



Pontificia Universidad
JAVERIANA
Bogotá

Introducción a IA

Taller 6

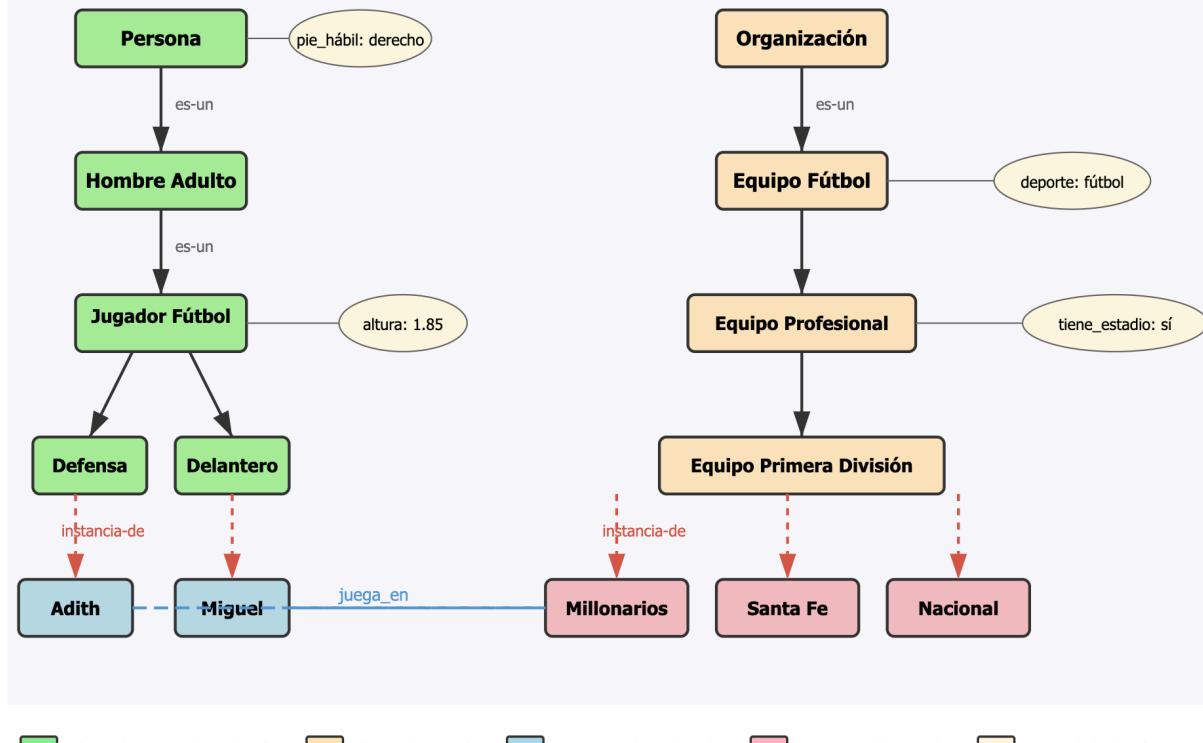
Grupo 5

Santos Alejandro Arellano Olarte // **Santos-arellano**
Alejandra Abaunza Suárez
Daniel Santiago Avila Medina
Jeison Camilo Alfonso Moreno

Taller 6.....	1
Descripción del Modelo.....	2
Funcionalidades Implementadas.....	4
Código.....	4

Descripción del Modelo

Esta red semántica representa el conocimiento sobre jugadores de fútbol y equipos, implementando un sistema de herencia de propiedades mediante relaciones **es-un** (jerarquía de clases) e **instanciade** (pertenencia a clases).



Ampliación Implementada

Nueva Jerarquía de Equipos: Se ha añadido una rama completa para representar equipos de fútbol con las siguientes características:

- **Organización** → Clase raíz para entidades organizativas
- **Equipo Fútbol** → Especialización deportiva
- **Equipo Profesional** → Equipos con estructura profesional
- **Equipo Primera División** → Máxima categoría
- **Instancias:** Millonarios, Santa Fe, Nacional

Los jugadores (Miguel y Adith) están vinculados al equipo Millonarios mediante la relación **juega_en**.

Funcionalidades Implementadas

- Herencia múltiple de propiedades a través de la jerarquía es-un
- Resolución de conflictos: las propiedades más específicas sobrescriben las heredadas
- Consultas bidireccionales: desde instancias hacia clases y viceversa
- Relaciones entre diferentes jerarquías (jugador-equipo)
- Sistema de inferencia para deducir propiedades no explícitas

Código

```
% =====
% TALLER 6 - REDES SEMÁNTICAS EN PROLOG
%
% =====
%
% Implementación de un sistema de redes semánticas que permite
% representar jerarquías de clases y herencia de propiedades

%
% Evitar advertencias de predicados separados
:- discontiguous es_un/2.
:- discontiguous instancia_de/2.
:- discontiguous propiedad/3.

%
% =====
%
% JERARQUÍA DE CLASES Y OBJETOS
%
% =====
```

```
% Jerarquía: persona -> hombre_adulto -> jugador_futbol ->
defensa/delantero
es_un(hombre_adulto, persona).
es_un(jugador_futbol, hombre_adulto).
es_un(defensa, jugador_futbol).
es_un(delantero, jugador_futbol).

% Instancias específicas de jugadores
instancia_de(miguel, delantero).
instancia_de(adith, defensa).

% =====
% PROPIEDADES DE CLASES E INSTANCIAS
% =====

% Propiedades generales por clase
propiedad(persona, pie_habil, derecho).
propiedad(hombre_adulto, altura, 1.80).
propiedad(jugador_futbol, altura, 1.85).
propiedad(jugador_futbol, numero_goles, 3).
propiedad(jugador_futbol, patea, balon).
propiedad(defensa, numero_goles, 1).
propiedad(delantero, numero_goles, 5).

% Propiedades individuales de los jugadores
propiedad(miguel, equipo, millonarios).
propiedad(adith, equipo, millonarios).

% =====
% REGLAS DE HERENCIA
% =====

% Determina si una clase hereda de otra (transitivo)
subclase_de(X, Y) :- es_un(X, Y).
```

```
subclase_de(X, Y) :-  
    es_un(X, Z),  
    subclase_de(Z, Y).  
  
% Verifica si un objeto pertenece a una clase  
pertenece_a(X, Y) :- instancia_de(X, Y).  
pertenece_a(X, Y) :-  
    instancia_de(X, Z),  
    subclase_de(Z, Y).  
  
% ======  
% SISTEMA DE HERENCIA DE PROPIEDADES  
% ======  
  
% Busca una propiedad, primero en el objeto, luego en su jerarquía  
obtener_propiedad(Objeto, Atributo, Valor) :-  
    propiedad(Objeto, Atributo, Valor), !.  
  
obtener_propiedad(Objeto, Atributo, Valor) :-  
    instancia_de(Objeto, Clase),  
    obtener_propiedad_clase(Clase, Atributo, Valor), !.  
  
obtener_propiedad(Clase, Atributo, Valor) :-  
    es_un(Clase, Superclase),  
    obtener_propiedad_clase(Superclase, Atributo, Valor).  
  
% Auxiliar para buscar en la jerarquía de clases  
obtener_propiedad_clase(Clase, Atributo, Valor) :-  
    propiedad(Clase, Atributo, Valor), !.  
  
obtener_propiedad_clase(Clase, Atributo, Valor) :-  
    es_un(Clase, Superclase),  
    obtener_propiedad_clase(Superclase, Atributo, Valor).
```

```
% =====
% CONSULTAS Y UTILIDADES
% =====

% Muestra todas las propiedades que tiene un objeto
listar_propiedades(Objeto) :-
    write('Propiedades de '), write(Objeto), write(':'), nl,
    forall(obtener_propiedad(Objeto, Atributo, Valor),
        (write(' '), write(Atributo), write(' = '), write(Valor), nl)).

% Verifica si un objeto puede realizar una acción
puede(Objeto, Accion, Sobre) :-
    obtener_propiedad(Objeto, Accion, Sobre).

% Lista todos los objetos que pertenecen a una clase
objetos_de_clase(Clase, Objeto) :-
    pertenece_a(Objeto, Clase).

% =====
% EQUIPOS DE FÚTBOL COLOMBIANO
% =====

% Jerarquía de equipos deportivos
es_un(equipo_futbol, organizacion).
es_un(equipo_profesional, equipo_futbol).
es_un(equipo_primera_division, equipo_profesional).

% Los tres grandes del fútbol colombiano
instancia_de(millonarios, equipo_primera_division).
instancia_de(santa_fe, equipo_primera_division).
instancia_de(nacional, equipo_primera_division).

% Características comunes de los equipos
propiedad(organizacion, tipo, deportiva).
```

Pontificia Universidad Javeriana de Bogotá

```
propiedad(equipo_futbol, deporte, futbol).
propiedad(equipo_futbol, numero_jugadores, 11).
propiedad(equipo_profesional, tiene_estadio, si).
propiedad(equipo_primera_division, division, primera).

% Datos específicos de cada equipo
propiedad(millonarios, ciudad, bogota).
propiedad(millonarios, fundacion, 1946).
propiedad(millonarios, colores, 'azul y blanco').
propiedad(santa_fe, ciudad, bogota).
propiedad(santa_fe, fundacion, 1941).
propiedad(santa_fe, colores, 'rojo y blanco').
propiedad(nacional, ciudad, medellin).
propiedad(nacional, fundacion, 1947).
propiedad(nacional, colores, 'verde y blanco').

% Encuentra en qué equipo juega alguien
juega_en(Jugador, Equipo) :-
    obtener_propiedad(Jugador, equipo, Equipo).

% Encuentra jugadores del mismo equipo
companeros(Jugador1, Jugador2) :-
    juega_en(Jugador1, Equipo),
    juega_en(Jugador2, Equipo),
    Jugador1 \= Jugador2.

% =====
% DEMOSTRACIÓN DEL SISTEMA
% =====

% Muestra las capacidades del sistema de redes semánticas
demo :-
    nl, write('== SISTEMA DE REDES SEMÁNTICAS =='), nl, nl,
```

```
write('1. Herencia de propiedades:'), nl,
write('    Altura de Adith: '),
obtener_propiedad(adith, altura, AlturaAdith),
write(AlturaAdith), nl,

write('    Pie hábil de Miguel: '),
obtener_propiedad(miguel, pie_habil, PieMiguel),
write(PieMiguel), nl, nl,

write('2. Diferencias entre posiciones:'), nl,
write('    Goles típicos de un defensa: '),
obtener_propiedad(defensa, numero_goles, GolesDefensa),
write(GolesDefensa), nl,
write('    Goles típicos de un delantero: '),
obtener_propiedad(delantero, numero_goles, GolesDelantero),
write(GolesDelantero), nl, nl,

write('3. Capacidades de los jugadores:'), nl,
write('    ¿Puede Miguel patear un balón? '),
(puede(miguel, patea, balon) -> write('Sí') ; write('No')), nl, nl,

write('4. Relaciones entre jugadores:'), nl,
write('    ¿Son Miguel y Adith compañeros? '),
(companeros(miguel, adith) -> write('Sí') ; write('No')), nl, nl,

write('5. Información del equipo Millonarios:'), nl,
listar_propiedades(millonarios), nl,

write('6. Perfil completo de Miguel:'), nl,
listar_propiedades(miguel).
```