

## Progetto ModSem A.A. 2024/2025

*Lorenzo Bonincontro*

*Albero Aiello*

### Descrizione del dominio

Il dominio scelto per l'ontologia riguarda il mondo dei videogiochi, con un focus particolare sulla narrazione, i personaggi e le relazioni tra i diversi elementi che compongono un gioco. Questo approccio permette di rappresentare le storie, i protagonisti e i contenuti narrativi che rendono ogni videogioco unico, offrendo agli utenti una visione completa e strutturata.

I videogiochi non sono solo prodotti commerciali ma anche strumenti narrativi complessi che possono raccontare storie profonde e coinvolgenti. La nostra ontologia è progettata per rappresentare:

- Personaggi: Le loro caratteristiche, come sesso, età, e il ruolo nella storia.
- Relazioni: Come i personaggi interagiscono tra loro o con il mondo del gioco.
- Contenuti aggiuntivi: Come DLC o espansioni che arricchiscono i giochi.

Il dominio dei videogiochi è estremamente vasto e complesso, offrendo innumerevoli possibilità di rappresentazione. Noi ci siamo concentrati su una specifica parte del dominio, privilegiando gli aspetti narrativi e le relazioni tra personaggi, giochi e contenuti aggiuntivi. Tuttavia, siamo consapevoli che questa non è l'unica possibile rappresentazione. Esistono molti altri aspetti, che non rientrano nell'ambito della nostra ontologia.

### Motivazioni

La struttura dell'ontologia rende più facile per gli utenti esplorare giochi, capire le loro storie e identificare collegamenti tra personaggi o serie. Non esistono molte ontologie che rappresentano questa dimensione narrativa del videogioco: ad esempio, la "[Videogame Ontology](#)" analizza principalmente eventi di gioco e risultati, come achievements, ma non si focalizza su personaggi, relazioni tra di essi...

I videogiochi oggi sono molto più che semplici passatempi. Sono:

- Prodotti vendibili: Un'industria multimiliardaria con tantissime opzioni per i consumatori.

- Racconti interattivi: Storie e mondi in cui i giocatori possono immergersi.
- Competizioni e sfide: Per chi ama mettere alla prova le proprie abilità.

### **Per chi è fatta?**

- **Neofiti:** Persone che non conoscono bene il mondo dei videogiochi e cercano informazioni facili da trovare.

L'obiettivo generale dell'ontologia è rendere l'accesso ai dati semplice e intuitivo, per utenti che non hanno una conoscenza approfondita del dominio. Un utente inesperto potrebbe, ad esempio, voler cercare informazioni su un videogioco per valutare un possibile acquisto. Questo potrebbe includere la possibilità di esplorare i giochi in base al genere o ai personaggi. Allo stesso modo, un utente potrebbe essere interessato ai giochi sviluppati da uno studio specifico a cui è particolarmente affezionato. Infine, potrebbe desiderare un elenco dei titoli appartenenti a una determinata serie, per capire quali ha già giocato e quali gli mancano per completarla.

### **Domande di competenza**

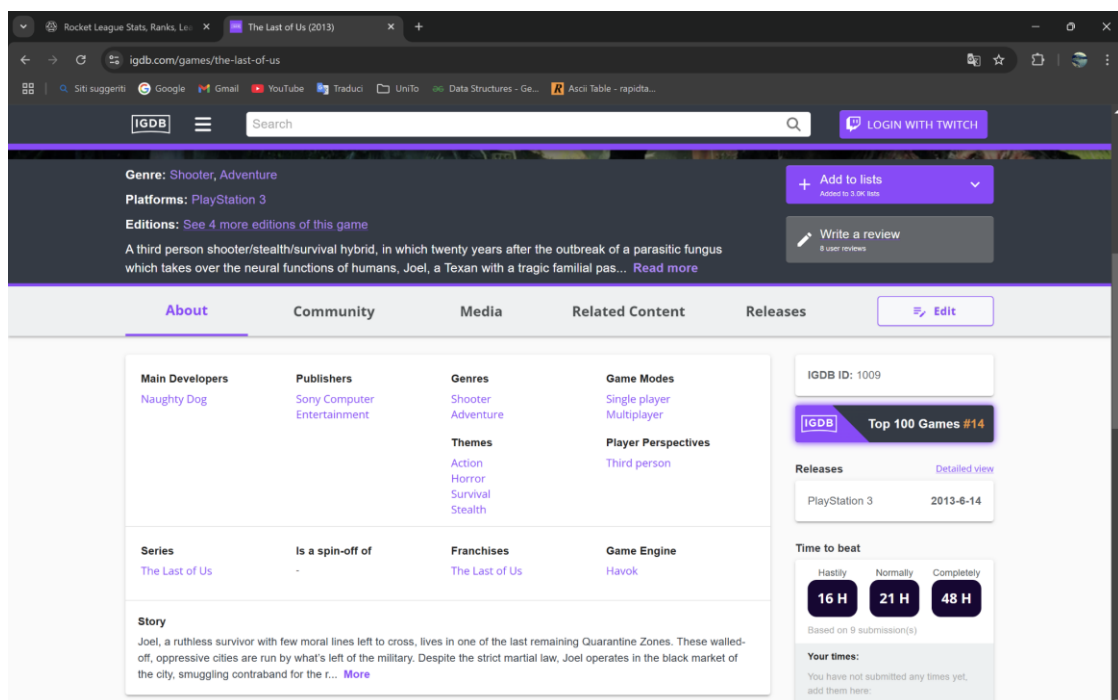
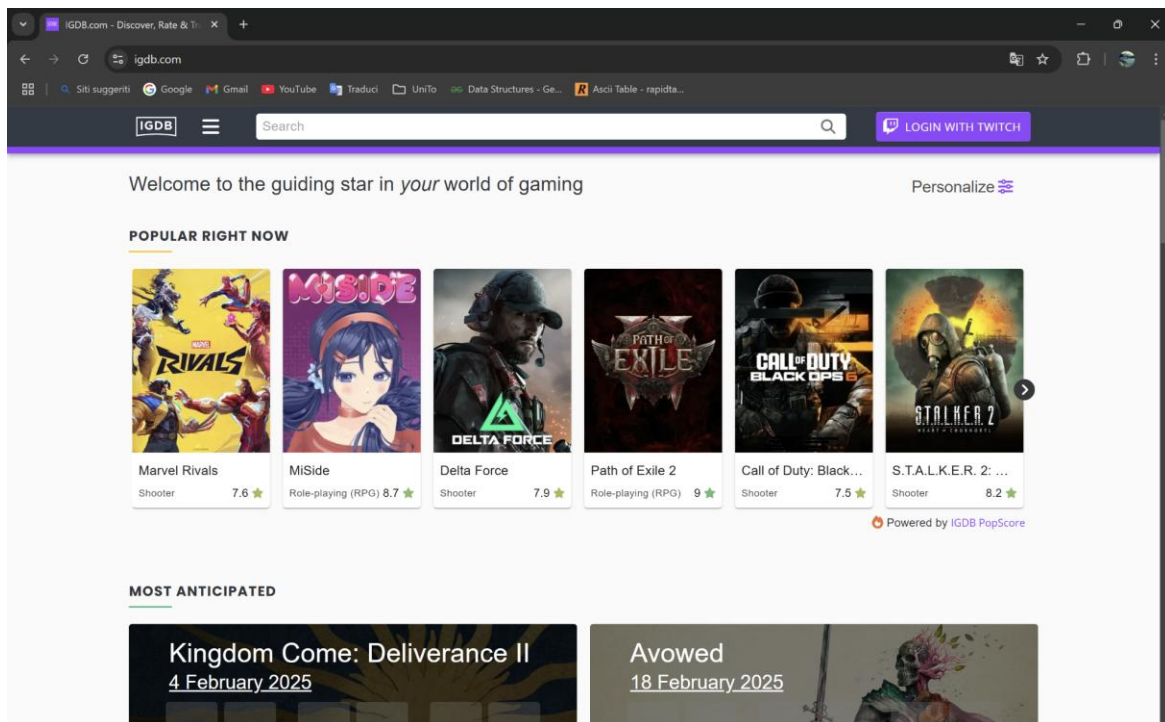
**L'ontologia può “rispondere” a numerose domande, qui ne riportiamo alcune**

- Qual è il titolo dei videogiochi di una serie?
- Di che serie fa parte un videogioco?
- Qual è lo studio che ha sviluppato un videogioco?
- Quanti giochi appartengono a una determinata serie?
- Chi è il protagonista di un videogioco?
- quali personaggi appaiono in un gioco x?
- Quanti sono i titoli di una serie?
- Qual è l'ordine cronologico dei videogiochi di una serie?
- Tutti i titoli con protagonista femminile/maschile?
- Tutti i titoli con storie d'amore contraccambiate?
- Quali sono i giochi dove i protagonisti uccidono qualcuno?
- Quanti tipi di giochi ci sono?
- Quali giochi sono multiplayer/singleplayer?
- Il gioco x possiede DLC/ quali giochi hanno il DLC?
- il gioco x che di che genere è?
- ...

# Documentazione

## Sitografia

IGDB: <https://www.igdb.com/>

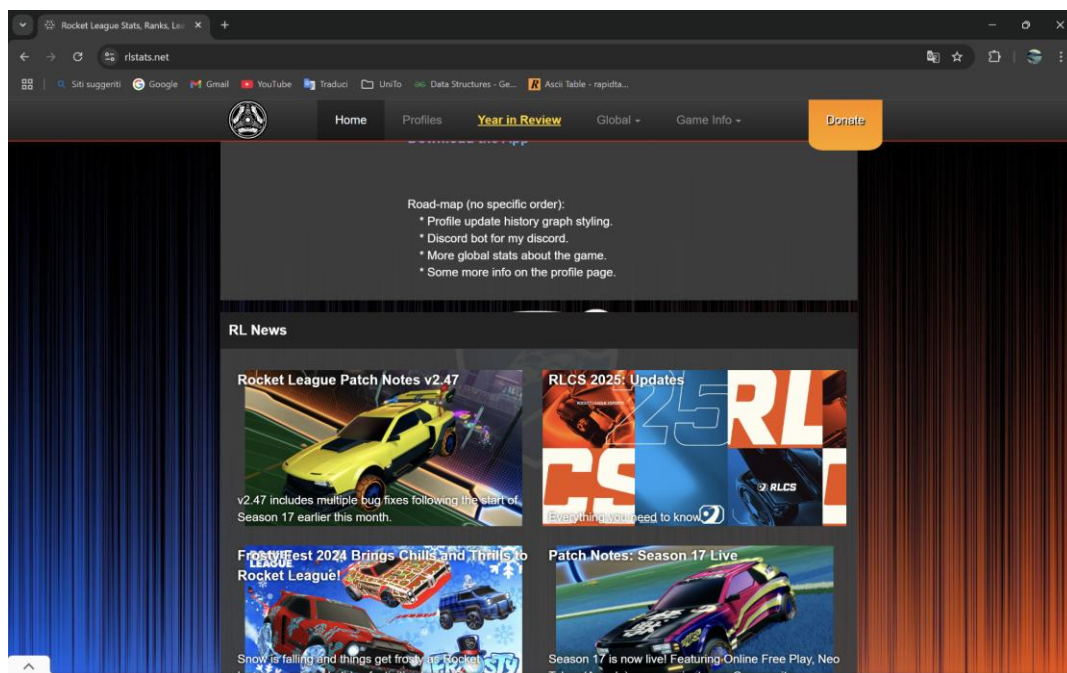


IGDB (Internet Game Database) è un database online dedicato a fornire informazioni dettagliate sui videogiochi. È utilizzato da appassionati, sviluppatori e aziende per accedere a una vasta quantità di dati su giochi, piattaforme, generi, sviluppatori e contenuti associati.

Esempi di dati estratti (es. relativi a The Last of Us):

- Nome del gioco (The Last of Us).
- Genere (Azione, Avventura).
- GameMode (SinglePlayer, MultiPlayer)
- Developers/Publishers (naughty-dog..)

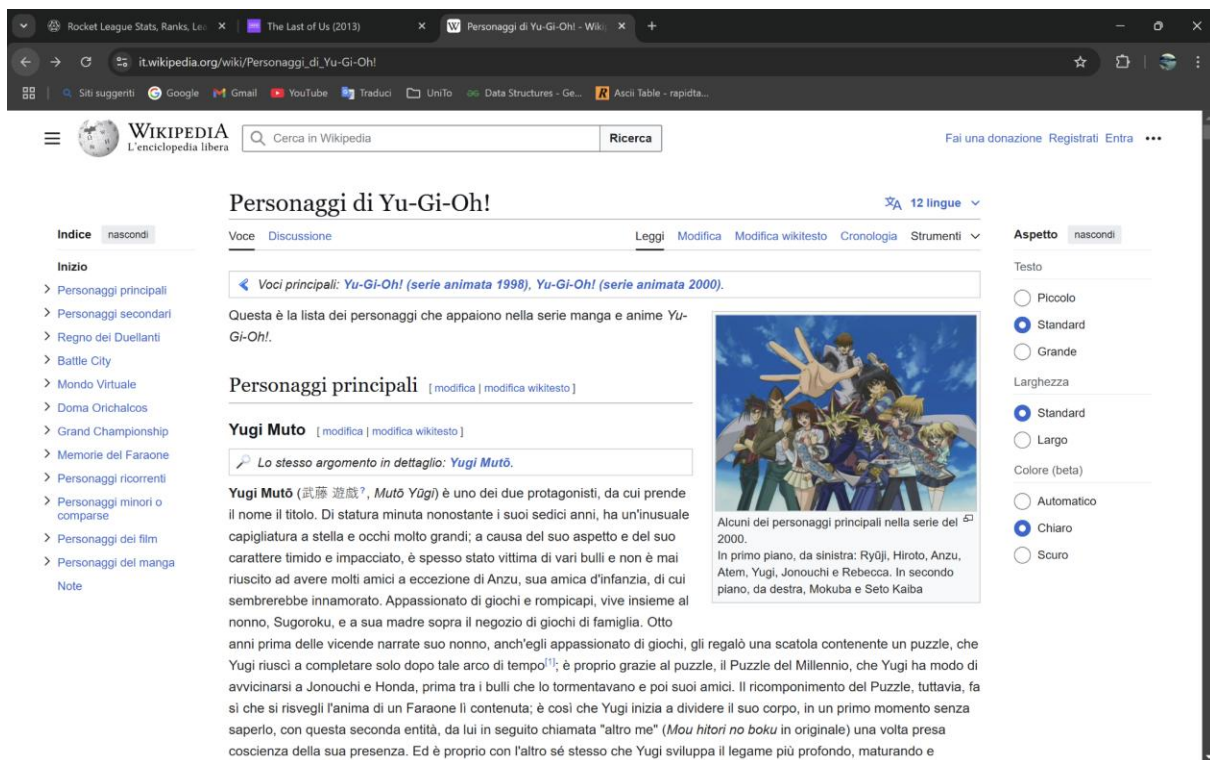
rlStats: <https://rlstats.net/>



rlStats è un sito web specializzato nel fornire statistiche e informazioni dettagliate sul gioco "Rocket League". La piattaforma è particolarmente utile per i giocatori e gli appassionati che desiderano approfondire aspetti specifici del gioco, come le mappe, le vetture, e le performance dei giocatori.

È stato utilizzato per ottenere un elenco completo e aggiornato delle vetture disponibili in Rocket League, che è stato integrato nella nostra ontologia per rappresentare i diversi tipi di auto utilizzabili nel gioco, e per raccogliere informazioni dettagliate sulle mappe di gioco, che sono state incluse per rappresentare le ambientazioni in cui si svolgono le partite.

**wikipedia:** [https://it.wikipedia.org/wiki/Personaggi\\_di\\_Yu-Gi-Oh!](https://it.wikipedia.org/wiki/Personaggi_di_Yu-Gi-Oh!)



The screenshot shows the Wikipedia page for "Personaggi di Yu-Gi-Oh!". The page is in Italian and includes a search bar at the top. The main content area is titled "Personaggi di Yu-Gi-Oh!" and contains a list of principal characters. The first entry is "Yugi Muto", which is expanded to show a detailed description of the character. The description mentions that Yugi Muto is a protagonist, is a shy and clumsy person, and is often a victim of bullies. It also mentions his relationship with his grandfather, Sugoroku, and his discovery of the Millennium Puzzle. The page also includes a sidebar with navigation links and a right-hand panel with settings for the page's appearance.

**Personaggi di Yu-Gi-Oh!**

Voci principali: *Yu-Gi-Oh! (serie animata 1998)*, *Yu-Gi-Oh! (serie animata 2000)*.

Questa è la lista dei personaggi che appaiono nella serie manga e anime *Yu-Gi-Oh!*.

**Personaggi principali** [modifica] [modifica wikitesto]

**Yugi Muto** [modifica] [modifica wikitesto]

Lo stesso argomento in dettaglio: *Yugi Muto*.

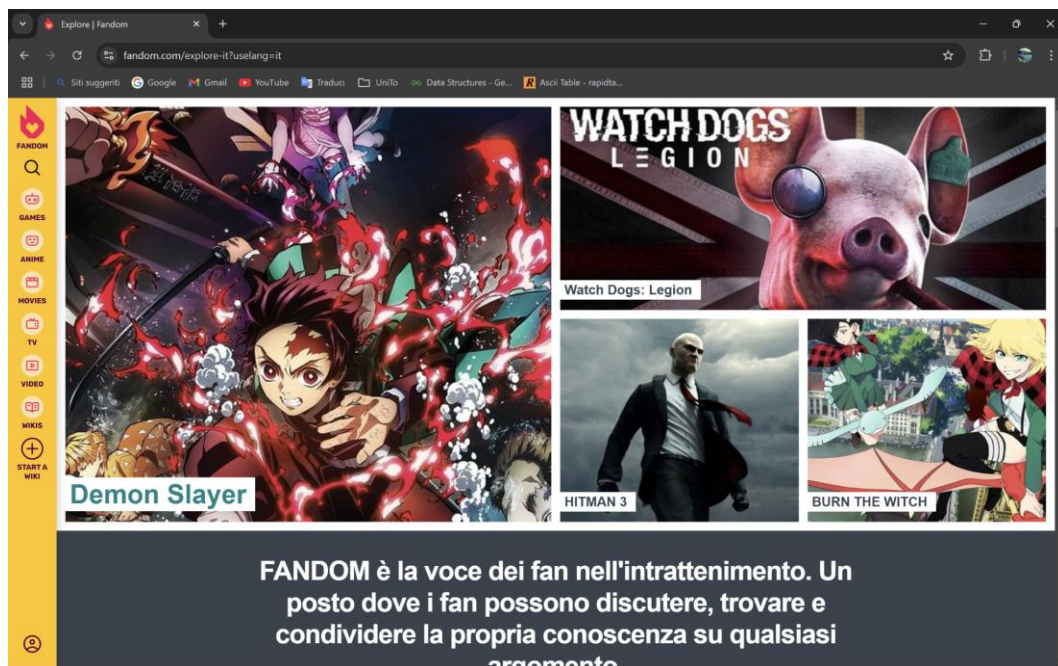
**Yugi Mutō** (武藤 遊戯?, *Mutō Yūgi*) è uno dei due protagonisti, da cui prende il nome il titolo. Di statura minuta nonostante i suoi sedici anni, ha un'inusuale capigliatura a stella e occhi molto grandi; a causa del suo aspetto e del suo carattere timido e impacciato, è spesso stato vittima di vari bulli e non è mai riuscito ad avere molti amici a eccezione di Anzu, sua amica d'infanzia, di cui sembrerebbe innamorato. Appassionato di giochi e rompicapi, vive insieme al nonno, Sugoroku, e a sua madre sopra il negozio di giochi di famiglia. Otto anni prima delle vicende narrate suo nonno, anch'egli appassionato di giochi, gli regalò una scatola contenente un puzzle, che Yugi riuscì a completare solo dopo tale arco di tempo<sup>[1]</sup>; è proprio grazie al puzzle, il Puzzle del Millennio, che Yugi ha modo di avvicinarsi a Jonouchi e Honda, prima tra i bulli che lo tormentavano e poi suoi amici. Il ricomponimento del Puzzle, tuttavia, fa sì che si risvegli l'anima di un Faraone il contenuta; è così che Yugi inizia a dividere il suo corpo, in un primo momento senza saperlo, con questa seconda entità, da lui in seguito chiamata "altro me" (*Mou hitori no boku* in originale) una volta presa coscienza della sua presenza. Ed è proprio con l'altro sé stesso che Yugi sviluppa il legame più profondo, maturando e

Tale sito è una pagina wikipedia dedicata ai personaggi dell'universo di Yu-Gi-Oh!, una delle serie anime e manga più iconiche. La pagina si propone come un archivio organizzato per introdurre e descrivere i personaggi principali e secondari che compongono l'universo narrativo, offrendo informazioni essenziali per i fan e gli appassionati.

Sono state prese informazioni utili per costruire l'ontologia:

- Ruoli e relazioni tra i personaggi
- Nomi dei protagonisti
- Storia e contesto narrativo
- Asset narrativi → carte

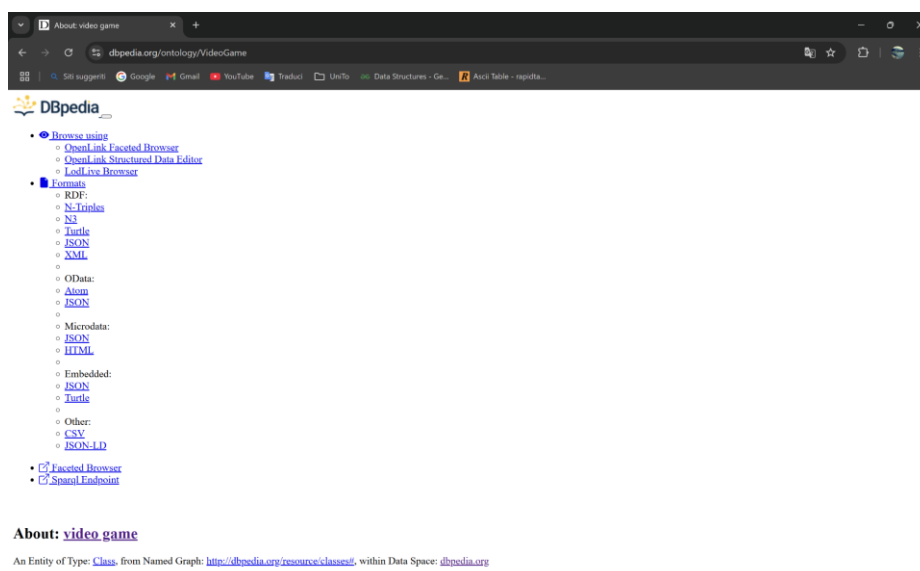
**Fandom:** <https://www.fandom.com/explore-it?uselang=it>



Fandom è una piattaforma collaborativa basata su wiki, dove le community di appassionati creano e condividono contenuti su una vasta gamma di argomenti, inclusi videogiochi, serie TV, film, libri e molto altro. È particolarmente utile per esplorare e comprendere contenuti narrativi e dettagli del mondo dei videogiochi, grazie alla sua ricchezza di informazioni sulle storie, personaggi e universi di gioco.

**Ontologie già esistenti:**

<https://dbpedia.org/ontology/VideoGame>





<https://www.wikidata.org/wiki/Q7889>

video game (Q7889)

electronic game with user interface and visual feedback

vg | computer game | computer games | videogame | videogames

Language	Label	Description	Also known as
English	video game	electronic game with user interface and visual feedback	vg computer game computer games videogame videogames

**Statements**

Instance of

- type of arts  
0 references
- software category  
0 references

subclass of

- electronic game  
0 references
- entertainment software

<http://vocab.linkeddata.es/vgo/>

The Video Game Ontology

Release 19 December 2014

This version:  
<http://vocab.linkeddata.es/vgo/>

Latest version:  
<http://url.org/net/VideoGameOntology>

Revision  
1.0

Authors:  
[Jari Parkkinen](#), Innovation and Software Department, School of Business and Management, Lappeenranta University of Technology, Finland  
[Elio Radulovic](#), Ontology Engineering Group, Universidad Politécnica de Madrid  
[María Poveda](#), Ontology Engineering Group, Universidad Politécnica de Madrid  
[Daniel Garzjo](#), Ontology Engineering Group, Universidad Politécnica de Madrid

This work is licensed under a [Attribution-NonCommercial-ShareAlike 2.0 Generic](#).

**Abstract**

The Video Game Ontology is aimed at modelling video game related information. The main goal is to capture knowledge about events that happen in video games and information about players. This ontology has been created as a joint work by researchers at Lappeenranta University of Technology, Finland and Universidad Politécnica de Madrid, Spain

The Video Game Ontology can be downloaded from [here](#).

**Table of Contents**

- 1. Introduction
  - 1.1 Namespace declarations

**Per essere in linea con le ontologie già esistenti, abbiamo utilizzato:**

- **Schema:** <https://schema.org/Product> → Concetto di prodotto come di “qualcosa che può essere venduto”

Game è sottoclasse di schema:Product → Game è un tipo di Product

Significa che ogni istanza di Game è anche un'istanza di Product, ma non tutte le istanze di Product sono Games.

schema:Product rappresenta un concetto più generico (un prodotto in senso ampio), mentre Game è un concetto più specifico.

- **wikidata:** <https://www.wikidata.org/wiki/Q7889> → Rappresentazione standard del concetto di "videogioco"

ex:Game owl:equivalentClass <<https://www.wikidata.org/wiki/Q7889>> →  
Game è concettualmente equivalente alla classe su wikidata, che rappresenta il concetto di VideoGame

Ogni istanza di Game sarà automaticamente anche un'istanza di Q7889.

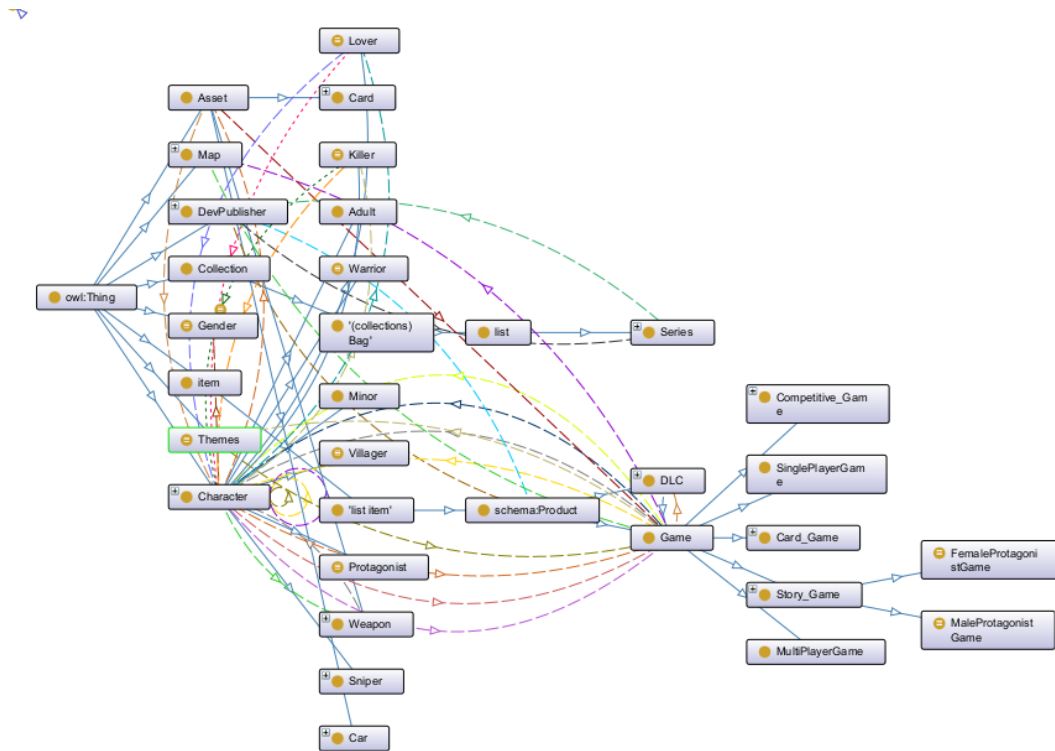
Ogni proprietà definita su Q7889 sarà applicabile anche a game

Inoltre abbiamo utilizzato l'Ontology Design Pattern lista per modellare i giochi di una Serie (es. Fifa).

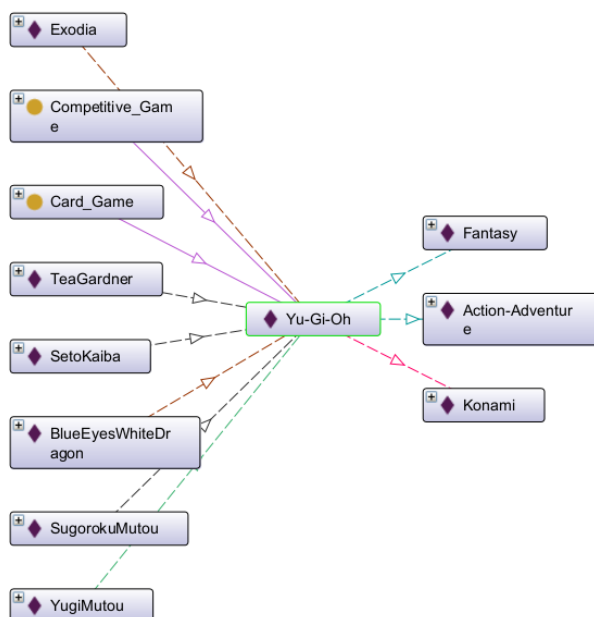


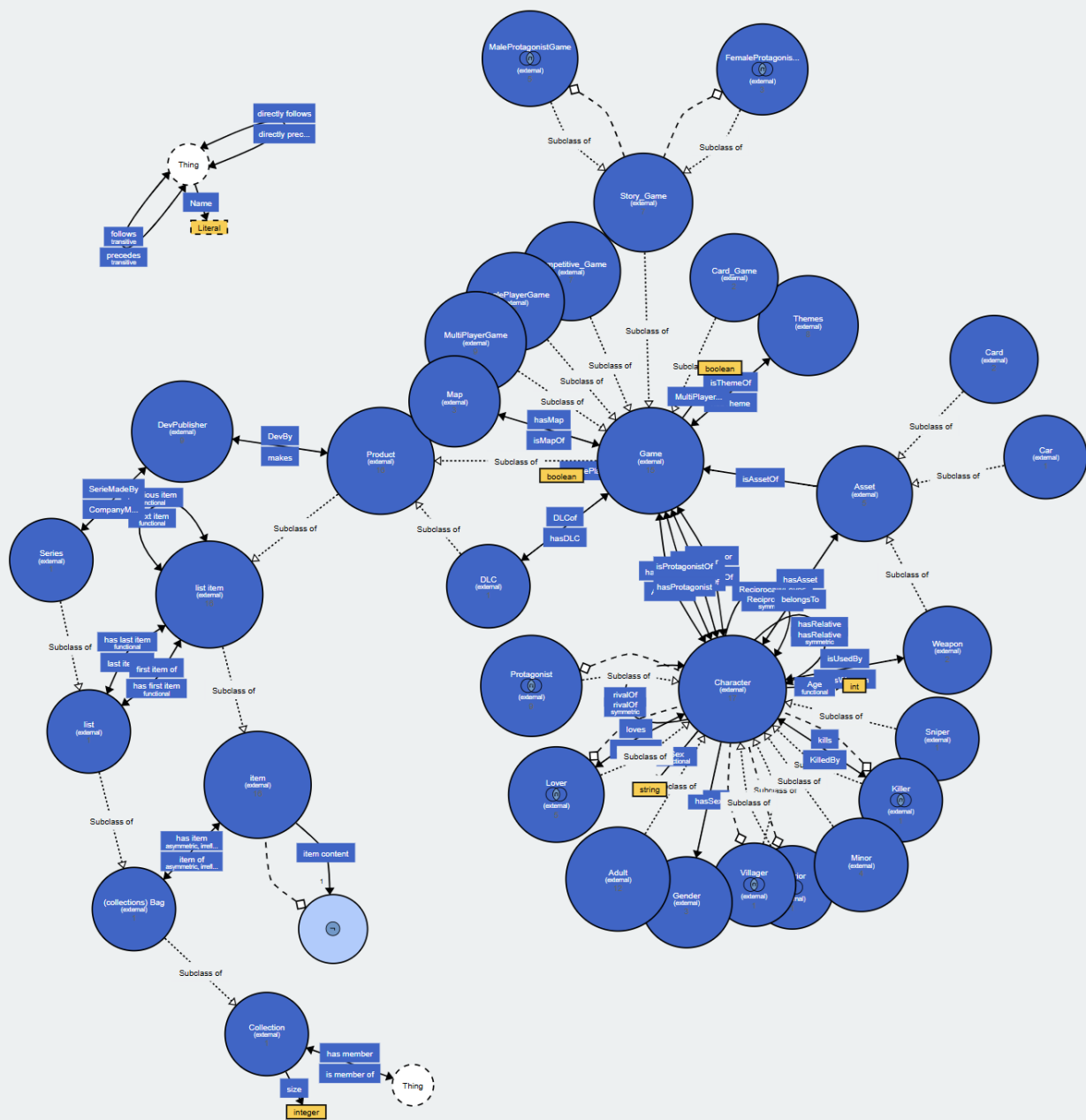
## Visualizzazione

Tassonomia:



A-Box:



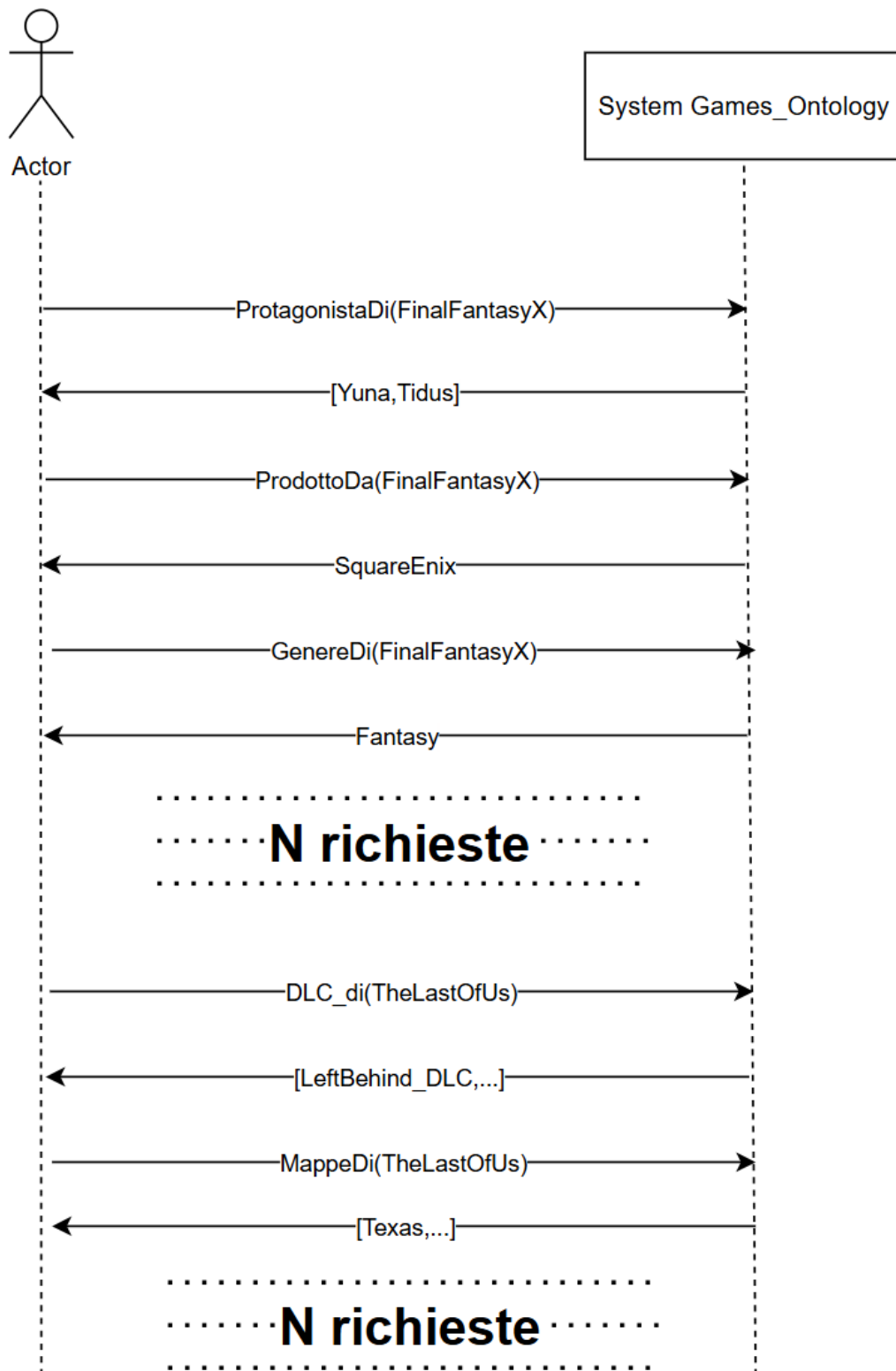


**Esempio Triple da sezione Descrizione Dominio.**

Soggetto	Predicato	Oggetto
Red Dead Redemption2	madeBy	Rockstar Games
Arthur Morgan	protagonistOf	Red Dead Redemption 2
Arthur Morgan	kills	AntonyForeman
AntonyForeman	AppearsIn	Red Dead Redemption 2
RocketLeague	hasTheme	Action-Adventure
Fifa2022	hasTheme	Sports
Fifa2023	DevBy	EASports
Tidus	isLovedBy	Yuna
Yuna	hasSex	Female
Yuna	isProtagonistOf	FinalFantasyX
TheLastOfUs	hasCharacter	Joel
PumpActionShotgun	isAsseOf	Red Dead Redemption2
PumpActionShotgun	isUsedBy	Arthur Morgan

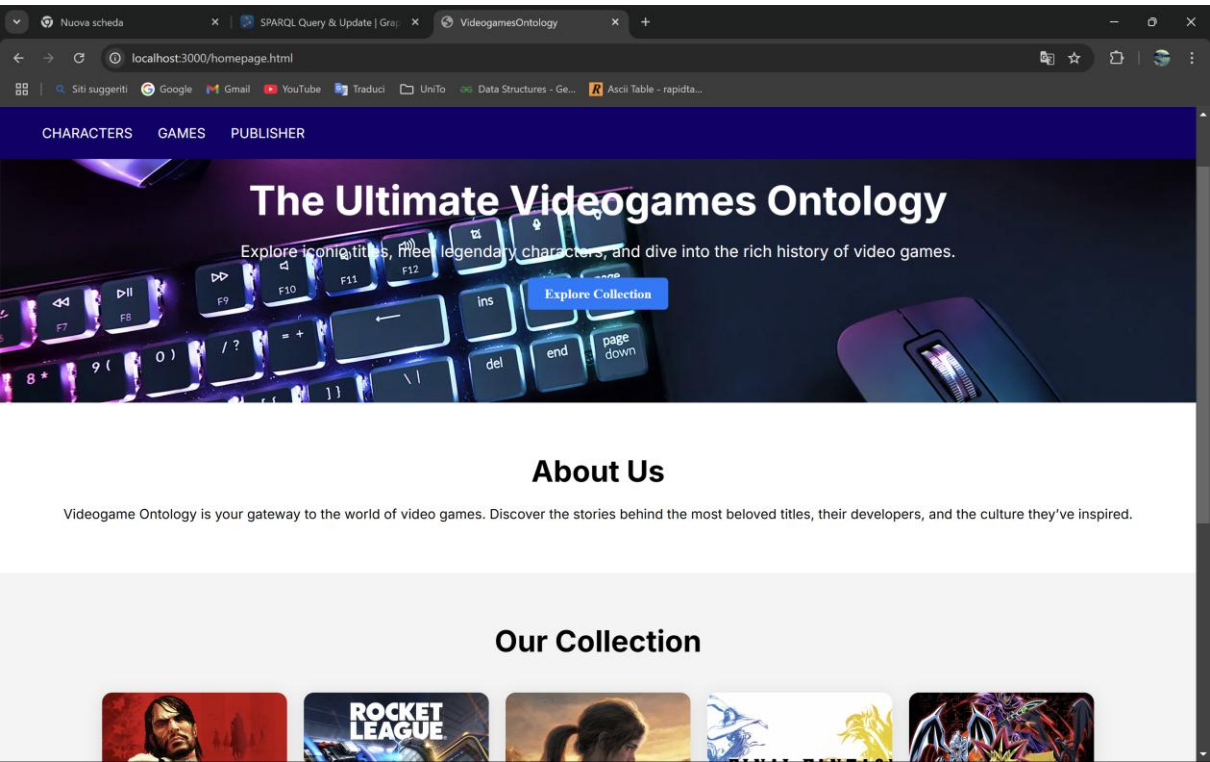
## Interazione Utente

Esempio Interazione dell'utente con il sistema

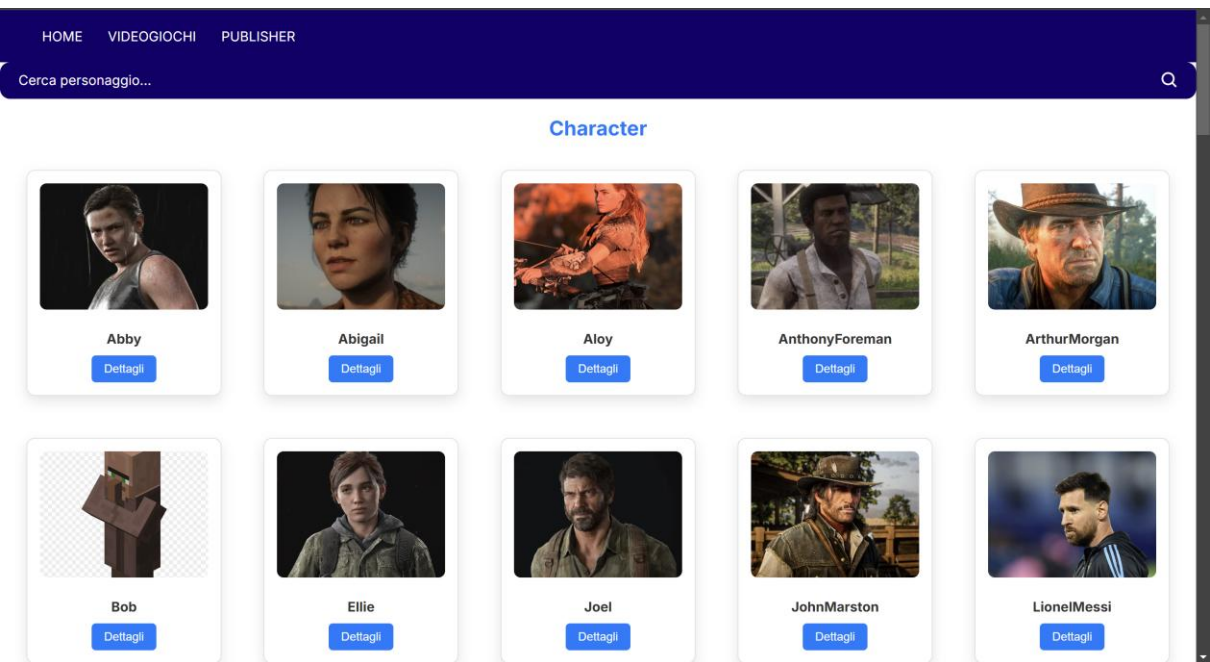


# Mock Up Interfaccia Client

## HomePage:



## Characters:



## Characters details:

HOME VIDEOGIOCHI PUBLISHER PERSONAGGI

### Tidus

#### Dettagli

Età: 21 anni

Sesso: Male

#### Asset Associati:

Nessun asset disponibile

#### Mappa Associata:

N/A

#### Appare in:

FinalFantasyX

#### Relazioni:

## Videogames:

HOME PUBLISHER PERSONAGGI

### HearthStone

[Dettagli](#)

### HorizonZeroDawn

[Dettagli](#)

### Minecraft

[Dettagli](#)

### RedDeadRedemption

[Dettagli](#)

### RedDeadRedemption2

[Dettagli](#)

### RocketLeague

[Dettagli](#)



Games details:

[HOME](#) [CHARACTERS](#) [GAMES](#) [PUBLISHER](#)

RedDeadRedemption

Game Information

**Platforms:** Microsoft Windows, Nintendo Switch, PlayStation 3, PlayStation 4, Xbox 360

**Comment:** RedDeadRedemption Game

**Theme:** Action-Adventure

**Main Characters:** Abigail, JohnMarston

**Release Year:** 2010

**multiplayerMode:** No

VIDEOGAMES WEBSITE

[Home](#)  
[Chi siamo](#)  
[Contatti](#)  
[Politiche di privacy](#)

LEGALE

[Condizioni generali](#)  
[Politica per i Cookie](#)

CONTATTI

123 456 789  
[Whatsapp](#)  
Dal Lunedì al Venerdì  
dalle 09:00 alle 20:00

LINK SOCIAL

[Facebook](#)  
[Twitter](#)  
[Instagram](#)

Publisher:

[HOME](#) [VIDEOGIOCHI](#) [PERSONAGGI](#)

Konami

Scopri i giochi sviluppati da questo publisher!

Dettagli

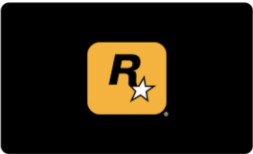
Scopri i giochi sviluppati da questo publisher!

Dettagli

PsyonixStudios

Scopri i giochi sviluppati da questo publisher!

Dettagli



RockstarGames

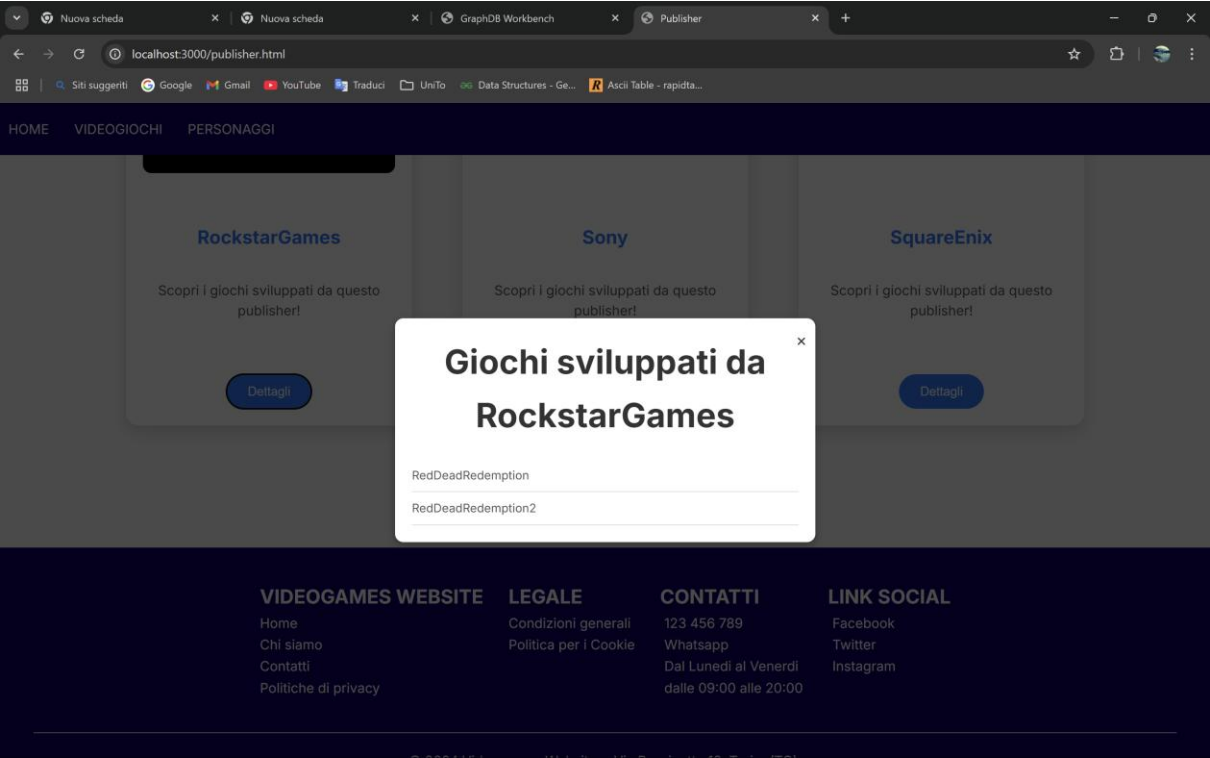
SONY

Sony

SQUARE ENIX.

SquareEnix

PublisherDetails:



# SPARQL Queries

## 1. Trova tutti i personaggi che appartengono a più di 3 giochi:

```
1 PREFIX owl: <http://www.w3.org/2002/07/owl#>
2 PREFIX ex: <http://www.semanticweb.org/loren_16pbl1h/ontologies/2025/0/Games_Ontology/>
3 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
4 PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
5
6 SELECT ?character (COUNT(DISTINCT ?game) AS ?gameCount)
7 WHERE {
8     ?game <http://www.semanticweb.org/loren_16pbl1h/ontologies/2025/0/Games_Ontology/hasCharacter> ?character .
9 }
10 GROUP BY ?character
11 HAVING (?gameCount > 3)
12
13
14
```

	character	gameCount
1	ex:LionelMessi	*5**xsd:integer

## 2. Elenco di giochi per ogni tema, ordinati per numero di giochi per tema:

```
1 PREFIX owl: <http://www.w3.org/2002/07/owl#>
2 PREFIX ex: <http://www.semanticweb.org/loren_16pbl1h/ontologies/2025/0/Games_Ontology/>
3 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
4 PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
5
6 SELECT ?theme ?game
7 WHERE {
8     ?game <http://www.semanticweb.org/loren_16pbl1h/ontologies/2025/0/Games_Ontology/hasTheme> ?theme .
9 }
10 ORDER BY ?theme
11
12
13
14
```

	theme	game
1	ex:Action-Adventure	ex:RedDeadRedemption
2	ex:Action-Adventure	ex:Yu-Gi-Oh
3	ex:Action-Adventure	ex:Minecraft
4	ex:Action-Adventure	ex:RocketLeague
5	ex:Action-Adventure	ex:TheLastOfUs
6	ex:Fantasy	ex:Yu-Gi-Oh
7	ex:Fantasy	ex:FinalFantasyX
8	ex:Fantasy	ex:Hearthstone
9	ex:Post-Apocalyptic	ex:TheLastOfUs
10	ex:Sandbox	ex:Minecraft
11	ex:Sports	ex:Fifa2021
12	ex:Strategy	ex:Hearthstone

## 3. Individui con un età maggiore di 30

```

1 PREFIX owl: <http://www.w3.org/2002/07/owl#>
2 PREFIX ex: <http://www.semanticweb.org/loren_16pbl1h/ontologies/2025/0/Games_Ontology/>
3 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
4 PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
5
6 SELECT ?individual ?age
7 WHERE {
8   ?individual <http://www.semanticweb.org/loren_16pbl1h/ontologies/2025/0/Games_Ontology/Age> ?age .
9   FILTER (?age > 30)
10 }
11

```

	individual	age
1	ex:Joel	"32"^^xsd:int
2	ex:AnthonyForeman	"35"^^xsd:int
3	ex:LionelMessi	"37"^^xsd:int
4	ex:ArthurMorgan	"53"^^xsd:int
5	ex:SugorokuMutou	"70"^^xsd:int

#### 4. Giochi sviluppati da “EASports”:

```

1 PREFIX owl: <http://www.w3.org/2002/07/owl#>
2 PREFIX ex: <http://www.semanticweb.org/loren_16pbl1h/ontologies/2025/0/Games_Ontology/>
3 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
4 PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
5
6 SELECT ?game
7 WHERE {
8   ?game <http://www.semanticweb.org/loren_16pbl1h/ontologies/2025/0/Games_Ontology/DevBy>
9         <http://www.semanticweb.org/loren_16pbl1h/ontologies/2025/0/Games_Ontology/EASports> .
10 }
11

```

	game
1	ex:Fifa2021
2	ex:Fifa2022
3	ex:Fifa2023
4	ex:Fifa2024
5	ex:Fifa2025

#### 5. Elenco dei personaggi con Età e Sesso:

```

1 PREFIX owl: <http://www.w3.org/2002/07/owl#>
2 PREFIX ex: <http://www.semanticweb.org/loren_16pbl1h/ontologies/2025/0/Games_Ontology/>
3 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
4 PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
5
6 SELECT ?character ?age ?sex
7 WHERE {
8   ?character a <http://www.semanticweb.org/loren_16pbl1h/ontologies/2025/0/Games_Ontology/Character> ;
9              <http://www.semanticweb.org/loren_16pbl1h/ontologies/2025/0/Games_Ontology/Age> ?age ;
10             <http://www.semanticweb.org/loren_16pbl1h/ontologies/2025/0/Games_Ontology/Sex> ?sex .
11 }

```

	character	age	sex
1	ex:Abby	"19"^^xsd:int	"Female"
2	ex:Abigail	"28"^^xsd:int	"Female"
3	ex:JohnMarston	"26"^^xsd:int	"Male"
4	ex:Aloy	"22"^^xsd:int	"Female"
5	ex:AnthonyForeman	"35"^^xsd:int	"Male"
6	ex:ArthurMorgan	"53"^^xsd:int	"Male"
7	ex:Ellie	"28"^^xsd:int	"Female"
8	ex:Joel	"32"^^xsd:int	"Male"
9	ex:LionelMessi	"37"^^xsd:int	"Male"
10	ex:SetoKaiba	"16"^^xsd:int	"Male"
11	ex:YugiMutou	"16"^^xsd:int	"Male"
12	ex:Steve	"20"^^xsd:int	"Male"
13	ex:SugorokuMutou	"70"^^xsd:int	"Male"
14	ex:TeaGardner	"16"^^xsd:int	"Female"
15	ex:Tidus	"21"^^xsd:int	"Male"
16	ex:Yuna	"14"^^xsd:int	"Female"

## 6. Giochi con sviluppatori che hanno creato più di 3 giochi:

```

1 PREFIX owl: <http://www.w3.org/2002/07/owl#>
2 PREFIX ex: <http://www.semanticweb.org/loren_16pbl1h/ontologies/2025/0/Games_Ontology/>
3 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
4 PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
5
6 SELECT ?developer (COUNT(?game) AS ?gameCount)
7 WHERE {
8     ?game <http://www.semanticweb.org/loren_16pbl1h/ontologies/2025/0/Games_Ontology/DevBy> ?developer .
9 }
10 GROUP BY ?developer
11 HAVING (?gameCount > 3)
12
13
14

```

	developer	gameCount
1	ex:EASports	"5"^^xsd:integer

## 7. Tutti i DLC e i giochi a cui sono associati:

```

1 PREFIX owl: <http://www.w3.org/2002/07/owl#>
2 PREFIX ex: <http://www.semanticweb.org/loren_16pbl1h/ontologies/2025/0/Games_Ontology/>
3 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
4 PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
5
6 SELECT ?dlc ?game
7 WHERE {
8     ?game <http://www.semanticweb.org/loren_16pbl1h/ontologies/2025/0/Games_Ontology/hasDLC> ?dlc .
9 }
10
11

```

	dlc	game
1	ex:LeftBehind_DLC	ex:TheLastOfUs

## 8. Top 5 giochi con il maggior numero di personaggi:

```

1 PREFIX owl: <http://www.w3.org/2002/07/owl#>
2 PREFIX ex: <http://www.semanticweb.org/loren_16pbl1h/ontologies/2025/0/Games_Ontology/>
3   PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
4   PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
5
6 SELECT ?game (COUNT(?character) AS ?characterCount)
7 WHERE {
8   ?game <http://www.semanticweb.org/loren_16pbl1h/ontologies/2025/0/Games_Ontology/hasCharacter> ?character .
9 }
10 GROUP BY ?game
11 ORDER BY DESC(?characterCount)
12 LIMIT 5

```

	game	characterCount
1	ex:Yu-Gi-Oh	"4^^xsd:integer
2	ex:TheLastOfUs2	"2^^xsd:integer
3	ex:RedDeadRedemption	"2^^xsd:integer
4	ex:RedDeadRedemption2	"2^^xsd:integer
5	ex:Minecraft	"2^^xsd:integer

## 9. Tema più comune nei giochi:

```

PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX ex: <http://www.semanticweb.org/loren_16pbl1h/ontologies/2025/0/Games_Ontology/>
  PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
  PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT ?theme (COUNT(?game) AS ?gameCount)
WHERE {
  ?game <http://www.semanticweb.org/loren_16pbl1h/ontologies/2025/0/Games_Ontology/hasTheme> ?theme .
}
GROUP BY ?theme
ORDER BY DESC(?gameCount)
LIMIT 1

```

	theme	gameCount
1	ex:Action-Adventure	"5^^xsd:integer



## 10. (WikiData) Anno di rilascio di un determinato videogioco:

```
1 PREFIX wd: <http://www.wikidata.org/entity/>
2 PREFIX wdt: <http://www.wikidata.org/prop/direct/>
3 PREFIX wikibase: <http://wikiba.se/ontology#>
4 PREFIX bd: <http://www.bigdata.com/rdf#>
5
6 SELECT ?gameLabel ?releaseYear WHERE {
7   VALUES ?game { wd:Q28062624 }
8   ?game wdt:P577 ?releaseDate .
9   BIND(YEAR(?releaseDate) AS ?releaseYear)
10  SERVICE wikibase:label { bd:serviceParam wikibase:language "[AUTO_LANGUAGE],en". }
11 } ORDER BY ?gameLabel
```

8 risultati in 406 ms

gameLabel	releaseYear
The Last of Us	2013
The Last of Us	2014
The Last of Us	2014
The Last of Us	2014

## 11. (WikiData) Su quali piattaforme posso giocare a questo videogioco:

```
1 PREFIX wd: <http://www.wikidata.org/entity/>
2 PREFIX wdt: <http://www.wikidata.org/prop/direct/>
3 PREFIX wikibase: <http://wikiba.se/ontology#>
4 PREFIX bd: <http://www.bigdata.com/rdf#>
5
6 SELECT ?platform ?platformLabel WHERE {
7   VALUES ?game { wd:Q28062624 }
8   ?game wdt:P400 ?platform .
9   SERVICE wikibase:label { bd:serviceParam wikibase:language "[AUTO_LANGUAGE],en". }
10 }
11
```

4 risultati in 193 ms

platform	platformLabel
wd:Q1406	Microsoft Windows
wd:Q10683	PlayStation 3
wd:Q5014725	PlayStation 4
wd:Q63184502	PlayStation 5

## Estensioni

### Applicazione Client

Il sito web è un'applicazione backend sviluppata in Node.js, utilizzando il framework Express. Il progetto è progettato per interagire con un'ontologia semantica dedicata ai videogiochi, ospitata su un server GraphDB. Attraverso l'uso di query SPARQL, l'applicazione recupera dati strutturati dall'ontologia, trasformandoli in contenuti fruibili e presentandoli agli utenti in formato HTML.

### Tecnologie Utilizzate

1. **Node.js – Express (javascript)**: Per la gestione lato backend.
2. **HTML, CSS e JS**: per la gestione del frontend
3. **GraphDB**: Database a grafo RDF per ospitare e interrogare l'ontologia.
4. **SPARQL**: Linguaggio di query per estrarre informazioni dall'ontologia RDF.

### Funzionamento Generale

1. **Backend con Node.js**:
  - L'applicazione è costruita su **Express**, che gestisce le richieste HTTP e le associa a specifiche rotte.
  - Le rotte inviano query SPARQL al server **GraphDB** utilizzando la libreria **axios**.
  - I dati restituiti da GraphDB vengono elaborati e resi disponibili al client.
2. **Interazione con GraphDB**:
  - Il server GraphDB ospita l'ontologia RDF, che rappresenta videogiochi, personaggi e le loro relazioni.
  - Utilizzando SPARQL, vengono estratte informazioni specifiche (ad esempio, nomi di giochi, tipi di personaggi, ecc.).
3. **Rendering dei dati**:
  - I risultati delle query vengono restituiti:
    - In formato HTML

### Possibili Sviluppi Futuri

- **Miglioramento dell'interfaccia utente**: Integrare un frontend interattivo.
- **Ricerca avanzata**: Implementare filtri e ricerche personalizzate su giochi o personaggi.
- **Gestione CRUD**: Permettere l'aggiunta o modifica di dati nell'ontologia tramite interfaccia.

## Regole SWRL

Abbiamo creato 6 regole SWRL per definire concetti di supporto per l'ontologia

- 1) Adult: Un personaggio di età maggiore di 18 anni:

*Character(?c) ^ Age(?c, ?cage) ^ swrlb:greaterThanOrEqualTo(?cage, 18) -> Adult(?c)*

**Spiegazione:**

Se un'entità è un personaggio (Character) e la sua età (Age) è maggiore o uguale a 18, allora quell'entità viene classificata come adulto (Adult).

- 2) Minor: Un personaggio di età minore di 18 anni:

*Character(?c) ^ Age(?c, ?cage) ^ swrlb:lessThan(?cage, 18) -> Minor(?c)*

**Spiegazione:**

Se un'entità è un personaggio (Character) e la sua età (Age) è minore di 18, allora quell'entità viene classificata come minorenne (Minor).

- 3) MutuallyLoves: Relazione tra personaggi che si amano a vicenda:

*Love(?x, ?z) ^ Love(?z, ?x) -> ReciprocallyLoves(?x, ?z)*

**Spiegazione:**

Se un personaggio x ama un altro personaggio z e il personaggio z ricambia l'amore verso x, allora esiste una relazione di amore reciproco (ReciprocallyLoves) tra x e z.

- 4) ThemeEquals - Propaga il tema di un gioco a un altro se quest'ultimo segue il primo:

*Game(?x) ^ Game(?y) ^ autogen6:follows(?x, ?y) ^ hasTheme(?y, ?theme) -> hasTheme(?x, ?theme)*

**Spiegazione:**

Se un gioco x segue (follows) un altro gioco y e il gioco y ha un tema (hasTheme), allora anche il gioco x eredita lo stesso tema.

- 5) MultiPlayerGame - Identifica un gioco come multiplayer se supporta la modalità multigiocatore:

*MultiPlayerMode(?x, true) ^ Game(?x) -> MultiPlayerGame(?x)*

**Spiegazione:**

Se un'entità è un gioco (Game) e ha abilitata la modalità multigiocatore (MultiPlayerMode con valore true), allora viene classificata come gioco multiplayer (MultiPlayerGame).

- 6) SinglePlayerGame - Identifica un gioco come singleplayer se supporta la modalità giocatore singolo:

*SinglePlayerMode(?x, true) ^ Game(?x) -> SinglePlayerGame(?x)*

**Spiegazione:**

Se un'entità è un gioco (Game) e ha abilitata la modalità giocatore singolo (SinglePlayerMode con valore true), allora viene classificata come gioco singleplayer (SinglePlayerGame).