "player",
5   "email": "xx@xxx.com",
6   "password": "1234",
7   "followers": []
8 }
```

Body Cookies Headers (8) Test Results

Status: 201 Created Time: 474 ms Size: 5.66 KB Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "newUser": {
3     "userName": "www",
4     "userLastName": "xxxx",
5     "role": "player",
6     "email": "xx@xxx.com",
7     "password": "$2b$10$.wphkGecZ.C72N0IGK71.L71txQqGfy9kZ61:F353tNXaA5Gg/D_",
8     "following": [],
9     "profileImage": [
10      {
11        "fieldname": "file",
12        "originalname": "favicon.ico",
13        "encoding": "7bit",
14        "mimetype": "image/x-icon"
```

+

≡

> New Collection

> tikitapp-club

> tikitapp-player

> tikitapp-user

> GET getPlayersList

> GET getClubsList

> GET getUserX

> POST signInUser

> POST signUpPlayer

> POST signUpClub

> PUT EditUser

> DEL deleteUser

tikitapp-user / signUpClub

POST localhost:3001/users/signup

Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1 {
2   "userName": "St Quirze 2",
3   "role": "club",
4   "email": "st@quirze2.com",
5   "password": "1234",
6   "phoneNumberClub": "123123123"
7 }
```

Body Cookies Headers (8) Test Results

Status: 201 Created Time: 211 ms Size: 5.67 KB Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "newUser": {
3     "userName": "St Quirze 2",
4     "role": "club",
5     "email": "st@quirze2.com",
6     "password": "$2b$10$2pJKN31RQ5sK2lteMAC3r.3hAyQxSK09ftjNm/o9e5zMuTVKeB2",
7     "following": [],
8     "profileImage": [
9       {
10        "fieldname": "file",
11        "originalname": "favicon.ico",
12        "encoding": "7bit",
13        "mimetype": "image/x-icon",
14        "buffer": "AAABAAQEBAAAAEATADJAQAARGAAABGYAAABACAAADMAACKAAAGIAAAQAGAdKAAA1BQAAQEAAAAEATAC@GAAbgAAI1QtkNchoKAAAAADUIRfIAAAQ
```

+

≡

> New Collection

> tikitapp-club

> tikitapp-player

> tikitapp-user

> GET getPlayersList

> GET getClubsList

> GET getUserX

> POST signInUser

> POST signUpPlayer

> POST signUpClub

> PUT EditUser

> DEL deleteUser

tikitapp-user / EditUser

PUT localhost:3001/users/put/62af1b15bb7509d1c5953bd4

Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1 {
2   "userName": "John"
3 }
```

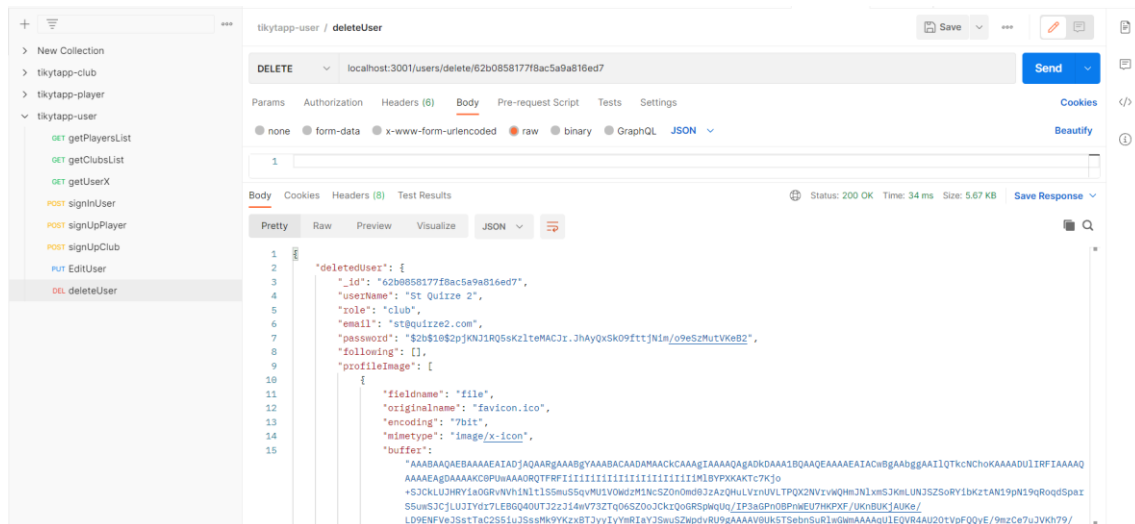
Body Cookies Headers (8) Test Results

Status: 200 OK Time: 205 ms Size: 5.69 KB Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "updatedUser": {
3     "_id": "62af1b15bb7509d1c5953bd4",
4     "userName": "John",
5     "userLastName": "Ramirez",
6     "role": "player",
7     "email": "lauraro@ramirez.com",
8     "password": "$2b$10$F1AsC23qv5k8pZE1ms17V.BXXaQ8g1K1SAhw1QE.Q.HFaXoUvQ88Cn",
9     "following": [
10      {
11        "62abaf77a6a8a9b1668dfb3"
12      }
13     ],
14     "profileImage": [
15       {
16        "fieldname": "file",
17        "originalname": "favicon.ico",
18        "encoding": "7bit",
19        "mimetype": "image/x-icon"
```

Jhon Freddy Estupiñan Caicedo: Tikitapp



A4 – Prueba Front End (Realizada por un usuario externo)

1. Registro en la aplicación con Rol 'Club'.

Home

Sign Up

Selecciona el Rol

Club

Club Name

Bonaire Club de Fútbol

Trainer Name

Wallace

Foundation Date

06/01/2020

Email

wallace1994@gmail.com

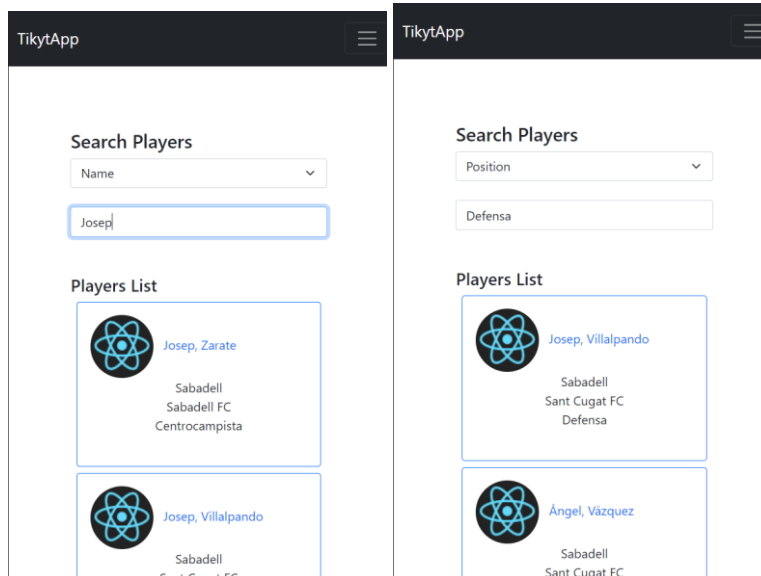
Password

Register

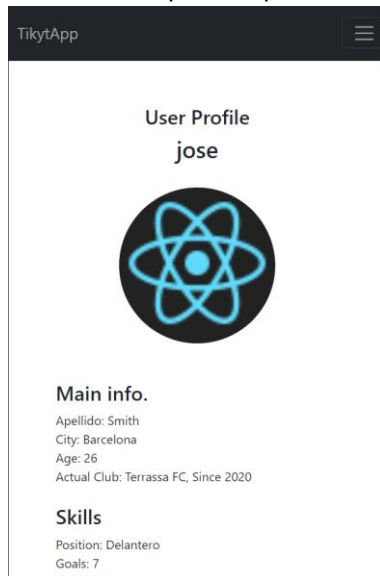
2. Iniciar sesión, y ver la lista de jugadores dados de alta.

The screenshot displays the TIKYT App interface. On the left, the 'Home' screen features a login form with fields for 'Email' (containing 'wallace1994@gmail.com') and 'Password' (masked with '*****'), a 'Login' button, and a link for 'Don't have account? Sign Up'. On the right, the 'TikytApp' screen shows a 'Search Players' section with a dropdown menu set to 'By'. Below this is a 'Players List' containing two entries: 'jose, Smith' and 'Miguel, Pizarro'. Each entry includes a blue atomic icon, the player's name, and their associated teams: 'Barcelona', 'Terrasa FC', and 'Delantero'.

3. Buscar usuarios haciendo uso del filtro.

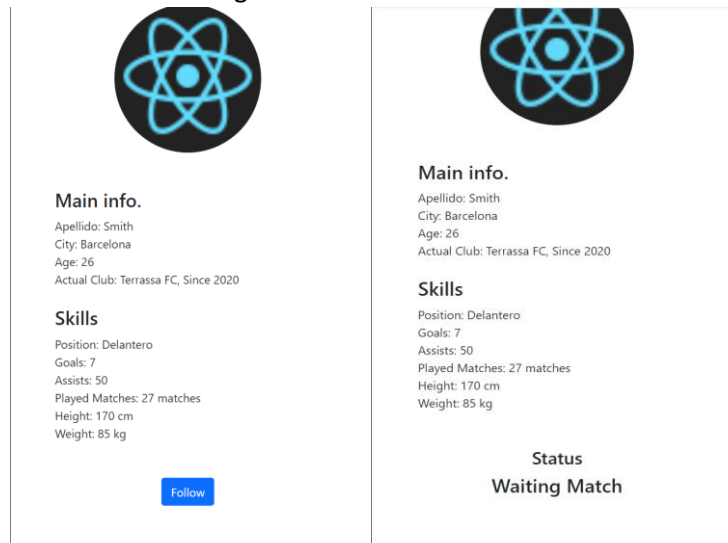


4. Consultar un perfil específico.



5. Seguir al usuario

a. El usuario no ha seguido al club



b. El usuario ha seguido al club

Main info.
Apellido: Smith
City: Barcelona
Age: 26
Actual Club: Terrassa FC, Since 2020

Skills
Position: Delantero
Goals: 7
Assists: 50
Played Matches: 27 matches
Height: 170 cm
Weight: 85 kg

Status
You both following eachother

Contact Data
Phone number: 600077219

Email: john@smith.com

c. Comprobar el funcionamiento de contacto por WhatsApp haciendo click en el logo de whatsapp.

Main info.
Apellido: Smith
City: Barcelona
Age: 26
Actual Club: Terrassa FC, Since 2020

Skills
Position: Delantero
Goals: 7
Assists: 50
Played Matches: 27 matches
Height: 170 cm
Weight: 85 kg

Status
You both following eachother

Contact Data
Phone number: 600077219

Email: john@smith.com

<https://wa.me/600077219>

Open WhatsApp
Chat on WhatsApp with 600077219
[Continue to Chat](#)
Don't have WhatsApp yet?
[Download](#)

WHATSAPP
Features
Security
Download
WhatsApp Web
Business
Privacy

COMPANY
About
Careers
Brand Center
Get in touch
Blog
WhatsApp Stories

6. Consultar el Menú y Acceder al apartado “mi cuenta”.

TikytApp
Players
Search
Account
Logout

Search Players
By

Players List

 jose, Smith
Barcelona
Terrassa FC
Delerantero

TikytApp



Main info.
Club Name: Bonaire Club de Fútbol
Trainer Name:
Role: club
City:
Address:
Category Team:
Number of players
Foundation on: 2020-06-01T00:00:00.000Z
Number of titles:

[Editar cuenta](#)

7. Modificar la foto de perfil y comprobar que se cambia.

TikytApp

Update Profile Data



Choose File 1366_2000.jpeg

Upload Pic

Club Name

Bonaire Club de Fútbol

Trainer Club

Role

club

Email

wallace1994@gmail.com

TikytApp



Main info.

Club Name: Bonaire Club de Fútbol

Trainer Name:

Role: club

City:

Address:

Category Team:

Number of players

Foundation on: 2020-06-01T00:00:00.000Z

Number of titles:

Editar cuenta

8. Modificar algún campo y comprobar que se cambia.

Choose File No file chosen

Upload Pic

Club Name

Bonaire Club de Fútbol

Trainer Club

Wallace

Role

club

Email

wallace1994@gmail.com

Club city

Terrassa

Address

Calle juaquin 1

Category Team

Infantil

Number of players

12

localhost:3000 says

User Modified

OK

Club city

Terrassa

Address

Calle juaquin 1

Category Team

Infantil

Number of players

12

Foundation Date

2020-06-01T00:00:00.000Z

Titles Number

13

Update

TikytApp



Main info.

Club Name: Bonaire Club de Fútbol

Trainer Name: Wallace

Role: club

City: Terrassa

Address: Calle juaquin 1

Category Team: Infantil

Number of players 12

Foundation on: 2020-06-01T00:00:00.000Z

Number of titles: 13

Editar cuenta