

# *EduSearch* : Web Pencarian Cerdas Berbasis Semantik untuk Mencari Data Seluruh Sekolah Formal di Kota Medan

<sup>1</sup>Chairil Umri, <sup>2</sup>Annisa Fadhillah Pulungan, <sup>3</sup>Aulia Halimatusyaddiah, <sup>4</sup>Irene Elisabeth Sinaga, <sup>5</sup>Leo Alfonso Tarigan, <sup>6</sup>Nurul Aini, <sup>7</sup>Said Muhammad Mazaya,  
<sup>1, 2, 3, 4, 5, 6</sup>Universitas Sumatera Utara, Medan, Indonesia  
<sup>7</sup>Universitas Battuta, Medan, Indonesia  
[lirmuliriahc@gmail.com](mailto:lirmuliriahc@gmail.com), [annisafpulungan@usu.ac.id](mailto:annisafpulungan@usu.ac.id),  
[auliahalimatusyaddiah@students.usu.ac.id](mailto:auliahalimatusyaddiah@students.usu.ac.id), [ireneelisabethsinaga@gmail.com](mailto:ireneelisabethsinaga@gmail.com),  
[leoalfonso2003@gmail.com](mailto:leoalfonso2003@gmail.com), [nurulaini22@students.usu.ac.id](mailto:nurulaini22@students.usu.ac.id),  
[saidmuhammad@students.usu.ac.id](mailto:saidmuhammad@students.usu.ac.id)

## ABSTRAK

Dengan perkembangan teknologi informasi pada saat ini, pencarian informasi mengenai banyak hal dapat lebih mudah dan efisien. Salah satu informasi yang sering dibutuhkan adalah informasi tentang sekolah pendidikan formal yang terdapat di daerah sekitar Kota Medan. Umumnya Informasi sekolah yang dibutuhkan berdasarkan domisili pengguna yang berada di sekitar Kota Medan. Ada banyak sekolah formal yang terdapat di Kota Medan yang membuat para orang tua yang ingin menyekolahkan anaknya bingung dalam mencari informasi sekolah di sekitar domisili mereka. Sehingga perlu membuat web berbasis teknologi semantik untuk mempermudah pengguna dalam melakukan pencarian informasi pada web tersebut. Web Pencarian ini kami sebut dengan *EduSearch*. Beberapa teknologi yang digunakan dalam pembuatan web semantik ini adalah SPARQL (*Simple Protocol and RDF Query Language*), RDF (*Resource Description Framework*), OWL, *Open Street Map*, dan Selenium Web Driver. Hasil dari pembuatan web berbasis semantik ini menghasilkan beberapa fitur yaitu fitur pencarian berdasarkan input *user*, fitur filter berdasarkan kecamatan, fitur filter berdasarkan jenjang pendidikan, fitur *open street map*, dan fitur detail sekolah. Fitur-fitur ini akan memudahkan pengguna dalam melakukan pencarian menggunakan web semantik ini dan mendapatkan informasi yang dibutuhkan pengguna. Dan berdasarkan hasil uji fungsional yang telah kami lakukan mendapatkan hasil yang baik pada fungsionalitas fitur-fitur yang kami rancang pada pencarian ini.

**Kata Kunci:** Mesin Pencarian, Web semantik, RDF, *Open Graph Protocol*, Apache Jena Fuseki, *Open Street Map*.

## PENDAHULUAN

Dalam era informasi digital yang terus berkembang, akses yang efisien terhadap seluruh data sekolah formal memiliki peran yang vital dalam mendukung masyarakat dalam mengakses informasi yang relevan dan penting. Di Medan, keragaman sekolah formal dari tingkat dasar hingga menengah atas menjadi landasan bagi perkembangan pendidikan di wilayah tersebut. Meskipun informasi tentang institusi-institusi ini umumnya tersedia di banyak platform lain, tetapi kami membangun sebuah akses yang lebih mudah, terstruktur, dan terintegrasi yang signifikan bagi masyarakat yang membutuhkan informasi terkait segala elemen data dari sekolah formal yang ada di Medan.

Dalam konteks ini, pengembangan sebuah platform “Web Pencarian Cerdas Berbasis Semantik Untuk Mencari Data Seluruh sekolah formal di Medan” menjadi sebuah inovasi spesifik untuk memenuhi kebutuhan akan akses yang efisien dan terperinci terhadap data institusi akademik di Medan. Penerapan sifat semantik dalam platform ini memungkinkan penyediaan data yang

terstruktur secara holistik, memperkenalkan pemahaman yang lebih dalam dan terkait antar berbagai entitas informasi yang ada. Dengan demikian, tujuan dari jurnal ini adalah untuk merancang dan mengimplementasikan sebuah sistem pencarian cerdas berbasis semantik yang secara khusus menyajikan seluruh data institusi akademik dari jenjang pendidikan dasar hingga menengah atas.

Tujuan utama dari penelitian ini adalah untuk menciptakan sebuah platform yang memungkinkan akses yang mudah, efisien, dan dapat dengan mudah dipahami oleh semua kalangan pengguna terhadap informasi dari berbagai sekolah formal. Sistem ini diharapkan dapat memfasilitasi pencarian yang terintegrasi, memberikan hasil yang relevan, dan meningkatkan aksesibilitas data bagi masyarakat luas. Dengan adanya platform ini, diharapkan dapat memudahkan orang-orang untuk menemukan informasi yang diperlukan mengenai sekolah formal di Medan, membantu mereka membuat keputusan yang lebih terinformasi terkait pendidikan diri mereka dan anak-anak mereka atau memperluas wawasan mengenai sekolah formal di sekitar mereka.

Menurut (Blobe, Engel, & Pharowe, 2006), prinsip-prinsip penting dalam pengembangan sistem informasi pendidikan, yang terkait dengan penekanan pada penggunaan standar dan konsep semantik untuk memastikan data dari berbagai sekolah formal dapat berinteraksi dengan baik melalui penerapan sifat interoperabilitas dalam sistem informasi pendidikan tersebut. Hal ini bertujuan untuk mempermudah pengguna dalam mendapatkan informasi yang relevan, cepat, dan akurat sesuai dengan kebutuhan.

## TINJAUAN PUSTAKA

### Mesin Pencarian

Menurut (Sadeeq, Zeebaree, & Author, 2021) mesin pencarian merupakan teknologi cerdas sebagai alat utama yang membantu pengguna dalam menjelajahi internet. Mesin pencarian memungkinkan pengguna untuk menemukan informasi dengan cepat di antara miliaran halaman web yang ada di internet. Terdapat berbagai macam jenis mesin pencarian sesuai dengan kebutuhan pengguna seperti Google sebagai mesin pencari umum serta ada beberapa mesin pencarian yang lebih spesifik seperti mesin pencari gambar, mesin pencari jurnal ilmiah, mesin pencari berita, dan banyak lagi.

Mesin pencarian mengumpulkan informasi dari berbagai sumber di internet melalui proses yang disebut "pengindeksan" yang melibatkan penjelajahan web oleh *bot* (atau *spider*) yang mengumpulkan data dari situs web dan menyimpannya dalam *database* mereka. Data yang dikumpulkan kemudian dianalisis dan diorganisasi sehingga dapat ditemukan dengan cepat saat pengguna mencari menggunakan mesin pencarian cerdas tersebut. Algoritma pencarian pada mesin pencarian berperan sebagai "otak" di balik mesin pencari untuk menentukan urutan hasil yang paling relevan untuk kueri pengguna.

### SPARQL

Menurut (Himawan, Harjanti, Supriati, & Setiyani, 2020) mendefinisikan SPARQL (*Simple Protocol and RDF Query Language*) merupakan bahasa query yang direkomendasikan oleh W3C (*World Wide Web Consortium*). Bahasa ini digunakan untuk mengambil informasi dari graph RDF dan berperan sebagai standar protokol yang memungkinkan akses sumber daya pada *Semantic Web*. SPARQL memungkinkan web untuk mengambil nilai dari data yang terstruktur dan semi- terstruktur. Hasil dari *query* SPARQL dapat dikembalikan dalam berbagai format, seperti XML, JSON, RDF, dan HTML.

### RDF (*Resource Description Framework*)

Menurut (Himawan et al., 2020) mendefinisikan RDF adalah bahasa yang menghasilkan standar metadata untuk sumber daya di halaman web. Metadata ini mendefinisikan informasi yang menjadi "pengetahuan" dalam Semantic Web. RDF menggunakan model data triples yang terdiri dari *subject*, *predicate*, dan *object*. *Subject* adalah resource yang diidentifikasi dengan URI, *predicate* adalah atribut yang menghubungkan resource dengan *object*, dan *object* menyimpan nilai atribut. Sebagai contoh, *resource* dalam Semantic Web dapat berupa halaman web, gambar, video, produk, organisasi, atau layanan yang dapat diidentifikasi dengan URI.

RDF menggunakan teknologi berbasis XML dengan URI sebagai pengidentifikasi sumber daya pada halaman web. RDF terdiri dari beberapa bahasa yang melengkapi satu sama lain, termasuk *RDF Schema* (RDFS), SPARQL sebagai bahasa query untuk RDF, dan OWL yang menggunakan *Ontology* berbasis RDF dan RDFS. Representasi *model graph* RDF dapat diwujudkan dalam bentuk XML.

### OWL (Web Ontology Language)

Menurut (Himawan et al., 2020) mendefinisikan *Ontology*, dalam konteks kecerdasan tiruan, adalah representasi konsep teoritis mengenai arti objek dalam suatu domain. *Ontology* menjelaskan domain tersebut dan memungkinkan data pada halaman web untuk dideskripsikan dengan lebih baik, menghasilkan hubungan yang jelas antar data. Dengan menggunakan *Ontology*, Semantic Web dapat menyimpan pengetahuan dan menambahkan metadata pada konten halaman web. Struktur *Ontology* terdiri dari komponen-komponen seperti konsep (termasuk *class*, *object*, dan kategori), relasi (mendeskripsikan objek-objek dalam *Ontology*), fungsi (relasi antar *class-class*), taksonomi (mengatur pengetahuan dengan generalisasi), aksioma (memodelkan kalimat yang selalu benar), dan instances (mewakili elemen dalam suatu domain, baik nyata maupun abstrak).

Model OWL memiliki tiga jenis:

1. *OWL Full*: Kompatibel sepenuhnya dengan RDF baik secara sintaks maupun semantik. Menggunakan RDF Schema sebagai dokumen OWL.
2. *OWL Description Logic (DL)*: Sub-bahasa dari *OWL Full* dengan kompleksitas lebih rendah dan tingkat kompatibilitas yang lebih rendah terhadap RDF.
3. *OWL Lite*: Versi OWL dengan batasan yang lebih ketat, lebih mudah dipahami dan digunakan oleh pengguna serta lebih mudah untuk dikembangkan oleh para programmer.

### Open Street Map

Menurut (*Mapping and the Citizen Sensor*, 2017) *Open Street Map* (OSM) merupakan proyek yang bertujuan untuk menciptakan map yang kompleks dan gratis, OSM didirikan pada tahun 2004, dalam beberapa tahun terakhir OSM telah berkembang pesat, OSM menghasilkan data spasial (data yang menyimpan informasi tentang lokasi atau letak geografis suatu objek di bumi) dimana banyak aplikasi yang menggunakannya untuk proyek mereka, selain itu OSM juga digunakan dalam penelitian, pengembangan perangkat lunak dan pengambilan putusan melalui data spasial yang mereka miliki. OSM menggunakan proses yang bersifat kolaboratif dan terbuka untuk mengumpulkan dan mengelola data spasial, proses ini melibatkan pengguna yang mengedit data, mengidentifikasi kesalahan, dan menggambarkan perubahan peta. OSM memberikan keleluasaan kepada pengguna untuk mengakses dan mengedit data geografis. OSM dapat digunakan untuk melakukan pemetaan lokasi serta membantu dalam navigasi dengan memberikan panduan arah/petunjuk jalan.

### Selenium Web Driver

Selenium adalah *tools auto testing* yang digunakan untuk mengotomatisasi tes aplikasi web yang dilakukan pada *browser*. Selenium akan melakukan validasi web *apps* pada berbagai *browser*

dan platform. Menurut (Nyamathulla, Ratnababu, Sultana Shaik, Lakshmi, & Professor, 2021) selenium adalah setelan pengujian otomatis sumber terbuka yang membantu dalam pengujian web otomatis aplikasi. *Tools auto testing* ini hanya dirancang untuk otomatisasi testing pada *Web Apps*, selain penggunaan untuk testing web *apps* maka tidak bisa digunakan. bahasa seperti C#, Java, Perl, PHP, Python, dan Ruby adalah bahasa favorit yang digunakan untuk Selenium. Namun beberapa penguji mulai lebih memilih 2 bahasa Python dan Ruby karena fitur Python dan Ruby yang mudah digunakan. Selenium ini diciptakan oleh Jason Huggins pada tahun 2004.

### **Apache Jena Fuseki**

Apache Jena Fuseki merupakan komponen dari proyek Apache Jena. suatu kerangka kerja sumber terbuka yang digunakan untuk memproses data semantik di web, Jena memungkinkan para pengembang untuk menciptakan serta mengelola data RDF (*Resource Description Framework*), yang digunakan untuk mewakili pengetahuan dalam bentuk graf berlabel. Sementara itu, Fuseki berperan sebagai layanan server SPARQL yang terhubung dengan Apache Jena. Apache jena dapat beroperasi sebagai layanan manajemen struktur dan seringkali diterapkan untuk membangun aplikasi web Semantik dan aplikasi data terkait (Chokshi & Panchal, 2022).

### **Open Graph Protocol**

Menurut (Halpin, 2023) *Open Graph Protocol* (OGP) adalah suatu protokol yang diungkapkan oleh David Recordon sebagai respons terhadap ketertarikan mendadak Facebook terhadap W3C dan *Semantic Web*. OGP memungkinkan tombol "Suka" baru dari Facebook untuk meluas ke luar batasan platform Facebook dan mencakup lebih luas di Web. Dengan OGP, siapa pun dapat dengan mudah menambahkan penandaan pada produk, film, dan item lainnya menggunakan serangkaian hubungan yang disederhanakan dalam gaya RDFa pada metadata setiap halaman web. Hal ini memfasilitasi integrasi informasi semantik dan memungkinkan objek-objek tersebut diakses dan diidentifikasi dengan lebih baik di seluruh lingkungan web yang lebih luas.

### **RDFLib**

Menurut (Brecher et al., 2021) RDFLib adalah suatu perpustakaan (library) dalam bahasa pemrograman Python yang digunakan untuk memanipulasi dan bekerja dengan data RDF (*Resource Description Framework*). Dalam konteks proyek ini, RDFLib digunakan sebagai penghubung (connector) ke ontologi karena proyek ini menggunakan kerangka kerja Python.

RDFLib memungkinkan pengguna untuk membuat, memodifikasi, dan mengonsumsi data RDF. Ontologi, yang merupakan representasi formal dari pengetahuan dalam suatu domain tertentu, dapat dihubungkan dan dimanipulasi menggunakan RDFLib. Proyek ini mungkin telah memanfaatkan fitur-fitur RDFLib untuk mengintegrasikan data semantik ke dalam aplikasinya.

### **JSON-LD**

Menurut (Charpenay, Käbisch, & Kosch, 2018) Transformasi dari JSON ke RDF bergantung pada mekanisme yang ditentukan oleh standar terbaru dari W3C yang disebut *JSON for Linked Data* (JSON-LD). JSON-LD bertujuan untuk menentukan transformasi melalui suatu konteks, yang merupakan objek JSON yang memberikan pemetaan dari istilah JSON sembarang ke RDF IRIs. Dalam kasus kami, jika istilah-istilah JSON untuk suatu standar tertentu diketahui sebelumnya, seseorang dapat membuat konteks JSON-LD untuk memetakan mereka ke IRIs yang dihasilkan dan dengan demikian mendapatkan representasi RDF dari dokumen JSON asli.

Dalam konteks ini, "JSON for Linked Data" atau JSON-LD merupakan suatu standar W3C yang mendefinisikan cara mengaitkan data semantik dalam format JSON dengan RDF, memungkinkan integrasi data semantik dengan mudah. JSON-LD menggunakan konteks untuk memberikan makna terhadap data JSON, sehingga dapat dihubungkan dengan RDF.

### Penelitian Terdahulu

Beberapa penelitian terdahulu yang relevan dengan tema *paper* yang merupakan acuan dalam pembuatan web pencarian cerdas "EduSearch" antara lain yaitu sebagai berikut :

Menurut (Li & Yang, 2008) dalam *paper* yang berjudul "A Semantic Search Engine for Spatial Web Portals" data geospasial menjadi sumber daya yang penting untuk berbagai aplikasi ilmu bumi. Portal Web Spasial (SWP) merupakan platform yang memfasilitasi pembagian, pertukaran, dan interoperabilitas data geospasial, metadata, dan layanan web. Namun, mesin pencari di SWP sebagian besar masih mengandalkan pencocokan kata kunci, yang seringkali tidak dapat memenuhi kebutuhan pengguna yang memiliki pengetahuan ilmu bumi yang terbatas atau permintaan yang kompleks. Hal ini berkaitan dengan tujuan pengerjaan tugas besar yaitu memudahkan pengguna melalui pencocokan kata kunci yang diinginkan untuk mendapatkan informasi mengenai sekolah formal di Medan secara efisien dan fleksibel.

Selanjutnya, (Wang, Sun, & Liang, 2020) dalam *paper* yang berjudul "JSON-LD Based Web API Semantic Annotation Considering Distributed Knowledge" mengusulkan sebuah pendekatan anotasi semantik Web API berbasis JSON-LD (JWASA), yang dapat meningkatkan kemampuan komposisi otomatis Web API yang telah dianotasi secara semantik. Komposisi otomatis Web API dapat mempercepat proses pengembangan sistem perangkat lunak. Namun, anotasi semantik Web API menghadapi berbagai kesulitan di dunia nyata, karena distribusi dan keragaman fungsi Web API, seperti format deskripsi API yang tidak seragam, hasil respons dengan struktur yang kompleks, kekurangan ontologi domain bisnis, konflik semantik antara pengetahuan yang tersebar, dan sebagainya. Kelebihan – kelebihan tersebut yang memberi pemikiran dalam pengembangan web pencarian yang kami kembangkan dengan metode pengambilan data berbasis JSON-LD.

Selain itu, menurut (Niemann, Mochol, & Tolksdorf, 2008) dalam *paper* yang berjudul "Enhancing Hotel Search with Semantic Web Technologies" membahas tentang tantangan yang dihadapi oleh penyedia layanan pariwisata dalam menawarkan produk yang sesuai dengan kebutuhan dan preferensi pelanggan yang beragam dan berubah-ubah. (Niemann et al., 2008) mengusulkan sebuah kerangka kerja yang menggunakan teknologi web semantik untuk meningkatkan pencarian dan penilaian hotel untuk pelanggan bisnis dengan mempertimbangkan rasio harga/manfaat dan peringkat hasil sesuai dengan kebutuhan pelanggan. Kerangka kerja ini juga menyediakan metode untuk memodelkan pengetahuan spesifik domain dan mengintegrasikan berbagai sumber data yang heterogen. Teknologi web semantik juga memungkinkan penyedia layanan pariwisata untuk menyesuaikan produk dan jasa mereka dengan kebutuhan dan preferensi pelanggan yang beragam dan berubah-ubah.

Serta menurut (Josi & Andretti Abdillah, n.d.) dalam *paper* yang berjudul "Penerapan Teknik Web Scraping Pada Mesin Pencarian Artikel Ilmiah" yang menjelaskan tentang search engine, yaitu sistem yang mengumpulkan informasi dari web dengan menggunakan program *bot* atau web *crawler*. (Josi & Andretti Abdillah, n.d.) juga menjelaskan tentang web *scraping*, yaitu teknik untuk menggali informasi dari situs web dengan mempelajari dokumen HTML dan Teknik navigasi dari situs web yang dituju. Web semantik dapat membantu web *scraping* dengan menyediakan standar untuk merepresentasikan dan mengakses data, seperti RDF, OWL, dan

SPARQL. Web semantik dapat memudahkan web *scraping* untuk menemukan, menggabungkan, dan memanfaatkan informasi dari berbagai sumber data yang heterogen. Hal ini berhubungan dengan metode pembangunan web pencarian cerdas yang kami lakukan yaitu *scraping* data dari data pokok pendidikan Kementerian Pendidikan dan Kebudayaan menggunakan teknologi Selenium dari *Java Script*.

## METODE PENELITIAN

### Data

Web Pencarian Cerdas Berbasis Semantik untuk Mencari Data Seluruh Sekolah Formal di Medan terdiri dari 2.471 data dari seluruh tingkat pendidikan formal. Sumber data berasal dari Data Pokok Pendidikan Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi Indonesia yang diambil dengan metode *scraping* menggunakan Selenium Web *Driver*. Data yang diambil kemudian diimplementasikan dalam bentuk informasi terformat dalam web pencarian yang dibuat.

### Arsitektur Umum

Web Pencarian Cerdas Berbasis Semantik untuk Mencari Data Seluruh Sekolah Formal di Medan yang kami buat dapat berelasi dengan sistem lain. Beberapa sistem yang dapat berelasi yaitu:

1. *Selenium Web Driver dengan JSONLD*

Selenium Web *Driver* adalah sebuah teknologi untuk mengambil data sekolah formal di medan melalui web dapodik. Dengan Selenium Web *Driver*, kita dapat mengotomatisasi tindakan-tindakan yang biasanya dilakukan oleh pengguna, seperti menyimpan informasi web dan menavigasi antar halaman web. Selenium akan menghasilkan file JSON yang akan dikonversi menjadi JSONLD.SPARQL *Endpoint Internal* (Fuseki). Web pencarian cerdas berbasis semantik untuk mencari data seluruh sekolah formal di Medan ini menggunakan SPARQL *Endpoint Internal* berupa Apache Jena Fuseki sebagai database dari *Ontology* yang kami gunakan.

2. *SPARQL Endpoint External* (<https://dapo.kemdikbud.go.id/sp>)

Aplikasi Pencarian sekolah formal di Medan kami menggunakan laman data pokok pendidikan Kemendikbud sebagai SPARQL *Endpoint External* yang bertujuan untuk menghubungkan data sekolah dari dataset yang kami buat dengan dataset yang berisi data sekolah lainnya dengan jumlah yang lebih banyak yang terdapat pada laman data pokok pendidikan kemendikbud. Data yang kami ambil berupa nama sekolah, NPSN, alamat sekolah, status sekolah, akreditasi sekolah, tanggal pendirian sekolah, link sekolah, serta garis bujur dan koordinat sekolah. Data diambil dalam bentuk JSON yang kemudian diubah menjadi JSONLD di Apache Jena Fuseki.

3. *Google Map dengan Leaflet.js*

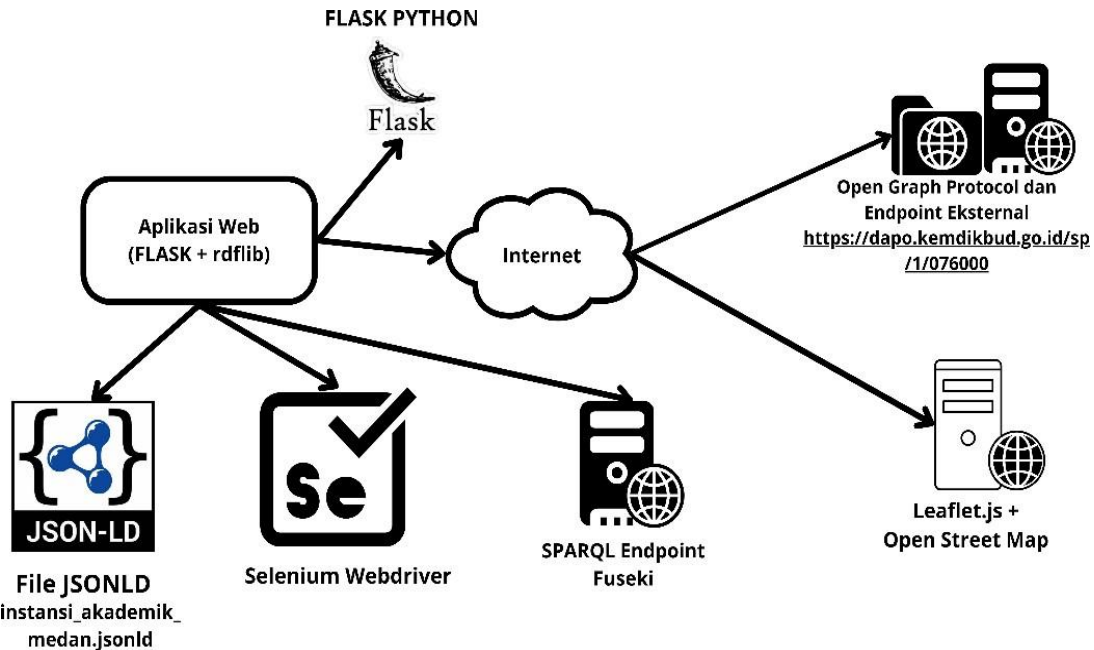
Web semantik ini dikembangkan menggunakan Leaflet.js untuk menyajikan data sekolah di Medan dalam bentuk peta interaktif. Leaflet.js memungkinkan kami dengan mudah menampilkan lokasi sekolah formal di Medan dengan tampilan yang bersih dan ramah pengguna. Pengguna dapat menjelajahi peta, melihat lokasi sekolah, dan mendapatkan informasi tambahan dengan interaktif. Leaflet.js memberikan pengalaman pengguna yang lebih dinamis dan memudahkan navigasi visual pada aplikasi ini dengan tambahan fitur yang terintegrasi langsung dengan google maps .

4. *Open Graph (Open Graph Protocol)*

Protokol Open Graph (*Open Graph Protocol*) adalah serangkaian tag HTML yang

ditambahkan ke halaman web untuk memberikan informasi lebih kaya kepada platform media sosial saat halaman tersebut dibagikan. Ini memungkinkan pengelolaan tautan yang lebih baik dan tampilan yang lebih menarik.

Relasi ke semua sistem diatas dapat digambarkan pada arsitektur umum aplikasi pada Gambar 1



dibawah ini :

Gambar 1. Arsitektur Umum Aplikasi

Web *EduSearch* Menggunakan *JSONLD* dan *rdflib* untuk mengakses permintaan dari *SPARQL* ke endpoint eksternal dan internal untuk mencari data sekolah. Web *EduSearch* Menggunakan *openstreet maps* dan *Google maps* berelasi dengan layanan google maps secara langsung untuk menampilkan data lokasi sekolah. Protokol Open Graph (*Open Graph Protocol*) adalah serangkaian tag HTML yang ditambahkan ke halaman web untuk memberikan informasi lebih kaya kepada platform media sosial saat halaman tersebut dibagikan. Ini memungkinkan pengelolaan tautan yang lebih baik dan tampilan yang lebih menarik.

### Fitur Utama pada Perancangan Web *EduSearch*

Dalam penelitian ini, kami merancang beberapa fitur pencarian utama dalam perancangan web pencarian cerdas *EduSearch* yaitu Fitur Pencarian Berdasarkan *Input User*, Filter berdasarkan Kecamatan, Filter berdasarkan jenjang pendidikan dan fitur. *Openstreet Map*

#### 1. Pencarian Berdasarkan *Input User*

Fitur pencarian yang diterapkan dalam aplikasi ini memberikan keleluasaan kepada pengguna untuk mencari sekolah berdasarkan Nama atau NPSN dengan mudah dan efisien. Dengan kemampuan ini, pengguna dapat memasukkan kata kunci berupa nama sekolah atau NPSN pada *website*, *EduSearch* menghasilkan hasil pencarian yang tepat dan relevan.

```
# SPARQL query with filter for bentukPendidikan if available
query_all_a = f"""
{PREFIXES}

SELECT DISTINCT ?namaSekolah ?npsn ?akreditasi ?bentukPendidikan ?kecamatan
WHERE {{
  ?individu instansi:namaSekolah ?namaSekolah ;
    instansi:NPSN ?npsn ;
    instansi:akreditasi ?akreditasi ;
    instansi:kindOf ?bentukPendidikan ;
    instansi:locatedIn ?kecamatan .

  FILTER(
    regex(STR(?namaSekolah), "{keyword}", "i") ||
    regex(STR(?npsn), "{keyword}", "i")
  )

  {f'FILTER(?bentukPendidikan = "{bentuk_pendidikan}")' if bentuk_pendidikan else ''}
}}
"""
```

Gambar 2. Query SPARQL untuk pencarian berdasarkan input

Gambar di atas merupakan Query SPARQL untuk pencarian berdasarkan nama sekolah atau NPSN, dimana *user* akan menginput *keyword* dan akan di cek pada filter lalu menampilkan data berdasarkan *keyword* yang dimasukkan.

## 2. Filter Berdasarkan Kecamatan

Fitur filter yang telah diimplementasikan dalam aplikasi ini memberikan kenyamanan kepada pengguna untuk memfilter sekolah berdasarkan Kecamatan dengan sederhana dan efisien. Melalui kemampuan ini, pengguna dapat memfilter nama kecamatan sebagai kriteria pencarian pada platform *EduSearch*, menghasilkan hasil pencarian yang sesuai dan relevan dengan kecamatan yang dimaksud.

```
query_by_kecamatan = f"""
{PREFIXES}

SELECT DISTINCT *
WHERE {{
  BIND("https://schema/Instansi_sumut#Kecamatan_{kecamatan}" AS ?targetKecamatan)
  OPTIONAL {{
    ?individu instansi:namaSekolah ?namaSekolah ;
      instansi:NPSN ?npsn ;
      instansi:akreditasi ?akreditasi ;
      instansi:kindOf ?bentukPendidikan ;
      instansi:locatedIn ?targetKecamatan .
  }}
}}
"""
```

Gambar 3. Query SPARQL Untuk Pencarian Berdasarkan Kecamatan

Gambar diatas merupakan *query* untuk mencari sekolah berdasarkan kecamatan, dimana akan di *bind* dari kecamatan yang didapatkan di *keyword*, lalu menampilkan sekolah berdasarkan kecamatan tersebut.

### 3. Filter Berdasarkan Jenjang Pendidikan

Fitur filter yang telah diintegrasikan dalam aplikasi ini memberikan kemudahan kepada pengguna untuk menyaring sekolah berdasarkan Jenjang Pendidikan seperti SD, SMP. dll, dengan simpel dan efektif. Dengan kemampuan ini, pengguna dapat memilih jenjang pendidikan sebagai kriteria penyaringan pada platform *EduSearch*, sehingga hasil pencarian yang dihasilkan sesuai dan relevan dengan jenjang pendidikan yang diinginkan.

```
def get_bentuk_pendidikan_classification():
    query = f"""
    {PREFIXES}

    SELECT DISTINCT ?bentukPendidikan
    WHERE {{
        ?individu instansi:namaSekolah ?namaSekolah ;
                instansi:kindOf ?bentukPendidikan .

        FILTER (
            !CONTAINS(STR(?bentukPendidikan), "SPK")
        )
    }}
    """
    results = run_query(query)
    return [{"bentukPendidikan": result['bentukPendidikan']['value']} for result in results]
```

Gambar 4. *Query* Untuk Mengambil Bentuk Pendidikan Yang Tersedia

Gambar diatas merupakan *query* yang digunakan untuk mengambil bentuk pendidikan yang tersedia di dataset.

### 4. Open Street Map dan Detail Sekolah

Pada website *EduSearch*, saat pengguna menekan tombol *details* akan menampilkan detail sekolah yang dipilih, dalam halaman ini terdapat juga lokasi sekolah tersebut yang menggunakan *Open Street Map* dan *Leaflet.js*.

```
query_detail = f"""
{PREFIXES}

SELECT DISTINCT *
WHERE {{
    BIND("{npsn}" AS ?targetNPSN)
    OPTIONAL {{
        ?individu instansi:NPSN ?targetNPSN ;
                instansi:namaSekolah ?namaSekolah ;
                instansi:alamat ?alamat ;
                instansi:lintang ?lintang ;
                instansi:Status ?status ;
                instansi:kabupaten ?kabupaten ;
                instansi:bujur ?bujur ;
                instansi:tanggalPendirian ?tanggalPendirian ;
                instansi:akreditasi ?akreditasi ;
                instansi:linkSekolah ?linkSekolah ;
                instansi:kindOf ?bentukPendidikan ;
                instansi:locatedIn ?kecamatan .
    }}
}}
"""
```

Gambar 5. *Query* Untuk Detail dan *Map* Sekolah Berdasarkan NPSN

Gambar di atas menampilkan *query* untuk menampilkan detail sekolah berdasarkan NPSN yang didapat dari parameter url untuk menampilkan *map* mengambil data dari lintang dan bujur.

```

<script>
var data = {{ data | safe }};

// Check if data is not empty before accessing properties
if (data.length > 0) {
var latitude = parseFloat(data[0]["lintang"]["value"]);
var longitude = parseFloat(data[0]["bujur"]["value"]);

console.log("Parsed latitude:", latitude);
console.log("Parsed longitude:", longitude);

// Initialize the map
var map = L.map('map').setView([latitude, longitude], 15);

// Add a tile layer (you can use any tile provider, here is OpenStreetMap)
L.tileLayer('https://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png', {
attribution: '© OpenStreetMap contributors'
}).addTo(map);

// Create a marker with a popup
var marker = L.marker([latitude, longitude]).addTo(map);
marker.bindPopup(`${data[0]["namaSekolah"]["value"]}<br>
<a href="https://maps.google.co.id/maps?expflags=enable_star_based_justifications:true&ie=UTF8&f=d&daddr=${latitude},${longitude}"
target="_blank">View on Google Maps</a>`);
} else {
console.error("No data available for the specified school.");
}
</script>

```

Gambar 6. Integrasi *Open Street Map* menggunakan Leaflet.js

Gambar di atas merupakan javascript untuk menampilkan *map* menggunakan Leaflet.js, dimana mengambil data dari lintang dan bujur pada dataset, lalu dimasukkan ke variable dan diletakkan di url leaflet.

## 5. Filter Berdasarkan Sekolah Terdekat

Dengan fitur Filter Berdasarkan Sekolah Terdekat di *EduSearch*, pengguna dapat dengan mudah menemukan sekolah terdekat dari lokasi mereka hanya dengan menekan tombol "sekolah terdekat." Informasi sekolah yang muncul mencakup jarak sekolah dari lokasi pengguna saat ini. Fitur ini tidak hanya memudahkan pengguna dalam memilih sekolah berdasarkan jarak fisik, tetapi juga dapat digunakan sebagai referensi untuk proses zonasi, memastikan bahwa pengguna memiliki aksesibilitas yang optimal ke pendidikan di daerah mereka.

```

app.route('/terdekat', { methods: ['GET', 'POST'] })
def terdekat():
    # SPARQL query to retrieve details for the specified NPSN
    query_terdekat = """
(PREFIXES)

SELECT DISTINCT ?namaSekolah ?npsn ?akreditasi ?bentukPendidikan ?kecamatan ?lintang ?bujur
WHERE {
OPTIONAL {
?individu instansi:namaSekolah ?namaSekolah ;
instansi:NPSN ?npsn ;
instansi:lintang ?lintang ;
instansi:bujur ?bujur ;
instansi:akreditasi ?akreditasi ;
instansi:kindOf ?bentukPendidikan ;
instansi:locatedIn ?kecamatan .
}
}
"""

user_latitude_str = request.args.get('latitude', '0')
user_longitude_str = request.args.get('longitude', '0')

try:
user_latitude = float(user_latitude_str)
user_longitude = float(user_longitude_str)
except ValueError:
# Handle the case where parsing to float fails
return "Invalid latitude or longitude", 400

# Print the values for debugging
print(f"User Latitude: {user_latitude}")
print(f"User Longitude: {user_longitude}")

# Assuming you have a function to query the SPARQL endpoint for schools
# Replace this with your actual SPARQL query and logic
data_terdekat = run_query(query_terdekat)

# Calculate distance for each school and add it to the data
for school in data_terdekat:
school_latitude_str = school["lintang"]["value"]
school_longitude_str = school["bujur"]["value"]

try:
school_latitude = float(school_latitude_str)
school_longitude = float(school_longitude_str)
except ValueError:
# Handle the case where parsing to float fails
school["distance"] = None
continue

distance = haversine(user_latitude, user_longitude, school_latitude, school_longitude)
school["distance"] = distance

# Execute the SPARQL query
bentuk_pendidikan_classification = get_bentuk_pendidikan_classification()

# Pass the results to the template
return render_template('terdekat.html', data=data_terdekat, bentuk_pendidikan_classification=bentuk_pendidikan_classification)

```

Gambar 7. Program Yang Digunakan Untuk Mencari Jarak Sekolah

Program diatas mengambil informasi lokasi user dari *latitude* dan *longitude* dari parameter url yang diambil dari lokasi pengguna saat ini, lalu dilakukan perhitungan jarak pengguna dengan *latitude* dan *longitude* sekolah dengan menggunakan metode *haversine*.

## HASIL DAN PEMBAHASAN

### 1. Pencarian Berdasarkan Input User

Pada fitur ini, Pengguna dapat memasukkan kata kunci berupa nama sekolah, NPSN pada website *EduSearch*.

Hasil Pencarian dari **smn**

Semua Jenjang ▾

Show 10 ▾ entries

Nama	NPSN	Akreditasi	Bentuk Pendidikan	Kecamatan	Action
SMAN 11 MEDAN	10210875	A	SMA	Medan Tembung	Details
SMAN 7 MEDAN	10210860	A	SMA	Medan Timur	Details
SMAN 17 MEDAN	10210851	A	SMA	Medan Tuntungan	Details
SMAN 14 MEDAN	10210865	A	SMA	Medan Denai	Details
SMAN 21 MEDAN	10210855	A	SMA	Medan Denai	Details
SMAN 13 MEDAN	10210877	A	SMA	Medan Johor	Details
SMAN 16 MEDAN	10210863	A	SMA	Medan Marelan	Details
SMAN 8 MEDAN	10210861	A	SMA	Medan Area	Details
SMAN 3 MEDAN	10210856	A	SMA	Medan Barat	Details
SMAN 4 MEDAN	10210857	A	SMA	Medan Petisah	Details

Showing 1 to 10 of 21 entries

Gambar 8. Hasil Pencarian "sman"

Gambar di atas merupakan tabel yang ditampilkan setelah user mencari dengan *keyword* "sman".

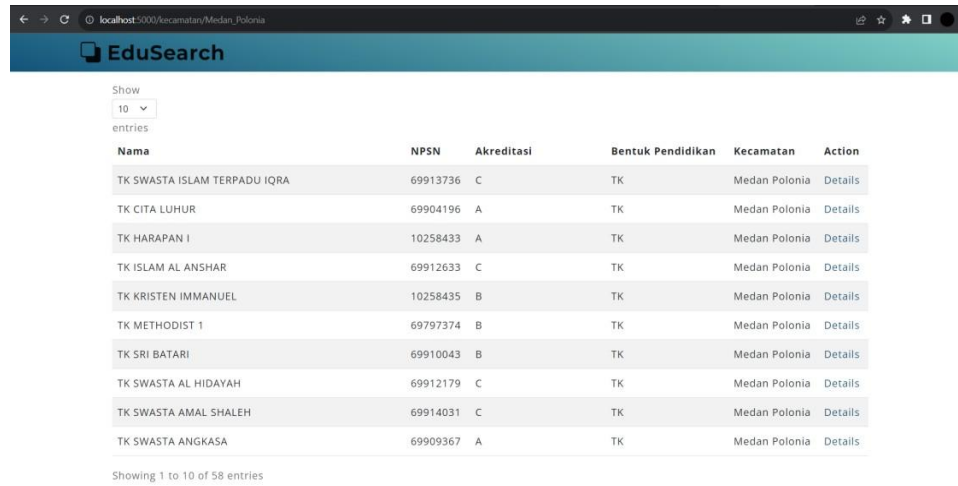
### 2. Filter Berdasarkan Kecamatan

Fitur ini dapat digunakan pengguna untuk melakukan filter sekolah berdasarkan kecamatan tempat tinggal pengguna.



Gambar 9. Kecamatan Yang Tersedia di Medan

Setelah *user* mengklik tombol kecamatan, user dapat menekan kecamatan yang diinginkan, dan akan ditampilkan kecamatan yang sesuai



Nama	NPSN	Akreditasi	Bentuk Pendidikan	Kecamatan	Action
TK SWASTA ISLAM TERPADU IQRA	69913736	C	TK	Medan Polonia	Details
TK CITA LUHUR	69904196	A	TK	Medan Polonia	Details
TK HARAPAN I	10258433	A	TK	Medan Polonia	Details
TK ISLAM AL ANSHAR	69912633	C	TK	Medan Polonia	Details
TK KRISTEN IMMANUEL	10258435	B	TK	Medan Polonia	Details
TK METHODIST 1	69797374	B	TK	Medan Polonia	Details
TK SRI BATARI	69910043	B	TK	Medan Polonia	Details
TK SWASTA AL HIDAYAH	69912179	C	TK	Medan Polonia	Details
TK SWASTA AMAL SHALEH	69914031	C	TK	Medan Polonia	Details
TK SWASTA ANGKASA	69909367	A	TK	Medan Polonia	Details

Showing 1 to 10 of 58 entries

Gambar 10. Hasil Filter Berdasarkan Kecamatan

Gambar di atas merupakan tabel kecamatan medan polonia yang ditampilkan setelah pengguna memilih kecamatan medan polonia.

### 3. Filter Berdasarkan Jenjang Pendidikan

Pada fitur ini, pengguna dapat menyaring sekolah berdasarkan jenjang pendidikan baik SD, SMP, SMA, dan lain-lain.



Nama	NPSN	Akreditasi	Bentuk Pendidikan	Kecamatan	Action
SMKN 2 MEDAN	10259588	Sertifikat Kedaluwarsa	SMK	Medan Amplas	Details
SMKN 3 MEDAN	10210763	A	SMK	Medan Amplas	Details
SMKN 7 MEDAN	10210978	A	SMK	Medan Amplas	Details
SMK-2 SWASTA NURHASANAH MEDAN	10211275	C	SMK	Medan Amplas	Details
SMKS AL WASHLIYAH 3	10211250	B	SMK	Medan Amplas	Details
SMKS AL WASHLIYAH 4 MEDAN	10211066	Sertifikat Kedaluwarsa	SMK	Medan Amplas	Details
SMKS AL WASHLIYAH AMPLAS MEDAN	69757134	C	SMK	Medan Amplas	Details
SMKS CIPTA KARYA	10210769	Sertifikat Kedaluwarsa	SMK	Medan Amplas	Details
SMKS MULTI KARYA	10211094	A	SMK	Medan Amplas	Details
SMKS NUR HASANAH	10259152	B	SMK	Medan Amplas	Details


Showing 1 to 10 of 160 entries (filtered from 2,471 total entries)

Gambar 11. Hasil Tabel Setelah Difilter Hanya SMK

Gambar di atas merupakan hasil tabel jika data sekolah di filter berdasarkan jenjang yang dipilih oleh pengguna.

### 4. Open Street Map dan Detail Sekolah


Fitur ini digunakan untuk melihat detail sekolah yang kita cari beserta alamat dan lokasi sekolah berada. Gambar 12 menunjukkan tampilan detail sekolah yang kita telah lakukan pencarian



**Link Sekolah**  
KUNJUNGI SEKOLAH

### SMAN 1 MEDAN

NPSN :	10210873
Status :	Negeri
Akreditasi :	A
Kecamatan :	Medan Polonia
Bentuk Pendidikan :	SMA
Tanggal Berdiri :	2021-12-08



**Alamat**

JL. TEUKU CIK DITIRO NO. 1  
MEDAN

Lintang 3.5813E0  
Bujur 9.86696E1

Gambar 12. Tampilan Detail Sekolah

Gambar di atas merupakan tampilan detail sekolah, dimana terdiri mengenai informasi sekolah yang dipilih, dan lokasi sekolah tersebut menggunakan Leaflet.js.

## 5. Filter Berdasarkan Sekolah Terdekat

Filter ini digunakan untuk melihat sekolah-sekolah yang terdekat disekitar domisili pengguna

**EduSearch**

Get My Location Semua Jelang \*

Show 10 entries

Nama	NPSN	Akreditasi	Bentuk Pendidikan	Kecamatan	Jarak	Action
SLB-A KARYA MURNI	10259865	A	SLB	Medan Johor	0.34 km	<a href="#">Details</a>
SMP SWASTA AL RAZI SINAR HARAPAN	69954870	A	SMP	Medan Johor	0.44 km	<a href="#">Details</a>
SMP SWASTA AL MANAR	69924576	B	SMP	Medan Johor	0.5 km	<a href="#">Details</a>
SMAS ISLAM ULUN NUHA	69758996	B	SMA	Medan Johor	0.67 km	<a href="#">Details</a>
UPT SD NEGERI 060927	10210495	A	SD	Medan Johor	0.92 km	<a href="#">Details</a>

Gambar 13. Hasil Pencarian Sekolah Terdekat

Di atas merupakan tampilan sekolah terdekat, dimana pengguna dapat melihat jarak antara sekolah dengan lokasinya saat ini.

## Uji Coba Validasi dan Fungsionalitas

Di bawah merupakan hasil uji coba kami terhadap fungsi-fungsi utama *website EduSearch*.

Tabel 1. Tabel Uji Coba Validasi dan Fungsionalitas

No.	Pengujian	Hasil Yang Diharapkan	Keberhasilan
1.	Pencarian berdasarkan input user	Menampilkan sekolah berdasarkan <i>input</i> yang diberikan <i>user</i>	Berhasil
2.	Filter berdasarkan kecamatan	Menampilkan tabel sekolah berdasarkan kecamatan yang dipilih <i>user</i>	Berhasil
3.	Filter berdasarkan jenjang pendidikan	Menampilkan tabel sekolah berdasarkan jenjang pendidikan yang dipilih <i>user</i>	Berhasil
4.	Menampilkan detail sekolah dan integrasi Open Street Map	Menampilkan detail sekolah, beserta lokasi sekolah berdasarkan lintang dan bujur	Berhasil
5.	Penggunaan Open Graph Protocol	Menampilkan informasi mengenai <i>website EduSearch</i> pada tautan	Berhasil
6.	Filter berdasarkan sekolah terdekat	Menampilkan sekolah terdekat dengan lokasi pengguna	Berhasil

Dari Tabel uji diatas menunjukkan bahwa fungsionalitas fitur yang dirancang pada sistem pencarian cerdas sekolah di sekitar kota Medan dapat berjalan dengan baik.

### KESIMPULAN

Web pencarian cerdas untuk mencari data seluruh sekolah formal di kota Medan “*EduSearch*” menerapkan konsep semantik dalam pengimplementasiannya untuk menyediakan informasi bagi para pengguna dengan aksebilitas yang mudah, efisien, dan fleksibel. Aksebilitas bagi pengguna saat menggunakan web pencarian ini yaitu kemudahan menggunakan mesin pencarian berdasarkan *keyword* yang diinginkan serta dapat menjelajahi data yang disajikan dengan menggunakan fitur filter berdasarkan bentuk pendidikan dan kecamatan yang tersedia, serta terdapat *open street map* untuk menampilkan lokasi secara *real time*. Data pada web pencarian *EduSearch* disajikan secara lengkap sesuai dengan kebutuhan pengguna seperti nama sekolah, NPSN, alamat, akreditasi, bentuk pendidikan, link sekolah, tanggal pendirian, serta maps lokasi sekolah. Teknologi yang digunakan untuk membangun web pencarian *EduSearch* yaitu terdiri dari Selenium Web *Driver* untuk mengambil data dari sumber data berupa Jsonld, SPARQL Internal dan External, Open Graph Protocol, Flask Python, dan Leaflet JS untuk Google Maps. Dan dari uji Validitas dan Fungsionalitas yang telah dilakukan pada web pencarian cerdas *EduSearch* menunjukkan keberhasilan yang baik pada fitur-fitur pencarian pada web cerdas *EduSearch*.

### REFERENSI

- Blobel, B. G. M. E., Engel, K., & Pharowe, P. (2006). Semantic Interoperability. *Methods of Information in Medicine*, 45(04), 343–353. <https://doi.org/10.1055/s-0038-1634087>
- Brecher, C., Buchsbaum, M., Muller, A., Schilling, K., Obdenbusch, M., Staudacher, S., & Albasatineh, M. C. (2021). Gaining IIoT insights by leveraging Ontology-based modelling of raw data and Digital Shadows. *2021 4th IEEE International Conference on Industrial Cyber-Physical*

- Systems (ICPS)*, 231–236. IEEE. <https://doi.org/10.1109/ICPS49255.2021.9468116>
- Charpenay, V., Käbisch, S., & Kosch, H. (2018). Semantic data integration on the web of things. *Proceedings of the 8th International Conference on the Internet of Things*, 1–8. New York, NY, USA: ACM. <https://doi.org/10.1145/3277593.3277609>
- Chokshi, H. J., & Panchal, R. (2022). Using Apache Jena Fuseki Server for Execution of SPARQL Queries in Job Search Ontology Using Semantic Technology. *International Journal of Innovative Research in Computer Science & Technology (IJIRCST)*, (10), 2347– 5552. <https://doi.org/10.55524/ijircst.2022.10.2.99>
- Halpin, H. (2023). The Hidden History of the Like Button: From Decentralized Data to Semantic Enclosure. *Social Media + Society*, 9(3). <https://doi.org/10.1177/20563051231195542>
- Himawan, Harjanti, T. W., Supriati, R., & Setiyani, H. (2020). Evolusi Penggunaan Teknologi Web 3.0 : Semantic Web. *Journal of Information System, Graphics, Hospitality and Technology*, 2(02), 54–60. <https://doi.org/10.37823/insight.v2i02.107>
- Josi, A., & Andretti Abdillah, L. (n.d.). *PENERAPAN TEKNIK WEB SCRAPING PADA MESIN PENCARI ARTIKEL ILMIAH*.
- Li, W., & Yang, C. (2008). A Semantic Search Engine for Spatial Web Portals. *IGARSS 2008 - 2008 IEEE International Geoscience and Remote Sensing Symposium*, II-1278-II–1281. IEEE. <https://doi.org/10.1109/IGARSS.2008.4779236>
- Mapping and the citizen sensor*. (2017). Ubiquity Press.
- Niemann, M., Mochol, M., & Tolksdorf, R. (2008). Enhancing Hotel Search with Semantic Web Technologies. *Journal of Theoretical and Applied Electronic Commerce Research*, 3(2), 82–96. <https://doi.org/10.4067/S0718-18762008000100008>
- Nyamathulla, S., Ratnababu, P., Sultana Shaik, N., Lakshmi, B. N., & Professor, A. (2021). *A Review on Selenium Web Driver with Python* (Vol. 25). Retrieved from <http://annalsofscsb.ro>
- Sadeeq, M. J., Zeebaree, S. R. M., & Author, A. (2021). *Semantic Search Engine Optimisation (SSEO) for Dynamic Websites: A Review Science and Business Journal homepage: ijsab.com/ijsb*. <https://doi.org/10.5281/zenodo.4536804>
- Wang, X., Sun, Q., & Liang, J. (2020). JSON-LD Based Web API Semantic Annotation Considering Distributed Knowledge. *IEEE Access*, 8, 197203–197221. <https://doi.org/10.1109/ACCESS.2020.3034937>