

Università degli studi di Torino

Dipartimento di Informatica



Modellazione concettuale per il Web Semantico

Relazione di progetto

Ontologia degli strumenti musicali

Studente: Lorenzo Botto

a.a. 2021/2022

Indice

1. Motivazioni	1
2. Requirements	2
3. Descrizione del dominio	3
4. Documentazione.....	4
4.1 Ontologie utilizzate.....	4
4.1.1 MO: The Music Ontology	4
4.1.2 FOAF	4
4.1.3 Dublin Core	5
4.1.4 Pattern Bag	6
4.2 Object properties, Data Properties e Individuals	7
4.3 Esempio reale	10
5. Documentazione LODE	15
6. Visualizzazione ontologia	15
6.1 Tassonomia classi	15
7. Interazione utente	17
7.1 Interazione con l'utente tramite navbar	17
7.2 Queries SPARQL.....	19
7.2.1 Ricerca delle chitarre con relative informazioni	19
7.2.2 Ricerca dei batteristi con relative informazioni.....	21
7.2.3 Ricerca delle band musicali (gruppi) con relative informazioni.....	22
7.2.4 Ricerca delle case produttrici di bassi elettrici con relative informazioni.....	23
7.2.5 Ricerca degli artisti (no batteristi) che hanno la casa produttrice di origine giapponese con relative informazioni	25
8. Applicazione Client	26
9. Sitografia.....	28

1. Motivazioni

Nessuno può negare che la musica accompagni gran parte dei momenti della nostra vita, riportandoci indietro nei ricordi, sensazioni, stati d'animo. Attorno a noi tutto è musica, anche quelli che percepiamo come rumori della città. E tutti questi suoni e melodie ci influenzano profondamente.

Recenti studi portati avanti dall'Università di Glasgow hanno anche individuato le canzoni che "fanno bene" all'umore: tra queste troviamo "**Sexual healing**" di Marvin Gaye, "**Comfortably numb**" dei Pink Floyd, "**Cold turkey**" di John Lennon, "**Healing hands**" di Elton John, "**More than a feeling**" dei Boston e "**Alright**" dei Supergrass. Ma, trattandosi di canzoni, la materia è soggettiva e ognuno potrebbe allungare la lista a dismisura.

La ricerca semantica permette ad un utente di raccogliere informazioni riguardanti il dominio degli strumenti musicali. Questi ultimi sono anche collegati agli artisti che suonano quel determinato strumento, così da poter ampliare il proprio bagaglio personale di informazioni. Inoltre, gli strumenti musicali hanno una relazione con la propria casa produttrice, la quale può essere utile anche per i negozi musicali che possono venire a conoscenza in maniera rapida di quali strumenti una casa produttrice produce. Da appassionato di musica ritengo che sia veramente utile avere un'ontologia che comprenda tutti questi aspetti e non solo.

2. Requirements

Il dominio di applicazione riguarda il fornire agli utenti, appassionati di musica, tutte le informazioni sugli strumenti musicali, sugli artisti che suonano lo strumento musicale, sulle case produttrici degli strumenti musicali e sulle band musicali (o solisti) in cui gli artisti suonano. In questo modo è molto facile per l'utente ampliare il proprio bagaglio di conoscenza perché direttamente dallo strumento musicale è possibile sapere da chi è suonato e in che band, così da poter andare ad ascoltare lo strumento.

Avendo inserito anche le case produttrici degli strumenti musicali, è anche utile per i negozi di musica quando vogliono recuperare le informazioni relative allo strumento musicale e alla casa produttrice, oppure per sapere quali altri prodotti sono venduti dalla casa produttrice per poterli acquistare.

Per distinguere tutte le varie categorie che possono interessare agli utenti, gli strumenti musicali sono suddivisi in:

- tastiere;
- chitarre;
- microfoni;
- batterie;
- bassi.

Le case produttrici, a loro volta, sono suddivise in:

- case produttrici tastiere;
- case produttrici chitarre;
- case produttrici microfoni;
- case produttrici piatti (per le batterie);
- case produttrici fusti (per le batterie);
- case produttrici bassi.

3. Descrizione del dominio

Come accennato nel paragrafo precedente, il dominio comprende gli strumenti musicali in relazione con gli artisti che lo suonano. Per restituire diverse informazioni all'utente si definiscono anche:

- **case produttrici**: descrive la casa produttrice di uno strumento musicale;
- **artisti**: gli artisti in campo musicale che suonano un determinato strumento musicale;
- **oggetti**: con quale strumento è suonato lo strumento musicale, se esiste;
- **generi musicali**: il genere musicale nel quale lo strumento musicale è suonato.

Gli strumenti musicali presi in considerazione si collocano in generi musicali differenti. Sono stati aggiunti anche artisti differenti, che suonano in una band musicale oppure come solisti ed inoltre che suonano tipologie di strumenti musicali differenti (per esempio diverse tipologie di chitarre). Infatti, alcuni strumenti musicali sono suddivisi in diverse categorie come per le chitarre:

- chitarre acustiche;
- chitarre elettriche:
 - chitarre a più corde;
 - chitarre Gibson;
 - chitarre Stratocaster;
 - chitarre Telecaster.

Per la suddivisione di queste categorie e per la relazione degli strumenti musicali con i generi musicali, ho analizzato un sito molto utile: https://kytary.it/come_scegliere_una_chitarra_elettrica.

Per tutto il resto, mi sono affidato alla mio bagaglio personale di conoscenza della musica.

4. Documentazione

4.1 Ontologie utilizzate

4.1.1 MO: The Music Ontology

The Music Ontology fornisce un vocabolario per la pubblicazione e il collegamento di una vasta gamma di dati relativi alla musica sul Web. L'ontologia musicale è specificata usando OWL, che fornisce un insieme di costrutti per descrivere i modelli di dominio in RDF. La utilizzo per gli strumenti musicali, il genere musicale, l'artista, band musicali e solisti.

4.1.2 FOAF

FOAF (acronimo di friend of a friend - Amico di un amico) è un'ontologia comprensibile dal computer atta a descrivere persone, con le loro attività e le relazioni con altre persone e oggetti. Chiunque può usare FOAF per descriversi. FOAF è un vocabolario descrittivo espresso in RDF ed è definita usando OWL. I computer possono usare FOAF, ad esempio, per cercare tutte le persone che vivono in Europa, o tutte le persone che hanno un amico in comune, questo appunto perché permette di definire le relazioni tra persone.

4.1.3 Dublin Core

Il Dublin Core (dal nome della città americana nello Ohio) è un sistema di metadati costituito da un nucleo di elementi essenziali ai fini della descrizione di qualsiasi materiale digitale.

È costituito dai seguenti elementi:

- **titolo**: Nome dato alla risorsa;
- **autore**: Entità che ha la responsabilità principale della produzione del contenuto della risorsa;
- **soggetto**: Argomento principale della risorsa;
- **descrizione**: Spiegazione del contenuto della risorsa;
- **editore**: Entità responsabile della pubblicazione della risorsa;
- **autore di contributo subordinato**: Entità responsabile della produzione di un contributo al contenuto della risorsa;
- **data**: Data associata ad un evento del ciclo di vita della risorsa;
- **tipo**: Natura o genere del contenuto della risorsa;
- **formato**: Manifestazione fisica o digitale della risorsa;
- **identificatore**: Riferimento univoco alla risorsa nell'ambito di un dato contesto;
- **fonte**: Riferimento a una risorsa dalla quale è derivata la risorsa in oggetto;
- **lingua**: Lingua del contenuto intellettuale della risorsa;
- **relazione**: Riferimento ad una risorsa correlata;
- **copertura**: Estensione o scopo del contenuto della risorsa;
- **gestione dei diritti**: Informazione sui diritti esercitati sulla risorsa.

4.1.4 Pattern Bag

Il pattern Bag è caratterizzato da una collezione che può avere più copie di ogni oggetto. Questa viene eseguita attraverso l'entità item. L'item collega esattamente una risorsa attraverso la relazione itemContent.

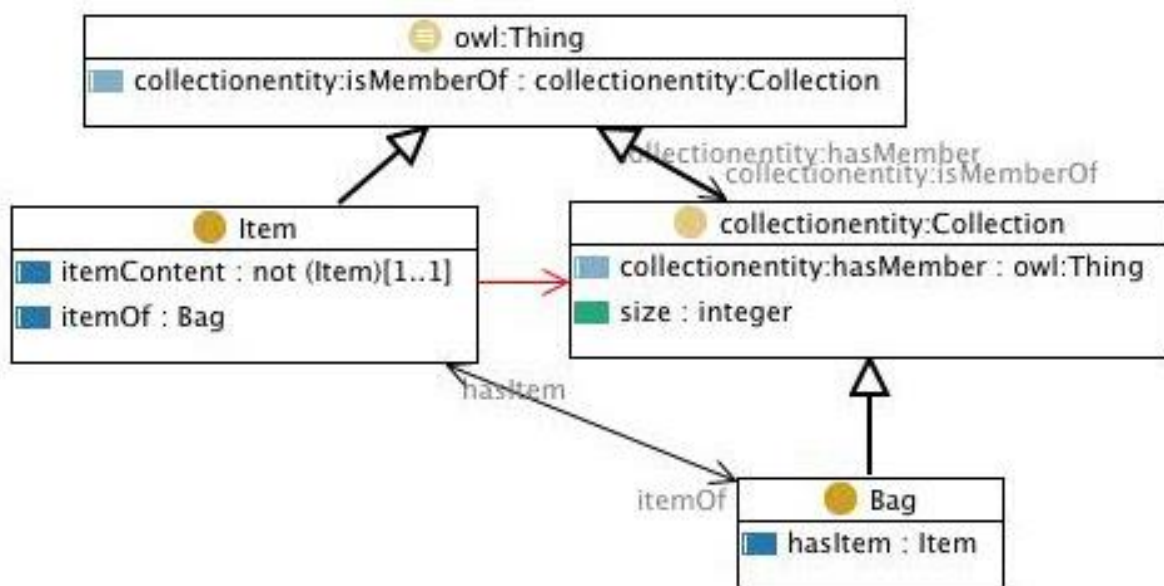


Figura 1: Rappresentazione Pattern Bag

4.2 Object properties, Data Properties e Individuals

Attraverso la definizione delle varie **Object Properties** si sono create le associazioni tra le varie classi, in modo da descrivere tutta l'ontologia:

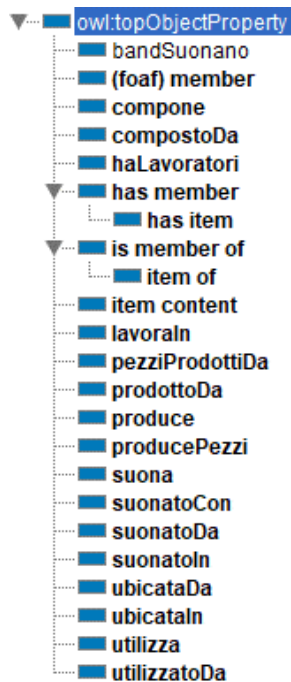


Figura 2: Rappresentazione Object Properties

Sono presenti proprietà funzionali, ovvero le proprietà con relazione 1:1 con gli individui, per esempio: “ubicataIn”, in quanto la sede principale di una Casa Produttrice sarà ubicata in una sola città.

Inoltre, è presente una property chain per le band (Gruppi e Solisti) dal nome **bandSuonano**. Questa property chain permette di collegare gli strumenti musicali che vengono suonati nella band, tramite gli artisti:

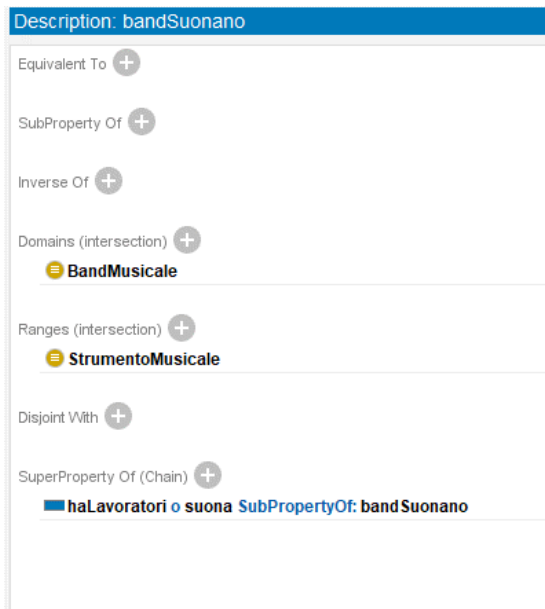


Figura 3: Property Chain 'bandSuonano'

Le **Data Properties** sono ulteriori proprietà che sono state utilizzate per descrivere le classi dell'ontologia:



Figura 4: Data Properties

Si nota l'utilizzo di FOAF per descrivere gli artisti, mentre le altre proprietà sono utilizzate per descrivere gli strumenti musicali, le band musicali, le case produttrici, ecc.

Gli **Individuals** sono tutti gli esempi reali che vengono associati alla classe relativa. In questo dominio sono stati aggiunti due solisti, con i relativi strumenti musicali, casa produttrice dello strumento musicale. Inoltre, è stata aggiunta anche una band musicale, con le relative informazioni:

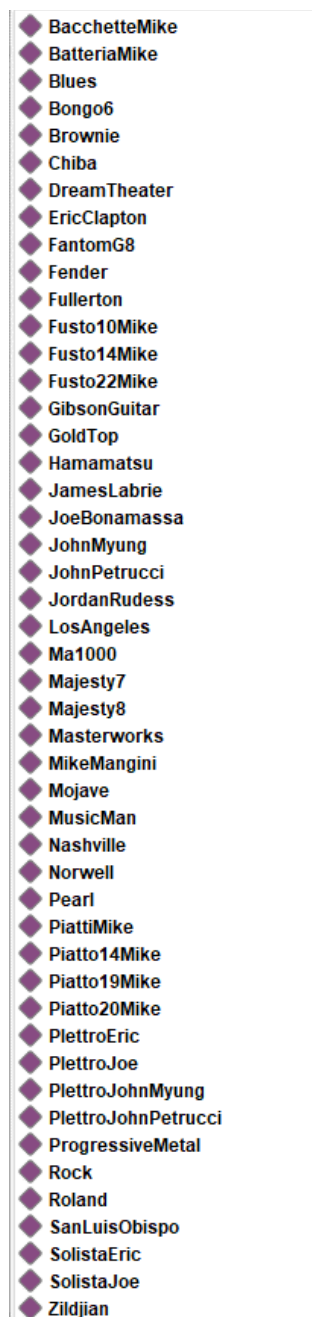


Figura 5: Individuals

4.3 Esempio reale

L'esempio reale che illustra il dominio si basa sull'inserimento di una band musicale con tutti i relativi strumenti e i suoi artisti. La band musicale in questione è "Dream Theater" che è composta da cinque artisti:

- John Petrucci;
- John Myung;
- Mike Mangini;
- James Labrie;
- Jordan Rudess.

A loro volta, ogni artista suona almeno uno strumento, il quale è prodotto da una certa casa produttrice. Ogni casa produttrice è ubicata in una determinata città. Inoltre, lo strumento musicale può essere suonato con uno strumento.

Di seguito la completa rappresentazione partendo dal gruppo musicale "Dream Theater":

The screenshot displays a Semantic Web editor interface. On the left, the 'Types' pane shows a hierarchy with 'Gruppo' selected. On the right, the 'Object property assertions' pane lists several assertions for the 'Dream Theater' group, including 'haLavoratori JohnPetrucci', 'bandSuonano Majesty7', 'bandSuonano Bongo6', 'bandSuonano Ma1000', 'bandSuonano FantomG8', 'bandSuonano BatteriaMike', 'bandSuonano Majesty8', 'haLavoratori JohnMyung', 'haLavoratori JordanRudess', 'haLavoratori MikeMangini', and 'haLavoratori JamesLabrie'. The 'Data property assertions' pane shows 'HaNumeroArtisti 5', 'NomeBandMusicale "Dream Theater"', and 'Immagine "https://i.libb.co/cNLLx1h/dreamtheater.jpg"'.

Object property assertions
haLavoratori JohnPetrucci
bandSuonano Majesty7
bandSuonano Bongo6
bandSuonano Ma1000
bandSuonano FantomG8
bandSuonano BatteriaMike
bandSuonano Majesty8
haLavoratori JohnMyung
haLavoratori JordanRudess
haLavoratori MikeMangini
haLavoratori JamesLabrie

Data property assertions
HaNumeroArtisti 5
NomeBandMusicale "Dream Theater"
Immagine "https://i.libb.co/cNLLx1h/dreamtheater.jpg"

Figura 6: Rappresentazione individuo "DreamTheater"

Come si può notare dalla Figura 7, ha tipo **Gruppo** e ha cinque artisti che lavorano nel gruppo musicale. Inoltre, ha tutti gli strumenti che vengono suonati dagli artisti, quindi anche nella band, e ha delle data properties che lo descrivono.

Per continuare l'esempio, andrò a mostrare l'artista "JohnPetrucchi", sarà uguale per tutti gli altri artisti:

The screenshot displays a software interface with two main panels. The left panel, titled 'Description: JohnPetrucchi', shows a 'Types' section with a plus icon and a list containing 'Chitarrista' with a yellow circle icon. Below this are sections for 'Same Individual As' and 'Different Individuals', each with a plus icon. The right panel, titled 'Property assertions: JohnPetrucchi', contains two sections: 'Object property assertions' and 'Data property assertions', each with a plus icon. The 'Object property assertions' section lists four items: 'suona Majesty7', 'lavoraIn DreamTheater', 'suona Majesty8', and 'utilizza PlettroJohnPetrucchi'. The 'Data property assertions' section lists six items: 'foaf:genere' with value 'Uomo', 'foaf:nome' with value 'John', 'foaf:cognome' with value 'Petrucchi', 'foaf:età' with value '54', 'Immagine' with value 'https://independentmusica', and 'foaf:data di nascita' with value '12/07/1967'. Each item in both sections has four circular icons to its right: a question mark, an '@' symbol, an 'x' symbol, and an 'o' symbol.

Figura 7: Rappresentazione Artista "JohnPetrucchi"

Si può notare che l'artista ha tipo **Chitarrista** e suona due strumenti musicali, lavora nel gruppo musicale introdotto precedentemente e suona gli strumenti musicali utilizzando un determinato oggetto. Inoltre, ha le proprietà di FOAF per la descrizione dell'artista e l'immagine.

Per proseguire con l'esempio, andrò a mostrare lo strumento musicale "Majesty7", sarà uguale per tutti gli altri:

The screenshot displays a Semantic Web editor interface with two main panels. The left panel, titled 'Description: Majesty7', shows the type 'ChitarraAPiuCorde' under the 'Types' section. The right panel, titled 'Property assertions: Majesty7', is divided into 'Object property assertions' and 'Data property assertions'. The 'Object property assertions' section includes 'suonatoCon PlettroJohnPetrucci', 'suonatoIn ProgressiveMetal', 'prodottoDa MusicMan', and 'suonatoDa JohnPetrucci'. The 'Data property assertions' section includes 'Legni "Mogano, Acero, Ebano"', 'Ponte "Tremolo Fluttuante"', 'Body "Solid Body"', 'HaNumeroCorde 7', 'Immagine "https://d1aeri3ty3izns.cloudfror.net/media/29/292871/600.jpg"', and 'NomeStrumentoMusicale "Music Man Majesty 7 Hydrospace"'.

Property	Value
suonatoCon	PlettroJohnPetrucci
suonatoIn	ProgressiveMetal
prodottoDa	MusicMan
suonatoDa	JohnPetrucci
Legni	"Mogano, Acero, Ebano"
Ponte	"Tremolo Fluttuante"
Body	"Solid Body"
HaNumeroCorde	7
Immagine	"https://d1aeri3ty3izns.cloudfror.net/media/29/292871/600.jpg"
NomeStrumentoMusicale	"Music Man Majesty 7 Hydrospace"

Figura 8: Rappresentazione Strumento Musicale "Majesty7"

Lo strumento musicale ha tipo **ChitarraAPiuCorde** ed è suonato con un determinato strumento, suonato in un determinato genere musicale, prodotta da una casa produttrice e suonata dall'artista che ho introdotto precedentemente. Inoltre, ha diverse proprietà per descrivere lo strumento musicale.

PlettroJohnPetrucchi è lo strumento con cui l'artista **JohnPetrucchi** suona la chitarra ed è rappresentato in questo modo:

The interface is divided into two main sections. The left section, titled 'Types', contains a list of types with 'Plettro' selected. Below the list are buttons for 'Same Individual As' and 'Different Individuals'. The right section, titled 'Object property assertions', shows a single assertion: 'utilizzatoDa JohnPetrucchi'.

Figura 9: Rappresentazione Plettro "PlettroJohnPetrucchi"

Lo strumento musicale **Majesty7** è prodotto dalla casa produttrice **MusicMan** così rappresentata:

The interface is divided into two main sections. The left section, titled 'Description: MusicMan', contains a list of types with 'CasaProduttriceBasso', 'CasaProduttriceChitarra', and 'OrigineAmericana' listed. Below the list are buttons for 'Same Individual As' and 'Different Individuals'. The right section, titled 'Property assertions: MusicMan', shows object property assertions for 'ubicataIn SanLuisObispo', 'produce Majesty7', 'produce Bongo6', and 'produce Majesty8', and data property assertions for 'NomeCasaProduttrice', 'Immagine', and 'FondataNel'.

Figura 10: Rappresentazione Casa Produttrice "MusicMan"

Si può notare che questa casa produttrice produce sia bassi che chitarre; infatti, produce **Majesty7** e **Majesty8** che sono due chitarre, mentre **Bongo6** è un basso. Possiede alcune data properties per la descrizione. Inoltre, ha la proprietà "ubicataIn" per indicare dove è situata la sede principale, in questo caso **SanLuisObispo** che ha questa rappresentazione:

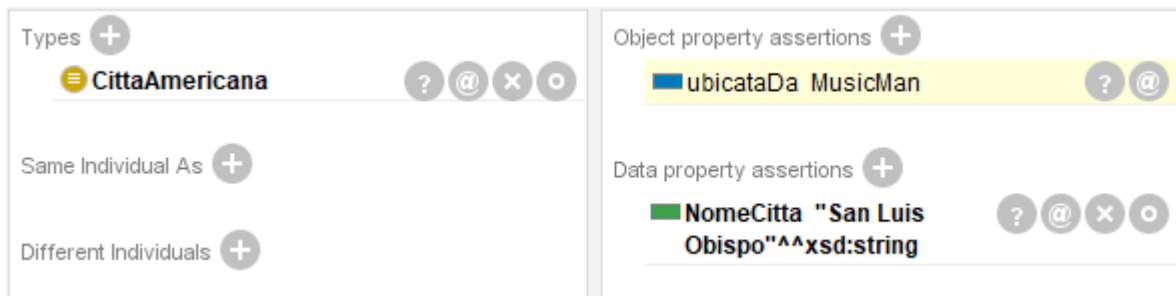


Figura 11: Rappresentazione Città "SanLuisObispo"

Per tutti gli altri artisti della band musicale si segue lo stesso procedimento, andando ad inserire gli strumenti musicali, le case produttrici con la città in cui risiedono e l'eventuale strumento che usano per suonare lo strumento musicale.

5. Documentazione LODE

Per la documentazione è stato utilizzato pyLODE, uno script Python per generare la documentazione dell'ontologia che estende il framework LODE.



6. Visualizzazione ontologia

6.1 Tassonomia classi

Come già accennato in precedenza, gli strumenti musicali sono suddivisi in base alle proprie caratteristiche:

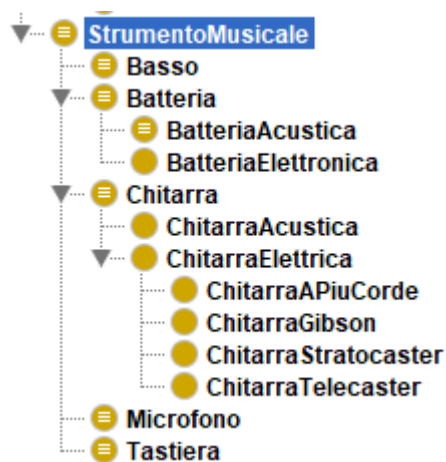


Figura 12: Rappresentazione Classi StrumentoMusicale

Si può notare che StrumentoMusicale ha delle sottoclassi, ovvero:

- **Batteria:** dove si suddivideranno le batterie acustiche e quelle elettroniche;
- **Chitarra:** dove si suddivideranno le chitarre acustiche da quelle elettroniche. A sua volta, le chitarre elettriche si suddivideranno a seconda della tipologia di chitarra elettrica.

Gli strumenti musicali potranno essere suonati con uno strumento (tranne quelli che si utilizzano la mano) e saranno rappresentati dalla classe **Oggetto** a cui appartengono **Bacchette** e **Plettro**.

Inoltre, nel dominio vi è la classe **GenereMusicale** per indicare in quale genere gli strumenti musicali sono più adatti e dove vengono maggiormente utilizzati.

Ogni strumento musicale è suonato da una persona che è rappresentata nel dominio come classe **Artista**, suddividendo le varie tipologie.

Ogni artista farà parte della classe **BandMusicale**, attraverso una delle due sue sottoclassi **Gruppo** o **Solista**:



Figura 13: Rappresentazione Classi Artista e BandMusicale

Infine, ogni strumento musicale è prodotto da una casa produttrice che nel dominio è la classe **CasaProduttrice**, suddividendo le varie tipologie. La casa produttrice ha una determinata classe **Origine** a seconda della città in cui risiede, descritta dalla classe **Citta**. Se risiede in una città in America, avrà come classe **CittaAmericana** e di conseguenza farà parte della classe **OrigineAmericana**, altrimenti se risiede in una città in Giappone, avrà come classe **CittaGiapponese** e di conseguenza farà parte della classe **OrigineGiapponese**.

La batteria ha una differenza rispetto agli altri strumenti musicali perché una batteria è composta dai fusti e dai piatti. Quindi ho deciso di sdoppiare le case produttrici perché sono diverse. Nel dominio è presente la classe **PezziBatteria** sottoclasse di una (**collections**)

Bag, che ha come due classi **Piatti** e **TuttiFusti**. È stata inserita come sottoclasse di **(collections) Bag** perché i piatti ed i fusti, sono una collezione di elementi anche duplicati. Infatti, in **item** ci saranno due classi **Fusto** e **Piatto** che serviranno a rappresentare ogni singolo piatto e fusto che apparterranno alle collezioni **Piatti** e **TuttiFusti**.

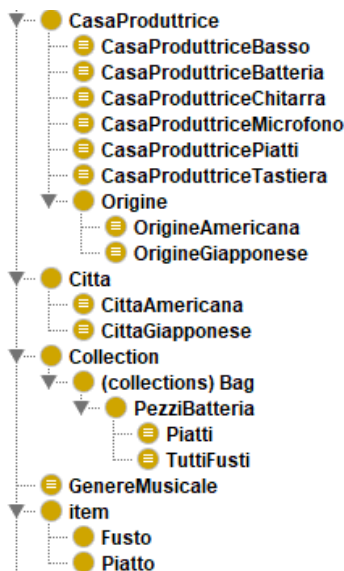


Figura 14: Rappresentazione CasaProduttrice, Citta, Collection e Item

7. Interazione utente

7.1 Interazione con l'utente tramite navbar

In questo progetto è stata creata un'applicazione client, di cui ne parlerò in dettaglio nella successiva sezione.

Per quanto riguarda l'interazione con l'utente avviene tramite un menù di navigazione, nel quale l'utente può ricercare lo strumento musicale, la casa produttrice, la band musicale o gli artisti sotto diverse categorie:

Il menù è così rappresentato:

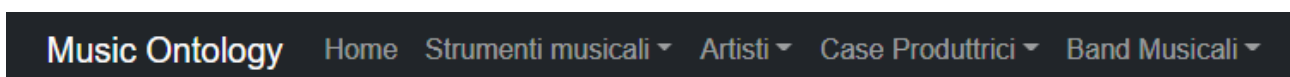


Figura 15: Menu di navigazione applicazione client

Quando l'utente seleziona una voce dal menù, verrà utilizzata una query SPARQL relativa alla voce che ha selezionato, in modo da restituire come output tutti gli elementi della categoria selezionata.

Tramite questo menù l'utente può ricercare velocemente ciò di cui ha bisogno attraverso le diverse categorie che compongono le diverse sezioni.

Un esempio con dati reali ed inseriti all'interno dell'ontologia: l'utente clicca sulla voce "Strumenti Musicali" e dal menù a tendina successivo clicca sulla voce "Tastiere". A questo punto viene utilizzata una query SPARQL per recuperare tutte le tastiere e le informazioni su di esse. Il risultato sarà il seguente:

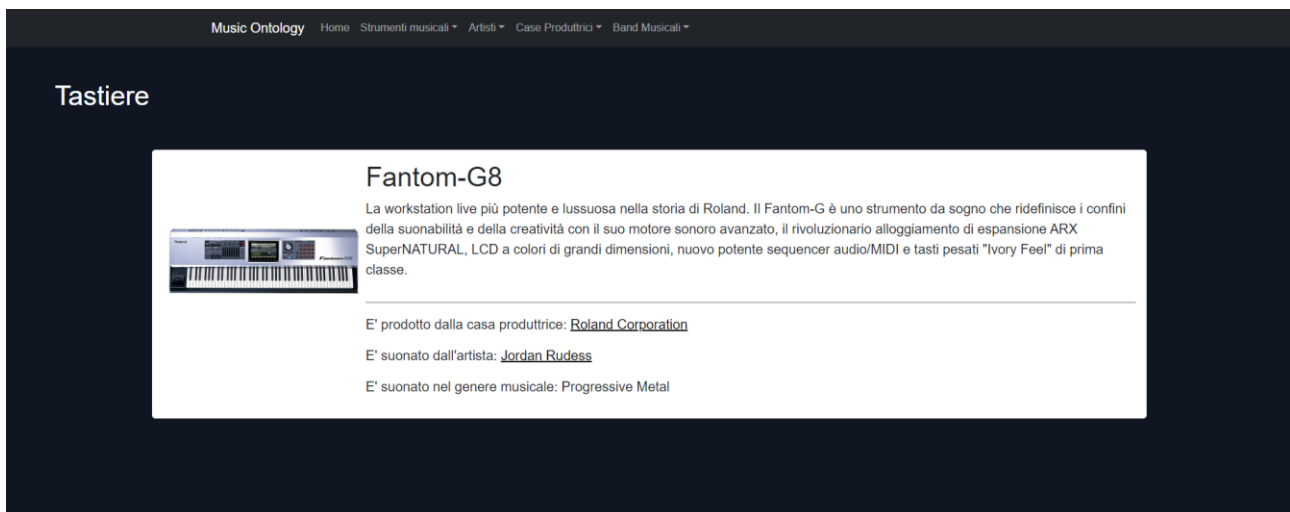


Figura 16: Risultato ricerca delle tastiere

7.2 Queries SPARQL

Le query SPARQL sono utilizzate per facilitare l'utente nella ricerca delle informazioni. L'ontologia è stata importata ed eseguita in GraphDB e successivamente interrogata tramite l'API fornita da GraphDB per effettuare le query SPARQL.

7.2.1 Ricerca delle chitarre con relative informazioni

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX music: <http://www.semanticweb.org/musical-instruments#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT ?chitarra ?nome ?immagine ?descrizione ?numCorde ?body ?ponte ?legni
?suonatoDaIRI ?suonatoCon ?suonatoIn ?prodottoDaIRI ?prodottoDa ?suonatoDa
WHERE { ?chitarra rdf:type music:Chitarra .
        ?chitarra music:NomeStrumentoMusicale ?nome .
        ?chitarra rdfs:comment ?descrizione .
        ?chitarra music:Immagine ?immagine .
        optional {?chitarra music:HaNumeroCorde ?numCorde} .
        ?chitarra music:Body ?body .
        ?chitarra music:Ponte ?ponte .
        ?chitarra music:Legni ?legni .
        ?chitarra music:suonatoCon ?suonato .
        ?suonato rdfs:label ?suonatoCon .
        ?chitarra music:suonatoIn ?genere .
        ?genere music:NomeGenereMusicale ?suonatoIn .
        ?chitarra music:prodottoDa ?prodottoDaIRI .
        ?prodottoDaIRI music:NomeCasaProduttrice ?prodottoDa .
        ?chitarra music:suonatoDa ?suonatoDaIRI .
        ?suonatoDaIRI foaf:firstName ?artistaNome .
        ?suonatoDaIRI foaf:lastName ?artistaCognome .
        BIND(CONCAT(?artistaNome, " ", ?artistaCognome) AS ?suonatoDa)
}
```

Fornisce tutte le chitarre con le relative informazioni. Sono salvati anche gli IRI dell'artista e della casa produttrice per generare successivamente dei link (verrà spiegato nella sezione successiva).

Risultati:

	chitarra ↕	nome ↕	immagine ↕	descrizione ↕	numCorde ↕	body ↕	ponte ↕	legni ↕	suonatoDaIRI ↕	suonatoCon ↕	suonatoIn ↕	prodottoDaIRI ↕	prodottoDa ↕	suonatoDa ↕
1	http://www.sei.instruments#Gibson	"Gold Top Personalizzata"	"https://d1aeri3ty3izns.cloudfront.net/media/47/473415/600/preview.jpg"	"Chitarra Gibson con il nome "Gold Top"®"		"Solid Body"	"Fisso, raramente tremolo"	"Mogano, Eban o Palissandro"	http://www.sei.instruments#Joe	"PiettoJoe"®	"Blues"	http://www.sei.instruments#Gibson	"Gibson Guitar Corporation"	"Joe Bonamassa"
2	http://www.sei.instruments#MusicMan	"Music Man Majesty 7 Hybrid"	"https://d1aeri3ty3izns.cloudfront.net/media/29/292871/600/preview.jpg"	"La chitarra "Majesty 7" utilizzata da John Petrucci"®	"7" xsd:integer	"Solid Body"	"Tremolo Fluttuante"	"Mogano, Acero, Ebano"	http://www.sei.instruments#John	"PiettoJohn Petrucci"®	"Progressive Metal"	http://www.sei.instruments#ErnieBall	"Ernie Ball Music Man"	"John Petrucci"
3	http://www.sei.instruments#MusicMan	"Music Man Majesty 8 Mystic Dream"	"https://media.guitarcenter.com/is/image/MMGS7/L88819000001000-00-720x720.jpg"	"La chitarra "Majesty 8" che utilizza John Petrucci"®	"8" xsd:integer	"Solid body"	"Ponte fisso Music Man"	"Tiglio, Mogano ed Ebano"	http://www.sei.instruments#John	"PiettoJohn Petrucci"®	"Progressive Metal"	http://www.sei.instruments#ErnieBall	"Ernie Ball Music Man"	"John Petrucci"
4	http://www.sei.instruments#Fender	"Fender Custom Eric Clapton "Brownie"™"	"https://www.tomassone.it/media/catalog/product/cache/4/image/9df78eab33525d08d6e5fb8d27136e95/P/R/PRO_FENCUSSTRECLASIGMB_01.jpg"	"La chitarra Fender Custom "Brownie" suonata da Eric Clapton"®	"6" xsd:integer	"Solid Body"	"Tremolo "Blocked" American Vintage Synchronized"	"Acero"	http://www.sei.instruments#Eric	"PiettoEric"®	"Rock"	http://www.sei.instruments#Fender	"Fender Musical Instruments Corporation"	"Eric Clapton"

Figura 17: Risultato Query

7.2.2 Ricerca dei batteristi con relative informazioni

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX music: <http://www.semanticweb.org/musical-instruments#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX mo: <http://purl.org/ontology/mo/>
SELECT DISTINCT ?batterista ?nome ?dataNascita ?genere ?eta ?cognome ?immagine
?bandIRI ?groupband ?lavoraIn (group_concat(distinct
?strumentoMusicaleURI;separator=", ") AS ?suonaIRI) (group_concat(distinct
?oggetto;separator=", ") AS ?suonaCon) (group_concat(distinct
?strumentoMusicale;separator=", ") AS ?suona)
WHERE {
    ?batterista rdf:type music:Batterista .
    ?batterista foaf:firstName ?nome .
    ?batterista foaf:lastName ?cognome .
    ?batterista foaf:birthday ?dataNascita .
    ?batterista foaf:age ?eta .
    ?batterista foaf:gender ?genere .
    ?batterista music:Immagine ?immagine .
    ?batterista music:lavoraIn ?bandIRI .
    ?bandIRI music:NomeBandMusicale ?lavoraIn .
    ?bandIRI rdf:type ?band .
    FILTER(?band in (mo:MusicGroup, mo:SoloMusicArtist))
    ?band rdfs:label ?groupband .
    ?batterista music:suona ?strumentoMusicaleURI .
    ?strumentoMusicaleURI music:NomeStrumentoMusicale ?strumentoMusicale .
    optional {
        ?strumentoMusicaleURI music:suonatoCon ?oggettoCon .
        ?oggettoCon rdfs:label ?oggetto
    }
}
GROUP BY ?batterista ?nome ?dataNascita ?genere ?eta ?cognome ?immagine ?bandIRI
?groupband ?lavoraIn
```

Restituisce tutti gli artisti che suonano la batteria (batteristi) con tutte le relative informazioni. Vengono salvati gli IRI degli strumenti musicali per generare successivamente dei link (verrà spiegato nella sezione successiva). Sono presenti delle funzioni "group_concat" nel caso in cui l'artista suoni più di uno strumento oppure utilizzi più di uno strumento.

Risultati:

	batterista	nome	dataNascita	genere	eta	cognome	immagine	bandIRI	groupband	lavoraln	suonalRI	suonaCon	suona
1	http://www.semanticweb.org/musical-instruments#Mike	"Mike"	"18/04/1963"	"Uomo"	"58"^^xsd:integer	"Mangini"	"https://metalitalia.com/wp-content/uploads/2017/01/dream-theater-mike-mangini-2017.jpg"	http://www.semanticweb.org/musical-instruments#Dire	"Gruppo"@it	"Dream Theater"	"http://www.semanticweb.org/musical-instruments#BatteriaMike"	"BacchetteMike"	"Mike Mangini Masterworks Kit and Zildjian Cymbals"

Figura 18: Risultato Query

7.2.3 Ricerca delle band musicali (gruppi) con relative informazioni

```

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX music: <http://www.semanticweb.org/musical-instruments#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX mo: <http://purl.org/ontology/mo/>
SELECT DISTINCT ?band ?descrizione ?nome ?numArtisti ?immagine
(group_concat(distinct ?artistiURI;separator=", ") AS ?artistiNomeURI)
(group_concat(distinct ?artisti;separator=", ") AS ?artistiNome)
(group_concat(distinct ?listaStrumenti;separator=", ") AS ?strumentiSuonatiURI)
(group_concat(distinct ?listaStrumentiNome;separator=", ") AS
?strumentiSuonatiNome) where {
    ?band rdf:type mo:MusicGroup .
    ?band rdfs:comment ?descrizione .
    ?band music:NomeBandMusicale ?nome .
    ?band music:HaNumeroArtisti ?numArtisti .
    ?band music:Immagine ?immagine .
    ?band music:bandSuonano ?listaStrumenti .
    ?listaStrumenti music:NomeStrumentoMusicale ?listaStrumentiNome .
    ?citta rdf:type ?origineTipo .
    ?band music:haLavoratori ?artistiURI .
    ?artistiURI foaf:firstName ?artistaNome .
    ?artistiURI foaf:lastName ?artistaCognome .
    BIND(CONCAT(?artistaNome, " ", ?artistaCognome) AS ?artisti)
}
GROUP BY ?band ?descrizione ?nome ?numArtisti ?immagine

```

Restituisce tutte le band musicali con le relative informazioni. Sono presenti delle funzioni "group_concat" perché nella band musicale suonano più artisti, così vengono concatenati in un unico risultato. Inoltre vengono anche concatenati tutti gli strumenti musicali suonati nella band. Salvo anche gli IRI di tutti gli artisti e di tutti gli strumenti per generare successivamente dei link (verrà spiegato nella sezione successiva).

Risultati:

	band	descrizione	nome	numArtisti	Immagine	artistiNomeURI	artistiNome	strumentiSuonatiURI	strumentiSuonatiNome
1	http://www.semanticweb.org/musical-instruments#DreamTheater	"Band musicale "Dream Theater""	"Dream Theater"	"5"^^xsd:integer	"https://i.lib.co/cnLLx1h/dreamtheater.jpg"	"http://www.semanticweb.org/musical-instruments#MikeMangini, http://www.semanticweb.org/musical-instruments#JohnMyung, http://www.semanticweb.org/musical-instruments#JohnPetrucchi, http://www.semanticweb.org/musical-instruments#JamesLabrie, http://www.semanticweb.org/musical-instruments#JordanRudess"	"Mike Mangini, John Myung, John Petrucci, James Labrie, Jordan Rudess"	"http://www.semanticweb.org/musical-instruments#BatteriaMike, http://www.semanticweb.org/musical-instruments#Bongo6, http://www.semanticweb.org/musical-instruments#FantomG8, http://www.semanticweb.org/musical-instruments#Majesty7, http://www.semanticweb.org/musical-instruments#Ma1000, http://www.semanticweb.org/musical-instruments#Majesty8"	"Mike Mangini Masterworks Kit and Zildjian Cymbals, Bongo John Myung 6 HH, Fantom-G8, Music Man Majesty 7 HydroSpace, MA-1000DS, Music Man Majesty 8 Mystic Dream"

Figura 19: Risultato Query

7.2.4 Ricerca delle case produttrici di bassi elettrici con relative informazioni

```

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX music: <http://www.semanticweb.org/musical-instruments#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT DISTINCT ?casaProdBassi ?descrizione ?nome ?immagine ?dataFond ?origine
?nomecitta (group_concat(distinct ?strumentoMusicaleURI;separator=", ") AS
?suonaIRI) (group_concat(distinct ?prod;separator=", ") AS ?produce)
WHERE {
    ?casaProdBassi rdf:type music:CasaProduttriceBasso .
    ?casaProdBassi rdfs:comment ?descrizione .
    ?casaProdBassi music:NomeCasaProduttrice ?nome .
    ?casaProdBassi music:Immagine ?immagine .
    ?casaProdBassi music:FondatoNel ?dataFond .
    ?casaProdBassi music:ubicataIn ?citta .
    ?citta music:NomeCitta ?nomecitta .
    ?citta rdf:type ?origineTipo .
    FILTER(?origineTipo IN (music:CittaAmericana, music:CittaGiapponese))
    ?origineTipo rdfs:label ?origine .
    optional {
        ?casaProdBassi music:produce ?strumentoMusicaleURI .
        ?strumentoMusicaleURI rdf:type ?strumentoTipo .
        FILTER(?strumentoTipo IN (music:Basso)) .
        ?strumentoMusicaleURI music:NomeStrumentoMusicale ?prod
    }
    optional {
        ?casaProdBassi music:producePezzi ?strumentoMusicaleURI .
        ?strumentoMusicaleURI music:NomeStrumentoMusicale ?prod
    }
}

```

GROUP BY ?casaProdBassi ?descrizione ?nome ?immagine ?dataFond ?origine
?nomecitta

Restituisce tutte le case produttrici di bassi elettrici con le relative informazioni. Vengono salvati gli IRI degli strumenti musicali prodotti così da generare successivamente dei link (verrà spiegato nella sezione successiva). Viene utilizzata la funzione "group_concat" nel caso in cui una casa produttrice produce più di uno strumento così da raggrupparlo in un unico risultato.

Risultati:

	band	descrizione	nome	numArtisti	immagine	artistiNomeIRI	artistiNome
1	http://www.semanticweb.org/instruments#DreamTheater	"Band musicale "Dream Theater""@it	"Dream Theater"	"5""xsd:integer	"https://i.libb.co/cNLLx1h/dreamtheater.jpg"	"http://www.semanticweb.org/musical-instruments#MikeMangini, http://www.semanticweb.org/musical-instruments#JohnMyung, http://www.semanticweb.org/musical-instruments#JohnPetrucchi, http://www.semanticweb.org/musical-instruments#JamesLabrie, http://www.semanticweb.org/musical-instruments#JordanRudess"	"Mike Mangini, John Myung, John Petrucci, James Labrie, Jordan Rudess"

Figura 20: Risultato Query

7.2.5 Ricerca degli artisti (no batteristi) che hanno la casa produttrice di origine giapponese con relative informazioni

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX music: <http://www.semanticweb.org/musical-instruments#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX mo: <http://purl.org/ontology/mo/>

SELECT DISTINCT ?artista ?nome ?dataNascita ?genere ?eta ?cognome ?immagine
?lavoraIn (group_concat(distinct ?oggetto;separator=", ") AS ?suonaCon)
(group_concat(distinct ?strumentoMusicale;separator=", ") AS ?suona) where{
    ?artista rdf:type mo:MusicArtist .
    ?artista rdf:type ?tipoArtista .
    FILTER(?tipoArtista in (music:Chitarrista, music:Bassista, music:Cantante,
music:Tastierista))
    ?artista foaf:firstName ?nome .
    ?artista foaf:lastName ?cognome .
    ?artista foaf:birthday ?dataNascita .
    ?artista foaf:age ?eta .
    ?artista foaf:gender ?genere .
    ?artista music:Immagine ?immagine .
    ?artista music:lavoraIn ?gruppo .
    ?gruppo music:NomeBandMusicale ?lavoraIn .
    ?artista music:suona ?strumento .
    ?strumento music:prodottoDa ?casaProd .
    ?casaProd music:ubicataIn ?cittaSede .
    ?cittaSede rdf:type ?tipoCitta .
    FILTER(?tipoCitta IN (music:CittaGiapponese))
    ?strumento music:NomeStrumentoMusicale ?strumentoMusicale .
    optional {
        ?strumento music:suonatoCon ?oggettoCon .
        ?oggettoCon rdfs:label ?oggetto
    }
}
GROUP BY ?artista ?nome ?dataNascita ?genere ?eta ?cognome ?immagine ?lavoraIn
```

Restituisce tutti gli artisti (no batteristi) che hanno lo strumento musicale prodotto da una casa produttrice che risiede in giappone.

Risultati:

	artista ↕	nome ↕	dataNascita ↕	genere ↕	eta ↕	cognome ↕	immagine ↕	lavoraIn ↕	suonaCon ↕	suona ↕
1	http://www.semantic-instruments#JordanF	"Jordan"	"04/11/1956"	"Uomo"	"65"^^xsd:integer	"Rudess"	"https://i.scdn.co/image/ab6761610000e5eb67422e82214a66f18cd5091c"	"Dream Theater"	"	"Fantom-G8"

Figura 21: Risultato Query

8. Applicazione Client

È stata realizzata un'applicazione client tramite il framework React, in modo da interrogare i dati caricati su GraphDB.

Il client prevede una pagina di Homepage dove viene presentata l'ontologia, con un menù di navigazione in alto nella pagina, dove l'utente può selezionare la voce che desidera per visualizzare le informazioni relative. Il menù è così rappresentato:

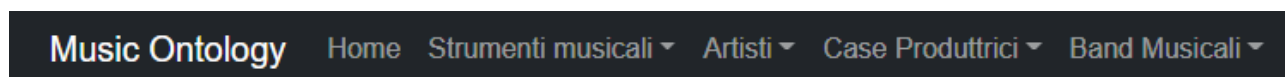


Figura 22: Navbar Client

Una volta che l'utente seleziona la voce dal menù, verrà utilizzata una query SPARQL, relativa alla voce selezionata, che tramite una chiamata HTTP a GraphDB (dato che espone le API) recupera le informazioni relative.

Inoltre, come accennato nella sezione precedente, vengono salvati gli IRI di determinati individui in alcune query SPARQL in modo da poter rendere quasi tutte le informazioni collegate tra di loro. Quando un link viene cliccato, viene richiamata una pagina di "ricerca IRI" che a seconda del tipo di IRI cercato (Artista, Strumento musicale, ecc) restituisce le informazioni relative a quell'IRI tramite una query SPARQL.

In questo modo l'utente può navigare tra un'informazione e un'altra in maniera molto rapida.

Un esempio pratico: l'utente clicca su "Strumenti Musicali" e nel menù a tendina successivo, seleziona la voce "Microfoni". Questo è il risultato che vede:



Figura 23: Risultati Microfoni nel Client

L'utente ora decide di cliccare sul link "Mojave" per andare a vedere le informazioni sulla casa produttrice. Tramite la pagina di ricerca IRI è possibile, altrimenti avrebbe dovuto andare nella pagina delle case produttrici di microfoni. Il risultato dopo aver cliccato il link:

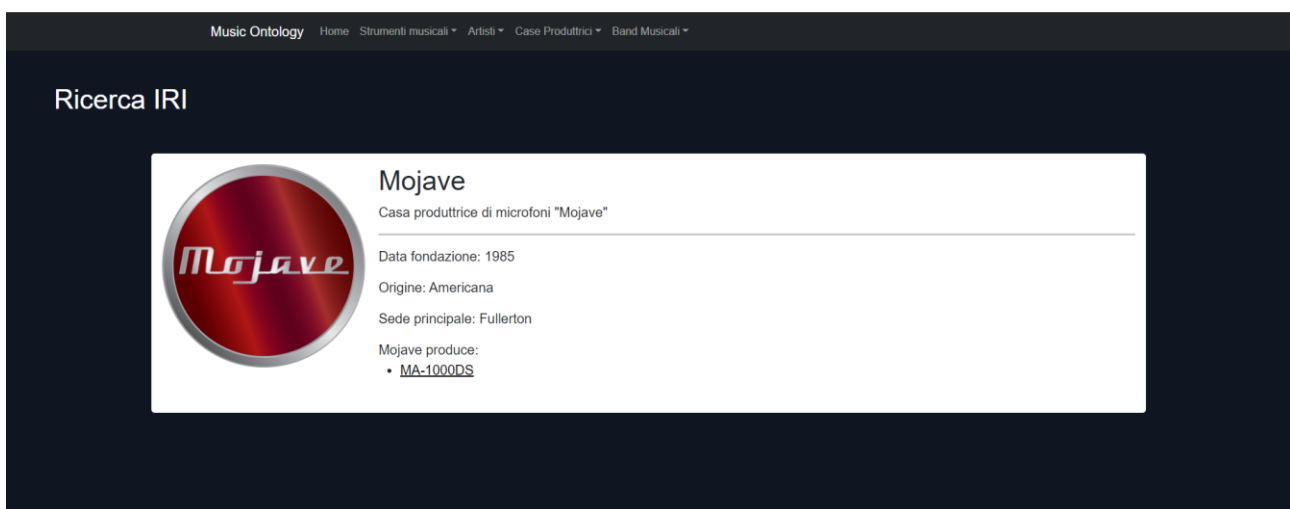


Figura 24: Casa Produttrice Mojave nel Client

In questo modo le informazioni sono collegate tra di loro.

9. Sitografia

(s.d.). Tratto da The Music Ontology: <http://musicontology.com/docs/faq.html>

Dublin Core - Wikipedia. (2020). Tratto da https://it.wikipedia.org/wiki/Dublin_Core

FOAF - Wikipedia. (2021). Tratto da <https://it.wikipedia.org/wiki/FOAF>

Per la maggior parte delle informazioni. (s.d.). Tratto da <https://it.wikipedia.org/>

Perchè la musica è importante? - *Sesto Daily News*. (2016). Tratto da <https://www.sestodailynews.net/rubriche/attitude-for-music-di-antonio-giovanditti/3482/perche-la-musica-e-importante>

pyLODE. (2021). Tratto da <https://github.com/RDFLib/pyLODE>

Submissions:Bag. (2010). Tratto da <http://ontologydesignpatterns.org/wiki/Submissions:Bag>