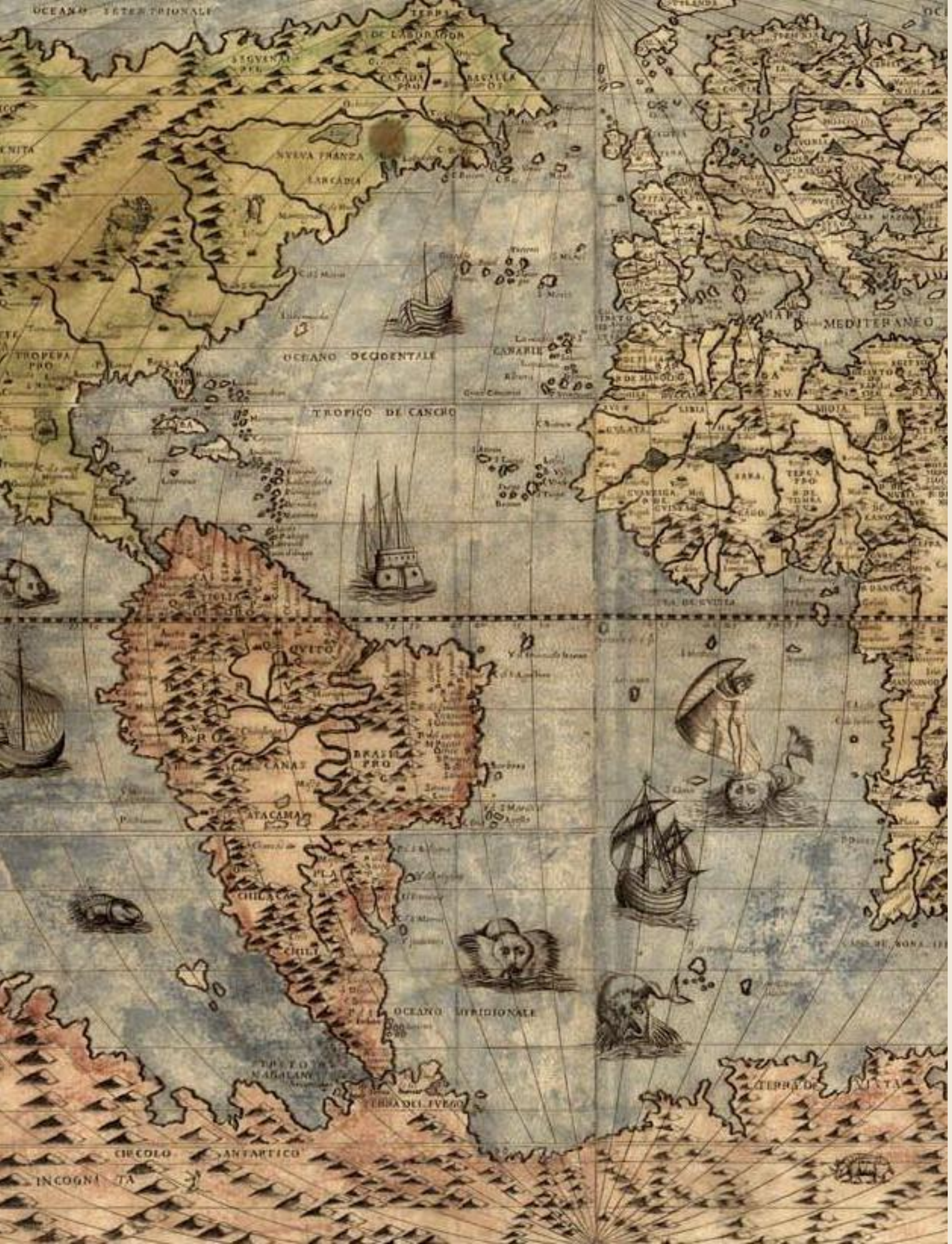


TripPlanner

A Web App to plan your journey

Open Data Management
and the Cloud

Final Project
Fall 2022
Erika Lena

https://github.com/erikalena/trip-planner

Overview

- Purpose statement
- Data discovery
- Data and Metadata model
- Interoperability
- Cloud solutions
- Tools
- Implementation challenges

Purpose statement

The idea was to provide a service which allows to automatically select a set of points of interest (POI) for a given place and find a clever way to visit them all in the few days we are going to spend there.

Problem delimitations:

- consider **Italian cities** only;
- **do not store all the data**, but find a way to access them;
- limit the **typology of POI** to: museums, archaeological sites, roads/squares and churches.

Data discovery

A research for **open data** provided by the European Commission was conducted, but no suitable datasets were found (https://data.europa.eu).

For this reason, instead of manually downloading and storing the data needed to run the application, different open data sources were used in order to methodically find what it is needed.

The proposed application deals with data about Italian cities POI by querying **Wikidata**, while the data it stores are those about users and their travel plans.

Data discovery

This choice has advantages and drawbacks.

The main advantages are:

- having a huge amount of data at our disposal;
- having data which are already **linked** and **open**;
- no need to deal with **data maintenance** for all the data.

Drawbacks are:

- we always need to be **connected to the internet** to investigate the data;
- **query formulation** is not so easy and straightforward.

Data Discovery

For the extraction of cities POI, resources from **Wikipedia** and **Wikidata [1]** were exploited as it will be explained.

DBpedia vs. Wikidata

Dbpedia [2] is a project aiming to extract structured content from the information created in the Wikipedia project. This structured information is made available on the World Wide Web.

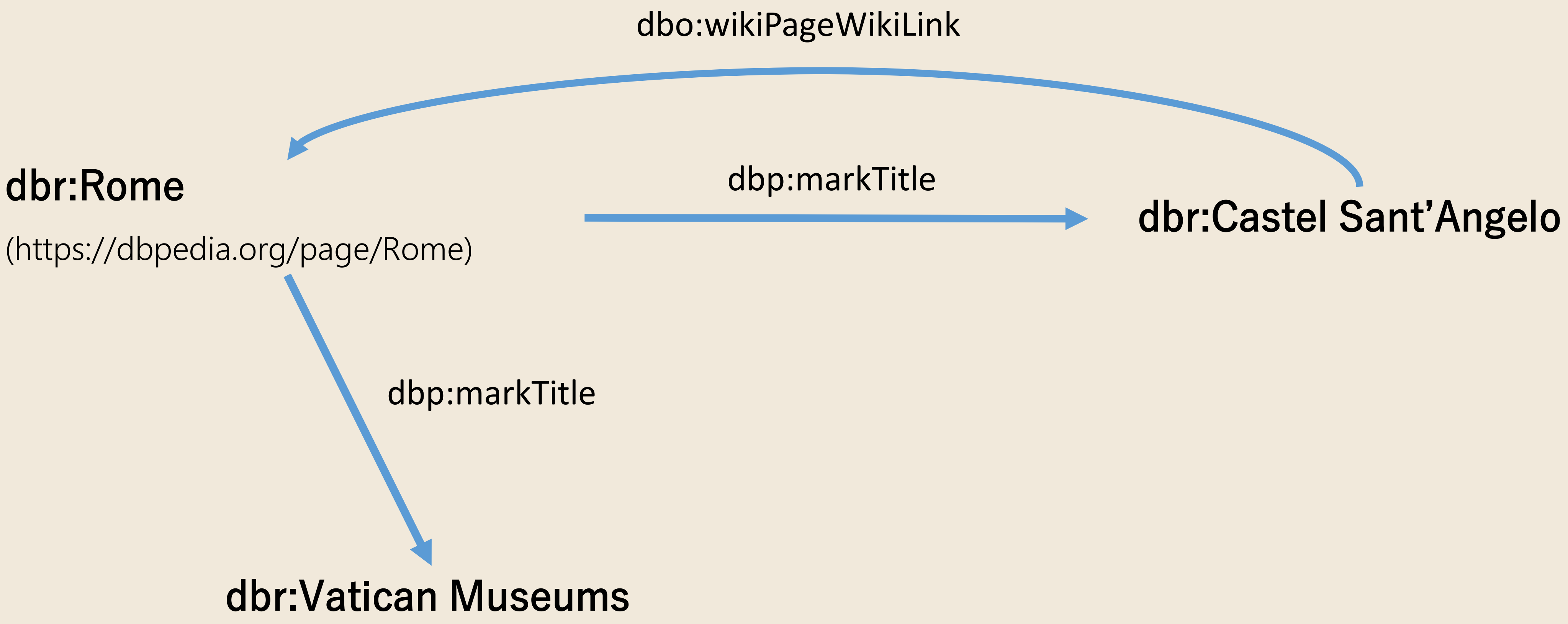
Wikidata is a free, collaborative, multilingual, **secondary database**, collecting structured data to provide support for Wikipedia, Wikimedia Commons, the other wikis of the Wikimedia movement, and to anyone in the world.

Data Discovery

To access data through **WikiData**, **SPARQL [3]** was used, which is a query language based on **Semantic Web standards** from the **World Wide Web Consortium**, W3C.

SPARQL queries run against **semantics-based data** or **ontologies** and it is used for querying data represented by **Resource Description Framework** (RDF) – a data model that describes information as triples of subjects, predicates and objects.

Data Discovery



dbr:Rome

(https://dbpedia.org/page/Rome)

Data Discovery

Example of SPARQL queries to find POI in a given city.

DBpedia

```
select ?poi ?type ?lat ?long ?img
where {
    VALUES ?city
{<http://dbpedia.org/resource/Bologna>}
    ?poi dbo:location ?city.
    optional
    {
        ?poi a ?type.
        ?poi geo:lat ?lat.
        ?poi geo:lat ?long.
        ?poi foaf:depiction ?img.
     }

}
```

Wikidata

```
select ?item ?itemLabel (sample(?coords) AS ?coords)
             (SAMPLE(?image) AS ?image)
where {
  ?item wdt:P131 wd:Q1891.
  optional { ?item wdt:P625 ?coords. }
  optional { ?item wdt:P18 ?image. }
  SERVICE wikibase:label
             { bd:serviceParam wikibase:language "it". }

}
group by ?item ?itemLabel
```

Data Discovery

The main differences are in:

- the **representation of entities**;
- the **ontologies used**, even if we can find correspondence between them.

In particular, the ontologies used by DBpedia are defined using the **Web Ontology Language** (OWL) and so they facilitate **semantic interoperability**.

While for Wikibase data model, this is possible, but requires a few more steps [4].

Data Discovery

We need to access and store the identifiers of Italian cities in order to query Wikidata databases:

```
SELECT ?city ?cityLabel WHERE {
    ?city wdt:P17 wd:Q38.  // country: Italia
    ?city wdt:P31 wd:Q515.       // instance of: city
    SERVICE wikibase:label { bd:serviceParam wikibase:language "it". }
}
```

Data Discovery

- The data we can find available on **Wikidata** satisfy all the requirements of deployment scheme given by Tim Berners-Lee for **Open Data**.
- Data are queried to be returned as a **JSON file**, which could also be saved in order to be used in the future.
- Each data item is queried along with its **Wikidata unique identifier**, which is made available to the user to find it easily and have the opportunity to further dig into data linked to that item.
- Wikidata entities are identified by strings of the type **Q[0-9]+**.

Data

An example of what is returned to our application.

```
{
    Results: {
        Bindings: [
            0: {
                coords: {datatype: 'http://www.opengis.net/ont/geosparql#wktLiteral', type: 'literal',
                    value: 'Point(11.343064 44.49373)'}
                image: {type: 'uri', value: 'http://commons.wikimedia.org/wiki/Special:FilePath/Bologna-vista02.jpg'}
                item: {type: 'uri', value: 'http://www.wikidata.org/entity/Q634917'}
                itemLabel: {xml:lang: 'it', type: 'literal', value: 'Piazza Maggiore'}
                label: {xml:lang: 'it', type: 'literal', value: 'piazza'}
                sitelink_count:  {datatype: 'http://www.w3.org/2001/XMLSchema#integer', type: 'literal',  value: '14‘}
            } …
          ]
      }
}
```

Data Model

This was about data discovery and access, the user can explore different cities' points of interest by simply querying Wikidata.

Once, the user starts to plan its trip, the application will **store the data of the user** along with the trip plans he/she desires to save.

These will be the data the application is fully in charge of.

Saving these for each user can also be useful if they want to visit the same city in the future, but see different places. In this case stored data will be compared to those returned by Wikidata.

Data Model

When the user decides to save data, data are stored accordingly to the physical data model presented in the following slide and managed through a **Relational Database System**.

For the moment just two types of resources are used, but others can be added and integrated; for this reason **metadata** could be stored as well, storing the data resource used, date and time as in standard log file.

This also simplifies **update operations** which, if needed, should involve mostly the description of cities obtained as **Wikipedia extract**.

Data Model

Trip Table

id	user	nDays	city
42	username	3	Q220

Visits Table

poiId	tripId	day
Q10285	42	1

POI Table

id	label	type	description
Q180212	Roman Forum	archaeological_site	«Il Foro Romano […]»

City Table

id	label
Q220	Rome

Data Model

For what concerns **semantic metadata**, those which provide relationships between entities in the **RDF schema**, one possibility is to use **Veritically Partioned Tables** (binary tables), where:

- each unique property create a two column table;
- each property is reconstructed through IDs for efficiency.

City

id	label
Q220	Rome

Instance of (P131)

id	value
Q220	Q515

Country (P17)

id	value
Q220	Q38

Interoperability

Since the proposed web application principal purpose is to dig into information available on Wikimedia and make them accessible as well as downloadable and usable, interoperability is needed.

- **Syntactic interoperability** is achieved by using standard data formats and standard communication protocols. In particular JSON and SQL are used, while the application is built as a RESTful Web Service.
- While **semantic interoperability** is provided by making the ontologies and vocabularies used by Wikidata interoperable and **compliant with OWL standards**. These information can be kept and maintained as metadata. Indeed, **RDF** is a standard model for data interchange on the Web, which ensures a way to provide semantic interoperability.

Cloud solutions

To build the web application, a traditional set of tools for web development was used: **html**, **CSS**, **Javascript**, **Vue** and **Node.js** for server-side.

Cloud solutions were used to simplify the deployment; in particular **AWS Elastic Beanstalk [5]** was used.

It orchestrates various AWS services like: Elastic Compute Cloud (EC2) and Simple Storage Service (S3).

Cloud Solutions

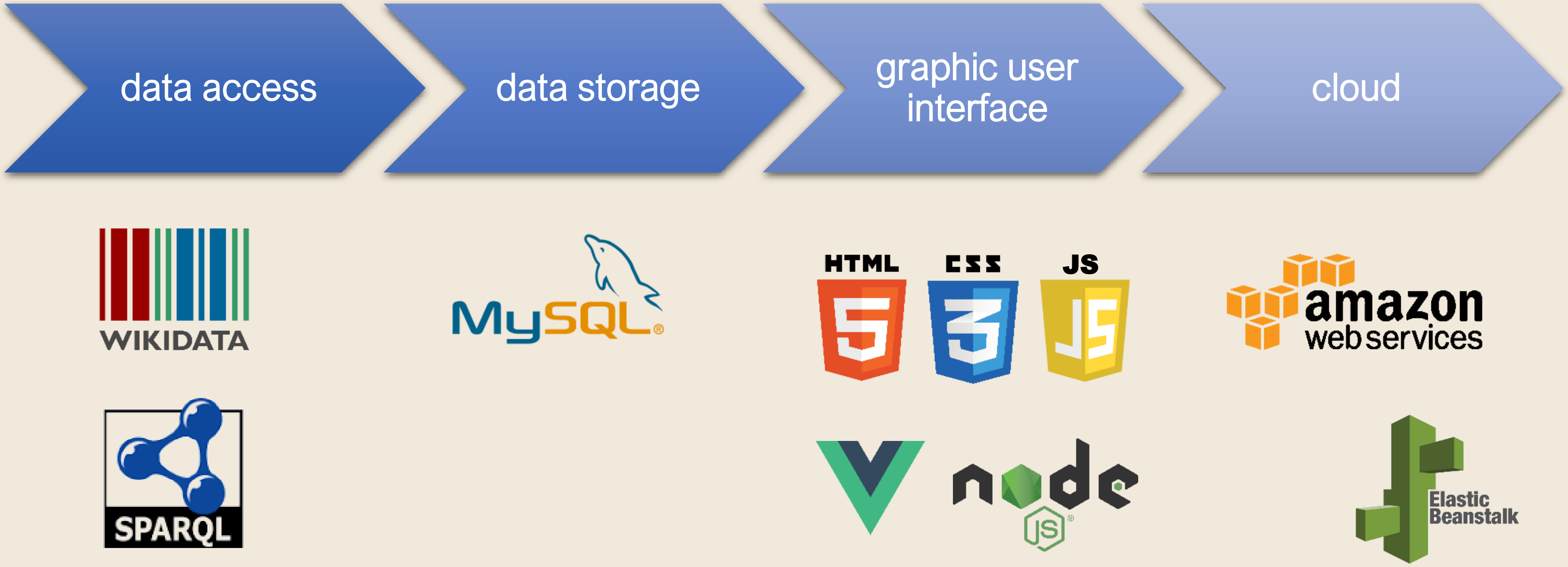
It provides a **web infrastructure management service**.

Essentially, it behaves as **Platform as a Service** (PaaS), providing the supporting architecture and computing resources required.

This result particularly helpful because it automatically takes care of **storage**, **scalability** and **underlying architecture** for us.

Elastic Bean gives you the possibility of choosing different levels of **security** for your files, so that you can block public access to your buckets.

Tools



data access

data storage

graphic user interface

cloud

Implementation challenges

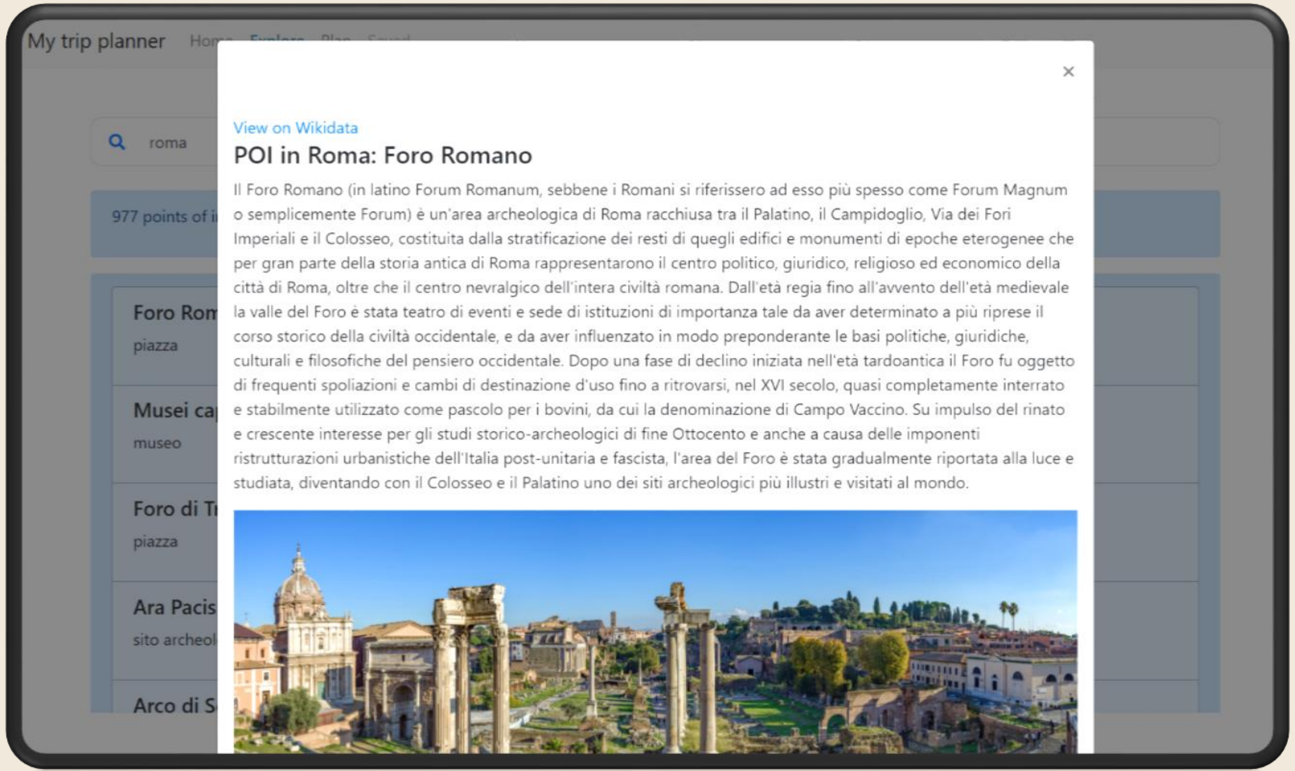
The main difficulty in developing the project was in **properly formulate** the SPARQL **queries** for Wikidata.

Obtaining results is also quite slow, so one thing could be to check if there is a way to improve **query efficiency**.

While, further developments to complete the application and exploit cloud solutions would be:

- use **clustering** (e.g. kmeans) to improve how POIs are split;
- **add** information also from other **resources**, like the time required to visit a specific POI;
- provide the possibility of exploiting previously saved trips.

Demo



View on Wikidata

POI in Roma: Foro Romano

Il Foro Romano (in latino Forum Romanum, sebbene i Romani si riferissero ad esso più spesso come Forum Magnum o semplicemente Forum) è un'area archeologica di Roma racchiusa tra il Palatino, il Campidoglio, Via dei Fori Imperiali e il Colosseo, costituita dalla stratificazione dei resti di quegli edifici e monumenti di epoche eterogenee che per gran parte della storia antica di Roma rappresentarono il centro politico, giuridico, religioso ed economico della città di Roma, oltre che il centro nevralgico dell'intera civiltà romana. Dall'età regia fino all'avvento dell'età medievale la valle del Foro è stata teatro di eventi e sede di istituzioni di importanza tale da aver determinato a più riprese il corso storico della civiltà occidentale, e da aver influenzato in modo preponderante le basi politiche, giuridiche, culturali e filosofiche del pensiero occidentale. Dopo una fase di declino iniziata nell'età tardoantica il Foro fu oggetto di frequenti spoliazioni e cambi di destinazione d'uso fino a ritrovarsi, nel XVI secolo, quasi completamente interrato e stabilmente utilizzato come pascolo per i bovini, da cui la denominazione di Campo Vaccino. Su impulso del rinato e crescente interesse per gli studi storico-archeologici di fine Ottocento e anche a causa delle imponenti ristrutturazioni urbanistiche dell'Italia post-unitaria e fascista, l'area del Foro è stata gradualmente riportata alla luce e studiata, diventando con il Colosseo e il Palatino uno dei siti archeologici più illustri e visitati al mondo.

Demo to show further details about data access and visualization.

https://github.com/erikalena/trip-planner

References

[1] https://www.wikidata.org/wiki/Wikidata:Main_Page

[2] https://www.dbpedia.org/

[3] https://www.w3.org/TR/rdf-sparql-query/

[4] Analysing and promoting ontology interoperability in Wikibase. D. Dobriy, A. Polleres, 2022.

[5] https://aws.amazon.com/it/elasticbeanstalk/