

# Fantasy of Legend

## Descripción Completa de la aplicación

Aplicación web enfocada a los e-sports del videojuego League of Legends en el que cada usuario tiene que seleccionar cinco jugadores profesionales de una misma liga para formar un equipo.

Después de cada jornada real se le asigna x puntos a cada jugador dependiendo del rendimiento/impacto que haya tenido en la partida. La suma de los puntos que hayan obtenido los jugadores elegidos por ese usuario será la puntuación final.

	TOP	JUNGLE	MID	ADC	SUPPORT	ENTRENADOR
ASESINATO	2	2	2	2	1,5	
MUERTE	-1	-1	-1	-1	-1	
ASISTENCIA	1	1	1	1	1,5	
MINIONS	0.01	0.015	0.01	0.01	0.02	
TRIPLE	2	2	2	2	2	
CUADRA	5	5	5	5	5	
PENTA	10	10	10	10	10	
10 ASIST / CADA 5	2/1	2/1	2/1	2/1	2/1	
CADA 5 MUERTES	-2	-2	-2	-2	-2	
0 MUERTES	2	2	2	2	2	
MAS PUNTOS DE VISION	1	1	1	1	1	
GANAR EQUIPO						5
PIERDE EQUIPO						-2
CADA DRAGON / ALMA						2/1
DRAGON ANCIANO / REPETIR						3/2
HERALDO						2
BARON						4
PRIMERA SANGRE						2
TORRE						1
GANAR < 30 MIN						3
GANAR < 20 MIN						7
ACE						3

La columna de la izquierda son los objetivos del juego, por los que tienen que pelear ambos equipos. En función de si se hacen con estos objetivos consiguen los puntos estimados dependiendo de la posición de cada jugador/entrenador.

En la liga hay un total de 18 jornadas, cada jornada tiene 5 partidos, de liga regular y 8 partidos de playoff incluyendo la final.

Los usuarios tendrán la oportunidad de crear una liga con hasta 5 amigos para ser un total de 6 usuarios por liga. Cada usuario solo puede pertenecer a una liga.

Los usuarios dispondrán de una cantidad de 30 Millones de \*moneda\* para formar su alineación (cantidad estimada, puede cambiar a lo largo del desarrollo de la aplicación). A cada jugador se le asignaran aleatoriamente 5 jugadores y un entrenador, los cuales podrá alinear o vender según le convenga al usuario.

Cada liga dispondrá de un apartado de mercado de jugadores, donde los usuarios podrán realizar pujas por los jugadores que quieran comprar, teniendo estos un precio mayor o menor según su rendimiento cada jornada. El usuario con la puja de mayor cuantía al final

del día se hará con los servicios de dicho jugador, restando el dinero ofrecido del saldo total del usuario.

En el mercado salen 5 jugadores aleatorios cada día.

Los usuarios podrán cambiar la alineación titular todas las veces que quieran antes de que empiece la jornada (viernes 18:00). A esa hora se guardará la alineación que este seleccionada en ese momento y serán esos los jugadores que puntúen en esa jornada.

Los usuarios no podrán tener un saldo negativo, ni dejar una posición sin alinear. Si una de estas 2 situaciones ocurre, el usuario en cuestión no puntuará esa jornada.

Esta idea se basa en la liga Fantasy Marca/Comunio de las ligas de futbol.

## Tecnologías

Las tecnologías a usar serán las siguientes:

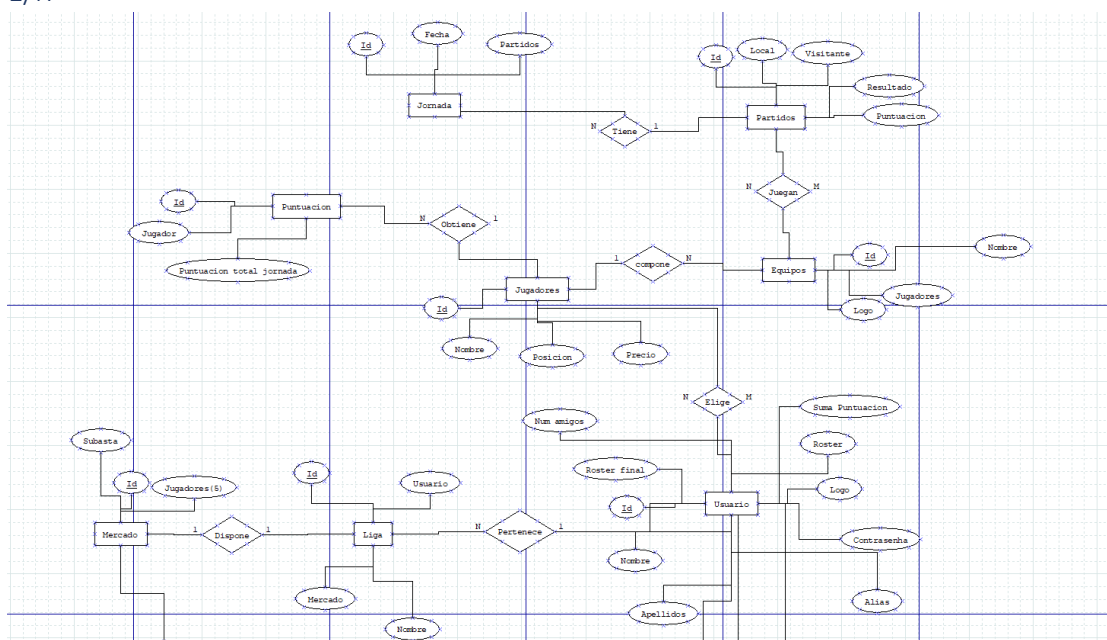
- Api riot developer, java, maven, angular, bootstrap, spring, hibernate, GitHub, Postman

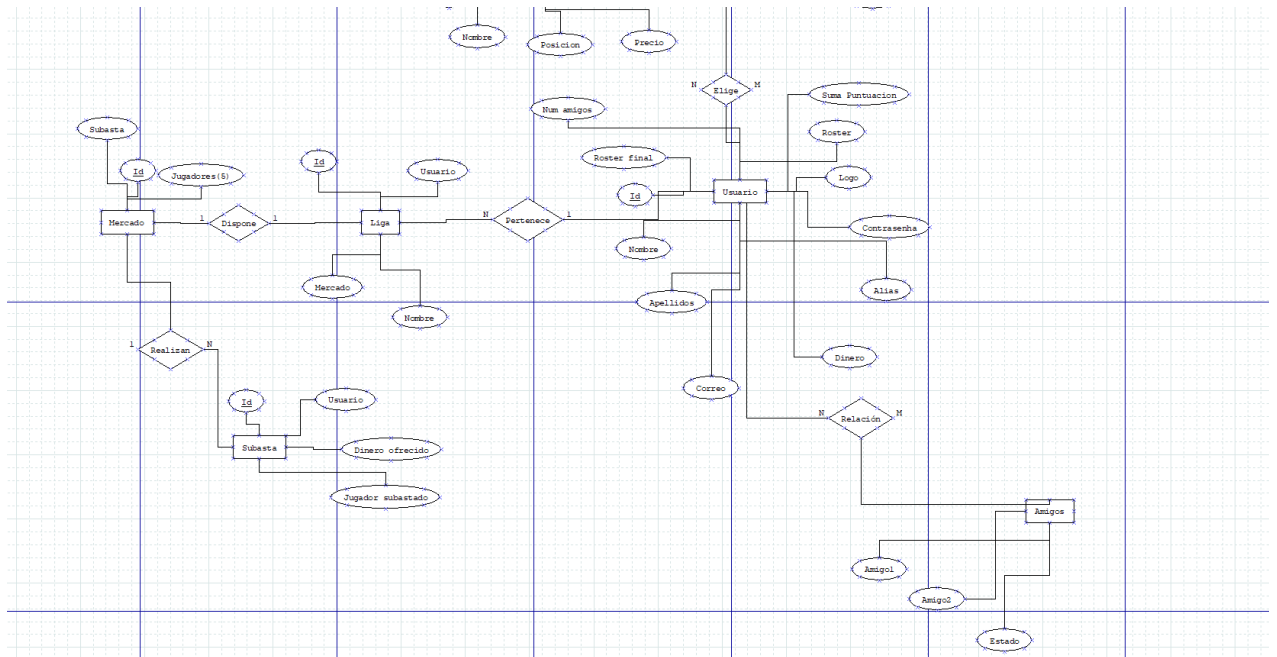
## Base de datos

### Resumen

La base de datos tiene más de trece tablas, cuenta con 4 relaciones N:M, 6 relaciones 1:N y una relación 1:1. Incluye también inserts de los jugadores/coach de la liga, inserts de todos los equipos y una vista donde se puede ver el número total de jugadores ordenado por posición.

E/R





## Codigo

```

DROP DATABASE FantasyBD;
create database FantasyBD;
USE FantasyBD;

create table if not exists equipo(
  id int not null auto_increment primary key,
  nombre varchar(40) not null,
  logo varchar(1000)
);

create table if not exists jugador(
  id int not null auto_increment primary key,
  nombre varchar(40) not null,
  posicion varchar(40) not null,
  precio int not null,
  fk_equipo int not null,
  foreign key (fk_equipo) references equipo (id)
);

create table if not exists jornada(
  id int not null auto_increment primary key,
  fecha timestamp
);

create table if not exists partido(
  id int not null auto_increment primary key,
  e_local varchar(100) not null,

```

```

visitante varchar (100) not null,
resultado int,
fk_jornada int not null,
foreign key (fk_jornada) references jornada (id)
);

create table if not exists puntuacion(
id int not null auto_increment primary key,
puntuacionTotal int default 0,
fk_partido int not null,
foreign key (fk_partido) references partido (id)
);

create table if not exists puntuacion_jugador(
id int not null auto_increment primary key,
fk_puntuacion int not null,
fk_jugador int not null,
foreign key (fk_puntuacion) references puntuacion (id),
foreign key (fk_jugador) references jugador (id)
);

create table if not exists liga(
id int not null auto_increment primary key,
nombre varchar(100) not null,
descripcion varchar(100)
);

create table if not exists usuario(
id int not null auto_increment primary key,
nombre varchar(100) not null,
apellido varchar(100) not null,
rol varchar(100) not null default "otakus",
correo varchar(100) not null,
alias varchar(100) not null ,
contrasena varchar(100) not null,
logo varchar(100),
dinero int default 30000000,
suma_puntuacion int default 0,
fk_liga int not null,
foreign key (fk_liga) references liga (id)
);

create table if not exists rosterFinal_usuario(
id int not null auto_increment primary key,
fk_usuario int not null,
fk_jugador int not null,
foreign key (fk_usuario) references usuario (id),
foreign key (fk_jugador) references jugador (id)
);

create table if not exists roster_usuario(
id int not null auto_increment primary key,
fk_usuario int not null,
fk_jugador int not null,
foreign key (fk_usuario) references usuario (id),

```

```

    foreign key (fk_jugador) references jugador (id)
);

create table if not exists mercado(
    id int not null auto_increment primary key,
    fk_liga int not null,
    foreign key (fk_liga) references liga (id)
);

create table if not exists mercado_jugador(
    id int not null auto_increment primary key,
    fk_mercado int not null,
    fk_jugador int not null,
    foreign key (fk_mercado) references mercado (id),
    foreign key (fk_jugador) references jugador (id)
);

create table if not exists subasta(
    id int not null auto_increment primary key,
    dinero_ofrecido int not null,
    fk_jugador_subastado int not null,
    fk_usuario int not null,
    fk_mercado int not null,
    foreign key (fk_jugador_subastado) references jugador (id),
    foreign key (fk_mercado) references mercado (id),
    foreign key (fk_usuario) references usuario (id)
);

#INSERTS DE EQUIPOS EN LA TABLA EQUIPOS
INSERT INTO equipo (nombre,logo)
values ("EXCEL","https://gamepedia.cursecdn.com/lolesports_gamepedia_en/8/85/Excel_Esportslogo_profile.png"),
("FNATIC","https://gamepedia.cursecdn.com/lolesports_gamepedia_en/tumb/f/fc/Fnaticlogo_square.png/1200px-Fnaticlogo_square.png"),
("G2","https://3hmd7z41jo8fr2m9o2xkrln5-wpengine.netdna-ssl.com/wp-content/uploads/G2-Esports-2020-Logo-1.png"),
("MISFITS","https://i.blogs.es/d06c20/misfits-8fenvxt30/1366_2000.png"),
("ORIGEN","https://gamepedia.cursecdn.com/lolesports_gamepedia_en/1/12/Origenlogo_square.png"),
("ROGUE","https://gamepedia.cursecdn.com/lolesports_gamepedia_en/a/a4/Rogue_%28European_Team%29logo_square.png"),
("SCHALKE
04","https://upload.wikimedia.org/wikipedia/commons/9/97/FC_Schalke_04_Logo.png"),
("SK","https://esports.clashroyale.com/uploaded-images/sk_logo.png?mtime=20190531212514"),
("MAD LIONS","https://www.giantsgaming.pro/wp-content/uploads/2019/05/600px-MAD_Lionslogo_profile.png"),
("VITALITY","https://images.prismic.io/rivalryglhf/ee46659e6ccb4e4d7dab39d05ca09176b6390ed7_team-vitality-lcs-league-of-legends.png?auto=compress,format");

```

```

#INSERTS DE JUGADORES EN LA TABLA JUGADOR

```

```

INSERT INTO jugador(nombre,posicion,precio,fk_equipo)
values ("EXPECT", "TOP", 1, 1),
      ("CAEDREL", "JUNGLE", 1, 1),
      ("MICKEY", "MID", 1, 1),
      ("PATRIK", "ADC", 1, 1),
      ("TORE", "SUPPORT", 1, 1),
      ("KASING", "SUPPORT", 1, 1),
      ("SENDO", "TOP", 1, 1),
      ("TAXER", "JUNGLE", 1, 1),
      ("SPECIAL", "MID", 1, 1),
      ("DEADLY", "ADC", 1, 1),
      ("BOASTER", "MID", 1, 1),
      ("YOUNGBACK/MAPACHE", "COACH", 1, 1);

INSERT INTO jugador(nombre,posicion,precio,fk_equipo)
values ("BWIPO", "TOP", 1, 2),
      ("SELFMADE", "JUNGLE", 1, 2),
      ("NEMESIS", "MID", 1, 2),
      ("REKKLES", "ADC", 1, 2),
      ("HYLISSANG", "SUPPORT", 1, 2),
      ("PRIDE", "TOP", 1, 2),
      ("DAN", "JUNGLE", 1, 2),
      ("MAGIFELIX", "MID", 1, 2),
      ("XMATTY", "ADC", 1, 2),
      ("BRAVADO", "SUPPORT", 1, 2),
      ("CHIBS", "MID", 1, 2),
      ("MITHY", "COACH", 1, 2);

INSERT INTO jugador(nombre,posicion,precio,fk_equipo)
values ("WUNDER", "TOP", 1, 3),
      ("JANKOS", "JUNGLE", 1, 3),
      ("PERKZ", "MID", 1, 3),
      ("CAPS", "ADC", 1, 3),
      ("MIKYX", "SUPPORT", 1, 3),
      ("OSCARININ", "TOP", 1, 3),
      ("NASSER", "JUNGLE", 1, 3),
      ("FEENIIXZ", "MID", 1, 3),
      ("SUPA", "ADC", 1, 3),
      ("AESEENAR", "SUPPORT", 1, 3),
      ("NANDISKO", "TOP", 1, 3),
      ("P1NOY", "ADC", 1, 3),
      ("GRABBZ", "COACH", 1, 3);

INSERT INTO jugador(nombre,posicion,precio,fk_equipo)
values ("DAN DAN", "TOP", 1, 4),
      ("RAZORK", "JUNGLE", 1, 4),
      ("FEBIVEN", "MID", 1, 4),
      ("BVOY", "ADC", 1, 4),
      ("DENYK", "SUPPORT", 1, 4),
      ("AGRESIVOO", "TOP", 1, 4),
      ("KIREI", "JUNGLE", 1, 4),
      ("RONALDO", "MID", 1, 4),
      ("JEZU", "ADC", 1, 4),
      ("DOSS", "SUPPORT", 1, 4),
      ("ROZARA", "JUNGLE", 1, 4),
      ("DECAY", "MID", 1, 4),
      ("JANDRO/AMAZING", "COACH", 1, 4);

INSERT INTO jugador(nombre,posicion,precio,fk_equipo)
values ("ALPHARI", "TOP", 1, 5),
      ("XERXE", "JUNGLE", 1, 5),

```

```

        ("NUKEDUCK", "MID", 1, 5),
        ("UPSET", "ADC", 1, 5),
        ("DESTINY", "SUPPORT", 1, 5),
        ("MYRWN", "TOP", 1, 5),
        ("CRONIIK", "JUNGLE", 1, 5),
        ("KAMIKAZE", "MID", 1, 5),
        ("CORVO", "ADC", 1, 5),
        ("BOLY", "SUPPORT", 1, 5),
        ("HIIVA", "SUPPORT", 1, 5),
        ("GUILHOTO/KAYYS", "COACH", 1, 5);
INSERT INTO jugador(nombre, posicion, precio, fk_equipo)
values ("FINN", "TOP", 1, 6),
        ("INSPIRED", "JUNGLE", 1, 6),
        ("LARSEN", "MID", 1, 6),
        ("HANS SAMA", "ADC", 1, 6),
        ("VANDER", "SUPPORT", 1, 6),
        ("SZYGENDA", "TOP", 1, 6),
        ("ZANZARAH", "JUNGLE", 1, 6),
        ("CZEKOLAD", "MID", 1, 6),
        ("WOOLITE", "ADC", 1, 6),
        ("MYSTIQUES", "SUPPORT", 1, 6),
        ("BLUEKNIGHT", "MID", 1, 6),
        ("BLUMIGAN/FREDY122", "COACH", 1, 6);
INSERT INTO jugador(nombre, posicion, precio, fk_equipo)
values ("ODOAMNE", "TOP", 1, 7),
        ("GILIUS", "JUNGLE", 1, 7),
        ("ABBEDAGGE", "MID", 1, 7),
        ("FORGIVEN", "ADC", 1, 7),
        ("DREAMS", "SUPPORT", 1, 7),
        ("SLEEPING", "TOP", 1, 7),
        ("LUROX", "JUNGLE", 1, 7),
        ("SERTUSS", "MID", 1, 7),
        ("INNAXE", "ADC", 1, 7),
        ("NEON", "ADC", 1, 7),
        ("SIRNUKESALOT", "SUPPORT", 1, 7),
        ("KAMITO", "ADC", 1, 7),
        ("DYLAN FALCO", "COACH", 1, 7);
INSERT INTO jugador(nombre, posicion, precio, fk_equipo)
values ("SACRE", "TOP", 1, 8),
        ("TRICK", "JUNGLE", 1, 8),
        ("JENAX", "MID", 1, 8),
        ("CROWNSHOT", "ADC", 1, 8),
        ("LIMIT", "SUPPORT", 1, 8),
        ("VENTAIR", "TOP", 1, 8),
        ("PHRENIC", "JUNGLE", 1, 8),
        ("REEKER", "MID", 1, 8),
        ("TIGER", "ADC", 1, 8),
        ("TRYMBI", "SUPPORT", 1, 8),
        ("BERTHO", "SUPPORT", 1, 8),
        ("UNLIMITED/KANANI", "COACH", 1, 8);
INSERT INTO jugador(nombre, posicion, precio, fk_equipo)
values ("OROME", "TOP", 1, 9),
        ("SHADOW", "JUNGLE", 1, 9),
        ("HUMANOID", "MID", 1, 9),
        ("CARZZY", "ADC", 1, 9),
        ("KAISER", "SUPPORT", 1, 9),
        ("WERLYB", "TOP", 1, 9),

```

```

        ("KOLDO", "JUNGLE", 1, 9),
        ("HATRIXX", "MID", 1, 9),
        ("FLAKKED", "ADC", 1, 9),
        ("PRIME", "SUPPORT", 1, 9),
        ("FALCO", "SUPPORT", 1, 9),
        ("MAC/KAAS", "COACH", 1, 9);
INSERT INTO jugador(nombre, posicion, precio, fk_equipo)
values ("CABOCHARD", "TOP", 1, 10),
        ("SKEANZ", "JUNGLE", 1, 10),
        ("SAKEN", "MID", 1, 10),
        ("COMP", "ADC", 1, 10),
        ("JACTROLL", "SUPPORT", 1, 10),
        ("MOWGLI", "JUNGLE", 1, 10),
        ("NJI", "JUNGLE", 1, 10),
        ("SELFIE", "MID", 1, 10),
        ("LUCKER", "ADC", 1, 10),
        ("TRAYTON", "SUPPORT", 1, 10),
        ("STEEELBACK", "SUPPORT", 1, 10),
        ("KREPO", "SUPPORT", 1, 10),
        ("XANI", "JUNGLE", 1, 10),
        ("DUKE/MEPHISTO", "COACH", 1, 10);

create or replace view num_posiciones
as select count(posicion) as "Total de supports",
(select count(posicion) from jugador where posicion="ADC") as
"Total de adcs",
(select count(posicion) from jugador where posicion="JUNGLE") as
"Total de junglas",
(select count(posicion) from jugador where posicion="MID") as
"Total de mids",
(select count(posicion) from jugador where posicion="TOP") as
"Total de tops",
(select count(posicion) from jugador where posicion="COACH") as
"Total de coaches"FROM jugador where posicion ="SUPPORT";

```

## Aclaraciones

La diferencia entre las tablas roster\_usuario y rosterFinal\_usuario:

roster\_usuario son todos los jugadores que ha llegado a comprar el usuario mientras que, roster\_final\_usuario corresponde al roster final que elige para cada jornada.

La vista num\_posiciones corresponde al número total de jugadores ordenado por posiciones

**Integrantes del equipo:** Alberto Esteban, Alejandro Martínez, Óscar Ledesma.

**Tutor/a del trabajo:** Davinia de la Rosa.





