



D4.6 Pilot III: First Report

Author(s): Pierre Baviera,
Swaroop Bhat, Katharine Cooney,
Mario Romera,

Date: 6/1/2021

H2020-ICT-29b

Grant Agreement No. 825182

Prêt-à-LLOD - Ready-to-use Multilingual
Linked Language Data for Knowledge
Services across Sectors

D4.6 Pilot III: First Report

Deliverable Number: D27

Dissemination Level: CO

Delivery Date:

Version: 1

Author(s): Pierre Baviera, Swaroop Bhat, Katharine Cooney, Mario Romera

Document History

Version Date	Changes	Authors
12/8/2020	Creation	Derilinx
11/12/2020	Updated	Derilinx
14/12/2020	Updated	Derilinx
21/12/2020	Internal Review	DFKI
6/1/2021	Incorporated changes	Derilinx



Executive Summary	4
1 Pilot Overview	6
1.1 Introduction and Goals	6
1.2 Pilot Schedule	7
1.3 Functional Requirements	8
1.4 Non-Functional Requirements	8
2 Pilot III.A – Chatbot providing access to Government and Health information	9
2.1 Motivation - Proficiency in English	9
2.2.1 Use Cases	9
2.2.2 Data Sources	10
2.2.3 Current languages of interest	10
2.2.4 Application Benchmarks	10
2.3 Technical Approach	11
2.3.1 Phase I	11
2.3.2 Phase II (Application Version V1.0):	13
2.3.3 Phase III	19
2.4 Application Benchmarks	23
2.4.1 KPI Contributions	24
2.5 Next steps	24
3 Pilot III.B Open Data Dashboard	25
3.1 Motivation	25
3.2.1 Use cases	25
Use Case 1	25
Use Case 2	25
Use Case 3	26
3.2.2 Data Sources	26
3.2.3 Current languages of interest	26
3.3 Technical Approach	26
3.4 Application Benchmarks	30
3.4.1 KPI Contributions	37
3.5 Next Steps	37
4 Summary and Next Steps	38



Executive Summary

The goal of Pilot III, run by Derilinx, is to improve access to government and health services by means of a) a chatbot and b) a dashboard. These will demonstrate the multi-lingual and cross-border facility for such applications.

There is huge potential in the application of Prêt-à-LLOD innovative approaches to language technologies within Derilinx' existing commercial service offerings, bringing ground-breaking advancements in language technologies to the wider sphere of the government and NGO sectors.

With this pilot, "Supporting the Development of Public Services in Open Government both within and across borders", Derilinx aims to provide tools and interfaces for more intuitive access to open data portals and public services using natural language.

In particular, the goal of this pilot is to explore and use state-of-the-art methods and techniques in computational semantics, data mining and machine learning for linking language data at the level of meaning. This pilot will free up, and connect, data that often exists in disparate silos, thus leading to a significant overhead in managing, enriching and reusing this data. In this respect, the work carried out here contributes directly to the 4th Challenge of Prêt-à-LLOD, which is "*Linking conceptual and lexical data for language services.*"

The tool developed by Derilinx as part of the Pilot is called Kweery.

Kweery is centred on the following key functionalities:

1. Handle queries/build responses in a selection of European languages
2. Handle queries regarding multiple European destinations
3. Retrieve data from the European Data Portal and other open data portals within Europe
4. Be based on an API so that it can be easily deployed to any platform and can incorporate Prêt-à-LLOD tools as they become available

These were delivered via the integrated sub-pilots:

1. Answer multilingual queries on Government and Health Services via Chatbot
2. Explore Open Data via a dashboard



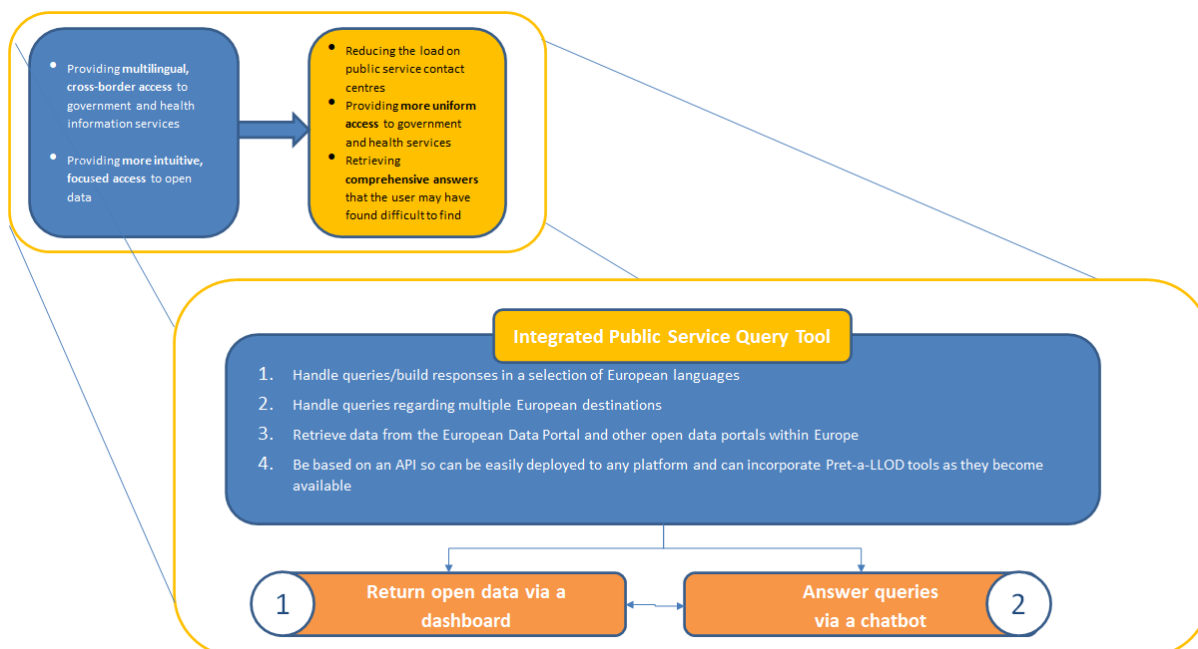


Figure 1. Integrated Public Service Query Tool

Further to thorough testing of the integrated pilot, Kweery has reached such level of maturity that Derilinx is now in a position to run demonstration sessions, building on the strong interest in the solution from existing Derilinx customers and other potential customers. Please visit <https://pret-a-llod.github.io/posts/derilinx-webinar-recording.html> to access the recording of an example webinar.

Key next steps linked to Kweery will be to effectively refine and optimise the performance of the tool through real customers install and configuration (including new data sources and languages).

The final version of this report, due month 36 (December 2021), will report on further development activities and the final results of the project.

1 Pilot Overview

1.1 Introduction and Goals

Derilinx provides services to government for high-quality data publishing and is a leading Linked & Open Data company, driving decision-making and providing insights in the public sector. Derilinx' clients include Government Bodies, Local Authorities and NGOs. Being able to discover, access and use data is key to ensure the effective delivery of public services, to foster trust in the organisations through transparency and to stimulate innovation, both internally and in the wider economy. Derilinx supports its clients in this mission through the publication of high-quality and reusable data.

There is huge potential in the application of Prêt-à-LLOD innovative approaches to language technologies within our existing commercial service offerings, bringing ground-breaking advancements in language technologies to the wider sphere of the government and NGO sectors.

With this pilot, "Supporting the Development of Public Services in Open Government both within and across borders", Derilinx aims to provide tools and interfaces for more intuitive access to open data portals and public services using natural language.

In particular, the goal of this pilot is to explore and use state-of-the-art methods and techniques in computational semantics, data mining and machine learning for linking language data at the level of meaning. It will incorporate LLOD/NLP technologies into our data sharing platform "datAdore"¹, and into Derilinx' wider knowledge management and data-access services. This pilot will free up, and connect, data that often exists in disparate silos, thus leading to a significant overhead in managing, enriching and reusing this data. In this respect, the work carried out here contributes directly to Challenge 4 of Prêt-à-LLOD, concerning the linking of conceptual and lexical data for language services.

The Pilot specifications distinguish between two parts (or sub-pilots):

Part A Improving the quality of responses and the multilingual capability of government and health information services.

Access to healthcare and government services can be complicated, even more so for the users trying to access information through a language that is not their own. To facilitate easier access to this information, AI/NLP systems can be used to complement other communication channels. Conversational User Interfaces (CUI) such as chatbots can provide more uniform (and multilingual) access to health information and government services and have the potential to optimize citizen interactions. Derilinx are using the pilot to develop a chatbot answering natural language queries regarding public services information that may be another language. This will provide cross-border and multilingual access to this information.

Part B Improving access to Open Data by facilitating natural language queries and providing multilingual access

Open data portals provide access to data, which is provided in open formats such as CSV, comma delimited files. Although this data is easily machine-readable, it can be more difficult for non-technical users to access, necessitating several steps to find the relevant piece of information inside a file. The open data query tool will allow the user to pose a natural language question in a user interface. The

¹ <https://derilinx.com/datadore/>



tool will then help identify the relevant file(s), query the contents of these files and return an answer to the question via a dashboard. This will eventually incorporate the ability to query data that is provided in a different language and return the results in the user’s own language, thus providing cross-border, multilingual access.

The two sub-pilots, consisting of the implementation of a chatbot and an open data dashboard aim to (1) answer natural language queries regarding public services information via a chatbot and (2) return open data associated with this query, and provide further search functionality via a dashboard. Both the public service information and, eventually, the open data will be returned in the language of the query, providing a web interface for users to access cross-border government services and open data in their native language.

These pilot tools have been integrated to provide a single interface which we have called “Kweery”.

1.2 Pilot Schedule

It was initially planned to have two separate applications for the two sub-pilots. These have subsequently been merged into a single application and so the development milestones have been changed to reflect this.

Milestone ID	Goal	Due date (Project Month)	Updated Due date (Project Month)
MS-III.1a	High level design agreed	M8	M8
MS-III.2a	Development of test cases for internal (DLX) testing	M10	M10
MS-III.3a	Alpha version of application ready for testing	M12	M12
MS-III.4a	Beta version of application ready for testing	M18	M24
MS-III.5a	Application completes internal testing	M20	M26
MS-III.6a	Pilot user satisfaction survey developed	M21	M26
MS-III.7a	User testing complete	M22	M28
MS-III.8a	Application fully implemented	M24	M30
Milestone ID	Goal	Due date (Project Month)	Updated Due date (Project Month)
MS-III.1b	High level design agreed	M22	M14
MS-III.2b	Development of test cases for internal (DLX) testing	M22	M14
MS-III.3b	Alpha version of application ready for testing	M26	M18



MS-III.4b	Beta version of application ready for testing	M30	M24
MS-III.5b	Application completes internal testing	M32	M26
MS-III.6b	User testing complete	M34	M28
MS-III.7b	Application fully implemented	M36	M30

Figure 2: Pilot milestones

1.3 Functional Requirements

Pilot III.A (Kweery chatbot) will focus in particular on:

- Interpretating and disambiguating natural language questions
- Providing a chatbot in multiple languages

Pilot III.B (Kweery dashboard) will focus in particular on:

- Interpretation and disambiguation of natural language queries
- Development of an API that enables interaction between the Kweery Chatbot and the Kweery Dashboard
- An API which will work in conjunction with the European data portal and other Open Data portals within Europe
- Extension of datAdore based on the API, to allow the user to identify data relevant to their search through selected (European) languages

The current version addresses points 1,2 and 3 above. Future developments will address point 4.

1.4 Non-Functional Requirements

The following are addressed in order to fulfil the proposed functional requirements:

- The Kweery chatbot is extensible (uses APIs) to facilitate the incorporation of Prêt-à-LLOD functionalities as they become available and evolve
- The extension of datAdore (Kweery dashboard) should allow the incorporation of appropriate Prêt-à-LLOD workflows as they become available and evolve
- The chatbot must comply with GDPR and ethical requirements – the chatbot does not store any user information
- The available license agreements should either allow integration of Prêt-à-LLOD components into the chatbot without requiring the chatbot code to be made open source (e.g., Apache or MIT license) or a commercial license needs to be available.

2 Pilot III.A – Chatbot providing access to Government and Health information

2.1 Motivation - Proficiency in English

The latest Irish Census (2016) includes in a summary of the top 10 non-Irish nationalities, their own assessment of their ability in English². Over 58,000 people assess their own ability as “not well, not at all or haven’t stated”. This is out of 535,475 non-Irish nationals usually resident in Ireland, a significant proportion of whom (103,113) are from the UK and so can be assumed to have ability in English.

Looking at the top 10 non-Irish nationalities alone, this suggests that of non-Irish nationalities, excluding those from the UK, 268,862, there could be some 22% who are significantly disadvantaged by their ability in English and so have limited access to services.

Nationality	Living in Ireland for 2016 Census
Polish	122,515 persons
UK	103,113 persons
Lithuanian	36,552 persons
Romanian	29,186 persons
Latvian	19,933 persons
Brazilian	13,640 persons
Spanish	12,112 persons
Italian	11,732 persons
French	11,661 persons
German	11,531 persons

Figure 3: Top 10 non-Irish nationalities from 2016 Irish Census

Similar analysis for England and Wales³ shows a significant proportion (864,000) who say they speak English “not well” or “not well at all”.

As demonstrated above, there is significant potential for providing cross-border multilingual access to health information and government services, particularly for some of the less well-served languages. In order to identify the languages which would have maximum impact, local census data can be used.

Covid-19 has highlighted the importance of reaching the largest possible audience by providing health and government information in multiple languages.

2.2.1 Use Cases

² <https://www.cso.ie/en/releasesandpublications/ep/p-cpnin/cpnin/introduction/>

³

<https://www.ons.gov.uk/peoplepopulationandcommunity/culturalidentity/language/articles/detailedanalysisenglishlanguageproficiencyinenglandandwales/2013-08-30>

Use cases that we address as part of this pilot are:

Actor	Goal	Extended Goal
General user	Understand how to access the health service in Ireland	I can access and use the health service, for example, how to access hospital services
General user without the language of the resource	Receive information in my own language on the public services available	I can access and use that public service, for example, understand what Social Welfare benefits, schemes and allowances are available to me.

Figure 4: Use cases for Kweery chatbot

2.2.2 Data Sources

- Irish citizens' information website⁴
- Spanish travel information website⁵

The Kweery chatbot will cover all topics covered by the Irish "Citizens' Information Website". Examples include:

- COVID-19
- Health
- Social Welfare
- Travel and Recreation

It will also cover all topics covered by the Spanish "Information for travelling to Spain" website. This includes:

- Visa and passport
- Security
- Health

2.2.3 Current languages of interest

- Spanish (ES)
- English (EN)

2.2.4 Application Benchmarks

- Quality of response:
 - Analysis of content of application responses against the search provided on the Citizens' Information website (this uses Google Search).
 - Validation of responses in Spanish by a native Spanish speaker
- Planned Beta Testing: User testing with trial users incorporating:
 - satisfaction of trial users with chatbot responses

⁴ <https://www.citizensinformation.ie/en/>

⁵ <https://www.spain.info/en/info/>



2.3 Technical Approach

Kweery Chatbot is implemented using a number of Artificial Intelligence technologies, including Natural Language Processing, Machine Learning Text Classification, Data Modelling and Information Retrieval.

- Natural language processing: NLTK⁶ (Natural Language Toolkit).
- Machine learning text classification: scikit-learn⁷ (Space Vector Model).
- Data modelling: Comma Delimited file as Knowledge base.
- Information retrieval: TF-IDF⁸ (Term frequency-inverse document frequency).

The application stack has been developed over 3 phases. Each phase has been refined from the drawbacks/limitations identified in the previous phase.

2.3.1 Phase I

In this phase we developed a basic bot which worked on the Information retrieval logic – term frequency-inverse document frequency (TF-IDF). All the application logic was developed as one single application.

Below figure 5 shows the application architecture diagram of Phase 1

⁶ <https://www.nltk.org/>

⁷ <https://scikit-learn.org/stable/index.html>

⁸ https://link.springer.com/referenceworkentry/10.1007/978-1-4899-7687-1_832



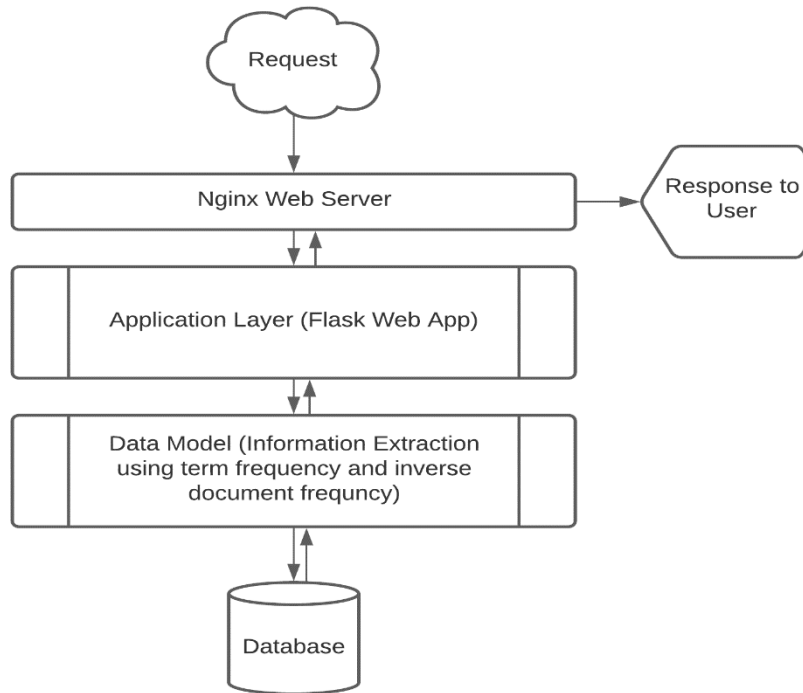


Figure 5. Kweery chatbot architecture (Phase 1)

Techniques used:

The data was pre-processed using a data pipeline, which takes the raw text and uses standard text cleaning methods, and was then stored in a database. The user-entered query goes through the same data pipeline before the text is analysed.

Information retrieval (TF-IDF) was used to search for relevant text and respond with relevant text. This approach did not show great results since we had a similar text in our knowledge base and we were not capturing the semantics of the user query. Also, calculating TF-IDF score of the entire text in memory resulted in a slow response and also limited the scalability of the application.

The application acted as search tool rather than an interactive chatbot. At this stage, the scalability of the application was also limited and we harvested data related to only one domain. These limitations resulted in poor search/response.

Limitations of Phase I:

- Covered data related to only one domain.
- Not scalable for different domains.
- Multi user not supported.
- Quality of response was poor.
- Not conversational.

- No multilingual functionality.

To address these limitations we planned Phase II, a scalable application with increased data volume.

2.3.2 Phase II (Application Version V1.0):

In this phase we aimed to address most of the limitations of phase I and add a suggestion/recommendation engine, semantic text analysis and text to linked data conversion.

In addition, we added the capability of answering questions in different languages. Currently, the application supports two languages, English (EN) and Spanish (ES).

Below figure 6 represents high level architecture diagram of the application on phase II and how it evolved.

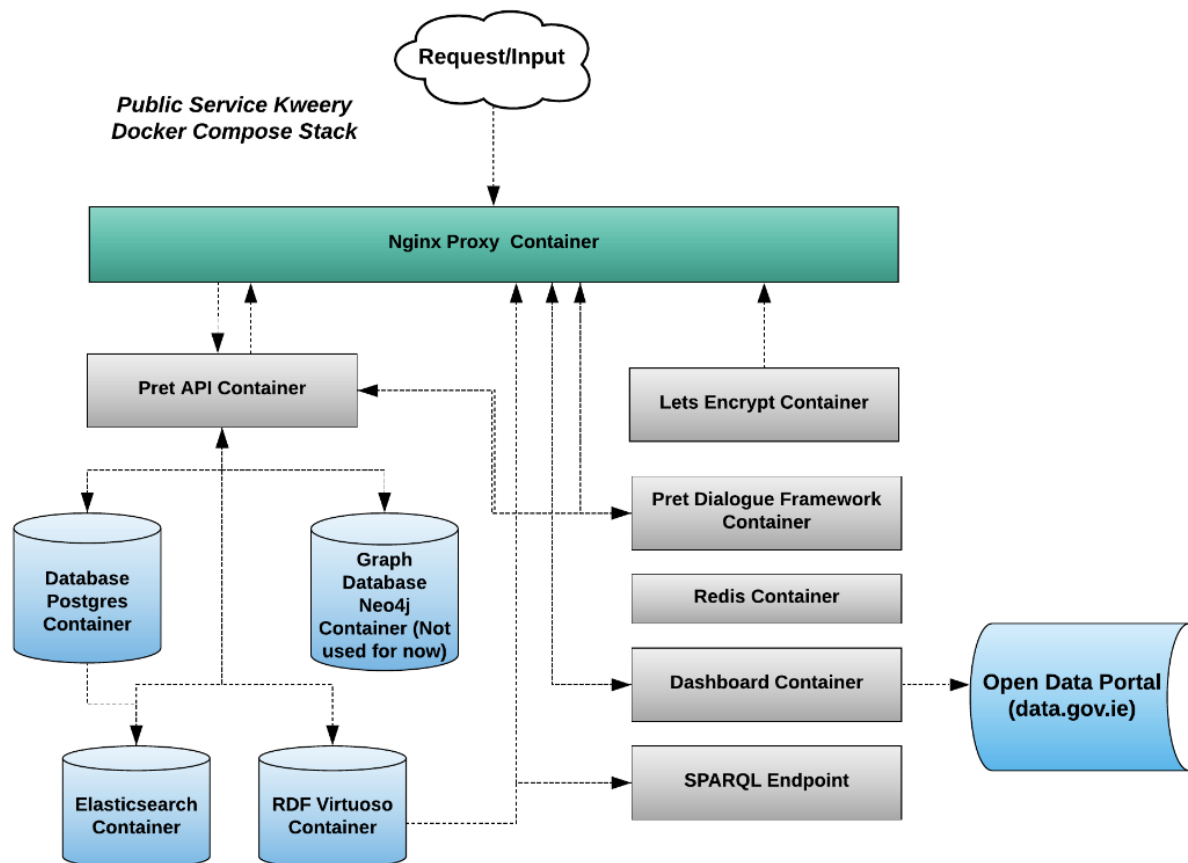


Figure 6. Kweery chatbot architecture (phase II)

The application is divided into two major components:

- Application programming Interface (API)

- Interactive web app (Dialog framework).

Component 1 - Kweery Application Programming Interface (API)

The APIs are designed as extensions, which allows us to plugin/remove/update any piece of functionality. In this phase we developed three major extensions - core plugin, NLP plugin and search plugin.

When the data is harvested to the application, it goes through the NLP extension which handles NLP tasks, and a data cleaning pipeline. In this extension we extract semantic meaning from the text and prepare the text for indexing. Information such as named entities, synonyms, hyponyms, hypernyms and text relationships is extracted in this stage.

Below figure 7 represents Kweery API software components and how each component is connected to each other.

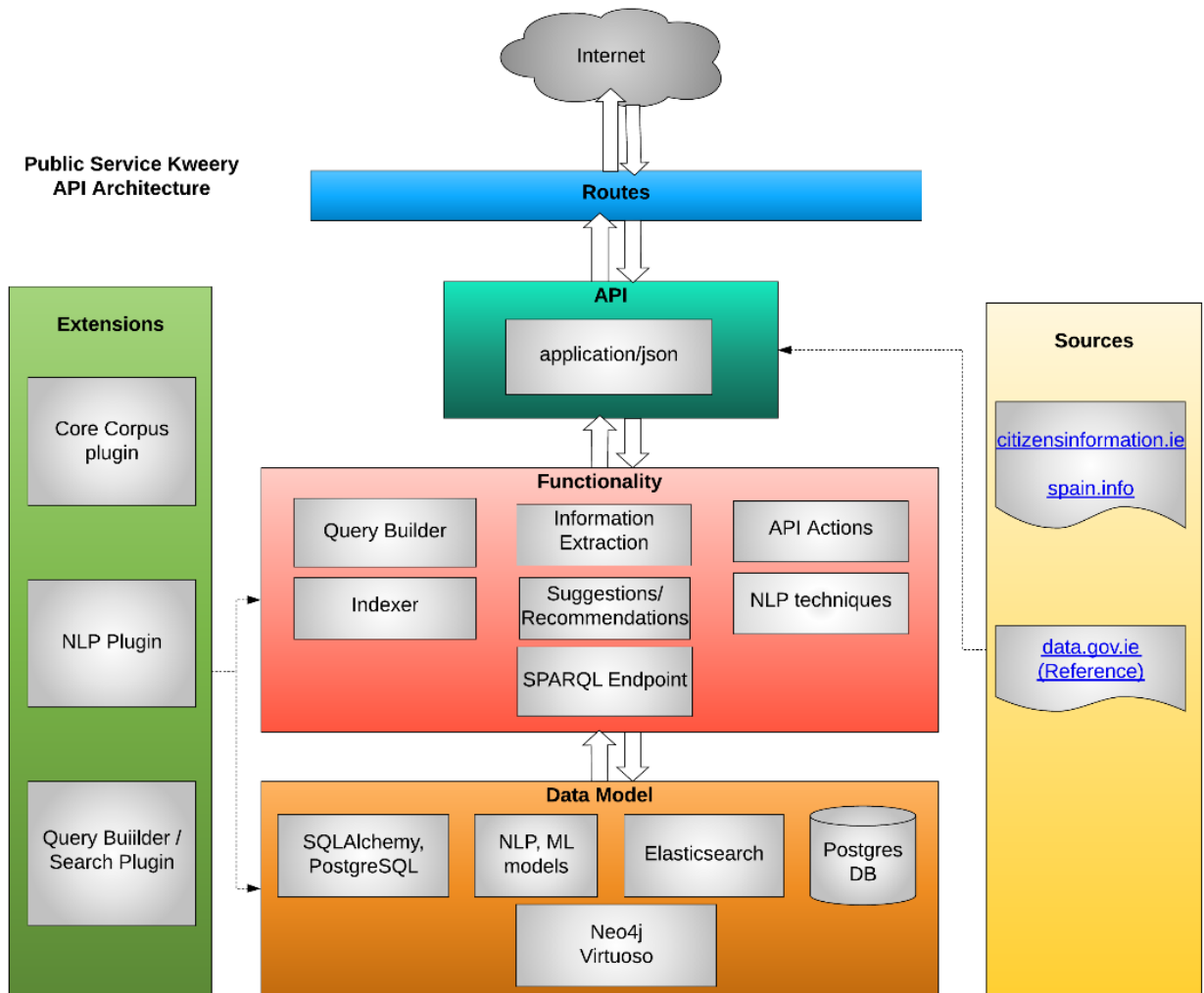


Figure 7: Kweery chatbot API Architecture:

Text Cleaning Pipeline

We retained all the data pipeline from phase I. In addition, each text cleaning process was made pluggable, allowing us to modify or add additional cleaning processes. The core cleaning process includes:

- Extra space removal.
- Stemmer (Using NLTK library).
- Lemmatization.
- Stop words removal.
- Regex pattern extractor.
- Domain specific stop words removal.

NLP Data Model

For entity extraction we use NLTK and SpaCy⁹ python packages. For detection of synonyms, hypernyms and hyponyms we use Wordnet¹⁰. This extracted information is then indexed to Elasticsearch¹¹ for easy searchability and scalability.

Further information on text relationship is extracted from a mix of SpaCy and NLTK tree parser with custom grammar rules. Information is extracted in the form:

subject -> predicate -> object.

This relationship extraction and understanding how the components of the text are related to each other increased our search results significantly and this can handle huge volumes of the search.

Extracted text relations are then stored in Graph Db (Neo4J¹²) for extracting suggestions and recommendations.

Recommendation System

We are using graph data model to extract the recommendation to the user. That is, for any user entered text we extract text relationship and find if the same relationship exists. If it does exist we extract all the child and parent components from the text and provide these key words as suggestion to the user.

Below figure 8 represents the workflow and components of the recommendation system or suggestion system.

⁹ <https://spacy.io/>

¹⁰ <https://wordnet.princeton.edu/>

¹¹ <https://www.elastic.co/>

¹² <https://neo4j.com/>

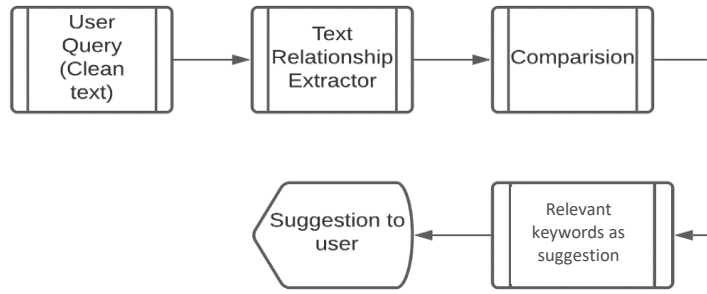


Figure 8. Recommendation System

The extracted text relationship is also stored in Linked Data Database (Virtuoso¹³). A Virtuoso RDF end point is made available for technical users.

Search and Build Query (Realtime)

This module/extension builds an ElasticSearch query when a user types a question. The query build uses the ranking mechanism to identify the important parts of a query and gives a higher rank if any match is found. Also, it searches user query text relationship with the available text relationship and ranks them accordingly.

There is also a configuration file to add/change current ranking systems and further analyzers/filters can be added or removed. This way we can change the behaviour of the search functionality depending on the harvested data.

Figure 9: Represents the workflow of the component - search query builder.

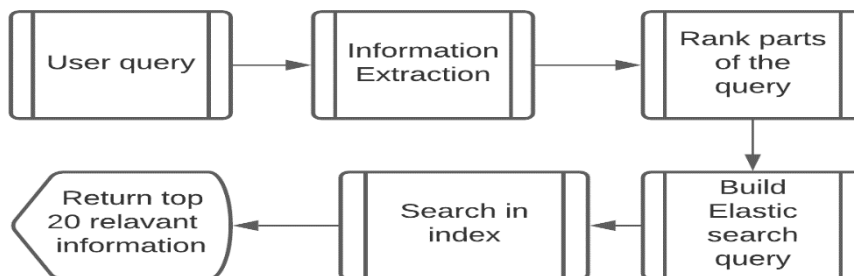


Figure 9. Return most relevant information

Below figure 10 represents the dialog framework components and how the response is sent to the given query.

¹³ <https://virtuoso.openlinksw.com/>

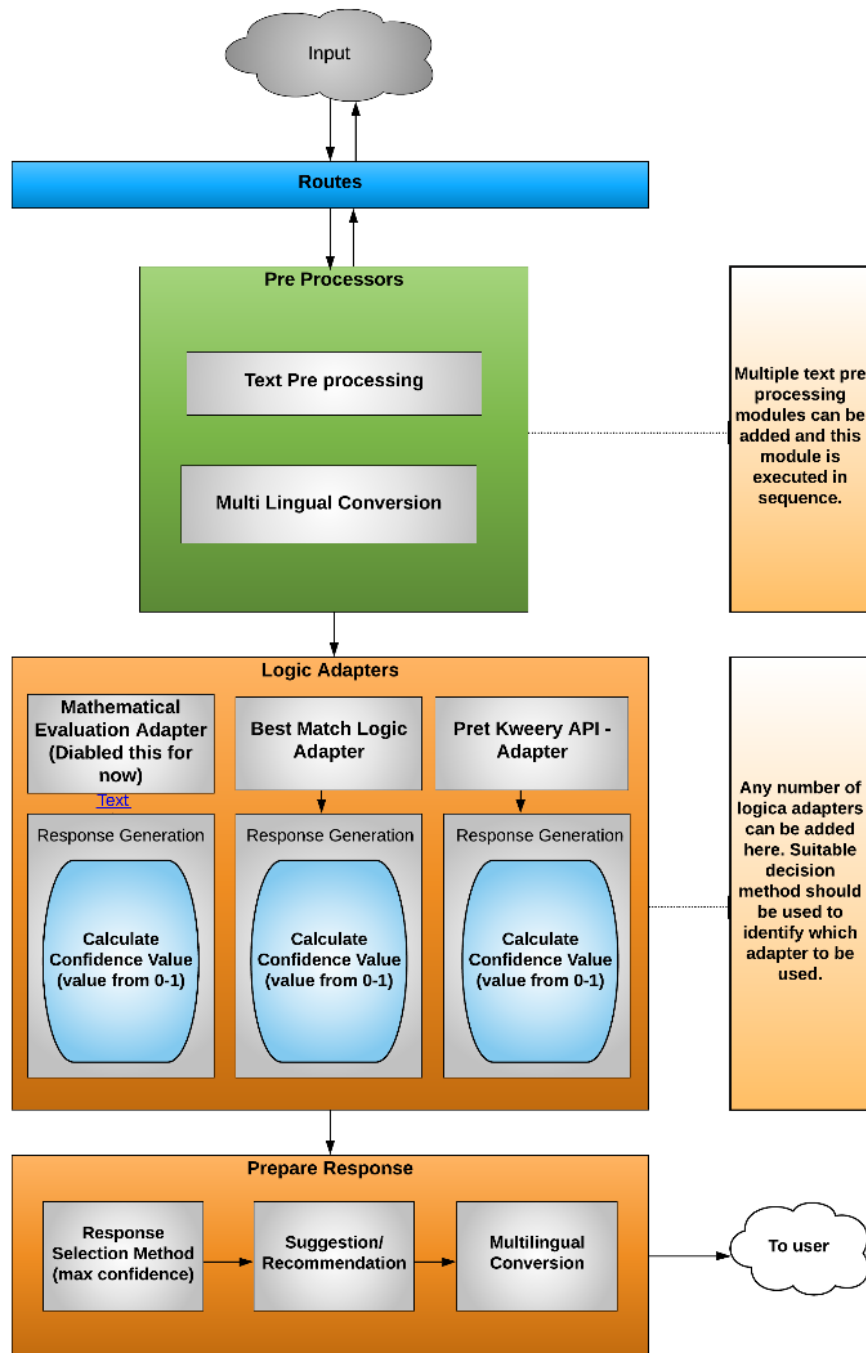


Figure 10. Kweery chatbot Dialog Framework Architecture

Multilingual Functionality

Currently Kweery chatbot supports two languages, English and Spanish. However, it can be made to handle most languages, depending on the translation engine we have. At the moment we are using the Google translation engine¹⁴ but we plan to use Prêt-à-LLOD translation tools as they become

¹⁴ <https://cloud.google.com/translate/docs/resources>

available. The translation API is part of the NLP extension and can easily be adapted to any translation API.

To switch to other languages, data should be harvested in those languages (knowledge base) and the language needs to be added to the list of supported languages in the config file.

Steps:

- We convert multilingual query to English
- Process the information in English.
- Extract relevant information in English and map to the required language.

Figure 11, below, represents the multilingual system, which is responsible for translating from one language to another (Supported English and Spanish).

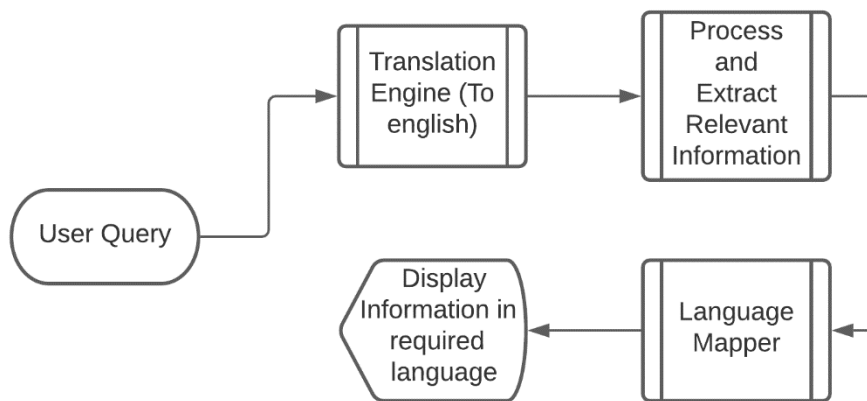


Figure 11. Translation process

The Language Mapper module maps the extracted relevant information to the knowledge base in respective language.

Phase II addressed all the limitations from phase I. Now the application is multilingual, scalable, conversational and has a better quality of response.

Limitations

- 1 Quality of response is lot better than phase 1. However, with more text cleaning/extracting information quality can be increased.
- 2 Need to account for the translation error (I.e., not all the statements are converted accurately). Need a way to handle the translation errors which is impacting on extracting the relevant information.
- 3 Application is not yet production ready and cannot handle multiple users.

2.3.3 Phase III

In this phase we have concentrated more on making the application production ready and improving the quality of response. In addition, we tried to address the limitations of the previous phase.

We included a more extensive text cleaning process with domain-specific words and also added the synonyms filter to the search index. In addition, we performed a lot of experimentation with indexing and capturing the semantic meaning of the text. Figures 12,13 and 14 document the JSON code for the Elastic Search Analysers, the Index filters and Search index schema used in Phase III.

Figure 12 represents the elastic search analyser configuration file and can be modified according to our needs.

```
analysis:
{
  analyzer:
  {
    pattern_analyzer:
    {
      filter:
      [
        "stemmer",
        "lowercase"
      ],
      pattern: "\\W+",
      flag: "CASE_INSENSITIVE",
      type: "pattern",
      stopwords: "_english_"
    },
    standard_stemming_lowercase_strip_html:
    {
      filter:
      [
        "stemmer",
        "lowercase",
        "synonym"
      ],
      char_filter:
      [
        "html_strip"
      ],
      type: "standard",
      stopwords: "_english_",
      tokenizer: "letter"
    },
    remove_stopwords:
    {
      type: "stop",
      char_filter:
      [
        "html_strip"
      ],
      stopwords: "_english_"
    },
    domain_analyzer:
    {
      filter:
      [
        "stemmer",
        "lowercase",
        "synonym"
      ],
```

```

        type: "standard",
        stopwords: "_english_"
    },
    country_tag_analyzer:
    {
        filter:
        [
            "lowercase"
        ],
        char_filter:
        [
            "html_strip"
        ],
        type: "standard",
        stopwords: "_none_"
    },
    standard_stemming_lowercase:
    {
        filter:
        [
            "stemmer",
            "lowercase"
        ],
        type: "standard",
        stopwords: "_english_",
        tokenizer: "letter"
    },
    standard_stemming_lowercase_shingle:
    {
        filter:
        [
            "stemmer",
            "lowercase",
            "synonym",
            "my_shingle_filter"
        ],
        char_filter:
        [
            "html_strip"
        ],
        type: "standard",
        stopwords: "_english_",
        tokenizer: "letter"
    }
}

```

Figure 12. Elastic Search Analysers for Phase III

Index Filters:

The below figure 13 represents the Elasticsearch filter configuration file and any new filter can be added easily.

```

filters:
{
  my_shingle_filter:
  {
    max_shingle_size: "5",
    min_shingle_size: "2",
    output_unigrams: "false",
    type: "shingle"
  },
  synonym:
  {
    type: "synonym",

```

```
    synonyms_path: "dlxbot/synonyms.txt"
  }
},
```

Figure 13. Index Filters (Phase III)

Search Index Schema:

The below figure 14 represents Elasticsearch mapping and data type of the knowledge base (corpus), from where information is extracted for the given query.

```
mappings:
{
  properties:
  {
    c_id:
    {
      type: "keyword"
    },
    country_tag:
    {
      type: "keyword"
    },
    document_id:
    {
      type: "keyword"
    },
    domain:
    {
      type: "text",
      analyzer: "domain_analyzer"
    },
    elastic_search_id:
    {
      type: "long"
    },
    extras:
    {
      type: "object",
      dynamic: "true"
    },
    features:
    {
      type: "nested",
      dynamic: "false",
      properties:
      {
        full_text:
        {
          type: "text",
          analyzer: "standard_stemming_lowercase_shingle"
        },
        named_entities:
        {
          type: "text",
          analyzer: "pattern_analyzer"
        },
        np:
        {
          type: "text",
          analyzer: "standard_stemming_lowercase_shingle"
        },
        text_relationship:
        {
```

```

        type: "text",
        analyzer: "standard_stemming_lowercase_shingle"
    },
    text_relationship_hyper:
    {
        type: "text",
        analyzer: "standard_stemming_lowercase_shingle"
    },
    text_relationship_hypo:
    {
        type: "text",
        analyzer: "standard_stemming_lowercase_shingle"
    },
    text_relationship_syn:
    {
        type: "text",
        analyzer: "standard_stemming_lowercase_shingle"
    }
    }
},
p_id:
{
    type: "keyword"
},
sub_domain:
{
    type: "text",
    analyzer: "domain_analyzer"
},
tags:
{
    type: "text",
    analyzer: "standard_stemming_lowercase"
}
}

```

Figure 14. Search index schema (Phase III)

In addition to the above search configuration, we have tried several other technologies to further narrow down the extracted related information with cosine similarity, semantic text similarity and sentence transformers.

Testing and bug fixing

In this phase most of the critical bugs have been fixed, however the software is not completely bug free.

2.4 Application Benchmarks

Note: Phase 1 Kweery chatbot scope only covered the “Moving Country” section of the Irish citizens’ information website so the accuracy figures are based on that limited scope. Phase 2 was tested twice, the first time with the same 26 questions to provide a comparison to Phase 1, then against the full scope (41 questions). All later phases included the whole of the Irish citizens’ information website (16 topics).

Phase	Model	Top answer is expected answer	Answer is correct or in suggestions	Score for search function on website	Comments
Phase 1	TF-IDF (Scikit-learn)	35.6%	55.7%	61.5%	Tested using 26 questions covering a single section of the Irish citizens’ information website
Phase 2	Elasticsearch Analysers and Boosting with Cosine similarity	35.6%	58.7%	61.5%	Tested using the same 26 questions
		35.4%	62.8%	73.2%	Over 41 questions covering all sections
Phase 2+	BERT Semantic text similarity ¹⁵	53.0%	72.0%	73.2%	Over the 41 questions
Phase 3	Sentence transformer (base nli mean tokens) ¹⁶	36.6%	62.8%	73.2%	Over the 41 questions

Figure 15: Comparative test results for models for Kweery chatbot

¹⁵ <https://github.com/AndriyMulyar/semantic-text-similarity>

¹⁶ <https://github.com/AndriyMulyar/semantic-text-similarity>

Conclusions

One of the models stood out, with significantly better results than the others for identifying the expected answer as the top answer. This model also returned similar results to the Google search used on the website, once the suggestions were taken into account. We decided to use this model (BERT Semantic text similarity).

2.4.1 KPI Contributions

This pilot will contribute to the following project KPI:

Nr	KPI		Benchmarks
4.4	Satisfaction level of early adopters of pilots in a survey	4-5/5	Chatbot pilot user satisfaction survey

2.5 Next steps

The next steps for Kweery chatbot include:

- Generate formal test results for the multilingual functionality
- Give interested clients access and get feedback on the application
- Explore expanding the multilingual functionality to other European languages

3 Pilot III.B Open Data Dashboard

3.1 Motivation

Open Data can provide useful information to users and is readily accessible to the more technical users who may be reading the data programmatically using an API or by other means. We identified a need to improve access to Open Data for less technical users. Open Data is generally formatted as comma-delimited files (CSV), which are machine-readable and can be opened in a program such as Excel or an alternative such as Google Sheets. To access open data information of interest, the user generally has to perform the following:

- Identify the dataset of interest
- Identify the associated resource of interest
- Download the resource
- Open the resource in the spreadsheet software of choice
- Search for the information required

This is a cumbersome and tedious process, in particular for users who are less technically minded, making it difficult to find a single piece of information.

The idea was to allow users to input a conversational question which can then be converted into a query which would:

1. Identify the dataset/resource containing relevant information
2. Select the relevant information from a CSV file
1. Return the information

3.2.1 Use cases

We identified three main use cases for the dashboard:

1. To answer simple user questions from information contained in Open Data CSV files
2. To apply a question to multiple files
3. To provide the capability to combine questions using Boolean functions (e.g. searching for a non-Catholic school in Donegal)

Use Case 1

The first use case simply means the user asks a question and the dashboard returns all the relevant datasets allowing the user to select one or two datasets to further inspect their resources. Selecting a resource, the user can explore the resource's content and apply automatically generated filters based on the user's original question.

Use Case 2

The second use case requires the user to select 2 files to apply the same filter to both of them and compare results side by side

Use Case 3

The third and final use case requires the user to input new custom filters to filter resource's contents.

3.2.2 Data Sources

- Ireland's Open Data Portal¹⁷
- Dublinked: Open Data for the Dublin Region¹⁸
- NLTK corpora, taggers, tokenizers¹⁹

3.2.3 Current languages of interest

Currently English but we plan to extend this to other European languages.

3.3 Technical Approach

Kweery Dashboard is implemented using a fairly common web application stack:

- Docker²⁰ for containerization
- Flask²¹ for the API functionalities
- Angular²² for the web application framework
- Natural language processing: NLTK²³ (Natural Language Toolkit).

The application stack has been developed over 4 phases.

Phase 1

Phase 1 covered the basic infrastructure design and key functionalities:

- Fetching data from datAdore
- Displaying the tabular data in the dashboard
- Included a basic filtering function

¹⁷ <https://data.gov.ie/>

¹⁸ <https://data.smartdublin.ie/>

¹⁹ http://www.nltk.org/nltk_data/

²⁰ <https://www.docker.com/>

²¹ <https://flask.palletsprojects.com/>

²² <https://angular.io/>

²³ <https://www.nltk.org/>



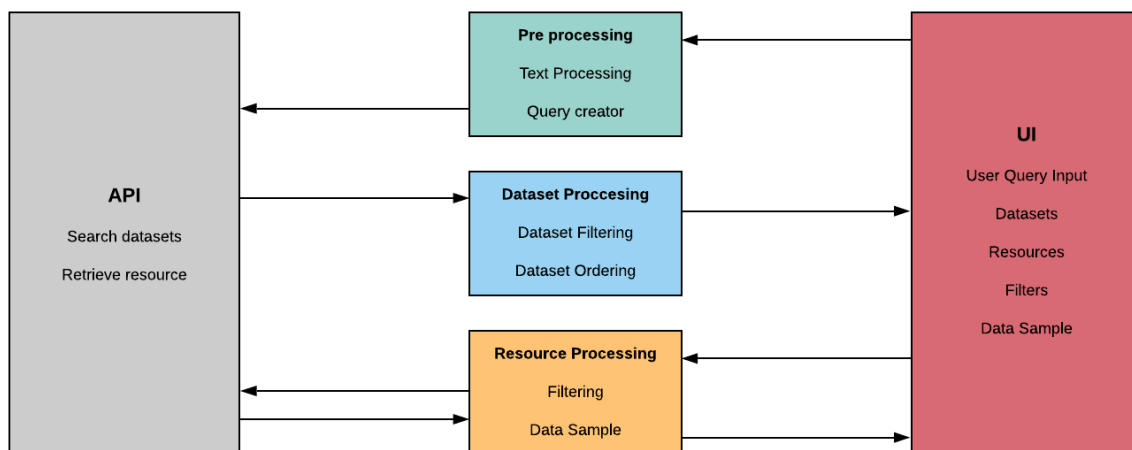


Figure 17: Basic design of Kweery dashboard

Text processing:

Extraction of text entities using NLTK library. Extracting verbs and nouns to create the filters.

Query creator:

Multiple combinations of entities from n to 1 length, ensuring we fetch all possible related datasets from the data source.

This process works creating a query with all the entities extracted, and if there are no results found, it will combine all the entities in groups of length of entities – 1, perform the search query and in case there is no results found it will keep forming groups of smaller length until all the entities are exhausted.

Dataset filtering and ordering, based on user input query similarity.

Resource processing:

Present query data based on automatically created filters (text entities extracted in the text processing step).

The infrastructure of the project is divided into an API backend made in Flask, an Angular dashboard and Docker for containerization.

The workflow:

- the user loads the dashboard and inputs a question.
- the dashboard extracts the texts entities and creates multiple data requests to the source server
- the dashboard weights and orders the results and displays them to the user
- the user selects one or two of the dataset results and the associated resources are displayed
- the user selects a resource
- the resource's contents are shown, and the user can filter the data

Once the basic functionality of the dashboard was working, we focused on extending its capabilities through the following phases.

Phase 2

Phase 2 extended the natural language processing capabilities by identifying combination keywords ('and', 'or' and 'not') to expand the functionality of the filtering function.

Apart from detecting verbs and nouns, the natural language processing was extended to detect 'and', 'or' and 'not' keywords and attach them to the relevant noun or verb, creating new filters.

These filters can be used the same way as in phase 1. In this phase we added the ability for the user to add new custom filters, giving the user full control over the filtering function.

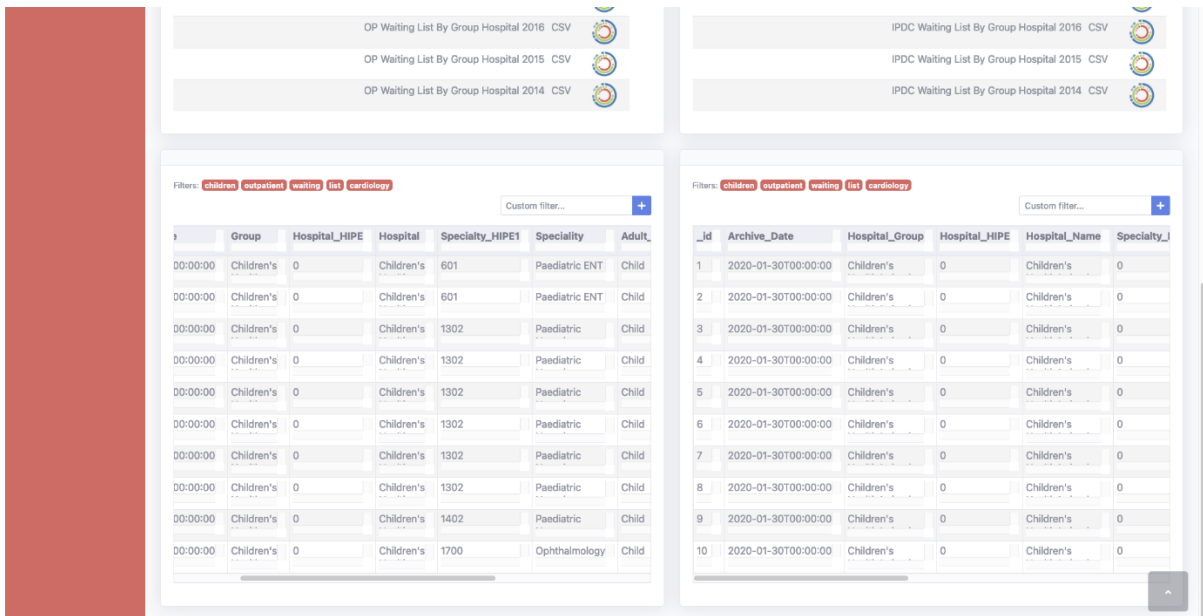
Example workflow in Phase 2:

In response to the query "What is the children's outpatient waiting list for cardiology?", the user is offered and can select the following two datasets, Outpatient and Inpatient waiting list by group hospital.

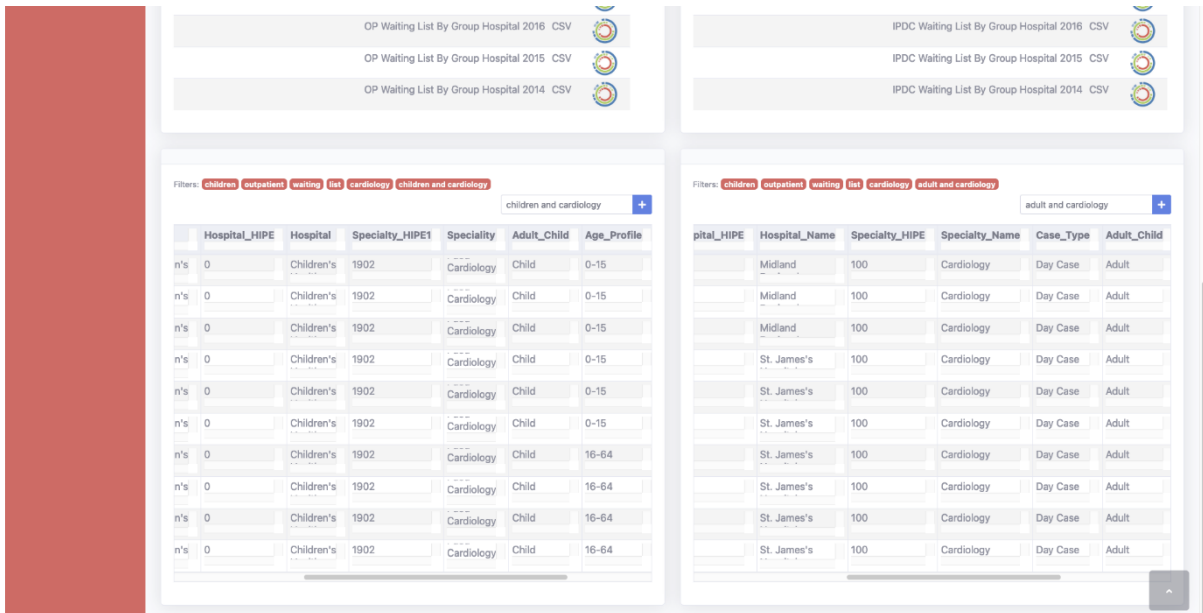
The screenshot shows a dashboard interface. On the left is a red sidebar with a 'Dashboard' header and a back arrow. The main content area is divided into two panels. The left panel is titled 'OP WAITING LIST BY GROUP HOSPITAL' and contains a table with columns: Name, Description, Format, and Source. The right panel is titled 'IPDC WAITING LIST BY GROUP HOSPITAL' and contains a table with columns: Filename, Description, Format, and Source. Below the tables are two filter bars. The left filter bar has buttons for 'children', 'outpatient', 'waiting', 'list', and 'cardiology', and a 'Custom filter...' input field. The right filter bar has the same buttons and a 'Custom filter...' input field.

The dashboard lists the various associated resources for each dataset, and the user can select the most recent data (2020).

The dashboard automatically generates filters that the user can select, based on the user's query, in this case, "children", "outpatient", "waiting list" and "cardiology":



The user can then select and combine the filters to apply and drill down to the information required, in this case they create a filter “children and cardiology” and apply it to both resources and the dashboard displays children’s cardiology information from the inpatients’ and outpatients’ files.



Phase 3

Phase 3 integrated the Kweery Chatbot and Dashboard allowing the Chatbot to show open data suggestions related to the chat conversation.

To create seamless integration between the Chatbot and the Dashboard we created two new API endpoints in the dashboard:



- Search endpoint: this endpoint allows the chatbot to ask the dashboard whether there are open data datasets relevant to the user question and provides a direct link to the dashboard where relevant.
- Show endpoint: hitting this URL will load the dashboard with the user question and the relevant resources already loaded.

Phase 4

In order to extend the dashboard features we want to make it work with as much open data as possible. The most numerous open data resources are in tabular format (comma delimited files that the dashboard can process). The second most numerous resources are geospatial data in geojson²⁴, kml²⁵ and other geospatial formats.

In phase 4 the dashboard also works with geospatial data, displaying it in a map and allowing the user to interact with the data directly in the map.

The map can display/hide multiple layers, aggregate layers into one and show extra data on every feature.

The library used to display the map is Leaflet²⁶.

3.4 Application Benchmarks

We will evaluate the sub-pilot for the quality and speed of its results against a manual process.

The time and quality benchmarks will be set by a user with a translation tool, measuring the time taken to identify and retrieve relevant data.

The datasets that have been identified by the application will be marked for comparative relevance and the time taken to identify and retrieve.

The criteria:

1. Were all the datasets manually identified as relevant identified by the application?
2. How relevant were any additional datasets that were retrieved using the application?
3. Was the data presented to the user in a meaningful form?
4. What was the time saving in delivering meaningful results with the application vs. the manual process?

The analysis will have both quantitative and qualitative components.

- Beta Testing: User testing with trial users incorporating:
 - Criteria 1,2 and 3 above
 - Satisfaction of trial users with functionality
- Metrics for Application Usage:
 - Number of datasets downloaded using the tool

²⁴ [GeoJSON Specification \(r-project.org\)](https://geojson.org/)

²⁵ [KML Tutorial | Keyhole Markup Language | Google Developers](#)

²⁶ <https://leafletjs.com/>

- Number of queries made using the tool

Test case 1:

User question: “What is the children’s outpatient waiting list for cardiology?”

Dashboard fetched the following datasets:

1. OP²⁷ Waiting List By Group Hospital ²⁸
2. IPDC²⁹ Waiting List By Group Hospital ³⁰

Entering the same query on Ireland’s Open Data Portal fetched the following datasets:

1. OP Waiting List By Group Hospital
2. IPDC Waiting List By Group Hospital
3. IPDC GI Endoscopy by Group Hospital ³¹

Test case 2:

User question: “Are there any non-Catholic schools in Donegal?”

Dashboard fetched the following datasets:

1. Schools³²
2. Primary Schools³³
3. School Enrolment 2008 – 2011³⁴
4. Data on Individual Schools³⁵
5. Secondary School Locations³⁶
6. Primary School Locations ³⁷
7. Post Primary Schools ³⁸
8. EDA70 - Pupils Enrolled in Second Level Schools (Number) by Sex, Type of School, School Programme and Year ³⁹
9. EDA49 - National Schools and Pupils by Teacher Size of School by Teacher Size of School, Year and Statistic ⁴⁰
10. Cork (Rochelle Schools) Rainfall Data⁴¹
11. Co. Donegal Municipal Districts ⁴²
12. Donegal (Dist.Hosp.) Rainfall Data ⁴³

²⁷ Outpatient

²⁸ <https://data.gov.ie/dataset/op-waiting-list-by-group-hospital>

²⁹ Inpatient

³⁰ <https://data.gov.ie/dataset/ipdc-waiting-list-by-group-hospital>

³¹ <https://data.gov.ie/dataset/ipdc-gi-endoscopy-by-group-hospital>

³² <https://data.gov.ie/dataset/schools1>

³³ <https://data.gov.ie/dataset/primary-schools>

³⁴ <https://data.gov.ie/dataset/school-enrollment-20082011>

³⁵ <https://data.gov.ie/dataset/data-on-individual-schools>

³⁶ <https://data.gov.ie/dataset/secondary-school-locations>

³⁷ <https://data.gov.ie/dataset/primary-school-locations>

³⁸ <https://data.gov.ie/dataset/post-primary-schools>

³⁹ <https://data.gov.ie/dataset/pupils-enrolled-in-second-level-schools-number-by-sex-type-of-school-school-programme-and-year>

⁴⁰ <https://data.gov.ie/dataset/national-schools-and-pupils-by-teacher-size-of-school-by-teacher-size-of-school-year-and-statistic>

⁴¹ <https://data.gov.ie/dataset/cork-rochelle-schools-rainfall-data>

⁴² <https://data.gov.ie/dataset/co-donegal-municipal-districts>

⁴³ <https://data.gov.ie/dataset/donegal-disthosp-rainfall-data>

13. Donegal Prompt Payment Return Q4 2017 ⁴⁴
14. Donegal Prompt Payment Return Q2 2017 ⁴⁵
15. Donegal Prompt Payment Return Q1 2017 ⁴⁶
16. Donegal (W.W.) Rainfall Data ⁴⁷
17. CD154 - Donegal Population by Private Households, Occupied and Vacancy Rate by Townlands, and Statistic⁴⁸
18. Revenue Net Receipts by County ⁴⁹
19. Tellus Airborne Geophysics - Total Count ⁵⁰
20. Tellus Airborne Geophysics - Magnetic Intensity ⁵¹

Entering the same query on Ireland's Open Data Portal fetched 24 datasets:

⁴⁴ <https://data.gov.ie/dataset/donegal-prompt-payment-return-q4-2017>

⁴⁵ <https://data.gov.ie/dataset/donegal-prompt-payment-return-q2-2017>

⁴⁶ <https://data.gov.ie/dataset/donegal-prompt-payment-return-q1-2017>

⁴⁷ <https://data.gov.ie/dataset/donegal-cullionboy-ww-rainfall-data>

⁴⁸ <https://data.gov.ie/dataset/gal-population-by-private-households-occupied-and-vacancy-rate-by-townlands-censusyear-and-statistic>

⁴⁹ <https://data.gov.ie/dataset/revenue-net-receipts-by-county>

⁵⁰ <https://data.gov.ie/dataset/tellus-airborne-geophysics-total-count>

⁵¹ <https://data.gov.ie/dataset/tellus-airborne-geophysics-magnetic-intensity>



Search Datasets : Search Tips Advanced Search

"Are there any non-Catholic schools in Donegal?" Q

24 datasets found for ""Are there any non-Catholic schools in Donegal?" " Order by: Relevance

INSPIRE Strategic noise maps. Noise contour map for railway noise in agglomerations – Lden Environment

Environmental Protection Agency

This is a polygon dataset of the strategic noise mapping of rail, which were identified as those rail exceeding the flow threshold of 30,000 vehicle passages per year, in the form of noise contours for the Lden (day, evening, night) period for Dublin and Cork agglomerations...

NULL WMS
★★★★☆
1 views

INSPIRE Strategic noise maps. Noise contour map for railway noise in agglomerations – Lnight Environment

Environmental Protection Agency

This is a polygon dataset of the strategic noise mapping of rail, which were identified as those rail exceeding the flow threshold of 30,000 vehicle passages per year, in the form of noise contours for the Lnight (night) period for Dublin and Cork agglomerations and the major...

NULL WMS
★★★★☆
6 views

INSPIRE Strategic noise maps. Noise contour map for roads noise in agglomerations – Lden Environment

Environmental Protection Agency

This is a polygon dataset of the strategic noise mapping of roads, which were identified as those roads exceeding the flow threshold of 3 million passages per year, in the form of noise contours for the Lden (day, evening, night) period for Dublin and Cork agglomerations and...

NULL WMS
★★★★☆
15 views

INSPIRE Strategic noise maps. Noise contour map for roads noise in agglomerations – Lnight Environment

Environmental Protection Agency

This is a polygon dataset of the strategic noise mapping of roads, which were identified as those roads exceeding the flow threshold of 3 million passages per year, in the form of noise contours for the Lnight (night) period for Dublin and Cork agglomerations and the major...

NULL WMS
★★★★☆
3 views

Figure 16: Top datasets returned from “Are there any non-Catholic schools in Donegal?” query

Test case 3

User question: “National parks to bring my kids to”

Kweery dashboard fetched the following datasets:

1. National Parks⁵²
2. National Parks - Points of Interest ⁵³
3. National Parks & Wildlife Service Estuaries⁵⁴
4. National Parks & Wildlife Service Sandbanks⁵⁵

Entering the same query on Ireland’s Open Data Portal fetched the following datasets:

1. Irish Semi-natural Grassland Survey 2007-2012⁵⁶

⁵² <https://data.gov.ie/dataset/national-parks>

⁵³ <https://data.gov.ie/dataset/national-parks-points-of-interest>

⁵⁴ <https://data.gov.ie/dataset/national-parks-and-wildlife-service-estuaries>

⁵⁵ <https://data.gov.ie/dataset/national-parks-and-wildlife-service-sandbanks>

⁵⁶ <https://data.gov.ie/dataset/irish-semi-natural-grassland-survey-2007-2012>

The results of these tests are collated in Figure 17 below.



Question 1				
“What is the children’s outpatient waiting list for cardiology?”				
	Using Kweery Dashboard		Using the same query on data.gov.ie	
		Percentage of identified datasets directly relevant to query		Percentage of identified datasets directly relevant to query
Were all the datasets manually identified as relevant, found by the application?	Yes	100%	Yes	29%
How relevant were any additional datasets that were retrieved using the application?	n/a, both datasets retrieved were used	100%	None	100%
Was the data presented to the user in a meaningful form?	Yes, the user could apply the filters suggested to drill down to the relevant data		n/a	
Question 2				
“Are there any non-Catholic schools in Donegal?”				
	Using Kweery Dashboard		Using the same query on data.gov.ie	
		Percentage of identified datasets directly relevant to query		Percentage of identified datasets directly relevant to query
Were all the datasets manually identified as relevant, identified by the application?	Yes	100%	No	0% - no school, Catholic or Donegal-related datasets were returned
How relevant were any additional datasets that were retrieved using the application?	Datasets were counted as relevant if related to schools, Catholic or Donegal	90%		0% - 24 irrelevant datasets returned
Was the data presented to the user in a meaningful form?	The most relevant data was at the top of the list. The user could apply filters to retrieve that information required			n/a

Question 3	"National parks to bring my kids to"			
	Using Kweery Dashboard		Using the same query on data.gov.ie	
		Percentage of identified datasets directly relevant to query		Percentage of identified datasets directly relevant to query
Were all the datasets manually identified as relevant, identified by the application?	Yes	100%	None of the relevant datasets were retrieved	0%
How relevant were any additional datasets that were retrieved using the application?	Datasets were counted as relevant if related to National parks	100%	n/a	0%
Was the data presented to the user in a meaningful form?	The user could apply filters to retrieve that information required		n/a	

Figure 17. Test results for Kweery dashboard

Conclusions

These preliminary results indicate that the dashboard search function is significantly more accurate than the standard search function available on the Irish National Open Data portal for natural language queries.

3.4.1 KPI Contributions

Nr	KPI		Benchmarks
4.2	Average reduction in the time (man hours) to convert a resource to linked data at end of project.	90%	Develop benchmarks based on sample input from users - manual method vs with application
4.4	Satisfaction level of early adopters of pilots in a survey	4-5/5	Survey of users

Figure 18: Applicable project KPIs

We will develop benchmarks to use with the early adopters of the application to measure these performance indicators.

3.5 Next Steps

We will continue to extend the dashboard features, making it work with more formats and improved understanding of the user query. We will expand the assessment and testing of the dashboard to incorporate assessment of the Key Performance Indicators and generate statistics for application usage among early adopters.

We will extend the dashboard to include some multilingual functionality. Initially the data will be returned with information at the resource metadata level translated to the language of the query.

The options we are looking at as next steps to improve the functionality of the dashboard include:

- Make use of chatbot natural language processing capabilities in the dashboard to improve the text entities extraction and multilingual translations.
- Multilingual retrieval of data, returning the contents (headers) of the data in translation as well as the resource metadata.
- Extend the search to other Open Data European Portals.
- Automatic selection of dataset and resource. Now the datasets weighting and order functions work well but we want to be able to automatically select the most relevant dataset and resource, saving time for the user.
- Investigation of a 'temporal query' meaning that we can detect dates, years, periods of time or any other date-related query and show only the data for that period, reducing the number of options shown to the user in the dashboard and the time needed to reach the information.

4 Summary and Next Steps

Further to thorough testing of the integrated sub-pilot, Kweery has reached such level of maturity that Derilinx is now in a position to run demonstration sessions, building on the strong interest in the solution from existing Derilinx customers and other potential customers. Please visit <https://pret-a-llod.github.io/posts/derilinx-webinar-recording.html> to access the recording of an example webinar.

The following pictures are screenshots of Kweery in action.

Integrated Kweery Chatbot

The image displays two screenshots of the Kweery chatbot interface. Both screenshots show a chatbot window titled "Prêt-à-LLOD: Public Service query tool" with language selection options (Spanish and English).
The top screenshot shows a user asking "Renting a car in Spain". The chatbot responds with a message: "If you are thinking of driving in Spain, you will surely be interested in finding out beforehand how to rent a car and whether your driving licence is compatible in this country, and what is required to get valid international insurance. Find out all about it below." Below this is a button labeled "OPEN DATA DASHBOARD". A follow-up message says "I found somewhat similar topics, please click on yes to see more" with a "Yes" button. A "Check this out" section follows, containing a grid of topic buttons such as "HOW DO YOU RENT A CAR?", "WHAT ARE THE MOST IMPORTANT RULES?", "WHAT DRIVING LICENCES CAN YOU USE IN SPAIN?", "WHICH ROADS HAVE TOLLS AND WHICH DO NOT?", "WHAT SHOULD YOU DO IF YOU GET A FINE?", "MONEY HOW CAN YOU PAY IN SPAIN?", "WHAT ARE THE MAIN OPTIONS FOR PHONING FROM SPAIN?", "HOW CAN I PAY IN SPAIN?", "WHAT CAN YOU BRING INTO SPAIN AND WITH WHAT LIMITATIONS?", "WHAT ARE NORMAL OPENING HOURS IN SPAIN?", "WHAT TIME IS IT IN SPAIN?", "WHAT ARE SPAIN'S NATIONAL PUBLIC HOLIDAYS?", "CONNECTIONS HOW TO GET MOBILE COVER AND INTERNET ACCESS", "TYPES OF VISAS AND WHERE TO GET THEM", "COMPLAINING HOW DO YOU MAKE A TOURIST COMPLAINT?", "CUSTOMS WHAT REGULATIONS ARE THERE?", "WHAT IS THE OFFICIAL CURRENCY?", "CUSTOMS AND MONEY WHAT IS VAT AND HOW TO RECOVER IT? TAX-FREE", and "CELIAC DISEASE IS IT EASY TO FIND FOOD FOR CELIACS AND HOW DO YOU ASK FOR IT?".
The bottom screenshot shows a user asking "¿Cómo obtener una licencia de conducir en Irlanda?". The chatbot responds with a message: "Puede obtener una copia de su registro de licencia de conducir del Servicio Nacional de Licencia de Conducir (NDLS). Consulte 'Contactos útiles' a continuación para conocer sus detalles. Hay un cargo de 15 €." Below this is a button labeled "HACER CLIC EN EL BOTÓN PARA VER LOS DATOS ABIERTOS RELACIONADOS EN EL PANEL" with "HACER CLIC EN EL BOTÓN PARA VER LOS DATOS ABIERTOS" below it. A follow-up message says "¿Te gustaría ver los temas relacionados?" with a "Si" button. A "Temas similares que pueden ser útiles" section follows, containing a grid of topic buttons such as "¿QUÉ SUCEDE SI COMETO UNA INFRACCIÓN DE CONDUCCIÓN EN EL EXTRANJERO?", "¿QUÉ SUCEDE SI CONDUZCO CON UNA LICENCIA EXTRANJERA EN IRLANDA?", "CONDUCCIÓN EN LA UE", "INFRACCIONES POR CONDUCCIÓN BAJO LOS EFECTOS DEL ALCOHOL", "PRUEBAS DE CONDUCCIÓN DE MOTOCICLETAS", "USAR SU TELÉFONO MÓVIL MIENTRAS CONDUCCIÓN", "PRUEBAS DE CONDUCCIÓN", "DELITOS DE CONDUCCIÓN", "CONDUCCIÓN FUERA DE LA UE", "PUNTOS DE PENALIZACIÓN POR INFRACCIONES DE TRÁNSITO", "DELITOS POR CONDUCCIÓN BAJO LOS EFECTOS DEL ALCOHOL O DROGAS", "PRUEBAS DE CONDUCCIÓN DURANTE COVID-19", "CONDUCCIÓN", "CONDUCCIÓN PELIGROSA", and "CONDUCCIÓN DESCUIDADA".

Prêt-à-LLOD Public Service query tool ireland ▼ English ▼

I am pre Chat Bot. How can I assist you?

how to get a driving license in ireland?

Please click the button to see related open data in the dashboard
[OPEN DATA DASHBOARD](#)

You can get a copy of your driving licence record from the National Driving Licence Service (NLS). See 'Useful contacts' below for their details. There is a charge of €15.

Driving Tests During Covid-19

Please click the button to see related open data in the dashboard
[OPEN DATA DASHBOARD](#)

Driving tests started again on 29 June 2020 for bus, trucks and motorcycles. Car testing started again on 20 July 2020 on a phased basis. If you are doing your driving test, you should read the safety procedures that are in place (pdf). Driving licences and learner permits that are due to expire before 30 June 2020 will be valid for a further 4 months. The following documents are also valid for a further 4 months: Certificates of competency received on passing a driving test Driver theory test certificates Motorcycle Initial basic training certificates This means that if your document expires between 1 March 2020 and 30 June 2020, you can add 4 months onto its expiration date.

HOW DO I APPLY TO TAKE A TEST? IF ENGLISH IS NOT YOUR FIRST LANGUAGE WHAT DO I NEED TO TAKE THE TEST? PREPARING FOR AND TAKING A TEST TESTS FOR PEOPLE WITH A DISABILITY

FURTHER INFORMATION MOTORCYCLE DRIVING TESTS APPEALING YOUR TEST RESULT

Write a message ✉

Integrated Kweery Dashboard

Prêt-à-LLOD Derilinx

are the children outpatient waiting list for cardiology?

OP WAITING LIST BY GROUP HOSPITAL
 The National Treatment Purchase Fund (NTPF) is responsible for the collection, collation and validation of Inpatient, Day Case and Outpatient waiting lists. The OP Waiting List report shows the total number of patients waiting across the various time bands, for a first appointment at a consultant and Consultant clinic. Each report lists the numbers waiting per Hospital in each time band.

IPDC WAITING LIST BY GROUP HOSPITAL
 The National Treatment Purchase Fund (NTPF) is responsible for the collection, collation and validation of Inpatient, Day Case and Outpatient waiting lists. The IPDC Waiting List report shows the total number of patients waiting across the various time bands, for treatment and Day case treatment in each Specialty (these numbers do not include GI Endoscopy, see separate reports). Each report lists the numbers waiting per Hospital in each time band.

Click a resource below to view data

Name	Description	Format	Source

Filters: **children** **outpatient** **waiting** **list** **cardiology** Custom Filter...

Right Click a dataset to add it for comparison

Filename	Description	Format	Source

Filters: **children** **outpatient** **waiting** **list** **cardiology** Custom Filter...

OP WAITING LIST BY GROUP HOSPITAL
 The National Treatment Purchase Fund (NTPF) is responsible for the collection, collation and validation of Inpatient, Day Case and Outpatient waiting lists. The OP Waiting List report shows the total number of patients waiting across the various time bands, for a first appointment at a consultant and Consultant clinic. Each report lists the numbers waiting per Hospital in each time band.

IPDC WAITING LIST BY GROUP HOSPITAL
 The National Treatment Purchase Fund (NTPF) is responsible for the collection, collation and validation of Inpatient, Day Case and Outpatient waiting lists. The IPDC Waiting List report shows the total number of patients waiting across the various time bands, for treatment and Day case treatment in each Specialty (these numbers do not include GI Endoscopy, see separate reports). Each report lists the numbers waiting per Hospital in each time band.

Click a resource below to view data

Name	Description	Format	Source
OP Waiting List by Group Hospital 2020	OP Waiting List by Group Hospital 2020	CSV	
OP Waiting List by Group Hospital 2019	OP Waiting List by Group Hospital 2019	CSV	
OP Waiting List By Group Hospital 2018	OP Waiting List By Group Hospital 2018	CSV	
OP Waiting List By Group Hospital 2017	OP Waiting List By Group Hospital 2017	CSV	
OP Waiting List By Group Hospital 2016	OP Waiting List By Group Hospital 2016	CSV	
OP Waiting List By Group Hospital 2015	OP Waiting List By Group Hospital 2015	CSV	
OP Waiting List By Group Hospital 2014	OP Waiting List By Group Hospital 2014	CSV	

Filters: **children** **outpatient** **waiting** **list** **cardiology** Custom Filter...

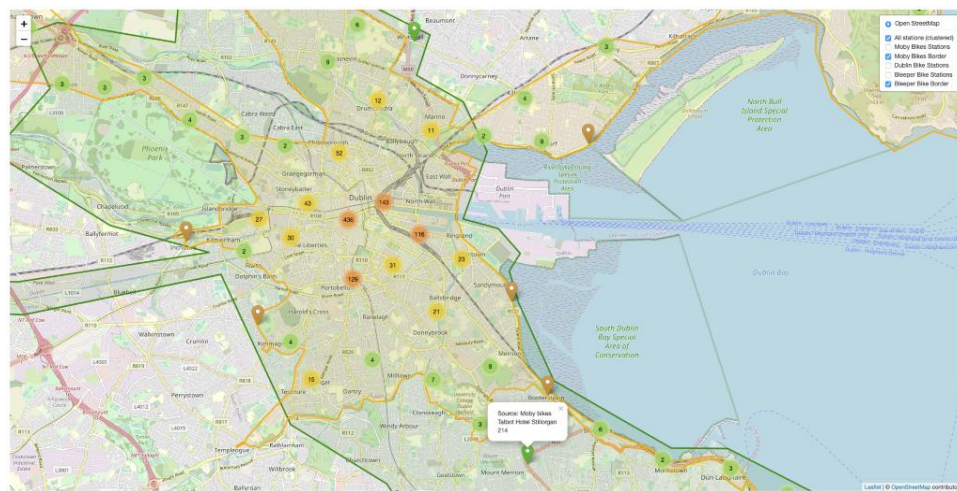
Right Click a dataset to add it for comparison

Filename	Description	Format	Source
IPDC Waiting List by Group Hospital 2020	IPDC Waiting List by Group Hospital 2020	CSV	
IPDC Waiting List By Group Hospital 2019	IPDC Waiting List By Group Hospital 2019	CSV	
IPDC Waiting List By Group Hospital 2018	IPDC Waiting List By Group Hospital 2018	CSV	
IPDC Waiting List By Group Hospital 2017	IPDC Waiting List By Group Hospital 2017	CSV	
IPDC Waiting List By Group Hospital 2016	IPDC Waiting List By Group Hospital 2016	CSV	
IPDC Waiting List By Group Hospital 2015	IPDC Waiting List By Group Hospital 2015	CSV	
IPDC Waiting List By Group Hospital 2014	IPDC Waiting List By Group Hospital 2014	CSV	

Filters: **children** **outpatient** **waiting** **list** **cardiology** Custom Filter...



OP Waiting List By Group Hospital 2016 CSV							IPDC Waiting List By Group Hospital 2016 CSV						
OP Waiting List By Group Hospital 2015 CSV							IPDC Waiting List By Group Hospital 2015 CSV						
OP Waiting List By Group Hospital 2014 CSV							IPDC Waiting List By Group Hospital 2014 CSV						
Filters: Children outpatient waiting list cardiology							Filters: Children outpatient waiting list cardiology						
Custom filter...							Custom filter...						
i	Group	Hospital_HPE	Hospital	Specialty_HPE1	Specialty	Adult	Id	Archive_Date	Hospital_Group	Hospital_HPE	Hospital_Name	Specialty_I	
00:00:00	Children's	0	Children's	601	Paediatric ENT	Child	1	2020-01-30T00:00:00	Children's	0	Children's	0	
00:00:00	Children's	0	Children's	601	Paediatric ENT	Child	2	2020-01-30T00:00:00	Children's	0	Children's	0	
00:00:00	Children's	0	Children's	1302	Paediatric	Child	3	2020-01-30T00:00:00	Children's	0	Children's	0	
00:00:00	Children's	0	Children's	1302	Paediatric	Child	4	2020-01-30T00:00:00	Children's	0	Children's	0	
00:00:00	Children's	0	Children's	1302	Paediatric	Child	5	2020-01-30T00:00:00	Children's	0	Children's	0	
00:00:00	Children's	0	Children's	1302	Paediatric	Child	6	2020-01-30T00:00:00	Children's	0	Children's	0	
00:00:00	Children's	0	Children's	1302	Paediatric	Child	7	2020-01-30T00:00:00	Children's	0	Children's	0	
00:00:00	Children's	0	Children's	1302	Paediatric	Child	8	2020-01-30T00:00:00	Children's	0	Children's	0	
00:00:00	Children's	0	Children's	1402	Paediatric	Child	9	2020-01-30T00:00:00	Children's	0	Children's	0	
00:00:00	Children's	0	Children's	1700	Ophthalmology	Child	10	2020-01-30T00:00:00	Children's	0	Children's	0	



Key next steps linked to Kweery will be to effectively refine and optimise the performance of the tool through real customers install and configuration (potentially including new data sources and languages).

The final version of this report, due month 36 (December 2021), will report on further development activities and the final results of the project.