

MODELLAZIONE SEMANTICA

VOYAGER

RELAZIONE DEL PROGETTO DI MODELLAZIONE CONCETTUALE PER IL WEB SEMANTICO



MOTIVAZIONE

Il lavoro portato avanti in questo progetto ha come obiettivo quello di creare una piattaforma che guidi, gli utenti iscritti, nel processo iniziale di ideazione ed organizzazione del loro prossimo viaggio. Di offrire loro maggiori informazioni riguardo i possibili itinerari da percorrere, spiagge da visitare, laghi, montagne e fiumi presenti nelle nazioni riportate. La piattaforma volge uno sguardo principalmente alle Nazioni Europee che si affacciano sul Mar Mediterraneo, vengono indicate le mete più rinomate dei vari stati come le isole italiane di "Capri" e "Lipari" o la greca "Santorini", ma anche meno gettonate come le catene montuose del Marocco e le spiagge Albanesi. Organizzare una vacanza è diventato un

processo sempre più lungo e faticoso, dovuto alla sempre più ampia scelta che ci viene posta di fronte e che ci porta ad uno stato di disorientamento, senza riuscire a scegliere in modo deciso e voluto, o alle volte senza nemmeno avere in mente bene ciò che si vuole. Con il ridursi delle distanze, oggi è possibile fare colazione in un bar nel centro grigio e piovoso londinese e la sera godersi un tramonto su di una spiaggia bianca nell'Atlantico orientale, viaggiare quindi assume un ruolo molto importante nella vita, poiché permette di staccare e riprendersi dagli impegni che la routine ci presenta davanti.

Voyager mira a questo, descrivere e proporre le scelte più valide, trascurando la miriade di offerta presenti.

REQUIREMENTS

L'ontologia creata qui tratta prevalentemente il contesto dei viaggi. La finalità della rappresentazione ontologica di questo dominio ha come obiettivo quello di guidare il cliente nell'esplorazione e nella ricerca di possibili itinerari e mete in un'area circoscritta al Mar Mediterraneo.

La funzionalità verso cui è orientata l'ontologia è quella di poter interrogare il sistema alla ricerca di soluzioni che soddisfano le proprie necessità.

L'entità principale che fa da perno per l'intero sistema è **GeographicEntity**, questa indica l'entità geografica generica che poi nel dettaglio può rappresentare **Region** o **Destination**, la prima si specifica in particolari regioni geografiche, che nel progetto interessa solo quelle inerenti il bacino del Mediterraneo ma che nel futuro può essere ampliata e arricchita con dettagli e destinazioni di altri Continenti o aree geografiche, la peculiarità è che devono contenere più stati anche non confinanti tra di loro ma comunque identificabili sotto uno stesso nome, come nel caso degli stati caraibici. La seconda entità "Destination" invece elenca le possibili mete turistiche (mari, laghi, isole ecc.) che è possibile visitare nello stato di destinazione.

Oltre all'aspetto geografico si pone un focus anche su quello culinario, infatti l'entità **Food** che si dirama in diverse sottoclassi va a trattare specifiche specialità culinarie di diverse

aree geografiche come quella della penisola balcanica e iberica, nonché quella del Bel Paese, elencando alcuni piatti tramite la creazioni di individui.

Infine la classe **AgriculturalProducts** va a concentrarsi su quelli che sono i prodotti agricoli tipici locali, che possono essere prodotti fruttiferi di stagione o meno e di bioprodotti.

Gli utenti a cui la piattaforma è rivolta sono molteplici, dagli utenti finali che sono i clienti che possono consultare il sistema alla ricerca di destinazione, a semplici appassionati di viaggi fino ad un aspetto prettamente imprenditoriale e di marketing dove realtà statali ed enti pubblici potrebbero voler utilizzare tale piattaforma col fine di pubblicizzare proprie località e piatti tipici locali. Non essendoci limiti per quanto riguarda la scelta della regione da trattare il progetto risulta scalabile e applicabile a diversi casi d'uso.

DESCRIZIONE E DOCUMENTAZIONE DOMINO

Il “viaggio” è sempre stato un fenomeno legato all’essere umano in quanto animale sociale e mobile. Per millenni l’uomo ha esplorato terre sconosciute con l’intento di scoprire nuove rotte o nuovi popoli da dominare e annettere ai propri imperi, con il risultato di essere riusciti a fondere le culture, lingue e usanze globalmente.

“Oggi il viaggio è un fenomeno non solo economico, ma anche psicologico. Il viaggio, infatti, nelle sue fasi (partenza, percorso e arrivo) rende l’idea della ciclicità della vita e del suo dinamismo. Il viaggio è, quindi, un’esperienza interiore dell’individuo che richiama la circolarità della vita: la nascita, l’adolescenza, la fase adulta e la morte.” [\[1\]](#)

La piattaforma si propone come un catalogo da cui prendere intuizioni ed elaborare il proprio percorso e la propria idea. Non vi è una piattaforma esistente a cui si ispira.

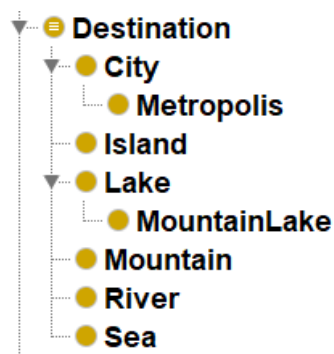
Nella rappresentazione del dominio è stata seguita la seguente struttura:

- la classe **GeographicEntity** indica in modo generico qualsiasi entità geografica localizzabile su una mappa
 - la classe **Destination** percorre un punto specifico della sua classe principale esplorando le destinazioni possibili, intese come meta finale del viaggio

- la classe **Region** invece raggruppa le varie regioni di interesse(Stati Mediterranei nel nostro caso)
- la classe **Food** si specifica sulla cucina delle varie aree trattate che a sua volta si articola in varie sottoclassi con possibili piatti e pietanze locali
- la classe **Country** invece fa il mapping delle nazioni con quelle presentate da Wikidata, con informazioni specifiche dei singoli stati come mari su cui confina ecc.

Voyager riassume le destinazioni disponibili in:

- ❖ CITY: le città che possono essere le capitali delle nazioni elencate o città turistiche secondarie ma molto conosciute.
- ❖ ISLAND: le isole principali da visitare, sono isole quasi tutte presenti nel Mediterraneo tranne l'isola di Tenerife che si trova nell'oceano Atlantico ma che fa sempre parte del territorio spagnolo, nazioni come l'Albania invece non hanno isole.
- ❖ LAKE: vi si trova un lago di ogni nazione, e anche laghi di montagna come il lago di Como.
- ❖ MOUNTAIN: le catene montuose si trovano tutte nel continente Europeo a parte il monte Afella che si trova nel continente Africano
- ❖ RIVER: tutti i fiumi che attraversano gli stati elencati nella piattaforma, tra i quali vi è anche il Danubio(secondo fiume più lungo d'Europa)
- ❖ SEA: dato che nel trattare questa ontologia si è deciso di incentrarsi nel bacino del Mediterraneo, i mari sono compresi tutti da quest'ultimo che però poi prende vari nomi a seconda della nazione su cui si affaccia



Queste sono le possibili destinazioni possibili che poi come è possibile vedere si articolano in Metropolis e MountainLake.

Annotations
Usage

Annotations: Metropolis

Annotations +

rdfs:label

Metropolis

rdfs:comment

Città metropolitane con una popolazione maggiore rispetto ad una città normale

Description: Metropolis

Equivalent To +

SubClass Of +

City

General class axioms +

SubClass Of (Anonymous Ancestor)

Instances +

Atene
Barcellona
Madrid
Milano
Roma

Target for Key +

Disjoint With +

Annotations
Usage

Annotations: MountainLake

Annotations +

rdfs:label

MountainLake

rdfs:comment

Laghi che si trovano situati in montagna

Description: MountainLake

Equivalent To +

SubClass Of +

Lake

General class axioms +

SubClass Of (Anonymous Ancestor)

Instances +

Como

Target for Key +

Disjoint With +

Disjoint Union Of +

Le Metropolis sono SubClass Of City e contengono le grandi capitali con più di tre milioni di abitanti, mentre MountainLake indica i laghi di montagna che si trovano in località montane, in questo caso solo il lago di Como nella catena montuosa delle Alpi. Entrambe le classi sono state inferite tramite l'utilizzo di SWIRL.

Un'altra classe molto rilevante dal punto di vista concettuale è State che contiene tutte le mete dell'ontologia asserita con la restrizione min in quanto vi sono nazioni nel mondo che contengono più di una capitale come nel caso del Sudafrica o di Israele, ma per i fini di questa ontologia questo dettaglio può essere trascurato in quanto le nazioni esposte contengono solo una capitale.

Description: State

Equivalent To

HasCapital min 1 City

?
@
x
o

Does some Food

?
@
x
o

Country

?
@
x
o

SubClass Of

MediterraneanState

?
@
x
o

General class axioms

SubClass Of (Anonymous Ancestor)

State

?
@
x
o

Instances

Albania

?
@
x

Croazia

?
@
x

Grecia

?
@
x

Italia

?
@
x

Marocco

?
@
x

Spagna

?
@
x

Target for Key

Disjoint With

Disjoint Union Of

Nella rappresentazione del dominio sono state usate diverse relazioni inverse come nel caso di Does e IsMadeIn in cui viene indicato che una nazione offre determinati cibi e che i cibi vengono preparati in una determinata nazione. Vi sono inoltre diverse relazioni transitive come nel caso di ContainsFood che va a coprire due classi diverse (BalkanCousine, GreekFood), e relazioni funzionali come IsCapitalOf.

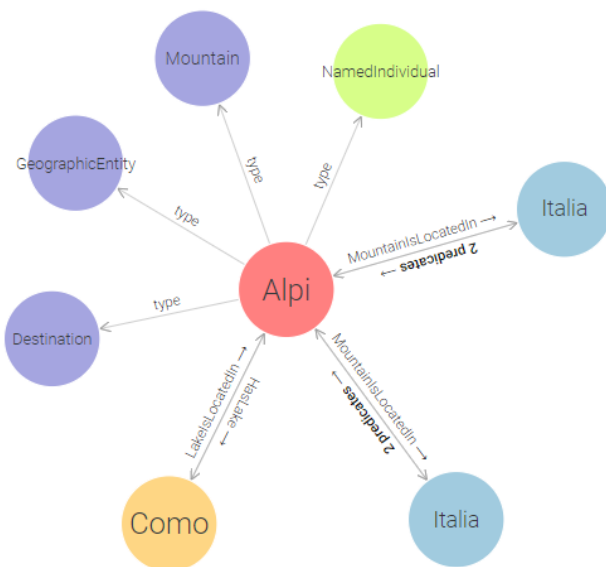
La property chain proposta è:

- HasDestination: che permette di dire “se uno stato contiene una montagna e quest’ultima contiene un lago allora possiamo inferire che lo stato abbia come destinazione il lago”

HasMountain o HasLake SubPropertyOf: HasDestination

Nella rappresentazione della A-box si va a rappresentare l’asserzione “Alpi contengono il lago di Como”.

The screenshot displays a Semantic Web editor interface. The top panel, titled "Direct instances: Alpi", shows a list of instances for the class "Mountain": Afella, Alpi (selected), Dajt, Dinara, Enos, and Teide. The bottom panel is divided into two sections. The left section, titled "Description: Alpi", shows the class "Mountain" under the "Types" tab, with options for "Same Individual As" and "Different Individuals". The right section, titled "Property assertions: Alpi", shows the "Object property assertions" tab with the assertion "HasLake Como" (represented by a blue square icon) and the "Data property assertions" tab with the assertion "Height 4809" (represented by a green square icon).



Qui è possibile vedere graficamente l'A-box.

Basandoci sull'esempio precedente si mostra il risultato tabellare associato all'entità Alpi:

	subject	predicate	object	context
1	:Alpi	voy:#HasLake	:Como	http://www.ontotext.com/explicit
2	:Alpi	voy:#Height	"4809" ^{xsd:integer}	http://www.ontotext.com/explicit
3	:Alpi	voy:#MountainsIsLocatedIn	:Italia	http://www.ontotext.com/implicit
4	:Alpi	voy:#MountainsIsLocatedIn	voy:#Italia	http://www.ontotext.com/implicit
5	:Alpi	rdf:type	voy:#Destination	http://www.ontotext.com/implicit
6	:Alpi	rdf:type	voy:#GeographicEntity	http://www.ontotext.com/implicit
7	:Alpi	rdf:type	voy:#Mountain	http://www.ontotext.com/explicit
8	:Alpi	rdf:type	owl:NamedIndividual	http://www.ontotext.com/explicit
9	:Alpi	rdfs:comment	"catena montuosa che si trova in Italia"	http://www.ontotext.com/explicit
10	:Alpi	rdfs:label	"Alpi"	http://www.ontotext.com/explicit

Allineamento con altre Ontologie

Per quanto riguarda la classe Food essa è stata allineata all'ontologia AGROVOC [\[2\]](#)

AGROVOC è un insieme di dati aperti collegati sull'agricoltura disponibile per l'uso pubblico e facilita l'accesso e la visibilità dei dati in tutti i domini e le lingue. Offre una raccolta strutturata di concetti agricoli, termini, definizioni e relazioni che vengono utilizzati per identificare inequivocabilmente le risorse, consentendo processi di indicizzazione standardizzati e rendendo le ricerche più efficienti. AGROVOC utilizza le tecnologie del web semantico, collegandosi ad altri sistemi di organizzazione della conoscenza multilingue e costruendo ponti tra i set di dati.

In questo caso abbiamo preso solo i prodotti agricoli strutturati come segue

- ❖ **Agricultural Products:** IRI http://aims.fao.org/aos/agrovoc/c_8678 questa classe contiene tutti i prodotti agricoli, prodotti nel loro stato originale o trasformati sufficientemente per essere preparati per la commercializzazione, Essa ha diverse sottoclassi di cui si è deciso di prendere solo quelle utili alla nostra ontologia
 - Bioproducts: IRI http://aims.fao.org/aos/agrovoc/c_8678 questa espone i prodotti BIO
 - Out Of Season Products: IRI http://aims.fao.org/aos/agrovoc/c_8678 che a sua volta contiene classi più specifiche suddivise in vegetali e frutta

La seconda ontologia utilizzata è Wikidata [\[3\]](#) che contiene classi interessanti che rappresentano le nazioni e che sono allineate alle nostre.

Di seguito si indicano solo due esempi di nazioni prese da Wikidata:

- ❖ **Italy:** ad esempio questa classe wikidata con codice Q38 che contiene informazioni già dichiarate come partof, life expectation, culture, continent e molte altre informazioni che possono essere allineate alle nazioni create nell'ontologia e che non necessitano di essere asserite nuovamente
- ❖ **Spain:** wikidata con codice Q29 che rappresenta lo stato spagnolo

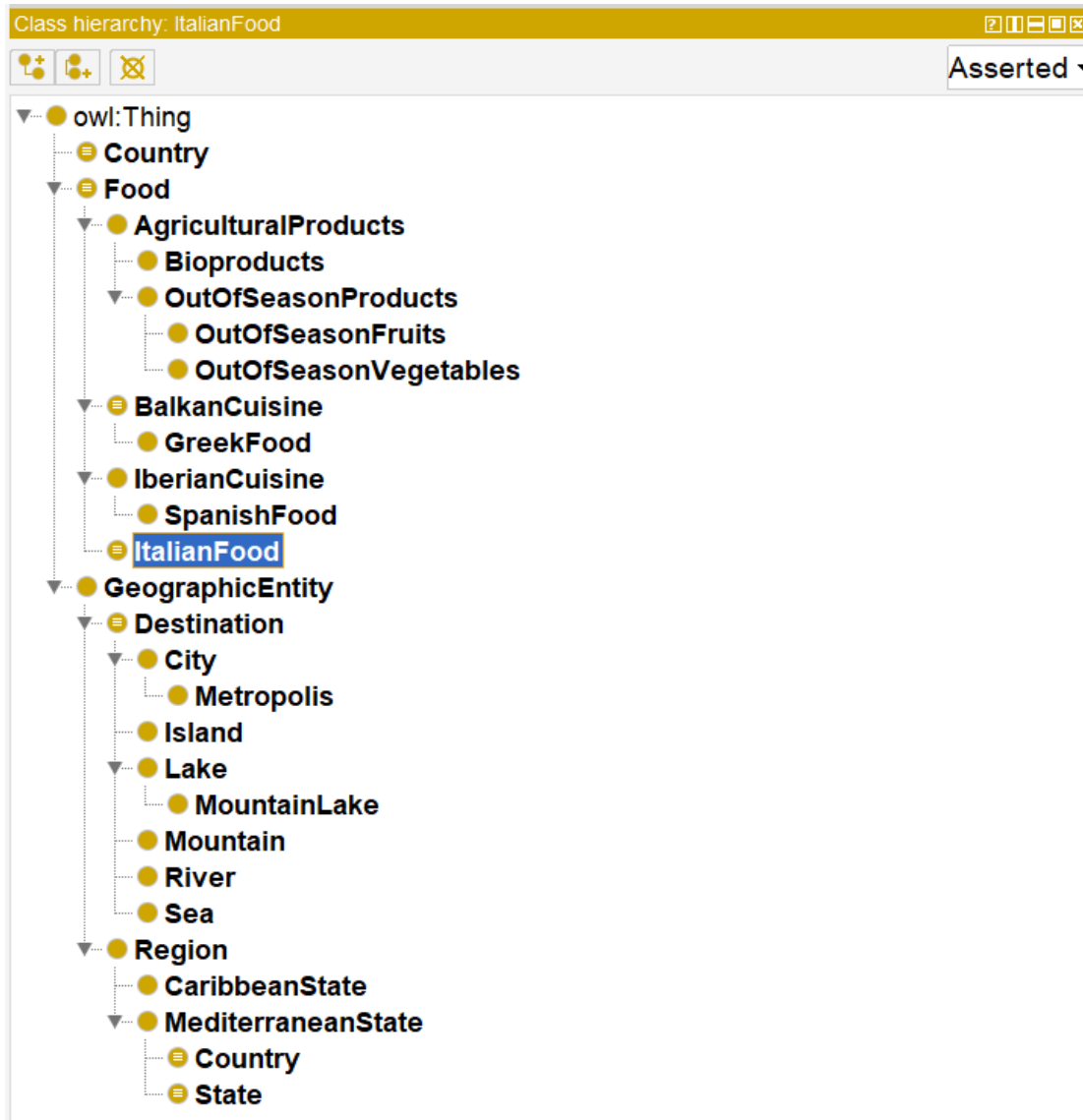
Pattern utilizzati

Al fine di modellare schemi ricorrenti nella modellazione ontologia è stato utilizzato il Pattern HasPart/IsPartOf [\[4\]](#) nella rappresentazione del lago Ohrid che si trova nel monte Dajt il quale poi viene inferito che è parte dell'Albania in quanto il monte che lo contiene si trova in Albania.

VISUALIZZAZIONE

Tassonomia delle classi

Di seguito viene presentata una panoramica delle classi principali utilizzate nella costruzione dell'ontologia [\[5\]](#)



Gerarchia delle classi

Di seguito possiamo vedere la gerarchia delle classi estratte tramite GraphDb dove possiamo notare che le classi più importanti sono GeographicEntity e Food

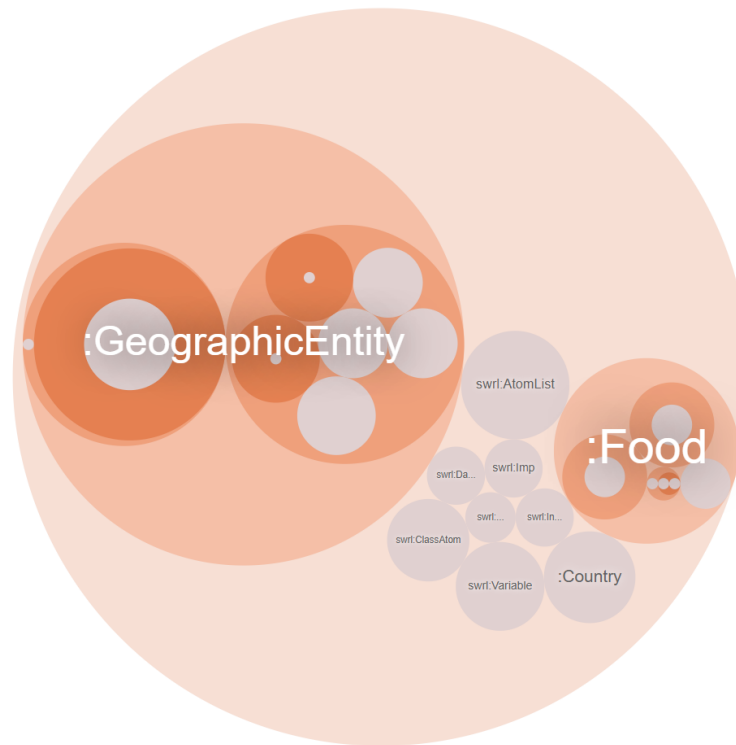
Class hierarchy ⓘ

All graphs ▾ 🔍 🔗 🔍 ↺ 🔍

Class Count ⓘ

33

1



Visualizzazione grafo

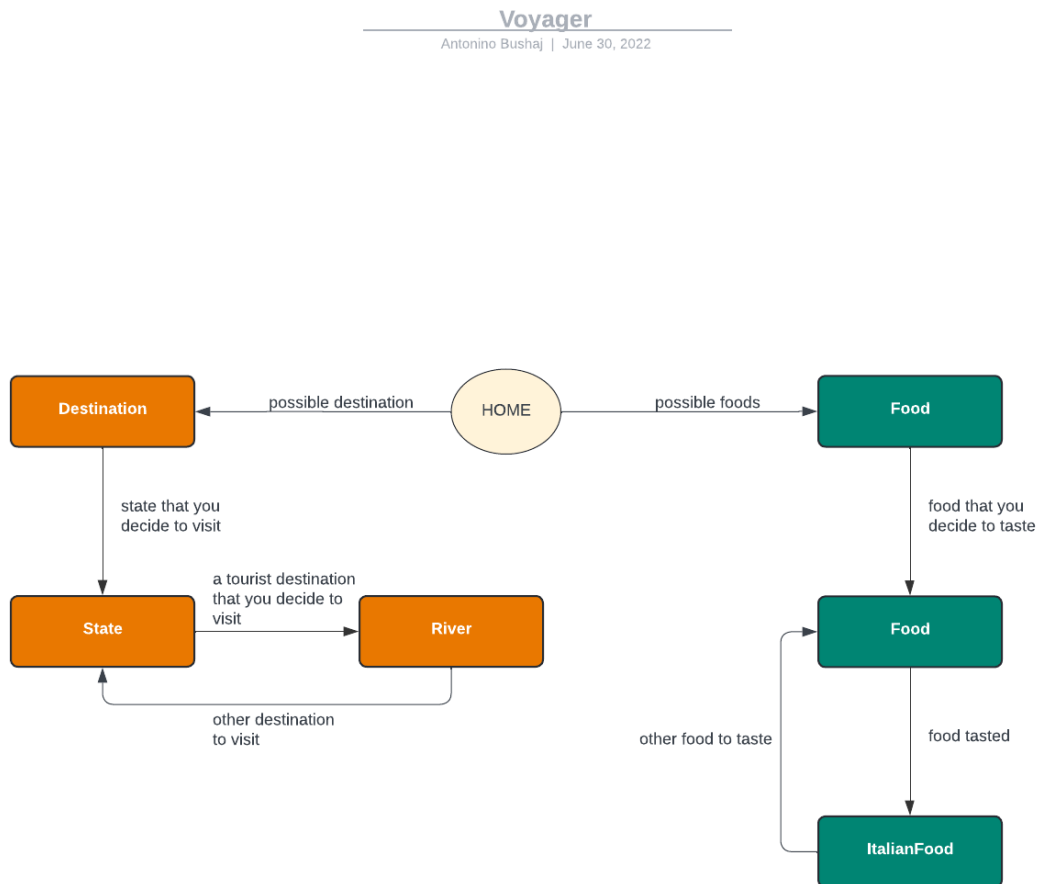
Il progetto è stato testato in vista di eventuali punti di criticità dell'ontologia tramite OntOlogy Pitfall Scanner (OOPS) [7].

Per la visualizzazione intera dell'ontologia si fa uso del seguente tool OWLGrEd che da una panoramica completa

http://owlgred.lumii.lv/online_visualization/vgn0

QUERIES SPARQL

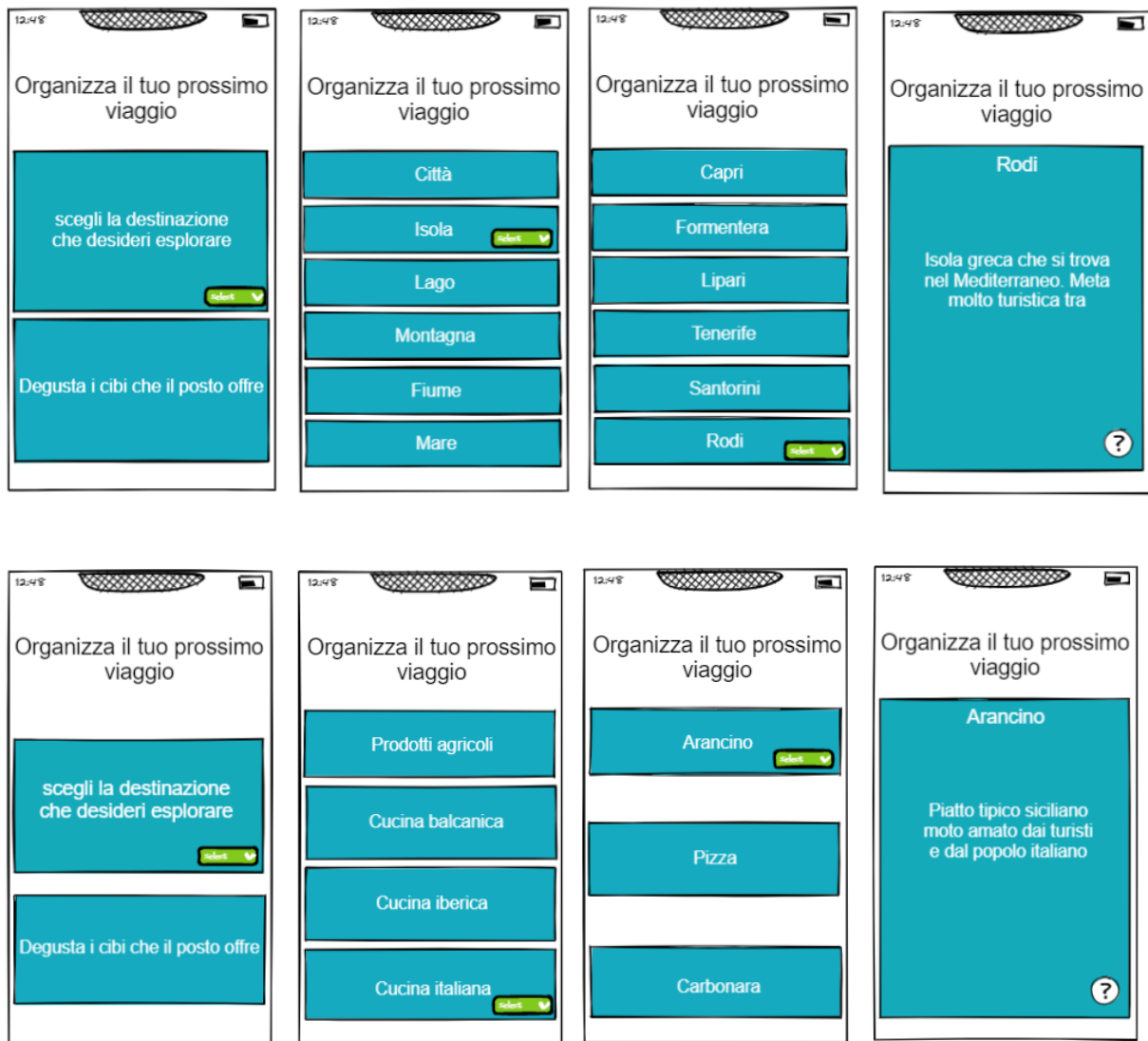
Flow chart dell'interazione



Un ipotetico utente potrebbe effettuare l'accesso alla piattaforma e volersi informare sulle destinazioni possibili. Nella pagina dedicata alle nazioni può scorrere tra tutte le scelte possibili e poi cercare la meta turistica che preferisce, effettuando anche scelte multiple.

L'utente ha la possibilità di esplorare anche i piatti che una destinazione offre e scorrere le proposte culinarie a seconda della regione precisa in cui viaggia, es(Balcanica, Iberica) e una volta trovato ciò che cerca può leggere informazioni riguardanti quel piatto.

Schema di interfaccia-interazione con dati reali



Come indicato sopra l'utente può accedere e scegliere di eseguire le operazioni che richiede seguendo il menù proposto.

Query

1. Query 1

Obiettivo:

Trovare tutti i cibi preparati in Spagna

Codice:

```
SELECT ?nation ?food ?taste WHERE {  
  ?nation :Does ?food;  
  rdfs:label ?label.  
  ?food :Taste ?taste  
  FILTER regex(?label, "Spagna")  
}
```

Risultato grafico:

	nation	food	taste
1	:Spagna	:Jamon	"salty"
2	:Spagna	:Paella	"salty"
3	:Spagna	voy:#Chorizo	"Spicy"
4	:Spagna	voy:#PatatasBravas	"Salty"

2. Query 2

Obiettivo:

Trovare le montagne più alte di ogni nazione

Codice:

```
SELECT ?nation (MAX(?mountain) AS ?HighestMountain)
```

```

WHERE {
    ?nation :HasMountain ?mountain
}

GROUP BY ?nation

```

Risultato grafico:

	nation	HighestMountain
1	:Albania	:Dajt
2	:Italia	:Alpi
3	:Grecia	:Enos
4	:Croazia	:Dinara
5	:Spagna	voy:#ElToro
6	:Marocco	:Afella
7	voy:#Italia	voy:#Appennini

3. Query 3

Obiettivo:

Trovare tutte le isole che fanno parte dell'Italia

Codice:

```

SELECT ?island
WHERE {
    :Italia :HasIsland ?island.
}

```

Risultato grafico:

	island
1	:Capri
2	:Lipari

4. Query 4

Obiettivo:

Mostrare tutti i cibi salati e la nazione che li produce

Codice:

```
SELECT ?food ?taste ?nation
WHERE {
    ?food :Taste ?taste;
        :IsMadeIn ?nation
    FILTER regex(?taste, "salty")
}
```

Risultato grafico:

	food	taste	nation
1	:Arancino	"salty"	:Italia
2	:Arancino	"salty"	voy:#Italia
3	:Carbonara	"salty"	:Italia
4	:Carbonara	"salty"	voy:#Italia
5	:Pizza	"salty"	:Italia
6	:Pizza	"salty"	voy:#Italia
7	:Feta	"salty"	:Grecia
8	:Jamon	"salty"	:Spagna
9	:Paella	"salty"	:Spagna

5. Query 5

Obiettivo:

Mostrare tutte le capitali europee con una popolazione maggiore di tre milioni di abitanti

Codice:

```
SELECT ?nation ?city ?population
WHERE {
    ?city :IsCapitalOf ?nation;
        :Population ?population.
    FILTER(?population > 3000000)
}
```

Risultato grafico:

	nation	city	population
1	:Grecia	:Atene	"5000000" ^{^^xsd:integer}
2	:Spagna	:Madrid	"5000000" ^{^^xsd:integer}

ESTENSIONE: BASE DI REGOLE SWRL

Sono state create regole SWRL utilizzando il plugin SWRL tab di Protege. Per l'interpretazione delle regole si è utilizzato il reasoner Pellet.

1. Se un lago si trova in montagna allora è un lago di montagna

voyager:Mountain(?m) ^ voyager:HasLake(?m, ?l) ^ voyager:Lake(?l) ->
voyager:MountainLake(?l)

2. Se lo stato contiene un'isola con molti turisti allora lo stato ha un'isola popolare

voyager:Island(?i) ^ voyager:HasIsland(?s, ?i) ^ voyager:State(?s) ^ swrlb:greaterThan(?t, 100000) ^ voyager:NumTourists(?i, ?t) -> voyager:HasPopularIsland(?s, ?i)

3. Se la città ha una popolazione superiore al milione di abitanti allora è una metropoli

voyager:City(?c) ^ swrlb:greaterThan(?p, 1000000) ^ voyager:Population(?c, ?p) ->
voyager:Metropolis(?c)

4. Se la nazione ha una montagna e la montagna ha un lago allora la nazione ha come destinazione il lago

voyager:Mountain(?m) ^ voyager:HasLake(?m, ?l) ^ voyager:HasMountain(?s, ?m) ^
voyager:State(?s) ^ voyager:Lake(?l) -> voyager:HasDestination(?s, ?l)

5. Se il numero di turisti della prima isola è più grande del numero di turisti della seconda isola allora inferiamo che Isola1 ha più turisti di Isola2

swrlb:greaterThan(?t1, ?t2) ^ voyager:Island(?i1) ^ voyager:NumTourists(?i1, ?t1) ^
voyager:NumTourists(?i2, ?t2) ^ voyager:Island(?i2) -> voyager:HasMoreTourist(?i1, ?i2)

CONCLUSIONE: POSSIBILI AMPLIAMENTI FUTURI

L'ontologia in questione si concentra solo su determinati aspetti del dominio, in questo caso le destinazioni possibili e le varie cucine che è possibile trovarvi.

Il progetto quindi ha una copertura limitata che può essere sicuramente ampliata andando a trattare anche aspetti come le tradizioni o le culture del luogo oltre alla possibilità di aggiungere percorsi specifici guidati in base alla destinazione scelta e categorizzarle a seconda della stagione o dell'età dei visitatori.

SITOGRAFIA E BIBLIOGRAFIA

- [1] <https://www.stateofmind.it/2019/09/viaggio-psicologia/>
- [2] <https://agrovoc.uniroma2.it/agrovoc/agrovoc/en/>
- [3] https://www.wikidata.org/wiki/Wikidata:WikiProject_Countries
- [4] http://ontologydesignpatterns.org/wiki/Main_Page
- [5] <http://vowl.visualdataweb.org/webvowl.html>
- [6] <https://oops.linkeddata.es/response.jsp#>
- [7] <http://oops.linkeddata.es/>