# Art Attack

André Flores
up201907001@fe.up.pt

Diogo Faria
up201907014@fe.up.pt

Tiago Rodrigues
up201906807@fe.up.pt

December 11, 2023

**Abstract**

*It is easy to get lost when entering the art world. Who are the artists? What are their works about? Where are the works exhibited? What artists are contemporary? Art Attack is a platform to search for artists and artworks, and discover their history and how they are related. For each item the user can find biographical information, exhibitions, and related items. The platform is based on linked open data avilable through SPARQL endpoints. The platform uses both general data, provided by DBPedia and Wikidata, and museum-specific data, currently only from the Getty Museum and the Smithsonian American Art Museum.*

## 1 Introduction

Being able to discover the history of a piece of art can be difficult. Who made it? In which museums was it in? Who's owned it? What was its path through its life? To learn this information, one needs to track an artwork's information through different museums and other knowledge sources, leading to a tedious and time-consuming process.

We want to make this information easily accessible in just one place so that people can learn more about both a piece of art and its artist.

Thus, Art Attack was born, a website that's able to join information from different museums and other knowledge sources in a single place, providing an intuitive and easy way for anyone interested in art to dive into this topic.

## 2 Requirements

In terms of requirements, we decided to split them between functional and non-functional ones. The functional requirements were:

- Users should be able to search for an artist and/or piece of art and go to an individual entity page.
- Users should be able to go from artist to piece of art and vice-versa via links on their respective pages.
- Users should see consistent data across the different data sources.
- Users should be able to see similar artists and be able to go to their respective pages on an artist page.
- Users should be able to see similar artworks and be able to go to their respective pages on the artwork page.

Aside from those, the non-functional requirements we had were:

- User-friendly web interface.
- High uptime.

## 3 Existing solutions

There exist platforms that can be considered similar to one of the aspects our website provides, for example, a museum website which allows a user to get information on its own about pieces of art and artists, though none have the complete information we provide.

DBPedia would probably come the closest in terms of similarity, though its unappealing interface, lack of more detailed data and of easily provided connections between different artists and artworks set it apart from our own solution.

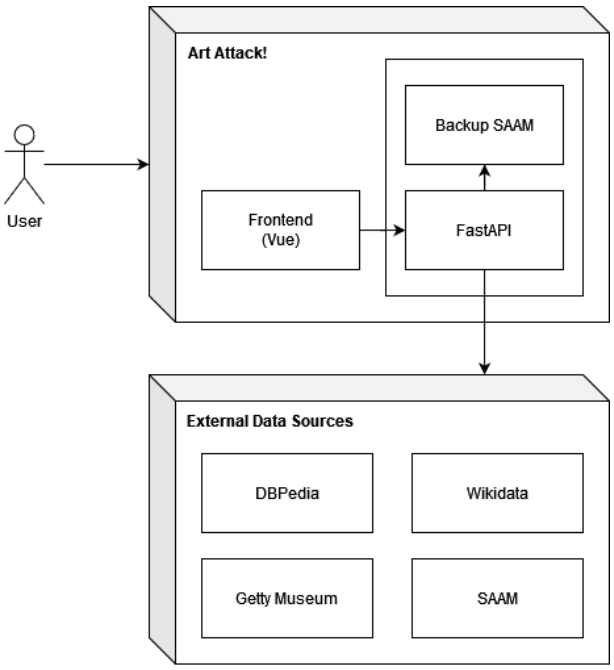


**Figure 1:** *Architecture of the solution*

# 4 Architecture of the solution

As described in fig. 1, a user can access our website, which we developed using Vue. Vue, in order to get its data, sends requests to our backend using FastAPI, and those requests are then translated into separate SPARQL queries, which are sent to the different SPARQL endpoints, one for each data source, and we compile the data and send it back to the frontend where it is presented to the user.

# 5 Development details

## 5.1 Knowledge Sources

We were able to find several knowledge sources for the information we wanted, more than we were expecting, even when limiting ourselves to data sources that conformed with W3C standards. In the end, due to non-functional endpoints, we ended with 4 different data sources: Getty Museum[1], Smithsonian American Art Museum (SAAM)[2], DBPedia[3], and Wikidata[4].

Getty and SAAM contain detailed information about artists, pieces of art, and exhibitions, though they focus on the latter two and are limited to entities that are or were present in the respective museums. On the other hand, DBPedia and Wikidata contain more general information about pieces of art and exhibitions while providing a more comprehensive range of information on artists, though still not always as in-depth as the museum ones.

## 5.2 Backend

Our backend was developed using Python with a connection to the frontend using FastAPI. The most important parts are the SPARQL queries to get our information.

For the search, we query 3 of the SPARQL endpoints: Getty, SAAM, and DBPedia, with 6 different SPARQL queries, with 2 for each endpoint, one for the artists, as shown in listing 1, and one for the artworks.

For artists, we join the results of the various queries. The results for DBPedia and SAAM are joined using the *owl:sameAs* predicate. To join DBPedia results with Getty Museum results, we use the *skos:exactMatch* predicate in the Getty Museum results, this gives us a URI with which we can query for another *skos:exactMatch* with the Wikidata URI. Since DBPedia has an *owl:sameAs* predicate to the Wikidata URI, we join the results on this value.

We could not find any definitive datapoint to join artworks between results from DBPedia, SAAM, and the Getty Museum. We join the results only if their names are an exact match.

We ended up not querying Wikidata in the search since it has a low timeout, which we usually set off, resulting in an error, though we still get information from them later on.

When a user enters an artist/artwork page, it's from one of the given results from a search, which already contains the URIs we found for each of the datasets, including Wikidata, as the URI is present in other data sources, and we query the 4 endpoints with separate queries. These queries get similar information, such as the name and date, but there are others which are specific to only one or a few data sources, such as the death manner for an artist from

```
# PREFIXES OMITTED
# %s = query_string.split(' ').join(.*)

SELECT DISTINCT ?uri (SAMPLE(?name) AS ?name)
↪ (SAMPLE(?image) AS ?image) ?dbpedia WHERE {
    ?uri rdf:type cidoc:E22_Man-Made_Object;
    ↪ cidoc:P102_has_title ?title.
    ?title rdfs:label ?name.
    OPTIONAL {
        ?uri owl:sameAs ?dbpedia.
        FILTER regex(str(?dbpedia),
        ↪ "^http://dbpedia.org/.*", "i")
    }
    OPTIONAL {
        ?uri cidoc:P138i_has_representation
        ↪ ?image.
    }
    FILTER regex(?uri, "^http.*")
    FILTER regex(?name, ".*%s.*", "i")
}
```

**Listing 1:** *Smithsonian American Art Museum SPARQL endpoint query to search for an artwork*

Wikidata, as can be seen in listing 2, the artist movement from both Wikidata and DBpedia, and the provenance of an artwork from Getty. We do these queries in a specific order so that certain information gets prioritized from one data source over another. For example, the date in Getty and SAAM usually refers to only the year, while DBpedia and Wikidata have the entire date.

```
# PREFIXES OMITTED
# %s = <artist_uri>

SELECT ?name ?birthDate ?birthPlace ?deathDate
↪ ?deathPlace ?deathManner ?movement WHERE {
    {
        SELECT (SAMPLE(?name) as ?name) WHERE {
            %s rdfs:label ?name.
            FILTER
            ↪ langMatches(lang(?name),'en').
        }
    }

    OPTIONAL {
        %s wdt:P569 ?birthDate.
    }

    OPTIONAL {
    {
        SELECT (SAMPLE(?birthPlace) as
        ↪ ?birthPlace) WHERE {
            %s wdt:P19 ?bPlace.
            ?bPlace rdfs:label ?birthPlace.
            FILTER
            ↪ langMatches(lang(?birthPlace),'en').
        }
    }
    }

    OPTIONAL {
    %s wdt:P570 ?deathDate.
    }

    OPTIONAL {
    {
        SELECT (SAMPLE(?deathPlace) as
        ↪ ?deathPlace) WHERE {
            %s wdt:P20 ?dPlace.
            ?dPlace rdfs:label ?deathPlace.
            FILTER
            ↪ langMatches(lang(?deathPlace),'en').
        }
    }
    }

    OPTIONAL {
        %s wdt:P1196 ?dManner.
        ?dManner rdfs:label ?deathManner.
        FILTER
        ↪ langMatches(lang(?deathManner),'en').
    }

    OPTIONAL {
    {
        SELECT (GROUP_CONCAT(?mov;separator=",")
        ↪ AS ?movement) WHERE {
            %s wdt:P135 ?movementPage.
            ?movementPage rdfs:label ?mov.
            FILTER langMatches(lang(?mov),'en').
        }
    }
    }
}
```

**Listing 2:** *Wikidata SPARQL endpoint query to search for an artist's information*

Aside from getting its own info, we then get extra information for both artists and artworks. For artists, we obtain their artworks, querying the data sources where the artist is present and getting all their artworks, and similar artists by movement, which we obtain both DBpedia. For artworks, we get similar artworks via their subject, obtained on DBPedia, and if they were a part of the same exhibitions on Getty. All this extra information is presented using cards which, when clicked, go to their own individual pages after queries to the endpoints.

### 5.3 Frontend

Our frontend was developed using Vue with Typescript, and we have three pages: the home page, where a user can search for anything; the artist page, where we present the information of an artist; and the artwork page, for the artwork information.

In terms of information for both the artist and artwork, while we display descriptions for them and are able to obtain them from different data sources, they are seen in a specific order, from the ones we find the most relevant to the least. We did this since most users would only look at the first description, and, in this specific case, the information we get from SAAM is much better when compared to the others.

## 6 Evaluation of the result

Our data is fully available on the web with an open licence, using open standards from the W3C, both RDF and SPARQL for all the knowledge sources, as they all have open SPARQL endpoints and even UI's to make it easier.

Aside from that, DBPedia has links to other sources, including Wikidata; With Getty, we can also find a link to Wikidata for further information; SAAM's data contains links to DBPedia and, rarely, to Wikidata too.

With this, due to our data being available with an open licence, using standards from the W3C, and containing outgoing links to other data, we believe it ranks 5 stars according to the Linked Open Data principles.

## 7 Conclusões

## References

[1] "Getty Museum". https://www.getty.edu/museum/. (accessed Dec. 7, 2023).

[2] "SAAM". https://americanart.si.edu. (accessed Dec. 7, 2023).

[3] "DBpedia". https://www.dbpedia.org. (accessed Dec. 7, 2023).

[4] "Wikidata". https://www.wikidata.org/wiki/Wikidata:Main_Page. (accessed Dec. 7, 2023).