preVIEW: from a fast prototype towards a sustainable semantic search system for central access to COVID-19 preprints

Lisa Langnickel (a, b)*, Johannes Darms (a)*, Roman Baum (a) and Juliane Fluck (a, c)
(a) ZB MED - Information Centre for Life Sciences, Cologne, Germany
(b) Graduate School DILS, Bielefeld Institute for Bioinformatics Infrastructure (BIBI), Faculty of Technology, Bielefeld University, Bielefeld, Germany
(c) University of Bonn, Bonn, Germany
* *these authors contributed equally to the work*

Abstract
The current COVID-19 pandemic emphasizes the use of so-called preprints - a type of publication that is not subject to peer review. Due to its global relevance, there is an immense number of COVID-19-related preprints every day. To help researchers find relevant information, we have developed the semantic search engine preVIEW which currently integrates preprints from seven different preprint servers. For semantic indexing, we implemented various text mining components to tag, for example, diseases or SARS-CoV-2 specific proteins. While the service initially served as a prototype developed together with users, we present a re-engineering towards a sustainable semantic search system, which was inevitable due to the continuously growing number of preprint publications. This enables easy reuse of the components and allows rapid adaptation of the service to further user needs.

Key words: *text mining; COVID-19; information retrieval.*

Introduction

The current COVID-19 pandemic poses new challenges for information providers due to the large volume of publications. The World Health Organization refers to the overabundance of correct and incorrect information during a disease outbreak as an *infodemic* (1). This makes it difficult for individual researchers to distinguish between research evidence and disinformation. According to (2), "more than 30,0000 of the COVID-19 articles published in 2020 were preprints". Since preprints are a form of publication that is not subject to peer review, a wide range of quality can be expected. On the other hand, publishing preprints allows for rapid dissemination of - potentially very valuable - information.

Since the outbreak of the COVID-19 pandemic, there has been a high demand for the immediate availability of COVID-19-related research results, making COVID-19-related preprints of great value. To assist researchers during this time, we developed the semantic search engine preVIEW, which specifically focuses on COVID-19-related preprints, and made it publicly available at an early stage of development at https://preview.zbmed.de (3). Currently, this search engine includes more than 36,000 preprints from seven different preprint servers.

Using specialized text mining components focused on SARS-CoV-2/COVID-19 related terms, which we have further extended over time, e.g., with mutant strain detection, we have developed a semantic search system and several additional features to support researchers.

While we started preVIEW as an *ad-hoc* prototype based on user needs, the long-term integration of preprint servers and text-mining-based semantic search engines into digital information services is essential. Therefore, we have invested in re-engineering the components to develop a modular software system that can be integrated into various library digital information services in the future.

Below we describe our architecture in more detail and provide a brief overview of preVIEW features.

Address for correspondence: Juliane Fluck, ZB MED - Information Centre for Life Sciences, Gleueler Str. 60, 50931 Cologne, Germany E-mail: fluck@zbmed.de

Journal of EAHIL 2021; Vol. 17 (3): 8-14 doi: 10.32384/jeahil17484

Methods

The re-engineered version of preVIEW has been developed with a microservices cloud-native architecture pattern in mind (4, 5). A focus of the re-engineering development is a clear separation between concerns. The refactoring resulted in three core microservices: *The User Interface Service*, the *Search Service*, and a *Terminology-Annotation Service*. In addition, two supporting services, the *Semantic Lookup Service* and an *Authentication Service*, are reused from the service landscape portfolio. This refers a set of (micro-) services that are shared within the department and are either services that cover cross-cutting topics such as authentication and authorization or can be used in different contexts. The basic idea of this approach is to shorten the time frame to release due to reuse. The services and their interconnection are shown in *Figure 1*. All services are encapsulated for deployment in an Open Container Initiative (OCI) (6) compatible image format and run in a compatible runtime environment (7). The *Terminology Annotation Service* focuses on retrieving preprints from various sources, harmonizing them into a common format, and enriching them with semantic concepts by applying terminology recognition using established named-entity recognition methods – including machine learning and rule-based approaches. A more detailed description of this service can be found here (3). In its current version, it annotates the following entity classes: disease names (mapped to Medical Subject Headings (MeSH) thesaurus concepts), human gene/protein names (mapped to HUGO Gene Nomenclature Committee (HGNC) concepts), SARS-CoV-2 specific proteins (mapped to Universal Protein Resource (UniProt) concepts if available) and, as a new entity type, SARS-CoV-2 virus variants classified as variants of concern or high concern according to (8). In order to detect those virus variant mentions, we developed a simple rule-based approach using manually curated terminology lists based on the information given in (8) and extended by further synonyms.

The *Search Service* and the *Index Service* form the central pillars of the semantic search engine. The semantic annotations provided by the *Terminology-Annotation service* are indexed by the Index Service, i.e., an Elasticsearch (9) instance. The *Search Service* was developed using Flask (10) and exposes and secures some functions of the Elsasticsearch index via a REST interface. To secure the interaction with the service, the Open ID Connect (OIDC) protocol [11] is used. The *Authentication Service*, a Keycloak instance (12), handles authentication and user management.

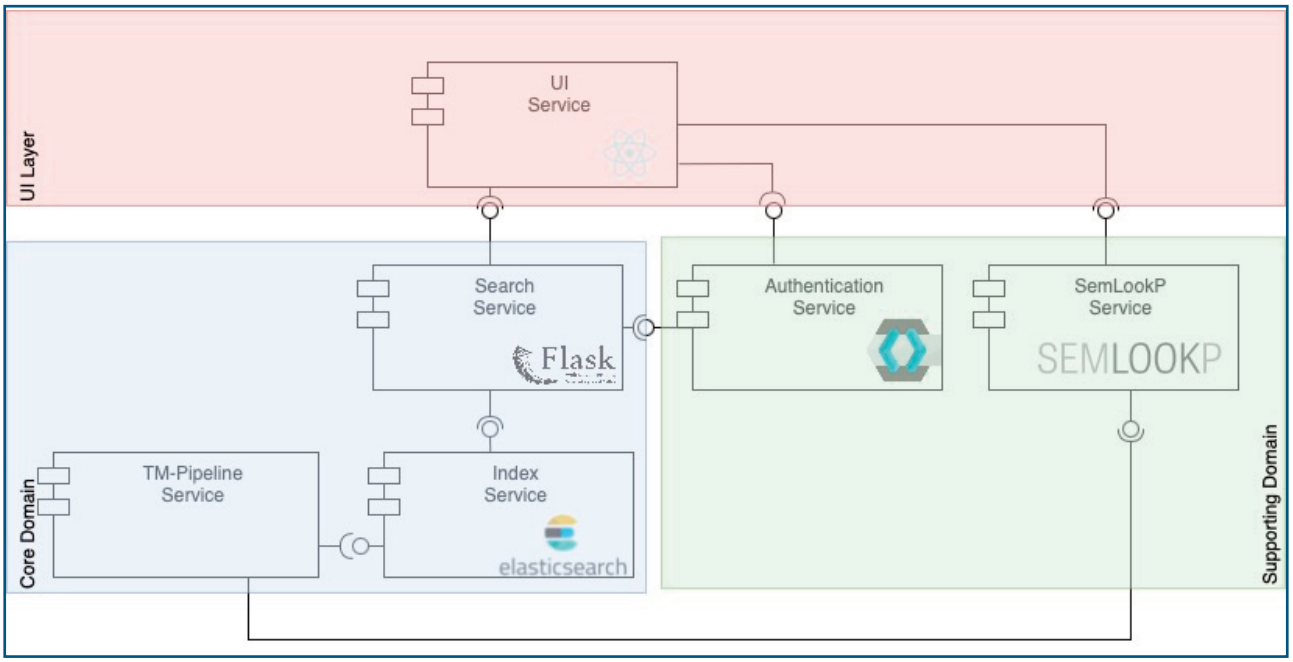

Fig. 1. *Component diagram of the preVIEW System.*

The *User Interface Service* uses the provided REST interface to create a suitable application. User authentication is integrated via OIDC and the *Authentication Service*. Information about semantic terms is retrieved via the *Semantic Lookup Service* using the provided REST API. The *Semantic Lookup Service* (release in preparation) builds on the Ontology Lookup Service (OLS) (13) and the Ontology Xref Service (OxO) (14) and provides a unified access to semantic resources.

The preVIEW *user interface* (UI) itself is developed using the React framework (15). The UI is a composition of loosely coupled components. Each component is developed and tested in isolation and is available through a component repository. To archive a similar look and feel to our application, all presentation components use the Elastic UI kit (16). Technically, each visible component consists of a representation component that handles the UI and an included "container" component that handles the business logic. This allows the UI to be migrated without changing the business logic. *Figure 2* shows a screenshot of the current preVIEW user interface with color-coded boxes representing some reusable UI components.

Results

The re-engineering focused on reusability without compromising the look and functionality of the service available at https://preview.zbmed.de. A brief description of the main features of preVIEW is given below.

The main page lists the available abstracts and displays relevant metadata such as title, authors, source (i.e., the preprint server), the date, and links to the original source (including full text). Examples are shown in *Figure 2* in the center, outlined in green. Abstracts can be expanded for a single document individually for all documents. Above the document list (outlined in blue) is a search bar that contains several functions, which are explained in more detail in the next section. In addition, we offer the following func-

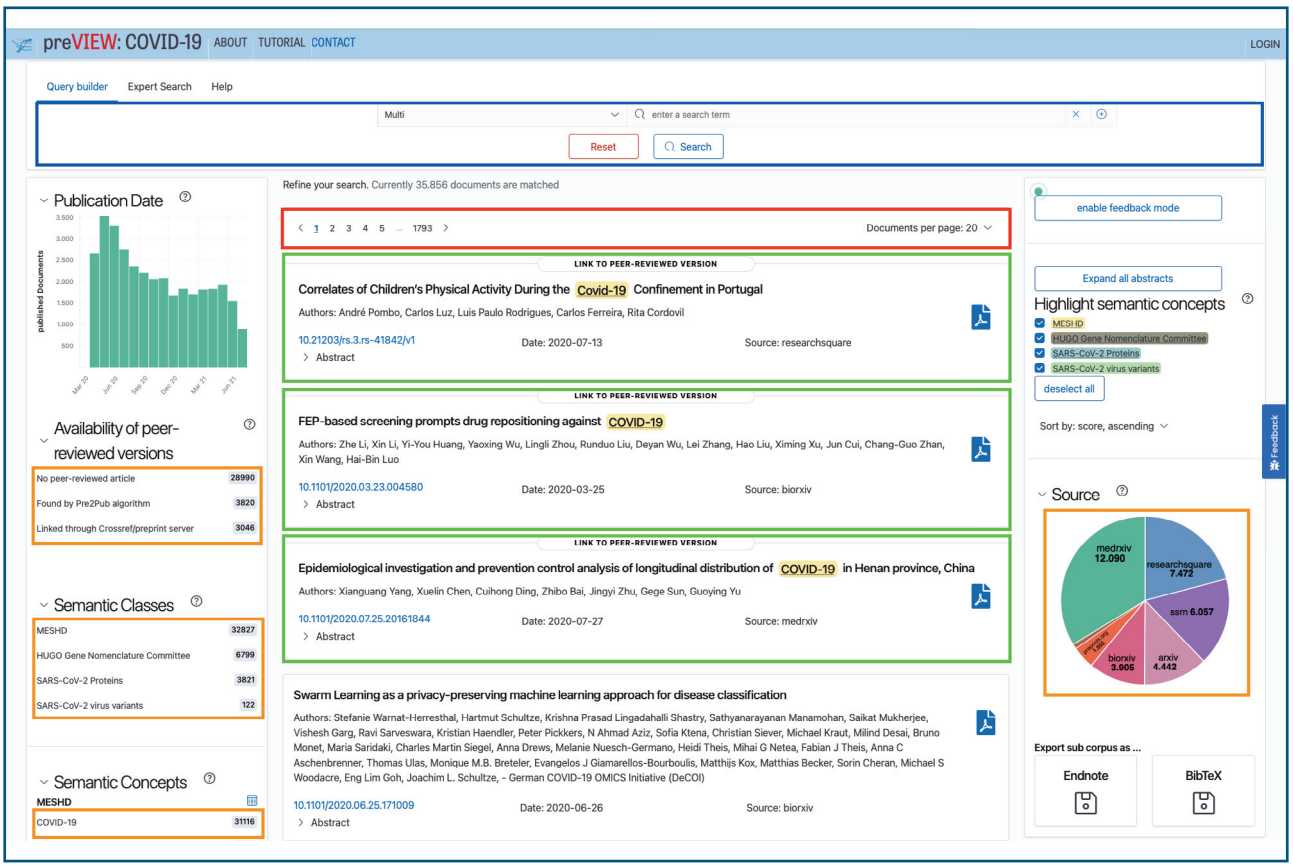

Fig. 2. *Screenshot of preVIEW with the major UI components highlighted.*

tions depicted in *Figure 2*: on the right side, you can select or deselect the highlighting of the found concepts; on the left side, there is an overview of the five most frequently occurring concepts for each terminology. These concepts can be easily added to the search query by clicking on them. In addition, you can also filter by semantic classes, i.e., you can, for example, select all preprints that contain at least one virus variant of interest. The bar chart on the left shows a distribution by publication date. Again, a specific month can be added to the search query by clicking on the corresponding bar. The documents found can be exported in currently two different formats, either EndNote or BibTeX. In the following, we would like to highlight two UI components of the application that are well suited for reuse in other library systems. First, the Search *Query component* and second, the *Semantic Information Widget component*.

The *Search Query Component* provides translation between different query languages and query construction methods. The primary purpose is to translate different query languages into the format used by our *Search Service*. The basis of this component is a composite pattern (17) where the branches are operations (AND, OR, NOT) and the leaves are predicates that allow defining a search on a specific field (e.g., title, author) or using a specific function (e.g. expand_concept). The composite is used to construct the abstract syntax tree (AST) of a search query. Several subcomponents are capable of constructing such an AST. First, an ANTLR-based (18) parser for a custom preVIEW-specific search query language. This allows advanced users to create more complex search queries in a domain-specific language. Second, a UI component, *Query Builder Component*, which allows interactive query building. Translations to or from other languages, like SRU/Z39.50 (19), can be easily added, as these additions are already provided for in

Fig. 3. *A Search Query shown in AST (I), ElasticSearch DSL(II), preVIEW DSL (III) and Query Builder Component (IV).*

the architecture. *Figure 3* shows a query displayed in the *Query Builder Component (IV)*, the custom query language of the preVIEW system (III), as an AST (I) and in the elastic search query syntax (II).

The *Semantic Information Widget Component* integrates semantic information provided by the *Semantic Lookup Service* into a handy widget. Preferred name, alternative spellings and a description of a semantic concept are displayed in a pop-up window. Cross-references - relationships to other semantic concepts - are also displayed, if available. Related hierarchical information about a concept is displayed as well.

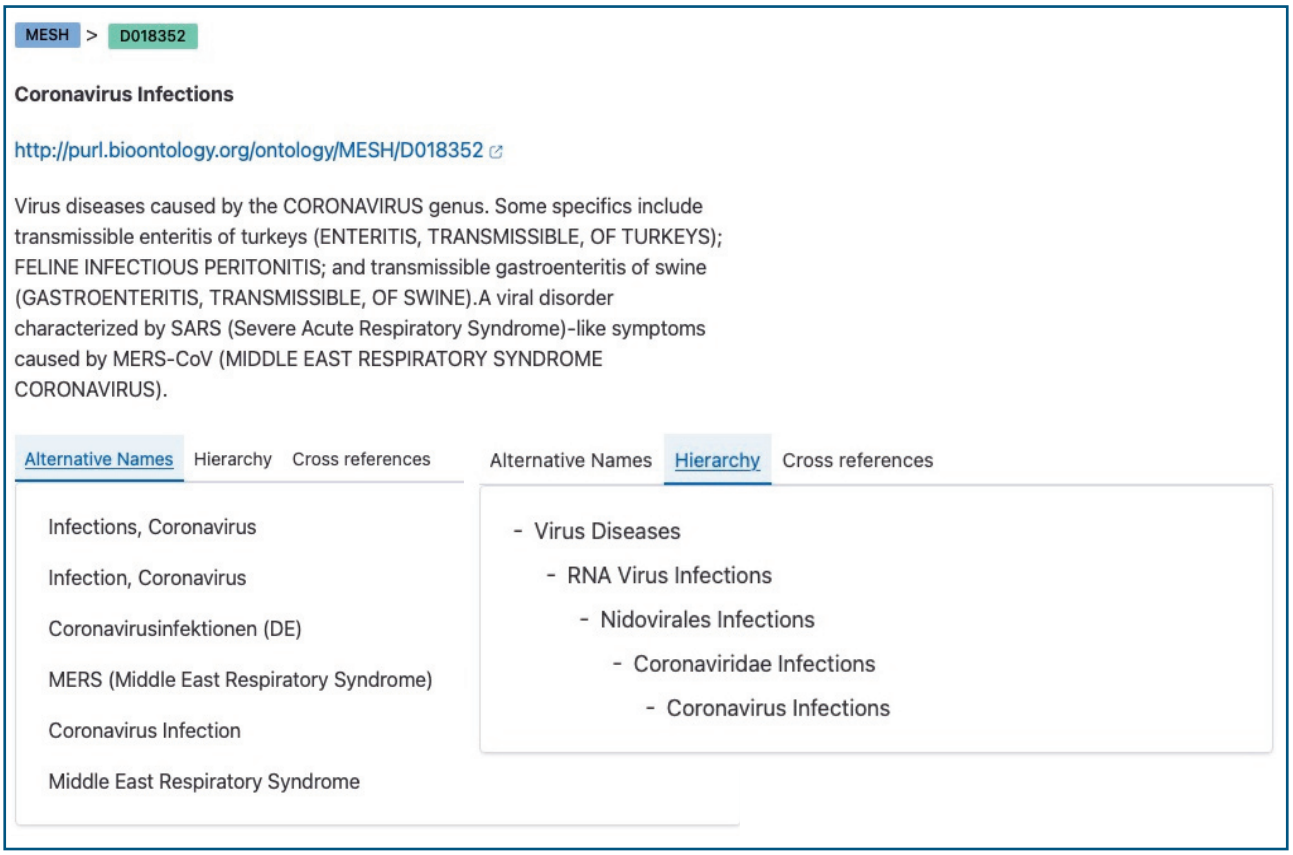

Fig. 4. *Semantic Information Widget Component. The Widget example shows information about the semantic concept "Coronavirus Infections" form MeSH. Within the widget, different information views of the semantic concept can be selected by the user via tab selection. The screenshot visualizes the tab "Alternative Names" on the left and the tab "Hierarchy" on the right.*

Within the application, the widget opens by clicking on a semantic annotation. *Figure 4* shows the widget.

Discussion

The semantic search engine preVIEW was developed due to the acute need during the current COVID-19 pandemic. As the number of so-called preprints has increased tremendously, there is a need for digital information services that facilitate the retrieval of these types of publications. Semantic search engines commonly used by information specialists, such as LIVIVO (20), do not integrate preprints. Therefore, we have developed a service that focuses exclusively on COVID-19 related preprints. Since preprints are not manually indexed by experts, text mining components were included to allow automatic indexing of relevant concept classes. Hence, information search and extraction must be facilitated by using automated methods based on machine learning and rules.

While the initial prototype was developed with a clear focus on rapid development and integration of various preprint resources, the architecture was already reaching its limits. This was partly due to the ever-increasing number of preprints, which affected the performance of the service, and partly due to increasingly complex application requirements, which were costly to implement. The architecture described in this paper emphasizes components that are reusable and can be shared among different services.

The development of the preVIEW system with loosely coupled, reusable components occur at the microservices level, in the backend code, i.e., the services that are invisible to the user, and at the user interface level, the UI components. One advantage that arises at both levels is that a component/service can be developed and tested in isolation. This allows a developer to focus on a single task, and the components can be viewed as LEGO bricks that are independent but have compati-

ble connection interfaces. Thus, they can be freely combined to build different application and software systems; similar to the LEGO system. However, this freedom also has the disadvantage of complex dependencies, as components may depend on each other and require specific versions. Package managers (such as helm, npm) are available to alleviate the problem, but compatibility between versions must be maintained or changes must be communicated efficiently, as package managers are no help in this regard.

Although developing a software system like the preVIEW application based on loosely coupled microservices and front-end components presents some challenges, the benefits of isolating concerns and the resulting independence are worth the effort. This is because the components developed for the preVIEW application can and are already being reused. For example, the *Semantic Information Widget Component* is being reused as part of the German Central Health Study Center COVID-19 (21). A service that bundles information about clinical, epidemiological and public health studies in Germany related to COVID-19. A component for highlighting semantic annotations in texts will also be integrated into another internal project.

Conclusions

We presented the re-engineering of our semantic search engine preVIEW COVID-19, which was initially developed as a prototype due to the acute need during the current pandemic. The transformation into a sustainable service - consisting of several microservices - not only resulted in a fast system, but also enables rapid action in the future. It simplifies the adaptation of the services to user needs and enables the reuse of components in other digital information services. With the set of services and components presented, we are also able to rapidly develop new prototype services in specialized areas to help researchers find relevant information to advance science.

Acknowledgements

This work was done as part of the NFDI4Health Task Force COVID-19 (www.nfdi4health.de). We gratefully acknowledge the financial support of the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – Project Number 451265285. In addition, we would like to thank Mhd. Hisham Hassoun for his contributions in re-engineering.

The authors of this paper received the award for best oral presentation by a first-time attendee at the EAHIL Virtual Conference "Crossing the Bridge: New Challenges, New Opportunities", 05-08 July 2021, Istanbul, Turkey.

Submitted on invitation.
Accepted on 6 September 2021.

REFERENCES

1. WHO. Infodemic [Internet]. [cited 2021 Sep 2]. Available from: https://www.who.int/westernpacific/health-topics/infodemic
2. Else H. How a torrent of COVID science changed research publishing — in seven charts. Nature. 2020 Dec 16;588(7839):553-553.
3. Langnickel L, Baum R, Darms J, Madan S, Fluck J. COVID-19 preVIEW: Semantic Search to Explore COVID-19 Research Preprints. Stud Health Technol Inform. 2021 May 27;281:78-82.
4. Kratzke N, Quint P-C. Understanding cloud-native applications after 10 years of cloud computing - A systematic mapping study. Journal of Systems and Software. 2017 Apr;126:1-16.
5. Alshuqayran N, Ali N, Evans R. A Systematic Mapping Study in Microservice Architecture. In: 2016 IEEE 9th International Conference on Service-Oriented Computing and Applications (SOCA) [Internet]. Macau, China: IEEE; 2016 [cited 2021 Sep 2]. p. 44-51. Available from: http://ieeexplore.ieee.org/document/7796008/
6. Open Container Initiative. Open Container Initiative - Open Container Initiative [Internet]. [cited 2021 Sep 2]. Available from: https://opencontainers.org/
7. Merkel D, others. Docker: lightweight linux containers for consistent development and deployment. Linux journal. 2014;2014(239):2.
8. CDC. Coronavirus Disease 2019 (COVID-19) [Internet]. Centers for Disease Control and Prevention. 2020 [cited 2021 Sep 3]. Available from: https://www.cdc.gov/coronavirus/2019-ncov/variants/variant-info.html

9. Elasticsearch B.V. Elasticsearch: The Official Distributed Search & Analytics Engine [Internet]. Elastic. [cited 2021 Sep 3]. Available from: https://www.elastic.co/elasticsearch
10. Pallets Organisation. Welcome to Flask — Flask Documentation (2.0.x) [Internet]. [cited 2021 Sep 2]. Available from: https://flask.palletsprojects.com/en/2.0.x/
11. OpenID Foundation. OpenID Connect | OpenID [Internet]. 2011 [cited 2021 Sep 2]. Available from: https://openid.net/connect/
12. Red Hat, Inc. Keycloak [Internet]. [cited 2021 Sep 2]. Available from: https://www.keycloak.org/
13. Côté RG, Jones P, Apweiler R, Hermjakob H. The Ontology Lookup Service, a lightweight cross-platform tool for controlled vocabulary queries. BMC Bioinformatics. 2006 Feb 28;7(1):97.
14. Jupp S, Liener T, Sarntivijai S, Vrousgou O, Burdett T, Parkinson HE. OxO-A Gravy of Ontology Mapping Extracts. In: ICBO. 2017.
15. Facebook Inc. React – A JavaScript library for building user interfaces [Internet]. [cited 2021 Sep 2]. Available from: https://reactjs.org/
16. Elasticsearch B.V. Elastic UI [Internet]. [cited 2021 Sep 2]. Available from: hhttps://elastic.github.io/eui
17. Gamma E, editor. Design patterns: elements of reusable object-oriented software. Reading, Mass: Addison-Wesley; 1995. 395 p. (Addison-Wesley professional computing series).
18. Parr TJ, Quong RW. ANTLR: A predicated-LL (k) parser generator. Software: Practice and Experience. 1995;25(7):789–810.
19. National Information Standards Organization (É.-U.), American National Standards Institute. Information retrieval (Z39.50): application service definition and protocol specification : an American national standard [Internet]. 2003 [cited 2021 Sep 2]. Available from: https://www.loc.gov/z3950/agency/Z39-50-2003.pdf
20. Müller B, Poley C, Pössel J, Hagelstein A, Gübitz T. LIVIVO – the Vertical Search Engine for Life Sciences. Datenbank Spektrum. 2017 Mar 1;17(1):29-34.
21. Schmidt CO, Darms J, Shutsko A, Löbe M, Nagrani R, Seifert B, et al. Facilitating Study and Item Level Browsing for Clinical and Epidemiological COVID-19 Studies. Studies in Health Technology and Informatics. 2021;281:794-8.