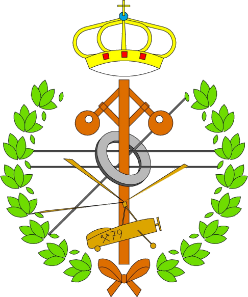
****

**Escuela de Ingenierías**

**Industrial, Informática y Aeroespacial**

**GRADO EN INGENIERÍA INFORMÁTICA**

Sistemas de Información de Gestión y Business Intelligence.

Memoria de la aplicación FIFAProAnalyzer.



Arturo García González

16 diciembre curso 2020/2021

Índice

1. Introducción.
2. Descripción del problema.
3. Herramientas para la resolución del problema.
4. Aplicación. Explicación de su funcionamiento.
5. Algoritmo de Recomendación.
6. Análisis de Resultados.
7. DAFO: análisis descriptivo de ventajas y limitaciones del trabajo realizado.
8. Líneas de futuro.
9. Lecciones aprendidas.
10. Fuentes bibliográficas y enlaces.

## Introducción.

En este documento se llevará a cabo una explicación a fondo de lo que consiste la aplicación de FIFAPro Analyzer, cómo utilizarla y cómo introducir la información que tenemos para obtener unos resultados óptimos en base a esta información. Además, es importante saber que esta no es una aplicación profesional, pero en un futuro podría serlo.

Para utilizar la aplicación, hay que tener claro que su finalidad es buscar y recomendar jugadores de ciertas características introducidas por el usuario previamente. Si bien a veces puede parecer que los resultados entre búsqueda y recomendación son muy parecidos, los jugadores obtenidos en la recomendación son los más adecuados según lo que quiere encontrar el usuario.

Además, dentro de este documento se encontrará un apartado que nos explica cómo utilizar esta aplicación, sus ventajas e inconvenientes, y varios ejemplos que nos muestran lo que debería salirnos como resultados en varios casos para que el usuario pueda corroborar si lo está haciendo bien a la hora de utilizarla.

## Descripción del problema.

## A la hora de buscar jugadores para incorporar a sus equipos, muchos clubes y entidades deportivas no conocen el valor de los mismos o no saben cómo enfocar la búsqueda de los mismos para encontrar la opción que más les convenga. Si bien es cierto que cuentan con personas especializadas para este tipo de actividades llamados directores deportivos, estos directores deportivos no suelen contar con programas de búsqueda y recomendación de jugadores, sino que delegan estas tareas en otras personas que tienen la misma función que ellos, pasando los directores a la única función de decidir y negociar si un jugador interesa o no al club, y por cuánto dinero podría firmarse ese traspaso.

## A menudo estas personas acuden a Transfermarkt (<https://www.transfermarkt.es/> ), **la Biblia del mercado de fichajes.** Periodistas y aficionados, pero también dirigentes y representantes acuden a beber de la fuente de **Transfermarkt** para consultar estadísticas, datos de fichajes, balances de mercado de los clubes y, sobre todo, **su Santo Grial: el valor de mercado de los futbolistas.** De todos los futbolistas. Más de 700mil jugadores tienen ficha en la web de origen alemán. Y un precio de mercado asignado en la misma.

Ese valor, lejos de ser una mera cifra orientativa, **es un objeto de estudio por parte de los directores deportivos** y demás personal encargado de fichar en los clubes, que acuden a Transfermarkt para peinar el mercado, para hacer cuentas, **para valorar posibles traspasos futuros.** Pero también son un arma de doble filo para los futbolistas, que acuden con cautela a comprobar sus datos.

Estos valores cambian, se actualizan, se analizan y se renuevan, como explica su responsable en la versión en castellano: "Normalmente cada seis meses, pero también hay excepciones para jugadores individuales, como al final de un mercado de fichajes o actuaciones especiales en torneos **(Kangin, en el Mundial Sub-20 o Ceballos en el Europeo Sub-21 por ejemplo).** Observamos el mercado de cerca, pero es tan dinámico que no todos los valores pueden ser siempre correctos. **Los valores tampoco deberían ser un valor a corto plazo,** entonces se tendrían que renovar todos los valores cada pocas semanas. Eso sería difícil de cumplir".

Pues bien, esta aplicación es una herramienta para ayudar a que este problema de búsqueda y recomendación de jugadores sea más eficiente y se pueda disfrutar de una diversidad de opciones mucho mayor de lo que se podría contar sin ella.

## Herramientas para la resolución del problema.

Para llevar a cabo la resolución del problema he utilizado 3 herramientas principales:

1. **Neo4J**.

Esta herramienta es un software libre de Base de datos orientada a grafos, implementado en Java.​​ Los desarrolladores describen a Neo4j como un motor de persistencia embebido, basado en disco, implementado en Java, completamente transaccional, que almacena datos estructurados en grafos en lugar de en tablas.

Además, esta herramienta fue la única impuesta por nuestro profesor Enrique para la realización del proyecto por lo que es la única de las 3 herramientas que no era de negociable utilización.

1. **Vuetify.**

Esta herramienta nos ha sido clave para llevar el proyecto a cabo porque es la que nos ha permitido crear una página web en aspectos tanto de diseño como de desarrollo y funcionalidad para recoger los datos que nos hacen falta.

Es muy fácil de utilizar ya que cuenta con varios manuales para aquellos que no hayan tenido contacto con esta tecnología aún, pero incluso sin esta documentación puede intuirse cómo funciona con solo mirar el código que se utiliza para crear las páginas.

Esta documentación se encuentra en <https://vuetifyjs.com/en/introduction/why-vuetify/> y, como digo, será de ayuda para cualquier persona que no haya tenido contacto anteriormente con esta tecnología.

1. **Visual Studio Code.**

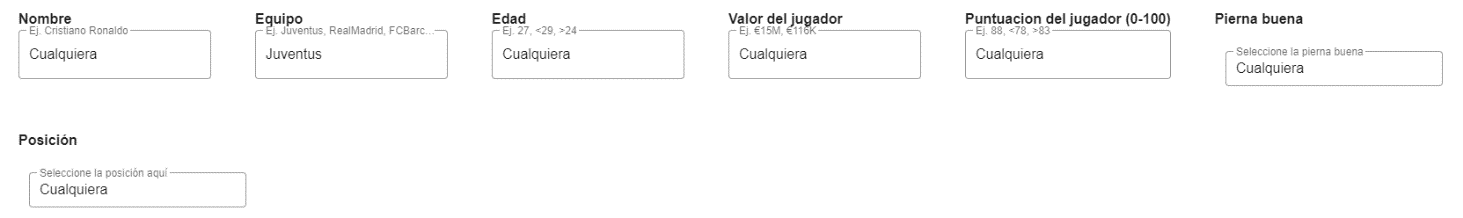
Como mucha gente del ámbito de la informática conoce, este entorno de desarrollo es muy atractivo tanto por su forma de mostrar el código como a la hora de ejecutarlo o trabajar con él. Por eso, creo que ha sido una buena elección para llevar a cabo el desarrollo de mi código y poder así crear la aplicación en cuestión. También instalé el pluggin ‘Vetur’ , pero no es más que un añadido para visualizar los datos con colores que ayuda a la hora de desarrollar código para ver los diferentes elementos con distintos colores.

## Aplicación. Explicación de su funcionamiento.

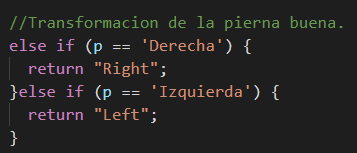
Para llevar a cabo esta explicación nos centraremos en sus 2 partes fundamentales:

1. **Frontend:**

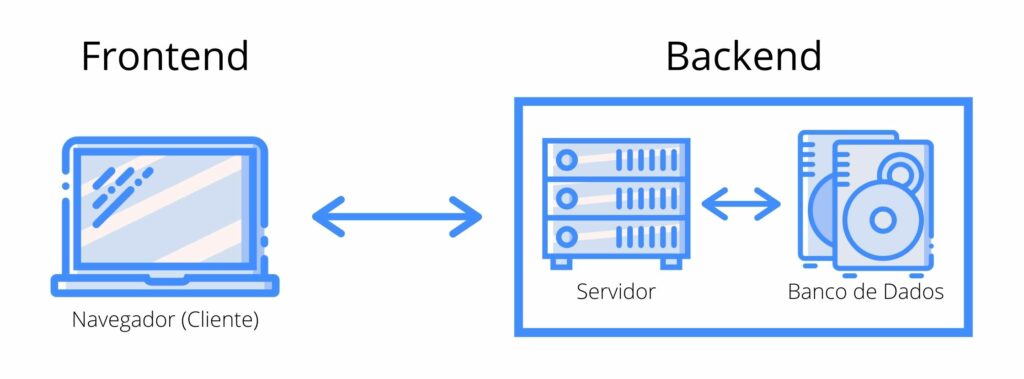
También conocida como la parte del software que interactúa con los usuarios, es decir, cómo los usuarios introducen datos o perciben la información de una aplicación. En nuestro caso nuestro Frontend tiene la misión de recoger datos introducidos por el cliente a través de varios campos como son el Nombre, Equipo o Posición del jugador. Estos datos se introducen en los cuadros que vemos en la imagen en forma de texto o seleccionando las opciones que nos deja la aplicación.



Una vez introducidos los campos de búsqueda deseados, nuestros datos se introducen en el sistema, se transforman mediante un método creado previamente para poder introducirlos posteriormente en la consulta a la base de datos, y se manda esta consulta o *query* a neo4j para esperar su respuesta. Por ejemplo, para transformar el dato de la pierna buena que queremos que tenga nuestro jugador lo haremos así:



*Right* y *Left* es la forma en que los datos se introducen en la query que viajará al Backend. Ahora bien, podríamos preguntarnos: ¿porque en inglés? Pues bien, la transformación se realiza en inglés y con la primera letra en mayúscula porque es así como tenemos los datos almacenados en nuestra base de datos, y de introducir la primera letra en minúscula ya no funcionaría nuestra aplicación.

De esta forma podemos asimilar el dinamismo de los datos del sistema a la siguiente imagen:

Donde nosotros actuaríamos como el cliente, que introduce los datos para realizar una consulta al Backend, en forma de *Request*, y este nos devolvería una *Response* o respuesta a nuestra consulta. Es importante dejar claro que, aunque no hubiera resultados de la búsqueda, el Backend devuelve igualmente una respuesta. Solo deja de enviarla cuando ha habido algún error al realizar la consulta o algún fallo en el proceso de realizarla.

Nuestro Frontend además nos mostrará también los resultados de nuestro sistema de recomendación una vez obtenidos. Es importante tener en cuenta que, en caso de que el sistema no obtenga recomendación alguna, no mostrará resultado alguno, por lo que no hay que estrañarse si en algún momento no se mostrase nada al darle al botón de recomendación.

1. **Backend:**

Esta es la parte de la aplicación que controla la lógica del sistema, es decir, cómo se van a tratar los datos una vez enviados desde el Frontend y cómo se tratarán para enviar de vuelta esa respuesta de la que hemos hablado anteriormente.

Aunque parezca más complicada, es en realidad más fácil de lo que parece, ya que realmente todo se basa en hacer consultas a la base de datos y enviar los resultados o reportar los errores que se encuentren al intentar llevar a cabo estas operaciones.

Para ello vamos a ver un ejemplo de consulta donde se tratan los datos de la edad de un jugador.

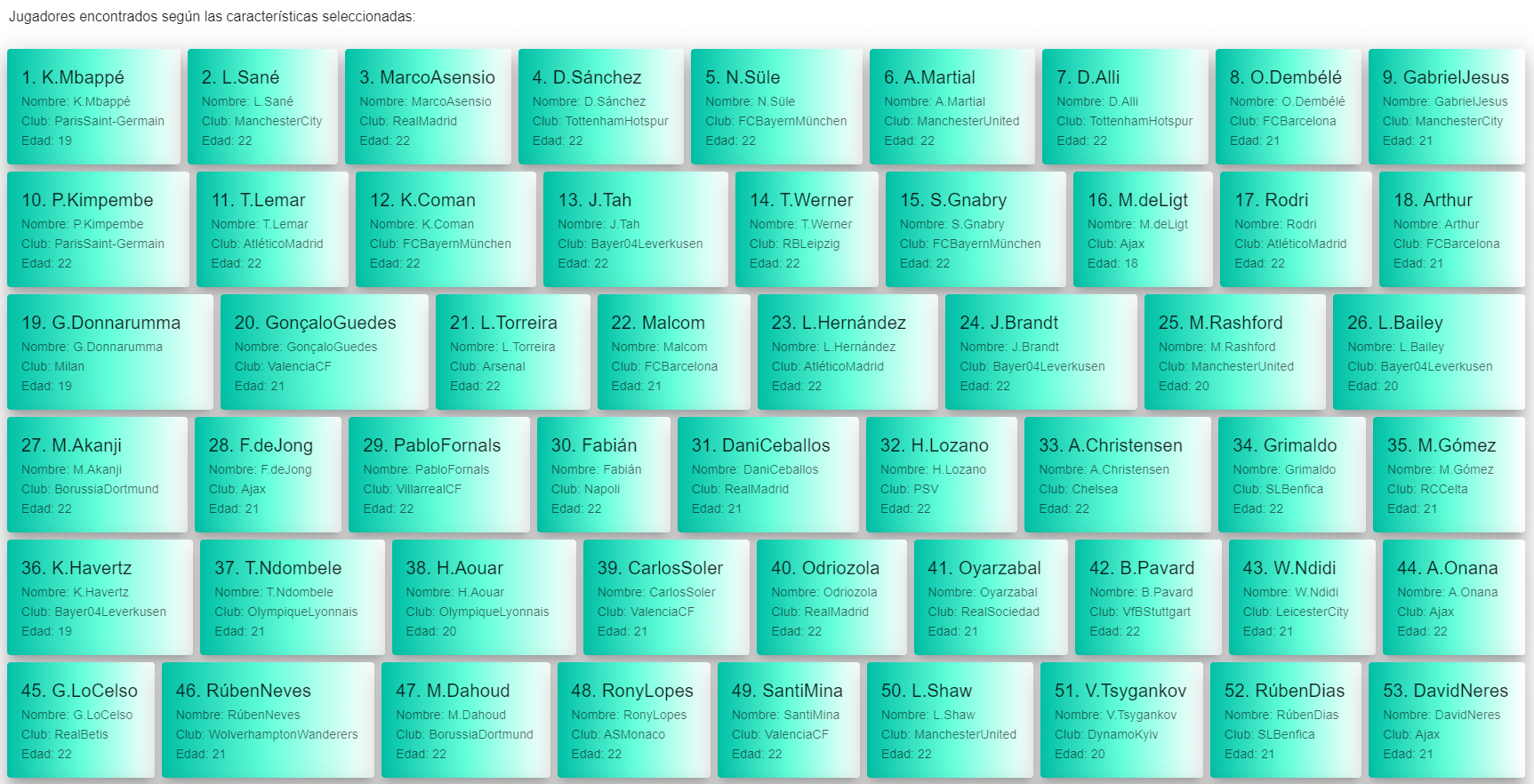


Para ello vemos cómo hemos de distinguir primero si el usuario desea buscar jugadores mayores o menores de una edad en concreto.

A continuación distinguimos entre si es el primer dato que se introduce a la query o no, para construir correctamente este inicio de query.

Por último, tenemos que comprobar que no se haya introducido ningún dato que no sea uno de los símbolos que contemplamos anteriormente. Si esto ocurre la query no devolverá nada y además se enviará un error al Frontend. La consulta que realizamos podríamos verla en la terminal donde estamos corriendo el Backend, y en nuestro caso sería la siguiente:



Por último, solo nos quedaría observar cómo se mostrarían los resultados en la interfaz de la aplicación, tanto si es una consulta como si es una recomendación.

## Algoritmo de Recomendación.

## Para llevar a cabo el algoritmo, lo primero que me planteé como futbolista que soy es lo siguiente: ¿Qué se busca hoy día en el fútbol? ¿Qué características ha de tener un jugador para poder formar parte de cualquier equipo? Pues bien, en base a estas 2 preguntas he basado mi algoritmo de recomendación, que quedará reflejado principalmente en 2 factores a tener en cuenta: la edad del jugador y su puntuación total, la cual nos la aporta directamente la FIFA y nos es de gran ayuda, ya que este número representa 'cómo de bueno' es un jugador en conjunto a sus otras características, lo que nos facilita muchísimo el trabajo a la hora de recomendar un jugador, ya que podríamos hacerlo directamente con esta variable, pero no sería del todo real ya que en la realidad no solo se busca que el jugador sea bueno.

A pesar de que los resultados obtenidos son buenos y que aplicando el sentido común, yo como futbolista también recomendaría los jugadores que se obtienen como resultado, me gustaría haber podido disponer de más variables para llevar a cabo esta recomendación como pueden ser:

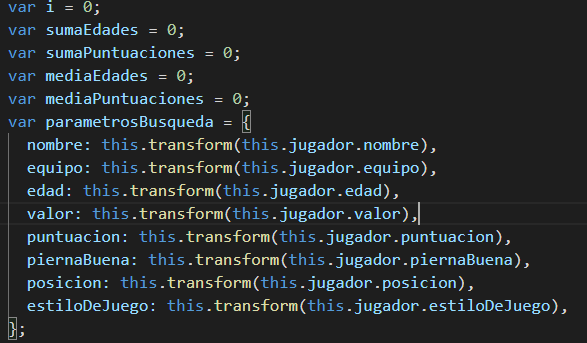
* La frecuencia lesiva del jugador: esta variable nos habría aportado mucha información relevante, ya que, un jugador que no tiene continuidad en el juego no puede llegar a formar parte de un proyecto siendo relevante en él.
* Medidas paramétricas reales de su fuerza, velocidad y resistencia: estas medidas nos podrían haber ayudado a la hora de buscar 'velocistas' que recorran la banda ofensivamente o por ejemplo mediocentros de mucho recorrido que aguanten casi todo el partido corriendo y nos cubriesen mucho terreno de juego tanto ofensiva como defensivamente.
* Sueldo: para muchos equipos esta es una limitación, ya que hay jugadores que cobran mucho dinero al año y no todos los clubes pueden permitirse tener a estas superestrellas en sus plantillas.

Dicho esto, repito que creo que los resultados obtenidos son buenos y tienen sentido bajo mi punto de vista como futbolista que soy.

Para explicar mi sistema de recomendación dividiré la explicación en 2 partes:

1.En base a qué datos obtengo mis jugadores y cómo los obtengo:

Para mi algoritmo de recomendación, como ya he comentado anteriormente, elijo la edad y la puntuación total del jugador principalmente como variables que determinarán que jugadores recomiendo finalmente. Pero no son los únicos valores que tengo en cuenta, ya que antes de recomendar un jugador he de tener en cuenta las características que el usuario que está siendo recomendado desea que tenga ese jugador. Para ello, lo primero que hago es transformar los datos como si de una búsqueda normal se tratara para posteriormente realizar una consulta a neo4j y obtener en un primer momento todos los jugadores que cumpla con estas características.

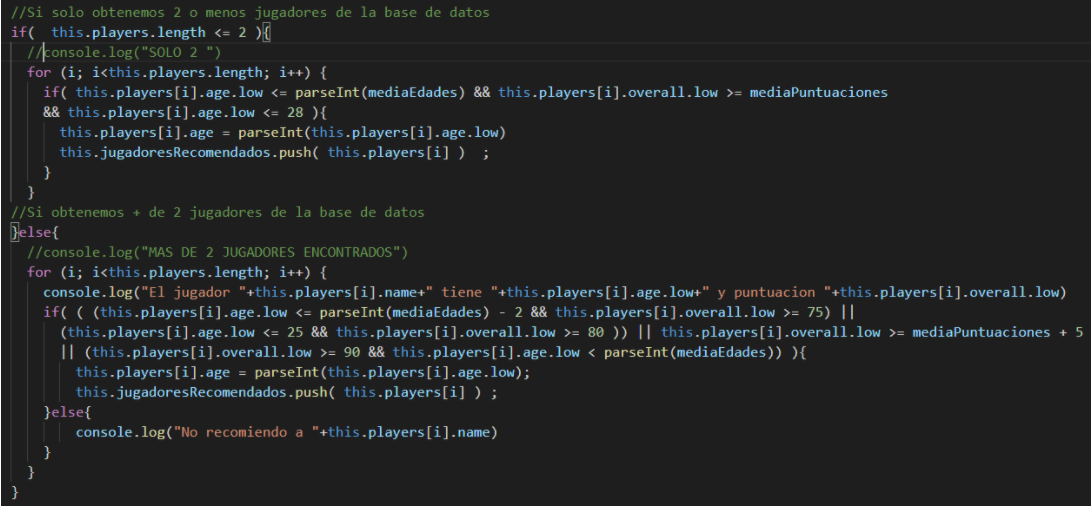


2.Cómo trato estos datos para llegar al resultado final:

Posteriormente y como muestra la imagen que podemos ver, obtengo la media aritmética de la edad de todos esos jugadores obtenidos y de sus correspondientes puntuaciones totales, y me guardo ambas 2 en 2 variables, mediaEdades y mediaPuntuaciones respectivamente. Y además crearé una variable donde posteriormente almacenaré los resultados obtenidos, es decir, los jugadores recomendados.



Ahora trataré de explicar cómo selecciono los jugadores que quiero recomendar al usuario. Para ello tendremos que fijarnos en estas pocas líneas de código, que aparentemente no parecen ser muy importantes pero es donde reside la esencia de nuestra recomendación de jugadores.



Esta imagen representa en esencia que:

1. Si la búsqueda devuelve solo 2 o menos jugadores, se recomendará aquel más joven de los 2, si es menor de 29 años y si tiene una puntuación mayor que el otro jugador encontrado. En caso de que alguna de estas 3 condiciones no se cumpla no se recomendaría ninguno de los jugadores.
2. Si la búsqueda obtiene más de 2 jugadores, entonces se recomendará.
   1. Aquellos cuya media de edad sea menor que la media de los jugadores obtenidos -2 y que su puntuación sea mayor que la media de puntuaciones de los jugadores obtenidos, o aquellos que tengan 25 años o menos y que tengan una puntación mayor de 79, la cual he considerado adecuada y que representa un jugador válido para casi cualquier equipo.
   2. Aquellos jugadores cuya puntuación sea 5 puntos mayor que la media, lo cual implica que no cualquier jugador pasará a ser recomendado.
   3. Aquellos jugadores cuya puntuación sea 90 o superior, y que además tengan una edad que sea menor que la media de edades de los jugadores obtenidos. Como 90 es una puntuación de jugador bastante alta, he considerado que aunque tenga 28 años un jugador es apto para jugar a un alto nivel y ofrecerá rendimiento en cualquier equipo.

Hecho esto, ya solo quedaría mostrar los datos obtenidos en algún sitio de la interfaz. Preferiblemente será debajo de todos los botones y campos a elegir y con unos colores similares a los que ya tiene la interfaz.

## Análisis de Resultados.

Finalmente y para concluir, nos centraremos en los resultados obtenidos y si son buenos o no. Obviamente y como ya eh comentado anteriormente, para obtener unos resultados óptimos deberíamos contar con muchas variables más que las que nos ha aportado nuestro archivo .csv y además quizás en otro formato, pero aún con este impedimento veremos que los resultados no son malos y de hecho se acercan bastante a lo que cualquier persona que sepa de fútbol recomendaría a otra que quisiera buscar los jugadores de las características seleccionadas.

Para comenzar a analizar los resultados vamos a dividirlo en 3 partes:

1. Una búsqueda sencilla en la que el sistema no debería de tener problema ninguno en encontrarnos jugadores.

Por ejemplo, buscaremos jugadores menores de 27 años y posteriormente del equipo de la Juventus.

Vemos que, para la búsqueda de jugadores de menos de 27 años el sistema responde perfectamente tanto en la búsqueda (celeste) como en la recomendación (azul):





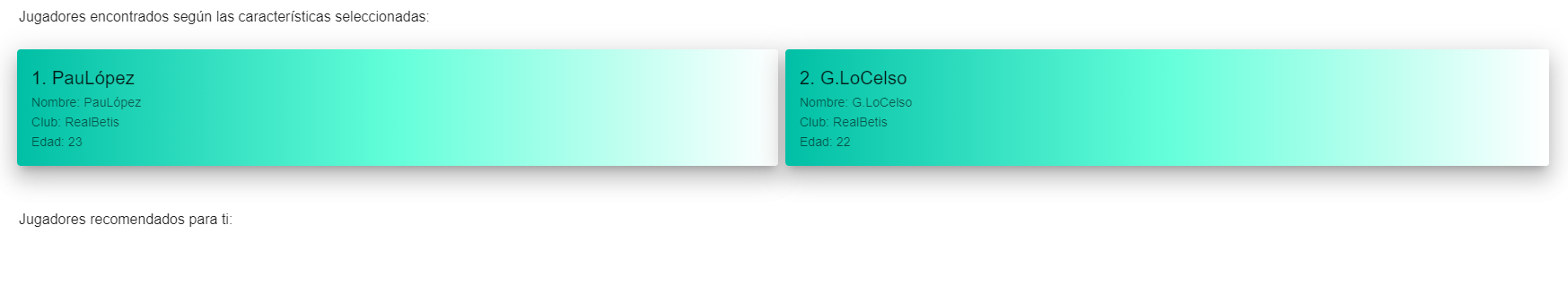
Aunque los resultados de ambas partes son muy parecidos, vemos que tiene sentido si recordamos que los jugadores se devuelven por orden descendente de puntuación, por lo que era muy probable que se recomendaran muchos de los encontrados.

Sin embargo vemos que cuando buscamos jugadores de la Juventus de Turín no se nos recomiendan tantos como antes.



1. Una búsqueda de nivel medio en la que el sistema debería funcionar bien aunque quizás nos gustaría que recomendará más o menos jugadores:

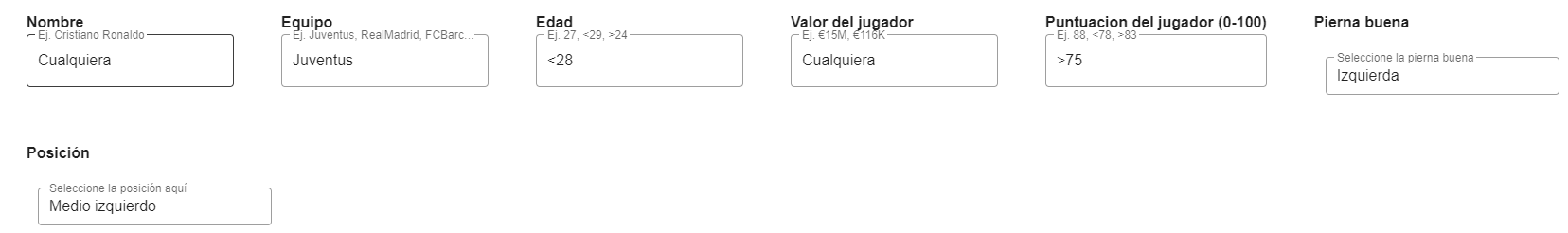
Para esta fase buscaremos jugadores zurdos del Real Betis y que además tengan menos de 27 años, por ejemplo.



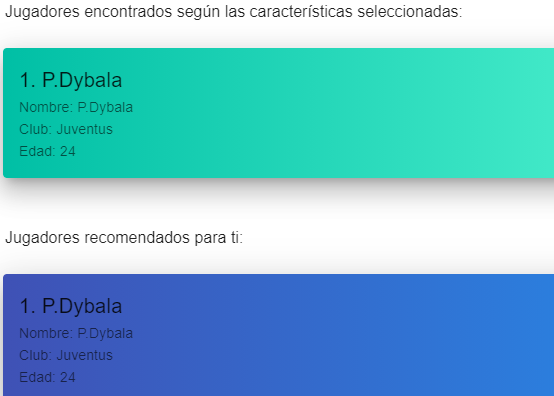
Como vemos, de 2 jugadores encontrados no se nos recomienda ninguno de los 2, lo cuál también puede pasar si buscamos en equipos que no son de alto nivel como pueden ser los grandes de Europa (FCBarcelona, Real Madrid, Bayern Munich, Liverpool, …).

1. Una búsqueda con muchos parámetros en la que pondremos a prueba el sistema y si de verdad funciona como se esperaría:

En esta ocasión buscaremos, por ejemplo, buscaremos un jugador de la Juventus, menor de 28 años, que tenga una buena puntuación como es más de 75, que sea zurdo y que juegue de medio izquierdo.



Como podemos ver en la imagen de debajo, con estas características encontramos un jugador que, además, se nos recomienda.



En base a los resultados obtenidos podemos decir que la aplicación funciona y, aun forzando mucho el resultado, vemos que si hay un jugador lo encuentra y lo recomienda perfectamente. Como hemos visto algunas variables como el nombre o el valor no se utilizan mucho porque usarlas sería prácticamente buscar un jugador en concreto, que conocemos a fondo, y damos por hecho que esto no puede suceder ya que, de ser así, no tendría mucho sentido utilizar la aplicación para buscar jugadores si ya sabemos de sobra al que queremos.

## DAFO

En primer lugar analizaremos las ventajas de esta aplicación.

**DEBILIDADES:**

Pero al igual que ventajas, este sistema tiene sus desventajas y sus limitaciones, lo cuál no viene dado solo por la falta de información para trabajar con ella como puede ser la ausencia de variables importantes como el ratio goles/partido o la frecuencia lesiva de un jugador, sino que además también se nos presentan dificultades en cuanto al formato de datos en el .csv que hemos escogido.

Como ejemplo, tenemos que el valor de los jugadores no se puede buscar en rangos de mayor que o menor que, ya que el formato de esta variable en el csv (€110.5M,€565K) nos impide buscar en la base de datos un jugador que valga más o menos que esto, ya que no se presenta el dato como un único número sino que trae caracteres “extraños” delante y detrás del número.

Tal y como he comentado antes, la falta de variables es la principal limitación de esta aplicación, y es que, aunque tenemos las necesarias para hacer un sistema con un nivel de precisión de recomendación aceptable, nos faltan muchas más para poder afinar bien en estas recomendaciones. Algunas de las variables que me gustaría haber tenido además de las ya mencionadas son, por ejemplo, la liga en la que juega un jugador, resultados de test de velocidad, resistencia y fuerza aportados por el propio club o más variables que considero básicas como pueden ser los robos de balón/partido o los pases/partido o las asistencias/partido.

**AMENAZAS:**

En general y que yo sepa, en el mercado del fútbol no existe ningún software como este o, al menos yo no lo conozco . Es por esto que creo que este software si se profesionalizara un poco más no tendría competidor alguno en el mercado. Los directores deportivos de los equipos realizan este trabajo con gente especializada que guarda esta información en bases de datos privadas y es extraño que la compartan. Además, existen empresas que venden esos datos a equipos para conocer estadísticas de otras ligas o jugadores.

**FORTALEZAS:**

Para empezar, comenzaremos con la principal ventaja: la búsqueda rápida de un abanico muy amplio de jugadores en poco tiempo y la obtención de una recomendación rápida en base a sus características.

Además, hay que tener en cuenta que se pueden encontrar jugadores por separado como ya hemos visto buscando variables aisladas como la edad, la posición del jugador o la pierna buena del mismo. Esto es de gran ayuda cuando queremos buscar jugadores sin importar mucho el equipo o el valor, obteniendo así un gran número de resultados de diversas características. También deducimos que, para encontrar jugadores que de verdad nos interesen deberemos aportar algo más de información a la aplicación que una o dos variables.

Si bien es cierto que normalmente los directores deportivos trabajan mucho para encontrar jugadores con las aptitudes deseadas, esta aplicación les permitiría hacer un filtrado rápido de los mismos y, además, obtener algunos de ellos que de alguna forma son más convenientes que los demás utilizando la funcionalidad de recomendar jugadores.

**OPORTUNIDADES:**

Dicho esto, he de decir que estoy satisfecho con los resultados obtenidos, ya que todos tienen sentido y son coherentes con lo que cualquier persona pensaría. Creo además firmemente que es una aplicación muy funcional y que con algunas horas de recopilación de variables, investigación y adecuación de las mismas para su posterior tratamiento en la base de datos podría llegar a ser un sistema usable a nivel profesional.

## Líneas de futuro.

En este aspecto la aplicación podría mejorar mucho en cuanto a calidad de recomendación, si se contase con más variables como las ya citadas en esta práctica.

Pese a este impedimento, hay que tener en cuenta que la tecnología utilizada cuenta con muchísimas más funcionalidades y rango de usabilidad para poder manejar la información en todos sus aspectos. Es por esto que creo que con tiempo y dedicación la aplicación podría llegar a profesionalizarse y siendo optimista, a comercializarse para equipos de alto nivel en el mundo del fútbol.

Salvando el obstáculo de la falta de variables creo que esta tecnología no tiene competencia alguna en el mundillo y llegaría a poderse obtener beneficio económico a través de prestar este servicio a equipos que lo requiriesen.

## Lecciones aprendidas.

Para concluir con la memoria se os ha requerido que pensemos en una serie de conclusiones o lecciones aprendidas durante la realización del proyecto.

Una de ellas y la más evidente es que, sin saber previamente nada de la tecnología de Neo4J ni Cypher, hemos sido capaces de desarrollar un proyecto completo desde su inicio hasta su final, y todo sin recibir ninguna ayuda por parte del docente de la asignatura, lo cuál tiene un mérito y nos recuerda que nunca debemos subestimarnos ni pensar que somos incapaces de llevar a cabo algo, aunque no tengamos conocimientos previos en el tema.

Otra de ellas y que cae por el propio peso de la primera, es que he aprendido a manejar la tecnología de Neo4J y Cypher, entendiendo por Cypher el lenguaje básico en el que realizamos las consultas a Neo4J y, aunque ha sido un aprendizaje básico y lento, ha sido productivo y creo que es algo fácil de entender y que cualquier persona con ganas de aprender a manejar este tipo de lenguaje sería capaz de entender.

## Fuentes bibliográficas y enlaces.

<https://vuejs.org/>

<https://www.electronjs.org/>

<https://nodejs.org/es/>

Tecnologías para utilizar la base de datos neo4j:

<https://neo4j.com/>

<https://neo4j.com/developer/javascript/>

<https://expressjs.com/es/>

<https://www.npmjs.com/package/body-parser>

<https://www.npmjs.com/package/axios>

DAFO, qúe es y para qué sirve:

<https://www.eaeprogramas.es/blog/negocio/empresa/analisis-dafo-que-es-y-para-que-sirve>

Fuente al archivo .csv escogido para el proyecto:

<https://www.kaggle.com/karangadiya/fifa19>