

UNIVERSITÀ DEGLI STUDI DI TORINO

DIPARTIMENTO DI INFORMATICA

SCUOLA DI SCIENZE DELLA NATURA

Corso di Laurea Magistrale in Informatica



Progetto Modellazione Concettuale del Web Semantico

Lorenzo SCIANDRA

Stefano Vittorio PORTA

ANNO ACCADEMICO
2020/2021

1 Pagina del progetto

Tutto quello che sarà presentato in questa documentazione può essere facilmente esplorato mediante il link pubblico del repository Github o attraverso la sua pagina HTML.

2 Motivazioni

Il dominio che abbiamo deciso di modellare riguarda il campo della ristorazione. Durante l'analisi abbiamo prestato particolare attenzione alla gestione di alcuni aspetti che abbiamo ritenuto particolarmente importanti:

- Gli Esercizi alimentari (Ristoranti, Bar, Paninoteche, ecc);
- Le Pietanze servite dagli esercizi alimentari e gli Ingredienti che le compongono;
- Le varie Attività collegate agli Esercizi alimentari ed effettuate da Persone, come Ordini, Consegne, Recensione e Amministrazione.

L'idea è nata pensando all'offerta sempre più diffusa dei servizi di consegna a domicilio. Infatti, date le restrizioni sanitarie vigenti, un numero in costante aumento di imprese di ristorazione sta ricorrendo a servizi di consegna a domicilio, non solo nelle grandi città, ma anche in centri urbani di più modeste dimensioni^{1 2}.

Esistono svariati servizi (alcuni dei quali molto rinomati e popolari) che fungono da mediatore tra la proposta dei locali e la richiesta dei clienti. Tuttavia, nessuno sembra fare affidamento ad una gestione semantica dei propri dati, preferendo piuttosto dei sistemi più standard.

L'idea di creare un'ontologia comune che modelli correttamente questo dominio potrebbe quindi risultare utile per offrire uno strumento potente, in grado di facilitare la gestione per tutte le parti e migliorare le possibili applicazioni.

Un secondo punto cruciale riguarda gli allergeni: tra le varie piattaforme di

¹Articolo de "Il Sole 24 ore": <https://www.ilsole24ore.com/art/coronavirus-e-boom-spesa-online-consegna-domicilio-AD3rKdC>

²Articolo de "la Repubblica": https://www.repubblica.it/economia/rapporti/osserva-italia/trend/2020/10/16/news/cresce_la_passione_per_il_cibo_a_domicilio-270779771/

delivery esistenti da noi provate, nessuna permette di filtrare i locali con la precisione che pensavamo di offrire (in funzione degli allergeni presenti nelle pietanze proposte³ o dell'alimentazione corrispondente). Questi criteri di ricerca sono stati da noi tenuti in considerazione durante la modellazione.

3 Requisiti

3.1 Finalità

Lo scopo dell'ontologia riguarda un'ipotetica piattaforma di delivery che possa permettere a dei locali di esporsi facilmente con le proprie proposte a un insieme di utenti a cui intendono rivolgersi per saziare le proprie esigenze alimentari. Il tutto attraverso una struttura su cui sia facile effettuare reasoning, per inferire pietanze incompatibili o meno con diete (veganismo, vegetarianismo) o con intolleranze ed allergie (come la celiachia, l'intolleranza al lattosio o il favismo).

Con l'ontologia, una volta effettuato reasoning, gli utenti potranno quindi filtrare i piatti non solo in base alla città o alla tipologia di locale, ma anche a seconda della presenza o meno di conformità a diete e disturbi o facendo riferimento a precedenti recensioni.

La nostra intenzione è di sfruttare il più possibile le potenzialità del reasoning, in modo da ridurre al minimo indispensabile il tempo e i dati necessari da introdurre manualmente per il funzionamento dell'ontologia.

Seguono due mockup dell'ipotetica piattaforma di delivery immaginata:



Figura 1: Pagina iniziale di ricerca

³Si veda ad esempio Just Eat: <https://www.justeat.it/help/article/115001318051/cosa-faccio-se-sono-allergico-a-qualcosa>

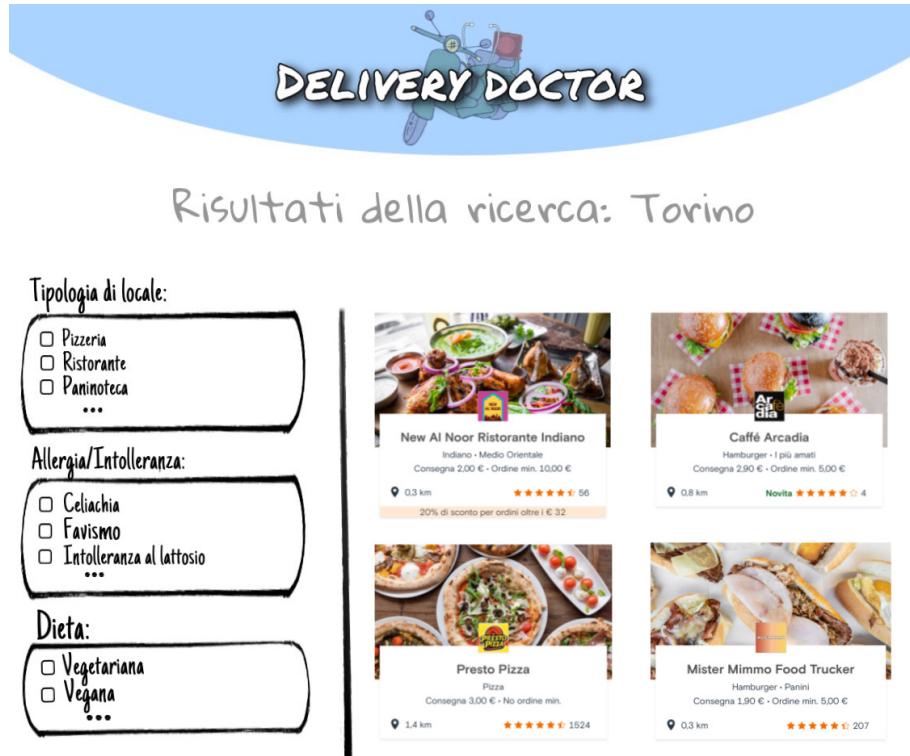


Figura 2: Risultati della ricerca di locali a Torino

3.2 Task specifici e contesto

I task principali che l'ontologia punta a rendere effettuabili sono:

- Ricerca di locali in una certa città, utilizzando filtri per i generi di pietanze servite e per gli allergeni;
- Gestione di un locale e delle pietanze servite;
- Introduzione e modifica di ingredienti e pietanze, indicabili come offerte da più locali;
- Pubblicazione di recensioni dei locali;
- Gestione delle attività di ordine e consegna da parte di addetti e proprietari delle attività.

3.3 Utenti target

- **Proprietari di esercizi alimentari** che vogliono gestire il traffico lavorativo del loro locale;
- **Persone che lavorano nel settore delle consegne** e che usano l'applicativo per trovare nuovi ordini da recapitare;
- **Clienti** che vogliono effettuare un ordine da un locale o recensire un posto dopo una consumazione.

4 Descrizione del Dominio e Fonti

4.1 Dominio

Il dominio, collocato nell'ambito dei servizi di consegna, ha come fine la rappresentazione di Locali, Pietanze, Ingredienti, Disturbi alimentari e Persone. Queste cinque classi principali sono correlate tra di loro:

- Le pietanze sono *composte da* vari ingredienti, oppure ne utilizzano alcuni come mezzo per la cottura (ad esempio frittura). Inoltre ereditano dagli ingredienti che le compongono le **incompatibilità** con gli stessi disturbi alimentari;
- Gli ingredienti possono essere *derivati* a loro volta da altri ingredienti, e in questo modo è possibile descrivere precisamente la composizione delle pietanze. In modo simile alle pietanze, anche gli ingredienti ereditano dagli ingredienti da cui sono derivati le **incompatibilità** con i disturbi alimentari specificati o inferiti;
- Multipli Locali possono *servire* le stesse pietanze. Inoltre, le Persone possono *piazzare ordini* (che saranno *consegnati* da altre Persone), *pubblicare recensioni* riguardo un certo ordine e *aprire nuovi Locali*.

4.2 Fonti

In modo da poter rappresentare adeguatamente il dominio, ci siamo basati prevalentemente su diversi siti web.

Per gli allergeni, abbiamo fatto affidamento a:

- ”Allergeni alimentari”, su *Wikipedia*;
- ”Carenza di glucosio-6-fosfato deidrogenasi (Favismo)”, su *Wikipedia*;
- ”Intolleranza al Lattosio”, su *Wikipedia*;
- ”Celiachia”, sul sito dell’*Associazione Italiana Celiachia*;
- ”Allergia alla frutta a guscio”, sul sito dell’*Ospedale Bambino Gesù*;
- ”Intolleranza al Fruttosio” sul *Portale delle malattie rare e dei farmaci orfani*;
- ”Allergia al Nickel”, sul sito della *Fondazione Veronesi*.

Per le informazioni sulle consegne abbiamo fatto affidamento ad alcuni siti che offrono servizi di delivery (Just Eat, Glovo, Deliveroo).

Le informazioni sulle città provengono da Wikipedia e Google Maps.

Infine per le Pietanze abbiamo utilizzato le ricette di Giallo Zafferano, di alcuni libri di cucina in nostro possesso (tra cui ”Muffin, Sccone e Torte”) e alcune preparazioni che conosciamo e prepariamo in casa.

5 Documentazione del Dominio

5.1 Risorse simili ed ispirazione

La nostra ontologia spazia su un buon numero di tematiche; possiamo affermare che le ispirazioni e le risorse utilizzate siano piuttosto variegate.

Cominciamo mostrando siti noti che offrono servizi di delivery, da cui abbiamo in parte tratto ispirazione. Iniziamo dalla schermata di ricerca di Just Eat una volta selezionata *Via Pessinotto 12*, a Torino:

The screenshot shows the Just Eat interface for a search in Turin. At the top, there's a navigation bar with the Just Eat logo, a search bar, and links for 'Accedi' and 'Aiuto'. Below the header, there are categories for different cuisines: Pizza, Hamburger, Cinese, Italiano, Panini, Giapponese, Indiano, and Americano. The main content area shows a list of 123 open restaurants. Each listing includes the restaurant name, cuisine type, average rating (from 1 to 5 stars), number of reviews, delivery information (e.g., delivery time, minimum order, cost), and a small thumbnail image. On the left side, there are filters for 'Tutte le cucine (A-Z)', 'Cambia zona', and various food categories like Americano, Asianfusion, and Cinese. On the right, there are buttons for 'Ordinato per Consigliati' and 'Sponsorizzato'.

Figura 3: Alcuni locali elencati su Just Eat nelle vicinanze del Dipartimento

Come è possibile notare, i filtri che un utente può applicare si allontanano da quelle che sono le nostre idee per l'ontologia. I locali vengono infatti filtrati solamente in base al tipo di cucina che offrono, alle stelle associate e alla presenza o meno della consegna gratuita. Non vi è alcun tipo di riferimento

a diete o allergie.

Spostiamoci ora su Glovo che sembra avvicinarsi di più alle nostre intenzioni:

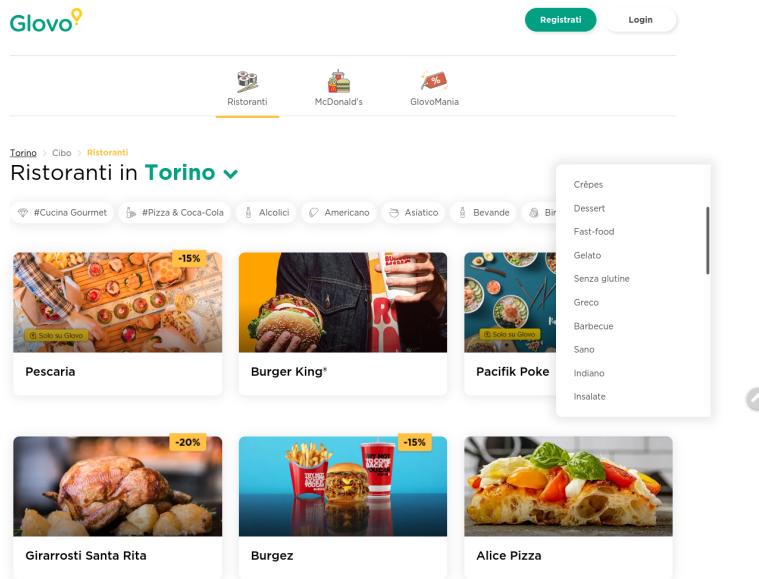


Figura 4: Alcuni locali registrati per il comune di Torino

È possibile notare come Glovo condivida alcune delle nostre idee. In primo luogo è data la possibilità di filtrare i locali in base alla città senza essere costretti, a differenza di Just Eat, a specificare un indirizzo in particolare. In secondo luogo, è possibile filtrare i ristoranti in una città anche in base alla dieta (è disponibile tuttavia solo quella Vegetariana) e all'attributo "senza glutine". Manca però in questo caso la possibilità di filtrare i locali in base alle valutazioni degli utenti e la scelta delle diete/allergie come criteri sono molto scarsi.

La piattaforma da noi immaginata presenta il meglio dei criteri di questi due siti, ampliandoli: la possibilità di scremare i risultati in base ad un ricco catalogo di allergie e diete senza però trascurare l'importanza delle recensioni dei clienti che avvalorano o meno un locale.

Cambiamo ora completamente tipologia di risorsa per analizzare un altro aspetto di Delivery Doctor: la tassonomia che descrive le pietanze e gli ingredienti che le compongono. Non possiamo non citare in questo caso Giallo Zafferano:

Giallo Zafferano

RICETTE ▾ BENESSERE ▾ CHEF BLOG SHOPPING ABBONATI

Cerca una ricetta

SPECIALE Lievitati dolci e salati

LIEVITATI PIZZE E FOCACCE

Pizza Margherita

4,2/5 VOTA

0:00 / 5:55

CONDIVIDI PREFERITI COMMENTI FATTE DA VOI

PRESENTAZIONE

Napoli è mille colori, si ma mille colori di pizza! Anche se chiedendo in giro vi diranno che l'originale è una sola, la pizza margherita, e non esistono altri tipi di condimento. Ogni pizzaiolo che si rispetti ha la sua ricetta e la custodisce gelosamente, ma noi ne abbiamo creata una tutta nostra e abbiamo scelto di condividerla con voi... rispettando passo passo le nostre indicazioni, anche a casa la pizza margherita sarà buona proprio come in pizzeria! I segreti sono pochi e ve ne accorgerete: per prima cosa gli ingredienti devono essere tutti di ottima qualità, il forno deve essere rovente e in ultimo ma non meno importante nell'impasto dovrete inserire tutto il vostro amore per la cucina; solo in questo modo la pizza margherita sarà speciale! E con lo stesso [impasto della pizza](#) potete realizzare le famose [pizze fritte](#), una ricca [parigina](#), delle soffici [pizzette rosse](#), un'insolita [clambella pizza!](#) Se, invece, siete alla ricerca di un impasto più digeribile e aromatico provate la [pinsa](#) da fare al gusto margherita oppure la nostra [pizza con lievito madre!](#)

INGREDIENTI

1679 Kcal Calorie per porzione +info

INGREDIENTI PER 2 PIZZE DI 28 CM DI DIAMETRO

• Farina Manitoba 200 g	• Farina 00 300 g
• Acqua 300 ml	• Olio extravergine d'oliva 35 g
• Sale fino 10 g	• Lievito di birra fresco 5 g

PER CONDIRE

• Polpa di pomodoro 500 g	• Fiordilatte 400 g
• Origano secco 1 cucchiaio	• Sale fino 1 cucchiaino
• Basilico q.b.	• Olio extravergine d'oliva q.b.

AGGIUNGI ALLA LISTA DELLA SPESA

Figura 5: Una ricetta per la Pizza Margherita descritta da Giallo Zafferano

Come si può vedere nell'immagine precedente, una pietanza come la pizza margherita (presente anche nella nostra ontologia) viene descritta con

moltissimi dettagli: non sono presenti solo gli ingredienti necessari (che ci permettono di classificarla come pietanza vegetariana) ma anche tempi di cottura/preparazione, calorie e tutte le fasi della preparazione, con l'aggiunta di immagini e contenuti video.

Da qui deriva la nostra idea di arricchire le pietanze proposte dai vari esercizi alimentari con gli ingredienti che le compongono. In questo modo anche la nostra ontologia è in grado di inferire la compatibilità o meno di un piatto con diete o disturbi alimentari, e come conseguenza è possibile per l'utente filtrare i locali in maniera coerente con le sue esigenze.

Per quanto riguarda invece la classificazione dei disturbi alimentari ci siamo ispirati alla decima versione del thesauro *"International Classification of Diseases"* dell'OMS, al quale ci siamo anche allineati:

The screenshot shows the BioPortal interface for the International Classification of Diseases, Version 10 - Clinical Modification. The page title is "International Classification of Diseases, Version 10 - Clinical Modification" and the last upload date is "January 6, 2021". The main content area displays the details of the "Celiac disease" entry. The "Summary" tab is selected, followed by "Classes", "Properties", "Notes", "Mappings", and "Widgets". The "Details" section contains the following information:

- Preferred Name:** Celiac disease
- Synonyms:** Nontropical sprue
- ID:** http://purl.bioontology.org/ontology/ICD10CM/K90.0
- altLabel:** Celiac; gluten-sensitive enteropathy; Celiac disease with steatorrhea
- CODE ALSO:** exocrine pancreatic insufficiency (K86.81)
- cui:** C4509268, C0007570, C4268664
- Inverse of SIB:**
 - Blind loop syndrome, not elsewhere classified
 - Tropical sprue
 - Other malabsorption due to intolerance
 - Intestinal malabsorption, unspecified
 - Pancreatic steatorrhea
 - Transient malabsorption
 - Other intestinal malabsorption
- notation:** K90.0
- Order number:** 14194
- prefLabel:** Celiac disease
- tui:** T047
- USE ADDITIONAL:** code for associated disorders including: dermatitis herpetiformis (L13.0); gluten ataxia (G32.81)
- subClassOf:** Intestinal malabsorption

The left sidebar lists various categories of diseases, such as "Certain conditions originating in the perinatal period (P00-P99)", "Certain infectious and parasitic diseases (A00-B99)", and "Diseases of the blood and blood-forming organs and certain disorders involving connective tissue, deformations and chromosomal aberrations (D00-D99)".

Figura 6: Descrizione della celiachia nell'ICD10.

5.2 Allineamenti

Oltre all'allineamento con l'*International Classification of Diseases*, già presentato nella sezione precedente, ci siamo allineati con Schema.org, da noi usato per la localizzazione geografica dei locali e per i concetti inerenti la recensione degli stessi.

Per quanto riguarda la modellazione delle azioni abbiamo fatto riferimento a Provenance e in particolare il PROV Model, dal quale abbiamo definito delle sottoclassi più specifiche e delle sottoproprietà che collegassero le nuovi classi.

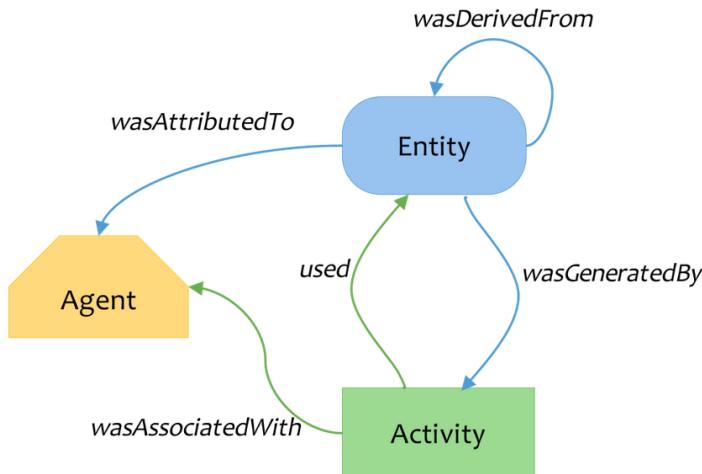


Figura 7: PROV Model

A partire dalle classi **Activity**, **Entity**, **Agent** e **Role** abbiamo introdotto le azioni *Recensire*, *Ordinare*, *Consegnare* e *Avviare Impresa* ponendole come `rdfs:subClassOf prov:Activity`. Attorno all'attività ruotano i concetti di:

- **Agente**, colui che fa l'azione;
- **Ruolo**, il compito svolto dalla persona nell'**Activity** ed entità che identifica elementi creati o usati nel corso dell'azione.

Per quanto riguarda l'allineamento con le varie risorse esterne abbiamo usato `rdfs:subClassOf` e `owl:equivalentClass` con questo criterio: là dove la nostra gerarchia si discostava da quella esterna abbiamo usato `owl:equivalentClass`

per collegare le classi che rappresentavano lo stesso concetto (riguarda il caso del nostro utilizzo di Schema.org).

Dove invece c'era il bisogno di introdurre classi più specifiche di quelle presenti e in cui poteva risultare utile usufruire della struttura delle relazioni della risorsa esterna abbiamo usato **rdfs:subClassOf** (come nel caso di Provenance).

6 LODE

Al momento della progettazione di Delivery Doctor, il progetto LODE sviluppato da Silvio Peroni presenta alcuni problemi tecnici, e spesso il servizio online è irraggiungibile. Per poter ottenere una documentazione completa dell'ontologia è stato quindi utilizzata un'alternativa che imita in tutto e per tutto LODE: pyLODE. Si tratta di un'implementazione in python, utilizzabile in locale.

Una copia aggiornata della documentazione è disponibile sul repository gitHub dell'Ontologia.

7 Visualizzazione Ontologia

Vediamo ora una versione grafica dell'ontologia che ne espliciti le relazioni tra le varie classi mediante sussunzione.

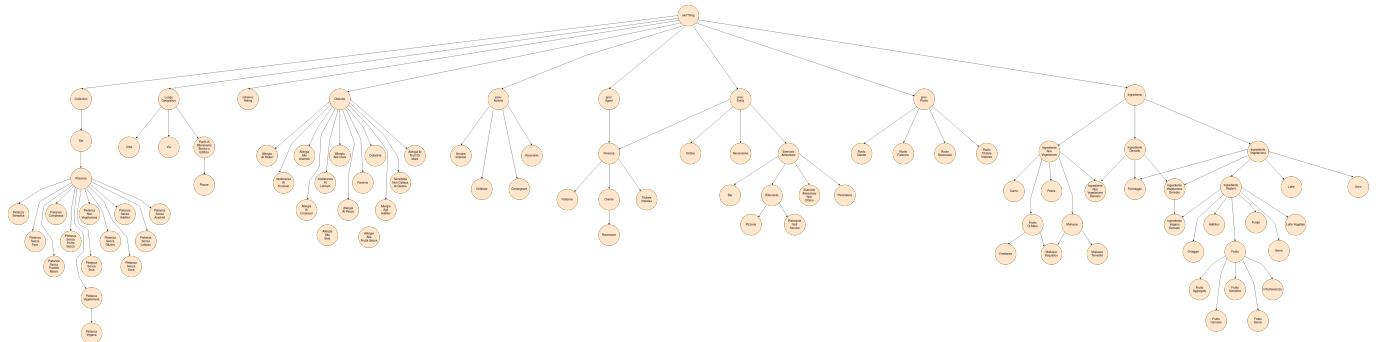


Figura 8: Gerarchia delle classi

Come si evince dalla tassonomia appena presentata, a cui si rimanda all'ultima sezione per una visione ingrandita, per quanto riguarda la modellazione delle pietanze come insieme di ingredienti che le compongono abbiamo usato l' Ontology Design Pattern Set:

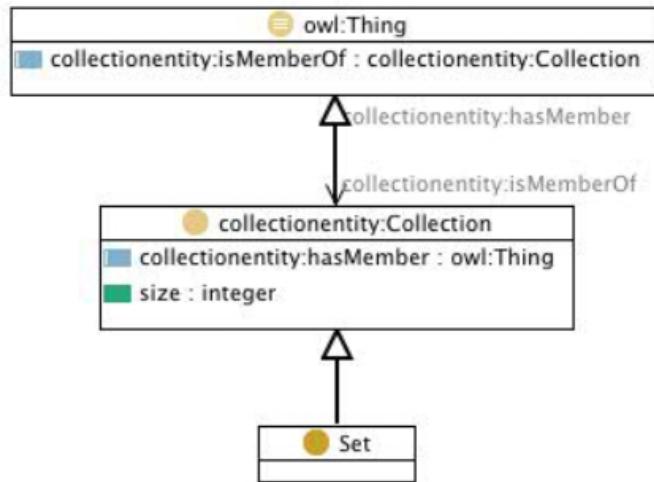


Figura 9: ODP Set

Abbiamo quindi definito la Pietanza come sussunta a Set ed usato la object property ereditata "hasMember" per la definizione delle nuove proprietà "utilizza ingrediente" e "utilizza indirettamente l'ingrediente" che collegano direttamente la pietanza con gli ingredienti di cui è composta.

Mostriamo ora due esempi tratti da Graph DB composti da un insieme di triple e dal Visual graph ad esse associato.

Partiamo dall'ordine effettuato da un utente generico Luca presso il Dubai Coffee Lounge:

	subject	predicate	object
1	deliverydoctor:CappuccinoECroissant	deliverydoctor:ordineCompostoDaPietanza	deliverydoctor:CappuccinoSpeciale
2	deliverydoctor:CappuccinoECroissant	deliverydoctor:ordineCompostoDaPietanza	deliverydoctor:Croissant
3	deliverydoctor:CappuccinoECroissant	deliverydoctor:ordineConsegnatoDa	deliverydoctor:Gianni
4	deliverydoctor:CappuccinoECroissant	deliverydoctor:ordineCreatoDaAzione	deliverydoctor:OrdinareDubai
5	deliverydoctor:CappuccinoECroissant	rdf:type	deliverydoctor:Ordine
6	deliverydoctor:CappuccinoECroissant	rdf:type	owl:NamedIndividual
7	deliverydoctor:CappuccinoECroissant	rdf:type	prov:Entity
8	deliverydoctor:CappuccinoECroissant	rdfs:comment	"Luca's order for a cappuccino and a croissant at "Dubai""@en
9	deliverydoctor:CappuccinoECroissant	rdfs:comment	"Ordine di un cappuccino e un croissant effettuato da Luca al "Dubai""@it
10	deliverydoctor:CappuccinoECroissant	rdfs:label	"Capp&croiss order Luca @ Dubai"@en
11	deliverydoctor:CappuccinoECroissant	rdfs:label	"Ordine capp&croiss Luca @ Dubai"@it
12	deliverydoctor:CappuccinoECroissant	skos:prefLabel	"Capp&croiss order Luca @ Dubai"@en
13	deliverydoctor:CappuccinoECroissant	skos:prefLabel	"Ordine capp&croiss Luca @ Dubai"@it
14	deliverydoctor:CappuccinoECroissant	prov:wasGeneratedBy	deliverydoctor:OrdinareDubai

Figura 10: Insieme delle triple usate per modellare l'ordine di Luca

Una visualizzazione grafica di questo è la seguente:

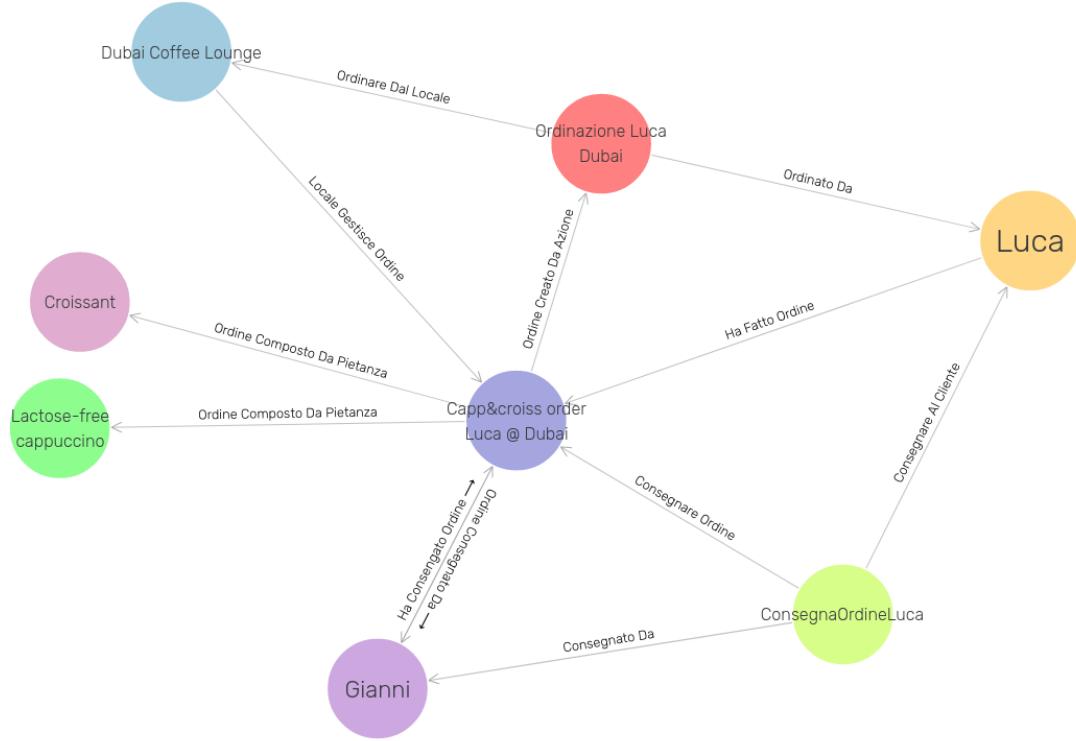


Figura 11: Visual Graph associato all'ordine di Luca

Come altro esempio presenteremo la modellazione dell'entità Pizza Margherita collegandoci con quanto presentato nella sezione 5 e la sua definizione in Giallo Zafferano.

Iniziamo mostrando le triple ad essa collegate:

	subject	predicate	object
1	deliverydr:PizzaMargherita	deliverydr:entitaWikidata	wiki:Q500724
2	deliverydr:PizzaMargherita	deliverydr:haNumerolngrediente	"5"^^xsd:positiveInteger
3	deliverydr:PizzaMargherita	deliverydr:utilizzalngrediente	deliverydr:Basilico
4	deliverydr:PizzaMargherita	deliverydr:utilizzalngrediente	deliverydr:ImpastoPizza
5	deliverydr:PizzaMargherita	deliverydr:utilizzalngrediente	deliverydr:Mozzarella
6	deliverydr:PizzaMargherita	deliverydr:utilizzalngrediente	deliverydr:OlioExtraOliva
7	deliverydr:PizzaMargherita	deliverydr:utilizzalngrediente	deliverydr:SugoPomodoro
8	deliverydr:PizzaMargherita	rdf:type	_:node3721
9	deliverydr:PizzaMargherita	rdf:type	owl:NamedIndividual
10	deliverydr:PizzaMargherita	rdfs:comment	"Pizza Margherita is a typical Neapolitan pizza, made with San Marzano tomatoes, mozzarella cheese, fresh basil and extra-virgin olive oil."@en
11	deliverydr:PizzaMargherita	rdfs:comment	"La pizza Margherita è la tipica pizza napoletana, condita con pomodoro, mozzarella (tradizionalmente è usato il fior di latte, non quella di bufala), basilico fresco e olio d'oliva; è, assieme alla pizza marinara, la più popolare pizza italiana."@it
12	deliverydr:PizzaMargherita	skos:altLabel	"Margherita"@it
13	deliverydr:PizzaMargherita	skos:prefLabel	"Margherita Pizza"@en
14	deliverydr:PizzaMargherita	skos:prefLabel	"Pizza Margherita"@it

Figura 12: Insieme delle triple usate per modellare la pizza margherita

Mostriamo ora, esattamente come prima, la loro trasposizione grafica:



Figura 13: Visual Graph associato alla pizza margherita

8 Ontologia in formato Turtle

Per visionare l'ontologia in formato Turtle rimandiamo il lettore alla pagina del progetto.

9 Queries SPARQL

Suddivideremo questa sezione in 5 parti: la prima contenente l'interazione e quindi le queries di un utente normale, la seconda per l'utente proprietario di un esercizio alimentare, la terza riguardano invece l'interazione di un fattorino e la quarta sarà dedicata alle queries dirette verso risorse ontologiche esterne. Nella parte finale sono mostrate delle semplici queries di utilità.

9.1 Utente Normale

Il flusso d'interazione dell'utente generico Luca riguarderà l'iniziale ricerca dei locali nella sua città, Torino, per poi filtrare i risultati sulla base degli esercizi alimentari che dispongono di almeno una pietanza compatibile con la celiachia. Terminerà l'interazione visionando tutti gli ordini compiuti.

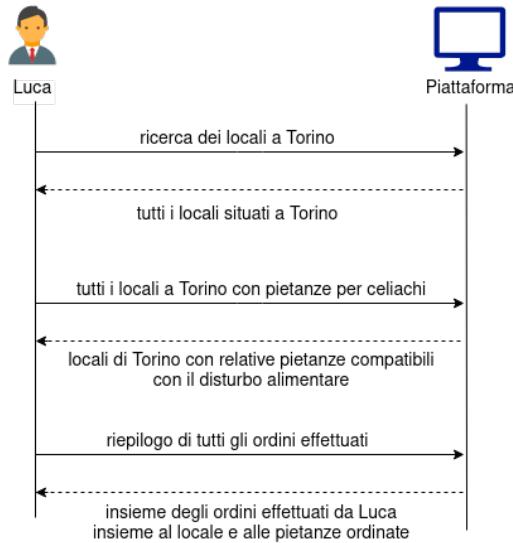


Figura 14: Interazione tra Luca e la piattaforma

Iniziamo con la Query che permette all'utente di ottenere tutti i locali situati a Torino.

```
1 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
2 PREFIX deliverydr:
3 <http://www.semanticweb.org/progettomodsem2020/deliverydoctor#>
4
5 SELECT ?localeNome WHERE{
6     ?locale deliverydr:localeSituatoInCitta ?citta;
7         deliverydr:haNome ?localeNome.
8     ?citta deliverydr:haNome "Torino".
9 }
```

Con risultato:

?localeNome
Dubai Coffee Lounge
Napples

L'utente ora filtrerà i risultati ottenuti cercando i locali e le rispettive pietanze servite compatibili con il disturbo celiachia.

```
1 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
2 PREFIX deliverydr:
3 <http://www.semanticweb.org/progettomodsem2020/deliverydoctor#>
4
5 SELECT ?localeNome ?pietanza WHERE {
6     ?locale deliverydr:localeSituatoInCitta ?citta;
7         deliverydr:localeServePietanza ?pietanza;
8         deliverydr:haNome ?localeNome.
9     ?pietanza deliverydr:pietanzaCompatibileConDisturbo
10            deliverydr:Celiachia,
11            deliverydr:AllergiaGlutine.
12     ?citta deliverydr:haNome "Torino".
13 }
```

Ottenendo come risultato:

?localeNome	?pietanza
<i>Dubai Coffee Lounge</i>	<i>deliverydr:CappuccinoSpeciale</i>
<i>Dubai Coffee Lounge</i>	<i>deliverydr:RisottoSugo</i>

Come possiamo notare dall'insieme dei locali verrà escluso Napples che nella nostra ontologia propone solo pizze e in maniera analoga tra le pietanze proposte dal Dubai verranno filtrate solo quelle compatibili con la celiachia/allergia al glutine.

Supponiamo ora che il nostro utente Luca voglia vedere lo storico degli ordini che ha effettuato.

```

1 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
2 PREFIX foaf: <http://xmlns.com/foaf/0.1/>
3 PREFIX deliverydr:
4 <http://www.semanticweb.org/progettoModsem2020/deliverydoctor#>
5
6 SELECT ?ordine ?localeNome ?pietanza WHERE{
7     ?ordinare deliverydr:ordinatoDa ?persona;
8         deliverydr:ordinareDalLocale ?locale.
9     ?ordine deliverydr:ordineCreatoDaAzione ?ordinare;
10    deliverydr:ordineCompostoDaPietanza ?pietanza.
11    ?locale deliverydr:haNome ?localeNome.
12    ?persona foaf:givenName "Luca".
13 }
```

?ordine	?locale	?pietanza
<i>deliverydr:CappuccinoECroissant</i>	<i>Dubai Coffee Lounge</i>	<i>deliverydr:CappuccinoSpeciale</i>
<i>deliverydr:CappuccinoECroissant</i>	<i>Dubai Coffee Lounge</i>	<i>deliverydr:Croissant</i>

9.2 Utente Proprietario

Nel caso dell'Utente Proprietario, il flusso d'interazione è più frammentato, dato che i compiti che deve assolvere sono molteplici, e come esempio può:

1. Controllare gli ordini che la sua attività ha ricevuto
2. Controllare le recensioni lasciate dagli utenti sulla sua attività

- Visionare il menù proposto dall'attività per valutare eventuali cambiamenti

Supponendo che il Proprietario diriga il locale "Dubai Coffee Lounge", è piuttosto immediata la ricerca di tutti gli ordini che gestisce e delle pietanze richieste:

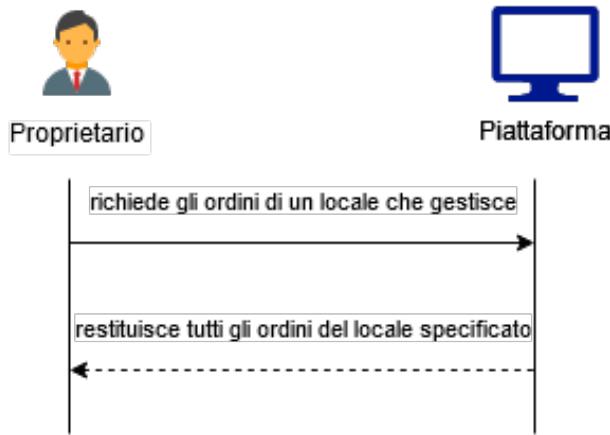


Figura 15: Il Proprietario richiede gli ordini gestiti per uno specifico locale

```

1 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
2 PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
3 PREFIX deliverydr:
4 <http://www.semanticweb.org/progettomodsem2020/deliverydoctor#>
5
6 SELECT ?ordine ?persona ?pietanza WHERE {
7     ?locale deliverydr:haNome "Dubai Coffee Lounge".
8     ?ordine deliverydr:ordineCompostoDaPietanza ?pietanza;
9         deliverydr:ordineCreatoDaAzione ?azione.
10    ?azione deliverydr:ordinatoDa ?persona;
11        deliverydr:ordinareDalLocale ?locale.
12 }
  
```

?ordine	?persona	?pietanza
<i>deliverydr: CappuccinoECroissant</i>	<i>deliverydr:Luca</i>	<i>deliverydr:Croissant</i>
<i>deliverydr: CappuccinoECroissant</i>	<i>deliverydr:Luca</i>	<i>deliverydr: CappuccinoSpeciale</i>

Assieme alla visualizzazione degli ordini gestiti, potrebbe essere utile sapere cosa pensano i clienti del locale e del servizio che offre. Per far ciò, è necessario recuperare alcuni dati:

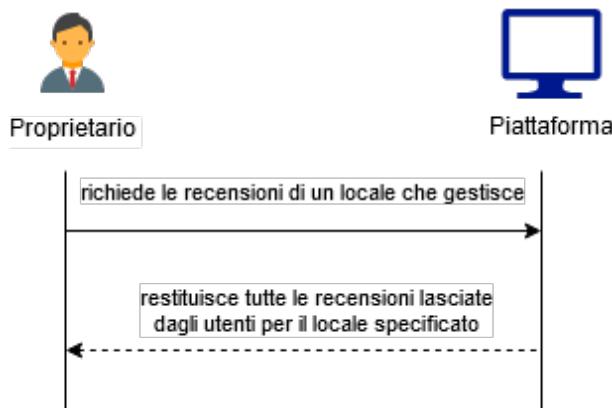


Figura 16: Il Proprietario richiede le recensioni pubblicate per uno specifico locale

```

1 PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
2 PREFIX schema: <http://schema.org/>
3 PREFIX deliverydr:
4 <http://www.semanticweb.org/progettomodsem2020/deliverydoctor#>
5
6 SELECT ?persona ?testo ?stelle WHERE {
7     ?locale deliverydr:haNome "Dubai Coffee Lounge".
8     ?azione deliverydr:recensireLocale ?locale;
9     deliverydr:recensitoDa ?persona;
10    deliverydr:scrivereRecensione ?recensione.
11    ?recensione schema:reviewBody ?testo;
12        schema:reviewRating ?rating.
13    ?rating deliverydr:possiedeStelle ?stelle
  
```

14 }

?persona	?testo	?stelle
deliverydr:Luca	<i>Servono un'ottima cheesecake e dei...</i>	5

Infine, se volesse vedere le pietanze che il locale serve al momento, per variare il menù (tenendo conto del numero di ingredienti che le compongono):

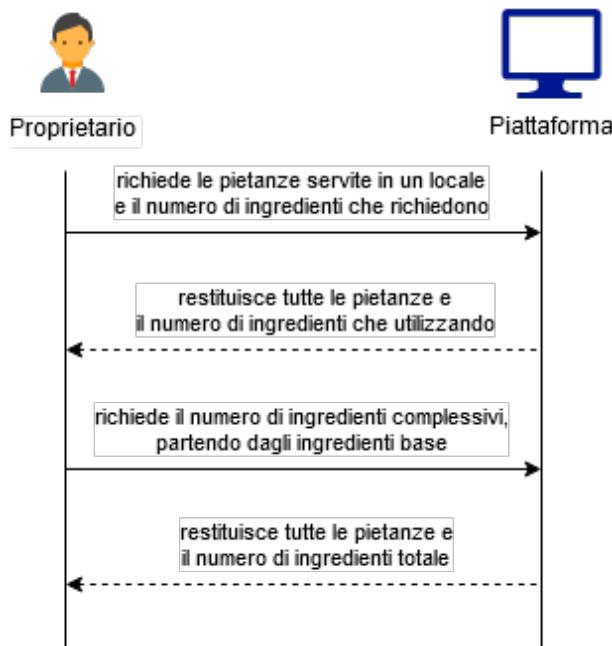


Figura 17: Il Proprietario richiede le pietanze servite in uno specifico locale, richiedendo successivamente anche il numero complessivo di ingredienti utilizzati

```
1 PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
2 PREFIX deliverydr:
3 <http://www.semanticweb.org/progettomodsem2020/deliverydoctor#>
4
5 SELECT ?pietanza
6 (count(DISTINCT ?ingrediente) AS ?numero_ingredienti)
```

```

7 WHERE {
8     ?locale deliverydr:haNome "Dubai Coffee Lounge".
9     ?locale deliverydr:localeServePietanza ?pietanza.
10    ?pietanza deliverydr:utilizzaIngrediente ?ingrediente.
11 }
12 GROUP BY ?pietanza

```

?pietanza	?numero_ingredienti
<i>deliverydr:PolpetteAlSugo</i>	3
<i>deliverydr:CappuccinoSpeciale</i>	2
<i>deliverydr:LasagneAlForno</i>	6
<i>deliverydr:Croissant</i>	4
<i>deliverydr:RisottoSugo</i>	2

Potrebbe essere utile conoscere anche il numero di ingredienti non direttamente utilizzati dalla pietanza. In tal caso la query richiede anche una UNION:

```

1 PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
2 PREFIX deliverydr:
3 <http://www.semanticweb.org/progettomodsem2020/deliverydoctor#>
4
5 SELECT ?pietanza
6 (count(DISTINCT ?ingrediente) AS ?numero_ingredienti)
7 WHERE {
8     ?locale deliverydr:haNome "Dubai Coffee Lounge".
9     ?locale deliverydr:localeServePietanza ?pietanza.
10    {?pietanza deliverydr:utilizzaIngrediente ?ingrediente}
11    UNION
12    {?pietanza deliverydr:utilizzaIngredienteIndirettamente
13      ?ingrediente}
14 }
15 GROUP BY ?pietanza

```

In questo caso i valori contenuti nella colonna del numero di ingredienti saranno leggermente differenti:

?pietanza	?numero_ingredienti
<i>deliverydr:PolpetteAlSugo</i>	14
<i>deliverydr:CappuccinoSpeciale</i>	4
<i>deliverydr:LasagneAlForno</i>	15
<i>deliverydr:Croissant</i>	6
<i>deliverydr:RisottoSugo</i>	3

9.3 Utente Fattorino

Per l'interazione con l'utente abilito alle consegne degli ordini, vediamo una tipica azione che potrebbe fare il fattorino Gianni, come controllare tutte le consegne che ha portato a termine.

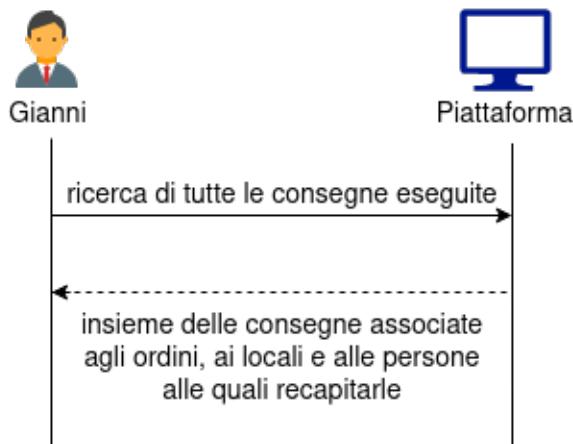


Figura 18: Interazione tra Gianni e la piattaforma

```

1 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
2 PREFIX foaf: <http://xmlns.com/foaf/0.1/>
3 PREFIX deliverydr:
4   <http://www.semanticweb.org/progettomodsem2020/deliverydoctor#>
5
6 SELECT ?consegna ?ordine ?localeNome ?destinatario WHERE{
7   ?consegna deliverydr:consegnareAlCliente ?destinatario;
8   deliverydr:consegnatoDa ?persona;
9   deliverydr:consegnareOrdine ?ordine.
10 ?ordine deliverydr:ordineCreatoDaAzione ?ordinare.
  
```

```

11 ?ordinare deliverydr:ordinareDalLocale ?locale.
12 ?locale deliverydr:haNome ?localeNome.
13 ?persona foaf:givenName "Gianni".
14 }

```

?consegna	?ordine	?localeNome	destinatario
<i>deliverydr: ConsegnaOrdine Luca</i>	<i>deliverydr: CappuccinoE Croissant</i>	<i>deliverydr: Dubai Coffee Lounge</i>	<i>deliverydr: Luca</i>

9.4 Queries verso risorse esterne

Verrà prima presentata un flowchart dell'interazione per poi soffermarci sulle due queries.

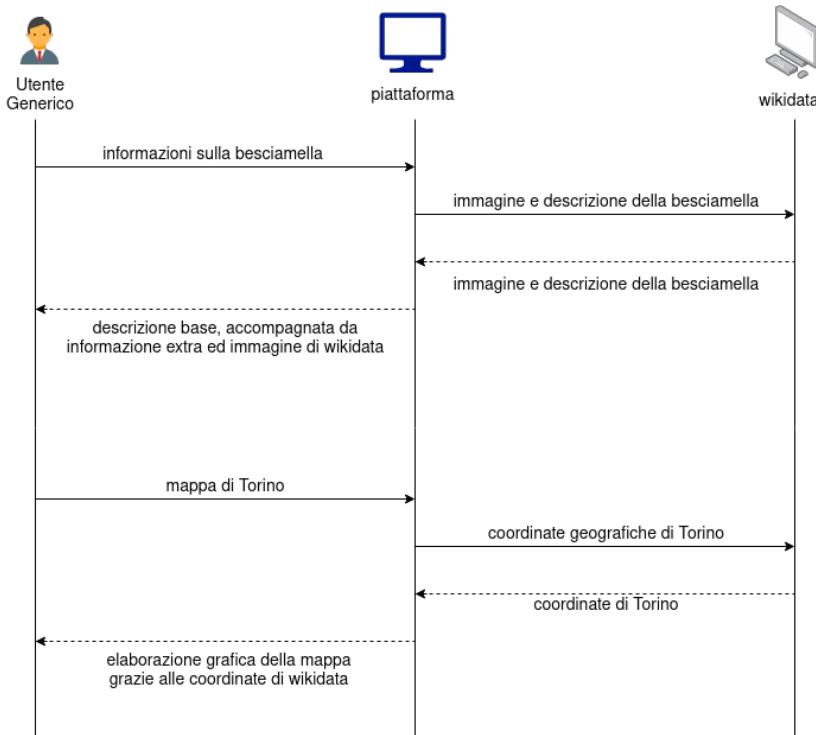


Figura 19: Schema interazione utente-piattaforma-wikidata

Come primo esempio di query indirizzata verso risorse esterne supponiamo la situazione in cui un utente, dopo aver visto gli ingredienti che compongono una pietanza, sia interessato ad ottenere informazioni su uno di essi. Supponiamo ad esempio che un utente, osservi le lasagne servite dal *Dubai Coffee Lounge* e voglia avere più informazioni riguardo la besciamella che le compone:

```

1 PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
2 PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
3 PREFIX schema: <http://schema.org/>
4 PREFIX deliverydr:
5 <http://www.semanticweb.org/progettoModsem2020/deliverydoctor#>
6
7 SELECT ?immagine ?descrizioneBase ?informazioneExtra
8 WHERE {
9     ?ingr skos:prefLabel "Besciamella";
10    rdfs:comment ?descrizioneBase.
11
12    FILTER ( lang(?descrizioneBase) = "it" ).
13
14    OPTIONAL {
15        ?ingr deliverydr:entitaWikidata ?entita.
16
17        SERVICE <https://query.wikidata.org/sparql> {
18            OPTIONAL {
19                ?entita <http://www.wikidata.org/prop/direct/P18>
20                           ?img.
21            }
22            OPTIONAL {
23                ?entita schema:description ?info.
24            }
25        }
26
27    FILTER ( lang(?informazioneExtra) = "it" )
28
29    BIND(COALESCE(?img, "n/a") AS ?immagine).
30    BIND(COALESCE(?info, "n/a") AS ?informazioneExtra).
31
32 }
```

33

34 [LIMIT 1](#)

Come risultato otteniamo l'immagine di Wikidata della besciamella, assieme alla descrizione base presente nell'ontologia e la descrizione dell'ingrediente estratta sempre da wikidata.

?immagine	?descrizioneBase	?informazioneExtra
https://upload.wikimedia.org/wikipedia/...	<i>Salsa base della cucina italiana, composta da farina, burro, latte e aromatizzata con noce moscata</i>	<i>salsa di base utilizzata per ricette più complesse</i>



Figura 20: Immagine referenziata dal link ricevuto

Come secondo caso immaginiamo invece un utente straniero in visita alla città di Torino.

Volendo trovare dei locali presso i quali poter ordinare, potrebbe essere interessato ad ottenere una mappa della città su cui poterli collocare assieme ad un'illustrazione della città, per una maggiore comodità nella ricerca.

A tale scopo interroghiamo Wikidata per ottenere le coordinate geografiche (sotto forma di Punto WKT di GeoSPARQL⁴) e l'illustrazione (come URL) della città, in modo poi da visualizzarli su una mappa di un'ipotetica applicazione finale centrata su di essa.

⁴<https://www.ogc.org/standards/geosparql>

```

1 PREFIX deliverydr:
2 <http://www.semanticweb.org/progettoMODSEM2020/deliverydoctor#>
3
4 SELECT ?img ?coordinate
5 WHERE {
6   ?citta deliverydr:haNome "Torino";
7   deliverydr:entitaWikidata ?ent.
8   SERVICE <https://query.wikidata.org/sparql> {
9     ?ent <http://www.wikidata.org/prop/direct/P625> ?coordinate;
10    <http://www.wikidata.org/prop/direct/P18> ?img.
11  }
12 }
```

?img	?coordinate
https://upload.wikimedia.org/wikipedia/...	"Point(7.7 45.066666666)"



Figura 21: Immagine referenziata dal link ricevuto

9.5 Query di utilità

Presentiamo ora delle query che usiamo nell'applicazione client per completare le query precedentemente mostrate. Queste query di utilità sono molto semplici ma essenziali e ci è parso opportuno riportarle anche nella documentazione.

Per ottenere tutte le città presenti nell'ontologia:

```
1 PREFIX deliverydr:  
2 <http://www.semanticweb.org/progettomodsem2020/deliverydoctor#>  
3  
4 SELECT ?nome  
5 WHERE {  
6     ?c a deliverydr:Citta;  
7     deliverydr:haNome ?nome  
8 }  
9 ORDER BY ?nome
```

?nome
Asti
Avigliana
Mondovì
Torino

Per ottenere tutti i nomi delle persone presenti nell'ontologia:

```
1 PREFIX foaf: <http://xmlns.com/foaf/0.1/>  
2 PREFIX deliverydr:  
3 <http://www.semanticweb.org/progettomodsem2020/deliverydoctor#>  
4  
5 SELECT ?nome  
6 WHERE {  
7     ?p a deliverydr:Persona;  
8     foaf:givenName ?nome  
9 }  
10 ORDER BY ?nome
```

?nome
<i>Federico Torrielli</i>
<i>Gianni</i>
<i>Ivan Spada</i>
<i>Lorenzo Sciandra</i>
<i>Luca</i>
<i>Stefano Vittorio Porta</i>

Per ottenere i nomi di tutti i locali presenti:

```

1 PREFIX deliverydr:
2 <http://www.semanticweb.org/progettomodsem2020/deliverydoctor#>
3
4 SELECT ?nomeLocale
5 WHERE {
6   ?locale a deliverydr:EsercizioAlimentare;
7   deliverydr:haNome ?nomeLocale.
8 }
9 ORDER BY ?nomeLocale

```

?nome
<i>Dubai Coffee Lounge</i>
<i>Galaxy Burger</i>
<i>Kingyo</i>
<i>Napples</i>
<i>Noir Caffè</i>

Infine, per ottenere i nomi di tutti gli ingredienti inseriti nell'ontologia:

```

1 PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
2 PREFIX deliverydr:
3 <http://www.semanticweb.org/progettomodsem2020/deliverydoctor#>
4
5 SELECT ?nome
6 WHERE {
7   ?i a deliverydr:ingrediente;
8   skos:prefLabel ?nome.
9   FILTER ( lang(?nome) = "it" )
10 }

```

11 ORDER BY ?**nome**

? nome
<i>Aceto di Vino Bianco</i>
<i>Acido tartarico</i>
<i>Acqua</i>
<i>Albicocca</i>
...

10 Applicazione Client

Il client è stato realizzato usando HTML5, Bootstrap 5, Vue.js e JQuery. Per la grafica abbiamo preso ispirazione dalle figure 1 e 2 in cui immaginavamo un mockup dell'ipotetica applicazione finale.

Il client si presenta quindi con una prima pagina html dedicata allo smistamento verso le 4 pagine che daranno la possibilità di eseguire le query introdotte nella selezione precedente, suddivise in base all'utente.

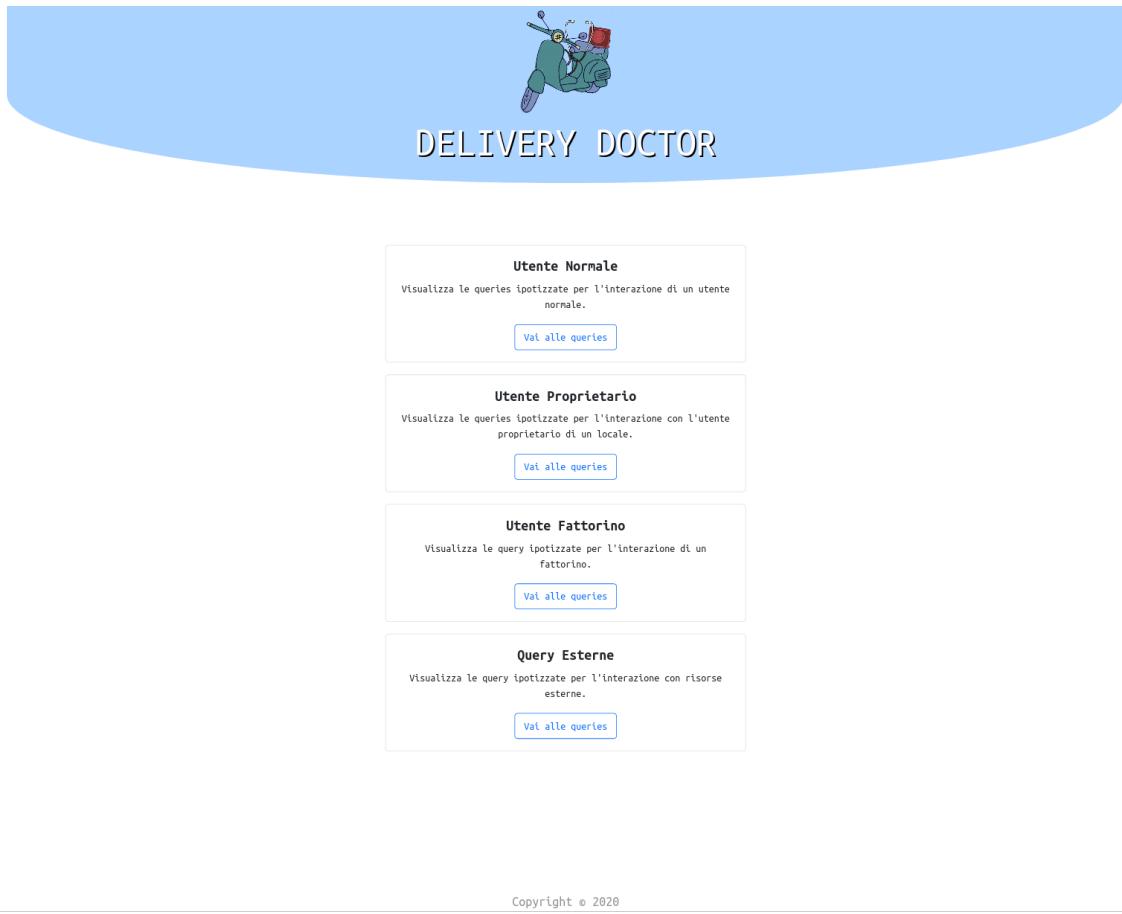


Figura 22: Pagina iniziale del client

Prendendo come esempio la sezione dell'utente normale, la nuova pagina presenterà una selezione tra le 3 queries introdotte.

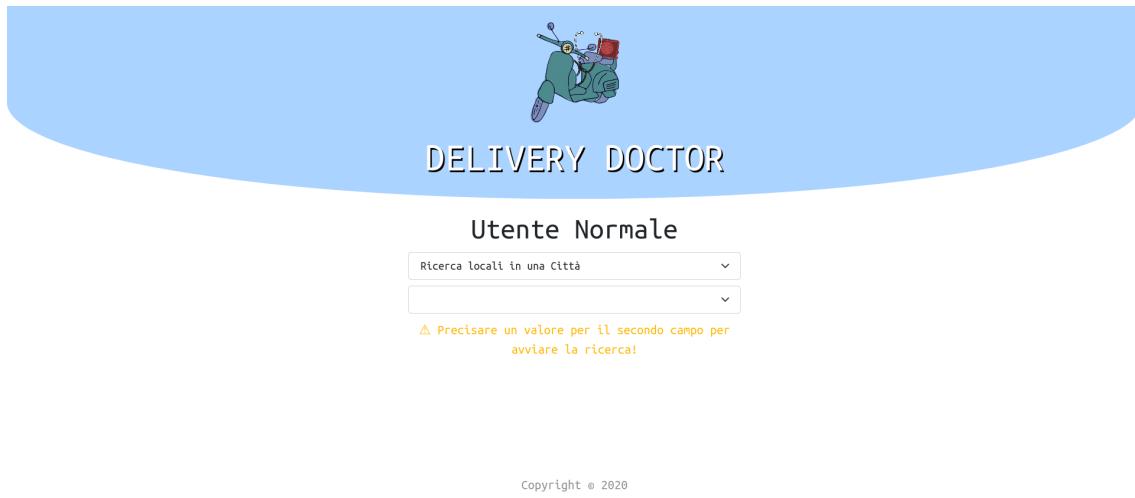


Figura 23: Pagina per eseguire le query di un utente normale

Come è possibile vedere, la pagina presenta due dropdown:

- Una prima, statica, dedicata alla scelta della query che si desidera effettuare tra quelle pre-impostate
- Una seconda, dinamica, i cui valori sono ottenuti con una richiesta http asincrona indirizzata al server GraphDB. Una volta recuperati, i valori permettono di completare la query scelta nella prima dropdown e di eseguirla.

Nel caso in cui non fosse selezionato nessun parametro nella seconda select, avremmo la situazione presentata in figura. Mostriamo ora un risultato totale di una query.

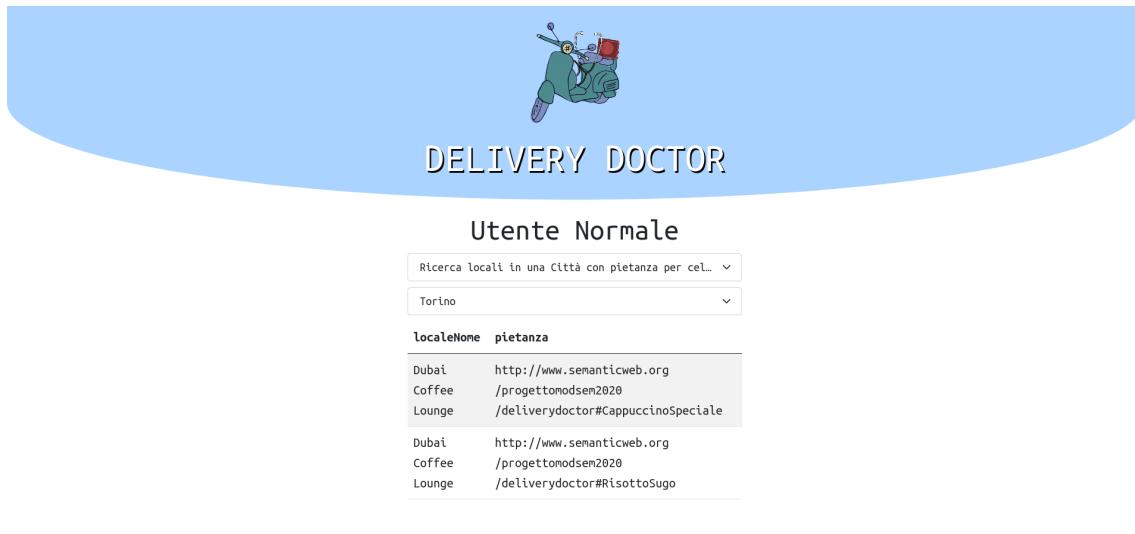


Figura 24: Risultati della query di ricerca di locali con pietanze senza glutine a Torino

Una volta scelto un valore per entrambe le dropdown, Vue si accorgerà di questo cambiamento ed eseguirà automaticamente la query. I risultati vengono richiesti al server GraphDB in formato CSV e JSON a seconda della complessità dei dati richiesti e, una volta ricevuti, saranno prima parsificati e successivamente inseriti in una tabella costruita dinamicamente grazie a Vue.js.

11 Regole SWRL

In questa sezione sarà presentata la base delle regole SWRL usate nel progetto. Vediamo come prime regole quelle che riguardano i ruoli assunti dalle persone. Le 4 regole che abbiamo usato per definire correttamente il ruolo di una persona si basano sulle seguenti considerazioni:

- se una persona ha eseguito almeno un ordine le verrà inferito il ruolo Ruolo Cliente;
- se un individuo ha recensito almeno un locale allora avrà associato il Ruolo Recensore;

- se una persona ha consegnato almeno un ordine avrà il Ruolo Fattorino;
- se un individuo ha avviato almeno un'impresa allora avrà il Ruolo Titolare.

```

1 RuoloCliente(?x) ^ ordinatoDa(?y, ?z) -> haRuolo(?z, ?x)
2
3 RuoloFattorino(?x) ^ consegnatoDa(?y, ?z) -> haRuolo(?z, ?x)
4
5 RuoloRecensore(?x) ^ recensitoDa(?y, ?z) -> haRuolo(?z, ?x)
6
7 RuoloTitolareImpresa(?x) ^ impresaAvviataDa(?y, ?z) ->
8                               haRuolo(?z, ?x)

```

Passiamo ora alla regola SWRL che classifica un locale come Esercizio Alimentare Non Ottimo nel momento in cui questo abbia associato anche una sola recensione con una stella.

```

1 recensioneCreatadaAzione(?x, ?y) ^ recensireLocale(?y, ?z) ^
2 schema:reviewRating(?x, ?s) ^ possiedeStelle(?s, ?c) ^
3 swrlb:equal(?c, 1) -> EsercizioAlimentareNonOttimo(?z)

```

Mostriamo ora le 2 regole usate per classificare le pietanze in base al numero di ingredienti di cui necessitano. Classificheremo una pietanza come semplice nel caso in cui richiedesse al più 3 ingredienti, mentre sarà considerata complessa nel momento in cui necessitasse di almeno 10.

```

1 Pietanza(?x) ^ haNumeroIngredienti(?x, ?n) ^
2                               swrlb:lessThanOrEqual(?n, 3) -> PietanzaSemplice(?x)
3
4 Pietanza(?x) ^ haNumeroIngredienti(?x, ?n) ^
5                               swrlb:greaterThanOrEqual(?n, 10) -> PietanzaComplessa(?x)

```

Concludiamo con 2 regole che permettono di inferire il fatto che un locale sia situato in una determinata città a partire dal fatto che sia situato in una sua via, o in una sua piazza.

```
1 localeSituatoIn(?x, ?y) ^ Via(?y) ^ viaDellaCitta(?y, ?z) ->
2                               localeSituatoIn(?x, ?z)
3
4 localeSituatoIn(?x, ?y) ^ Piazza(?y) ^ piazzaDellaCitta(?y, ?z)
5                               -> localeSituatoIn(?x, ?z)
```

12 Allegati Full Size

Saranno presentate, in questa sezione finale della documentazione, alcune delle immagini ingrandite inserite nelle sezioni precedenti che a causa dello spazio limitato potevano risultare di difficile lettura.

