



**KEMENTERIAN
KESEHATAN
REPUBLIK
INDONESIA**

Dokumentasi Teknis SATUSEHAT Kamus Farmalkes (KFA) ReST API

Pusat Data dan Teknologi Informasi - Digital Transformation Office

Versi 3.1.1, 06 Februari 2024
(master-1dca3ee, 2024-03-11)

KERAHASIAAN INFORMASI

Informasi yang terkandung dalam dokumen ini bersifat:

PUBLIK

Yang berarti bahwa semua pihak/entitas yang mendapatkan dokumen ini dan/atau sebagian/keseluruhan informasi di dalam dokumen ini, **DIPERBOLEHKAN** untuk menggunakan, menerbitkan, menyebarluaskan, menduplikasi, atau mendistribusikan kembali dokumen ini dan/atau informasi di dalamnya dalam bentuk apapun, baik secara elektronik maupun mekanik tanpa izin tertulis dari **Pusat Data dan Teknologi Informasi – Digital Transformation Office (DTO) Kementerian Kesehatan Republik Indonesia**.

Pusat Data dan Teknologi Informasi – Digital Transformation Office (DTO) Kementerian Kesehatan Republik Indonesia tidak membuat pernyataan dan tidak memberikan jaminan dalam bentuk apapun sehubungan dengan informasi di dalam dokumen ini, termasuk tetapi tidak terbatas pada keakuratan atau kelengkapan informasi, fakta dan/atau pendapat yang terkandung di dalam dokumen ini.

Pusat Data dan Teknologi Informasi – Digital Transformation Office (DTO) Kementerian Kesehatan Republik Indonesia, para direktur, para karyawan, dan unit-unit di bawahnya tidak dapat dimintai pertanggungjawaban atas penggunaan dan kepercayaan atas opini, perkiraan, dan temuan dalam dokumen ini.

RIWAYAT PERUBAHAN

Versi	Tanggal Pembaruan	Deskripsi Perubahan	Author/Editor	Versi Aplikasi
3.1.1	06 Februari 2024	<p>Rilis Dokumentasi Teknis SATUSEHAT: Kamus Farmalkes (KFA) ReST API versi AsciiDoc pdf</p> <ol style="list-style-type: none"> 1. <i>Hide endpoint development</i> 2. Ubah <i>endpoint</i> ReST API dari <i>development</i> ke <i>production</i> 	Ilham	KFA v2.0
3.1	04 Desember 2023	<p>Rilis Dokumentasi Teknis SATUSEHAT: Kamus Farmalkes (KFA) ReST API versi AsciiDoc pdf</p> <ol style="list-style-type: none"> 1. Penambahan <i>endpoint development</i> dan <i>staging</i> di produk KFA 2. Perubahan <i>endpoint</i> produk dan token KFA 	Ilham	KFA v2.0
3.0	17 Oktober 2023	<p>Rilis Dokumentasi Teknis SATUSEHAT: Kamus Farmalkes (KFA) ReST API versi AsciiDoc pdf</p> <ol style="list-style-type: none"> 1. Perubahan <i>Collection</i> 2. Perubahan <i>Endpoint</i> versi 1 dan versi 2 3. Penambahan Perubahan param pada <i>collection</i> Pencarian Produk dengan Paginasi 	Ilham	KFA v2.0

Versi	Tanggal Pembaruan	Deskripsi Perubahan	Author/Editor	Versi Aplikasi
2.0	15 September 2023	<p>Rilis Dokumentasi Teknis SATUSEHAT: Kamus Farmalkes (KFA) ReST API versi AsciiDoc pdf</p> <div style="border: 1px solid black; padding: 10px; margin-top: 10px;"> <ol style="list-style-type: none"> 1. region_code : isi dengan kode regional JKN yang diinginkan 2. document_ref : Isi dengan dokumen referensi atau dasar hukum yang berlaku. 3. Service yang baru diantaranya : GET Price JKN, GET Product All Detail Paginasi, GET Tag dan GET ATC </div>	Ilham	
1.2	05 September 2023	Pemakaian <i>placeholder</i> untuk <i>value client_id</i> , <i>client_secret</i> , <i>access_token</i> dan <i>developer.email</i>	Miko	
1.1	26 Mei 2023	<p>Update Dokumentasi</p> <div style="border: 1px solid black; padding: 10px; margin-top: 10px;"> <ul style="list-style-type: none"> • Penyesuaian urutan <i>request</i> pada <i>Products Integration</i>. • Update respon untuk beberapa <i>request</i> pada <i>Products Integration</i>. • Tambahkan beberapa <i>request</i> baru pada <i>Products Integration</i>. </div>	Ilham	
1.0	28 Maret 2023	<p>Rilis Dokumentasi Teknis SATUSEHAT: Kamus Farmalkes (KFA) ReST API versi AsciiDoc pdf</p>	Miko	

Versi	Tanggal Pembaruan	Deskripsi Perubahan	Author/Editor	Versi Aplikasi
0.0	27 Maret 2023	Penyusunan Dokumentasi Teknis SATUSEHAT: Kamus Farmalkes (KFA) ReST API versi AsciiDoc pdf	Miko	

DAFTAR ISI

KERAHASIAAN INFORMASI	1
RIWAYAT PERUBAHAN	2
PENGANTAR TEKNIS	9
1. Kata Kunci Prioritas	9
2. Simbol dan Istilah	10
3. ReST API	10
3.1. HTTP Method	10
3.2. HTTP Status Codes	11
4. ISO 8601	12
4.1. Format Tanggal dan Waktu	12
5. Pesan Prioritas	13
5.1. Catatan	13
5.2. Tips	13
5.3. Penting	14
5.4. Perhatian	14
5.5. Peringatan	14
A. PENDAHULUAN	15
1. Latar Belakang	15
2. Tujuan	15
3. Sistematika Dokumentasi	15
B. AUTENTIKASI	17
1. Endpoint	17
2. Postman Collection	17
3. Pengajuan dan Verifikasi Akses	18
4. Autentikasi - Mendapatkan Token	24
4.1. Request	24
URL	24
HTTP Verb/Method	24
Header	24
Query String	24
Body (<i>application/x-www-form-urlencoded</i>)	24
4.2. Response	25
2xx <i>Success</i>	25
4xx <i>Client Error</i>	26
5xx <i>Server Error (Content-Type: text/plain)</i>	26
4.3. Contoh Penggunaan/Kode	26
cURL (Windows)	26

cURL (Linux)	26
Postman	27
C. PRICE	28
1. <i>Integration</i> - Mendapatkan Harga Produk JKN	29
1.1. Request	29
URL	29
HTTP Verb/Method	29
Header	29
Query String	29
1.2. Response	30
2xx Success	30
4xx Client Error	32
5xx Server Error (Content-Type: text/plain)	32
1.3. Contoh Penggunaan/Kode	33
cURL (Windows)	33
cURL (Linux)	33
Postman	33
D. PRODUCTS VERSION 1	34
1. <i>Integration</i> - Pencarian Produk Sesuai ATC dengan Paginasi	35
1.1. Request	35
URL	35
HTTP Verb/Method	35
Header	35
Query String	35
1.2. Response	37
2xx Success	37
4xx Client Error	39
5xx Server Error (Content-Type: text/plain)	39
1.3. Contoh Penggunaan/Kode	39
cURL (Windows)	39
cURL (Linux)	40
Postman	40
2. <i>Integration</i> - Mendapatkan ATC Metadata	41
2.1. Request	41
URL	41
HTTP Verb/Method	41
Header	41
Query String	41
2.2. Response	43
2xx Success	43
4xx Client Error	44

5xx Server Error (Content-Type: text/plain)	44
2.3. Contoh Penggunaan/Kode	44
cURL (Windows)	45
cURL (Linux)	45
Postman	45
3. Integration - Pencarian Produk Sesuai Tag dengan Paginasi	46
3.1. Request	46
URL	46
HTTP Verb/Method	46
Header	46
Query String	46
3.2. Response	47
2xx Success	48
4xx Client Error	50
5xx Server Error (Content-Type: text/plain)	50
3.3. Contoh Penggunaan/Kode	50
cURL (Windows)	50
cURL (Linux)	51
Postman	51
4. Integration - Mendapatkan TAG Metadata	52
4.1. Request	52
URL	52
HTTP Verb/Method	52
Header	52
Query String	52
4.2. Response	53
2xx Success	53
4xx Client Error	54
5xx Server Error (Content-Type: text/plain)	55
4.3. Contoh Penggunaan/Kode	55
cURL (Windows)	55
cURL (Linux)	55
Postman	55
E. PRODUCTS VERSION 2	57
1. Integration - Mendapatkan Detail Produk	58
1.1. Request	58
URL	58
HTTP Verb/Method	58
Header	58
Query String	58
1.2. Response	59

2xx Success	59
4xx Client Error	61
5xx Server Error (Content-Type: text/plain)	62
1.3. Contoh Penggunaan/Kode	62
cURL (Windows)	62
cURL (Linux)	62
Postman	62
2. Integration - Pencarian Produk dengan Paginasi	64
2.1. Request	64
URL	64
HTTP Verb/Method	64
Header	64
Query String	64
2.2. Response	65
2xx Success	65
4xx Client Error	67
5xx Server Error (Content-Type: text/plain)	67
2.3. Contoh Penggunaan/Kode	68
cURL (Windows)	68
cURL (Linux)	68
Postman	68
DAFTAR ISTILAH	69
REFERENSI	70

PENGANTAR TEKNIS

Beberapa simbol, istilah, dan konvensi digunakan pada pedoman ini agar penyampaian konsep, struktur data, baik ke tim *developer* maupun non *developer* menjadi lebih jelas. Pada bagian ini akan dijelaskan lebih lanjut terkait hal tersebut.



Setiap teks yang berwarna **biru muda**, dapat diklik untuk melompat ke bagian yang direferensikan.

1. Kata Kunci Prioritas

Dalam banyak dokumen terkait standar, beberapa kata digunakan untuk menandakan persyaratan dalam spesifikasi. Kata-kata ini sering dikapitalisasi dan/atau ditebalkan. Pada bagian ini akan didefinisikan kata-kata ini seperti yang dilakukan pada banyak dokumen RFC (*Request for Comment*), dengan beracuan pada standar di RFC 2119 [<https://www.ietf.org/rfc/rfc2119.txt>] beberapa kata-kata tersebut dialihbahasakan ke bahasa Indonesia.

- **WAJIB**, berarti definisi tersebut merupakan persyaratan mutlak dari spesifikasi.
- **TIDAK BOLEH**, berarti definisi tersebut merupakan larangan mutlak terhadap spesifikasi.
- **SEBAIKNYA**, berarti bahwa ada alasan yang sah dan kuat dalam keadaan tertentu untuk mengesampingkan hal tertentu lainnya, tetapi implikasi dari perilaku tersebut harus dipahami dan dipertimbangkan dengan cermat sebelum memilih alur yang berbeda.
- **SEBAIKNYA TIDAK**, berarti bahwa ada alasan yang sah dan kuat dalam keadaan tertentu ketika perilaku tertentu dapat diterima atau bahkan berguna, tetapi implikasi dari perilaku harus dipahami dan kasusnya dipertimbangkan dengan cermat sebelum menerapkan perilaku apa pun yang dijelaskan dengan label ini.
- **DIREKOMENDASIKAN**, sama definisinya dengan **SEBAIKNYA**.
- **TIDAK DIREKOMENDASIKAN**, sama definisinya dengan **SEBAIKNYA TIDAK**.
- **BOLEH**, berarti bahwa suatu spesifikasi benar-benar opsional. Satu vendor dapat memilih untuk memasukkan spesifikasi tersebut karena kondisi tertentu membutuhkannya atau karena vendor merasa bahwa itu meningkatkan performa tertentu sementara vendor lain mungkin menghilangkan spesifikasi yang sama. Implementasi yang tidak menyertakan opsi tertentu **PERLU** siap untuk beroperasi dengan implementasi lain yang menyertakan opsi, meskipun mungkin dengan fungsionalitas yang berkurang. Dalam hal yang sama, implementasi yang menyertakan opsi tertentu **PERLU** siap untuk beroperasi dengan implementasi lain yang tidak menyertakan opsi (kecuali, tentu saja, untuk fitur yang disediakan opsi.)
- **OPSIONAL**, sama definisinya dengan **BOLEH**.

Penggunaan kata-kata tadi biasanya untuk panduan dari suatu spesifikasi agar berjalan sesuai yang diharapkan. Bisa juga terkait pertimbangan keamanan, dimana kata-kata tersebut sering digunakan untuk menentukan perilaku dengan implikasi keamanan. Efek pada keamanan karena tidak menerapkan **WAJIB** atau **SEBAIKNYA**, atau melakukan sesuatu yang menurut spesifikasi **TIDAK BOLEH** atau **SEBAIKNYA TIDAK** dilakukan mungkin sangat bijak.

2. Simbol dan Istilah

Berikut beberapa simbol dan istilah yang digunakan dalam dokumentasi ini.

Tabel 1. Penamaan dan Simbol

Penamaan	Simbol/Nilai	Penjelasan
<code>string</code>	<code>"</code> atau <code>'</code>	Representasi dari kumpulan karakter/huruf/symbol.
<code>number</code>	<code>"1"</code> , <code>"-1"</code> , <code>"1.0"</code> , <code>'1'</code> , <code>'-1'</code> , <code>'1.0'</code> ,	Representasi dari kumpulan angka baik dari bilangan bulat atau desimal, dalam bentuk <code>string</code> .
<code>uuid</code>	<code>uuid</code> , <code>uuid1</code> , <code>uuid2</code> , <code>uuid3</code> , <code>uuid4</code>	Merepresentasikan ID universal unik dari suatu entitas, yang terdiri dari 36 karakter (32 karakter heksadesimal dan 4 tanda minus), lebih lengkapnya silakan lihat di https://en.wikipedia.org/wiki/Universally_unique_identifier . Contoh data <code>uuid: 123e4567-e89b-12d3-a456-426614174000</code>

3. ReST API

ReST adalah akronim dari *Representational State Transfer* dan salah satu tipe arsitektur untuk sistem hypermedia terdistribusi. Roy Fielding pertama kali mempresentasikannya pada tahun 2000 dalam disertasinya yang terkenal [restfulapi.net] [wikipedia-rest].

ReST melakukan pertukaran data sesuai standar yang ada pada *HTTP Specification*, sehingga perlu dijelaskan secara singkat terkait beberapa *HTTP method* dan *HTTP status code* yang diantaranya digunakan dalam dokumentasi teknis ini.

3.1. HTTP Method

HTTP Method mendefinisikan metode untuk menunjukkan aksi yang sesuai yang akan dilakukan pada *resource* yang telah teridentifikasi. *Resource* sendiri merepresentasikan data yang sudah ada atau data dinamis yang tergantung dari apa yang sudah disepakati dan diterapkan pada *server* yang dituju [wikipedia-http-method].

Tabel 2. HTTP Method

Method	Penjelasan
<code>GET</code>	Meminta representasi <i>resource</i> dari sumber tertentu untuk di READ oleh peminta.
<code>POST</code>	Meminta server menerima entitas yang terlampir dalam <i>request</i> sesuai struktur yang disepakati sebelumnya dan memprosesnya sebagai proses CREATE .
<code>PUT</code>	Meminta server menerima entitas yang terlampir dalam <i>request</i> sesuai struktur yang disepakati sebelumnya dan memprosesnya sebagai proses UPDATE dari data yang dimaksud.

Method	Penjelasan
PATCH	Meminta server menerima sebagian entitas yang terlampir dalam <i>request</i> sesuai struktur yang disepakati sebelumnya dan memprosesnya sebagai proses PARTIAL UPDATE dari data yang dimaksud.
DELETE	Meminta agar <i>resource</i> yang dimaksud di- DELETE .

3.2. HTTP Status Codes

HTTP Status Codes dikeluarkan dari *server* sebagai bentuk *response* dari *request* yang berasal dari *client* [wikipedia-http-status-code]. Kode-kode ini dikategorikan menjadi 5 kelas yang dibedakan dari digit pertamanya yaitu: **1xx informational response**, **2xx successful**, **3xx redirection**, **4xx client error**, dan **5xx server error**. Pada bagian ini, hanya akan dijelaskan 3 *HTTP Status Codes* yang sering digunakan di ReST yaitu **2xx**, **4xx**, dan **5xx**.

Tabel 3. HTTP Method

Method	Penjelasan
200 OK	Respon standar untuk permintaan yang berhasil.
201 Created	Permintaan berhasil dan <i>resource</i> yang dimaksud telah dibuat. Biasanya akan ada header location yang menunjukkan letak <i>resource</i> yang sudah dibuat.
202 Accepted	Permintaan berhasil dan <i>resource</i> yang dimaksud telah dijadwalkan akan diproses. Biasanya akan ada header location yang menunjukkan letak <i>resource</i> yang akan menunjukkan status atau hasil proses yang sudah atau akan dikerjakan.
204 No Content	Permintaan berhasil dan <i>resource</i> yang dimaksud telah dibuat, tetapi server tidak menyediakan data kembalian/response apapun (pesan di body kosong).
206 Partial Content	Biasanya digunakan dalam proses <i>download</i> data secara partial (<i>byte serving</i> [https://en.wikipedia.org/wiki/Byte_serving]) sesuai rentang yang didefinisikan di- <i>header</i> .
400 Bad Request	Permintaan gagal dikarenakan ada kesalahan di bagian <i>client</i> , seperti struktur data yang tidak sesuai, masukan data yang tidak sah, atau ukuran <i>resource</i> yang dikirim terlalu besar.
401 Unauthorized	Permintaan gagal dikarenakan perlu ada proses autentikasi (<i>authentication</i>) tertentu untuk mengaksesnya atau hasil autentikasi tidak sah.
403 Forbidden	Permintaan gagal dikarenakan perlu ada proses otorisasi (<i>authorization</i>) tertentu untuk mengaksesnya atau otorisasi tidak ditolak.
404 Not Found	Permintaan gagal dikarenakan <i>resource</i> yang dimaksud tidak ditemukan.
405 Method Not Allowed	Permintaan gagal dikarenakan <i>HTTP Method</i> yang digunakan untuk meminta <i>resource</i> tidak sesuai atau tidak diperbolehkan.

Method	Penjelasan
409 Conflict	Permintaan gagal dikarenakan adanya konflik terkait <i>resource</i> yang diakses, misal pembaruan <i>resource</i> yang sama secara bersamaan.
413 Payload Too Large	Permintaan gagal dikarenakan ukuran <i>payload</i> yang dikirimkan melebihi batas yang bisa diproses oleh <i>server</i> .
415 Unsupported Media Type	Permintaan gagal dikarenakan <i>server</i> tidak mendukung jenis <i>resource</i> yang dikirimkan.
422 Unprocessable Entity	Permintaan gagal dikarenakan adanya galat terkait semantik dari data atau <i>resource</i> yang dikirimkan saat diproses.
429 Too Many Requests	Permintaan gagal dikarenakan <i>client</i> terlalu cepat atau banyak mengakses <i>server</i> , biasanya galat ini terkait dengan ketentuan dari skema <i>rate-limiter</i> .
500 Internal Server Error	Permintaan gagal dikarenakan ada kesalahan dibagian <i>server</i> .
501 Not Implemented	Target permintaan yang dimaksud belum diimplementasikan di <i>server</i> .
503 Service Unavailable	<i>Server</i> dalam kondisi mati atau dalam pemeliharaan (<i>maintenance</i>).
504 Gateway Timeout	<i>Server</i> hanya bertindak sebagai <i>gateway</i> atau <i>proxy</i> dan menerima <i>response</i> yang salah dari <i>server</i> sumber.

4. ISO 8601

ISO 8601 adalah sebuah standar internasional yang digunakan pada pertukaran data terkait tanggal dan waktu [wikipedia-iso8601]. Format tanggal yang digunakan pada dokumentasi teknis ini sebagian besar mengacu pada standar ISO 8601 untuk merepresentasikan tanggal dan waktu.

4.1. Format Tanggal dan Waktu

Tabel 4. Format ISO8601 untuk tanggal dan waktu

Simbol	Penjelasan
YYYY	Merepresentasikan 4 digit tahun, <i>zero-padded</i> dibagian kiri, mulai dari 0000 hingga 9999. Tahun 0000 sama dengan tahun 1 Sebelum Masehi (SM). Representasi tahun Sebelum Masehi harus menggunakan tanda minus -0000 = 1 SM, -0001 = 2 SM, dst.
MM	Merepresentasikan 2 digit bulan, <i>zero-padded</i> dibagian kiri, dalam satu tahun mulai dari 01 hingga 12. Secara berurutan merepresentasikan bulan: 01 Januari, 02 Februari, 03 Maret, 04 April, 05 Mei, 06 Juni, 07 Juli, 08 Agustus, 09 September, 10 Oktober, 11 November, 12 Desember.
DD	Merepresentasikan 2 digit hari, <i>zero-padded</i> dibagian kiri, dalam satu bulan mulai dari 01 hingga 31 tergantung dari bulan yang dimaksud.

Simbol	Penjelasan
hh	Merepresentasikan 2 digit jam, <i>zero-padded</i> dibagian kiri, dari 00 hingga 23.
mm	Merepresentasikan 2 digit menit, <i>zero-padded</i> dibagian kiri, dari 00 hingga 59.
ss	Merepresentasikan 2 digit detik, <i>zero-padded</i> dibagian kiri, dari 00 hingga 60.
.SSS	Merepresentasikan 3 digit milidetik, <i>zero-padded</i> dibagian kiri.
.SSSSSS	Merepresentasikan 6 digit nanodetik, <i>zero-padded</i> dibagian kiri.

Contoh penggunaan dari format yang telah dijelaskan sebelumnya:

- YYYY-MM-DD untuk merepresentasikan tanggal, contoh: 2021-01-01 (2 Januari 2021).
- hh:mm:ss untuk merepresentasikan waktu lokal, contoh: 22:10:58 (pukul 22 lewat 10 menit 58 detik).
- YYYY-MM-DD hh:mm:ss untuk merepresentasikan tanggal dan waktu lokal, contoh: 2021-01-02 22:10:58 (2 Januari 2021, pukul 22 lewat 10 menit 58 detik).
- hh:mm:ssZ atau hh:mm:ss+0000 untuk merepresentasikan waktu UTC/GMT, contoh: 10:10:58Z (pukul 10 lewat 10 menit 58 detik UTC) bila dikonversi ke zona Asia/Jakarta (UTC+7) menjadi 17:10:58+0700.
- YYYY-MM-DD hh:mm:ssZ untuk merepresentasikan tanggal dan waktu lokal, contoh: 2021-01-02 10:10:58Z (2 Januari 2021, pukul 10 lewat 10 menit 58 detik UTC) bila dikonversi ke zona Asia/Jakarta (UTC+7) menjadi 2 Januari 2021, 17:10:58+0700.
- YYYY-MM-DD hh:mm:ss.SSSSS untuk merepresentasikan tanggal dan waktu lokal hingga ke 3 digit mili detik, contoh: 2021-01-02 10:10:58.353 (2 Januari 2021, pukul 10 lewat 10 menit 58 detik 353 mili detik).

5. Pesan Prioritas

Beberapa pesan prioritas berisi pernyataan, peringatan, perhatian, atau teguran tertentu sebagai salah satu cara untuk menekankan beberapa penjelasan yang dianggap penting, digunakan pada dokumentasi ini. Tipe-tipe dari kotak prioritas tersebut antara lain:

5.1. Catatan

Memberikan catatan singkat terkait informasi tertentu. Contoh:



Postman adalah salah satu aplikasi untuk melakukan *testing* terkait ReST API yang *user-friendly* dibandingkan cURL.

5.2. Tips

Memberikan saran kepada pembaca yang terkait *best practices* atau tips tertentu. Contoh:



Untuk versi cURL terbaru, gunakan parameter `--oauth2-bearer` untuk melakukan *request* yang membutuhkan *token bearer*.

5.3. Penting

Memberikan saran kepada pembaca bahwa anjuran yang disebutkan penting untuk dibaca. Contoh:



Selalu pakai *token bearer* yang didapatkan setiap melakukan *request* API yang bukan untuk autentikasi.

5.4. Perhatian

Memberikan saran kepada pembaca agar anjuran yang telah dijelaskan dilakukan dengan hati-hati. Contoh:



Mohon diperhatikan saat melakukan *request* data, dikarenakan ada perbedaan target URL untuk versi *production* dan *development*.

5.5. Peringatan

Memberikan tekanan kepada pembaca agar anjuran yang telah dijelaskan dilakukan dengan serius dan hati-hati, dikarenakan kemungkinan akan ada efek samping. Contoh:



Cek apakah *payload* yang akan dikirimkan sudah sesuai standar yang disepakati sebelumnya, karena bila tidak sesuai *payload* tersebut tidak akan dapat diproses.

A. PENDAHULUAN

1. Latar Belakang

Disamping adanya dokumentasi secara umum terkait Kamus Farmasi dan Alat Kesehatan (KFA), maka diperlukan satu dokumentasi khusus yang membahas semua ReST API yang disediakan oleh Kamus Farmasi dan Alat Kesehatan (KFA) yang dikembangkan oleh tim *developer* **Pusat Data dan Teknologi Informasi - Digital Transformation Office (DTO) Kementerian Kesehatan Republik Indonesia**. Dokumentasi ReST API Kamus Farmasi dan Alat Kesehatan (KFA) ini berisi spesifikasi teknis ReST API tersebut mencakup parameter yang tersedia, contoh *request*, contoh hasil *response*, penjelasan dari kode status HTTP yang mungkin diterima, serta cara penggunaannya menggunakan cURL dan Postman.

2. Tujuan

Dokumentasi **ReST API Kamus Farmasi dan Alat Kesehatan (KFA)** dibuat untuk kebutuhan ReST API yang tersedia dari Kamus Farmasi dan Alat Kesehatan (KFA) yang telah dikembangkan oleh tim *developer* SATUSEHAT. Penjelasan ReST API Kamus Farmasi dan Alat Kesehatan (KFA) didokumentasi ini hanya terbatas dari spesifikasi dari ReST API itu sendiri, penjelasan lengkap terkait Kamus Farmasi dan Alat Kesehatan (KFA), akan dijelaskan pada dokumentasi tersendiri.

3. Sistematika Dokumentasi

Sistematika pembahasan ReST API Kamus Farmasi dan Alat Kesehatan (KFA) pada dokumentasi ini dibagi menjadi beberapa bagian yang terdiri dari:

Lembar Kerahasiaan Informasi, berisi pernyataan terkait siapa saja yang boleh memiliki akses terkait dokumentasi ini.

Lembar Riwayat Perubahan, berisi riwayat perubahan terkait isi dari dokumentasi ini dan detail singkat dari perubahan tersebut.

Daftar Isi, berisi daftar nama bab, sub bab, dan halaman terkait pada dokumentasi teknis ReST API Kamus Farmasi dan Alat Kesehatan (KFA) ini.

Pengantar Teknis, berisi pengantar teknis singkat terkait beberapa konvensi yang digunakan pada dokumentasi teknis ini. Di bagian ini juga terdapat penjelasan singkat mengenai ReST API serta format tanggal dan waktu sesuai ISO 8601.

Pendahuluan, berisi penjelasan mengenai latar belakang, tujuan, dan sistematika penyusunan dokumentasi teknis terkait ReST API Kamus Farmasi dan Alat Kesehatan (KFA) ini.

Autentikasi, berisi penjelasan dan spesifikasi teknis terkait ReST API yang digunakan untuk proses autentikasi ke Kamus Farmasi dan Alat Kesehatan (KFA) melalui ReST API yang disediakan.

Daftar Istilah, berisi istilah-istilah yang mungkin ada dan perlu penjelasan lebih lanjut yang sebagian atau selalu digunakan pada dokumentasi teknis ReST API Kamus Farmasi dan Alat Kesehatan (KFA) ini.

Referensi, berisi daftar referensi yang digunakan atau untuk penjelasan lebih lanjut dari suatu topik yang dibahas pada dokumentasi teknis ReST API Kamus Farmasi dan Alat Kesehatan (KFA) ini.

B. AUTENTIKASI

Untuk melakukan transaksi data dari Kamus Farmasi dan Alat Kesehatan (KFA), perlu dilakukan proses autentikasi terlebih dahulu agar mendapatkan akses yang tersedia. Autentikasi yang digunakan oleh Kamus Farmasi dan Alat Kesehatan (KFA) mengikuti standar protokol OAuth 2 dengan tipe pemberian akses (*grant type*) adalah `client_credentials`.

Autentikasi menggunakan *grant type* `client_credentials` adalah proses autentikasi yang dilakukan antara *server to server*, sehingga tidak ada proses registrasi atau *log in* di sini. Autentikasi dengan tipe tersebut hanya memerlukan data berupa `client_id` dan `client_server`, dimana nilai tersebut didapatkan ketika pihak yang ingin menggunakan atau mengakses Kamus Farmasi dan Alat Kesehatan (KFA) ini telah melakukan pengajuan, terdaftar, serta mendapatkan persetujuan dari **Pusat Data dan Teknologi Informasi - Digital Transformation Office (DTO) Kementerian Kesehatan Republik Indonesia**.

Cara Mendapatkan Nilai dari `client_id` dan `client_secret`

Silakan terlebih dahulu melakukan **Pengajuan dan Verifikasi Akses** pada website <https://satusehat.kemkes.go.id/platform> melalui web *browser* Anda. Untuk informasi lebih lanjut dapat dilihat pada [Pengajuan dan Verifikasi Akses](#)



Setiap teks yang berwarna **biru muda**, dapat diklik untuk melompat ke bagian yang direferensikan.

1. Endpoint

Pada bagian ini akan dijelaskan spesifikasi untuk **ReST API Kamus Farmasi dan Alat Kesehatan (KFA) Terkait Autentikasi**, yang mempunyai tiga *endpoint* berdasarkan jenis lingkungan pengembangannya (*development environment*) yaitu:

- *staging* <https://api-satusehat-stg.dto.kemkes.go.id/oauth2/v1>
- *production* <https://api-satusehat.kemkes.go.id/oauth2/v1>



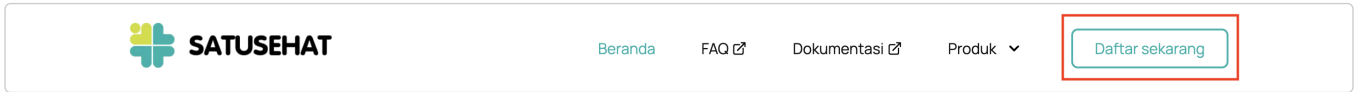
Semua penerapan, penjelasan, dan contoh yang akan dibahas akan menggunakan *endpoint development*.

2. Postman Collection

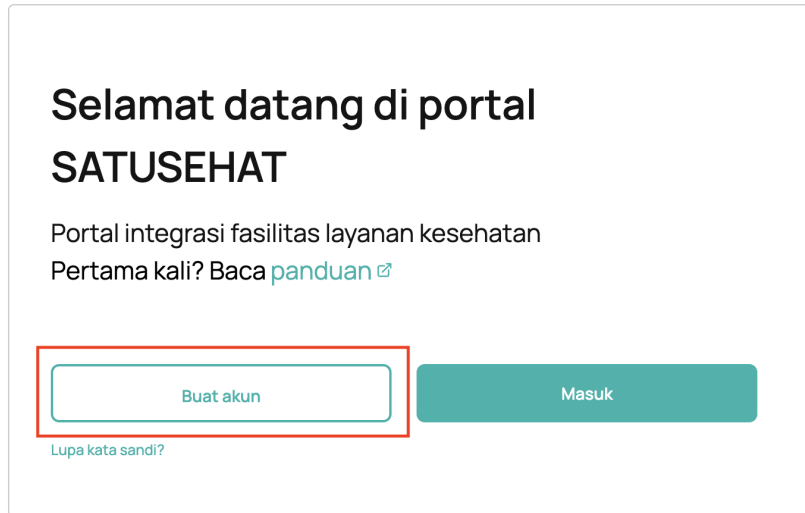
Silakan mengunduh Postman Collection untuk melihat contoh/melakukan workshop secara mandiri pada website [Postman Collection KFA](https://link.kemkes.go.id/MasterDataAPI) [<https://link.kemkes.go.id/MasterDataAPI>] melalui web *browser* Anda.

3. Pengajuan dan Verifikasi Akses

1. Untuk bergabung dengan Ekosistem FASKES SATUSEHAT, akses laman <https://satusehat.kemkes.go.id/platform> melalui web *browser* Anda, kemudian klik **Integrasi Faskes**.



2. Pada laman <https://satusehat.kemkes.go.id/platform/welcome> klik **Integrasi Faskes** untuk membuat akun SATUSEHAT.



3. Lengkapi data identitas berdasarkan *form* yang disediakan, kemudian klik **Buat Akun** untuk mendapatkan konfirmasi aktivasi akun pada *e-mail* yang didaftarkan.


Buat Akun FASKES SATUSEHAT Anda

Nama Pengelola Faskes

Email


Nomor telepon

Kata sandi

Minimal 8 karakter termasuk huruf kapital (A-Z), huruf kecil (a-z) dan angka (0-9)

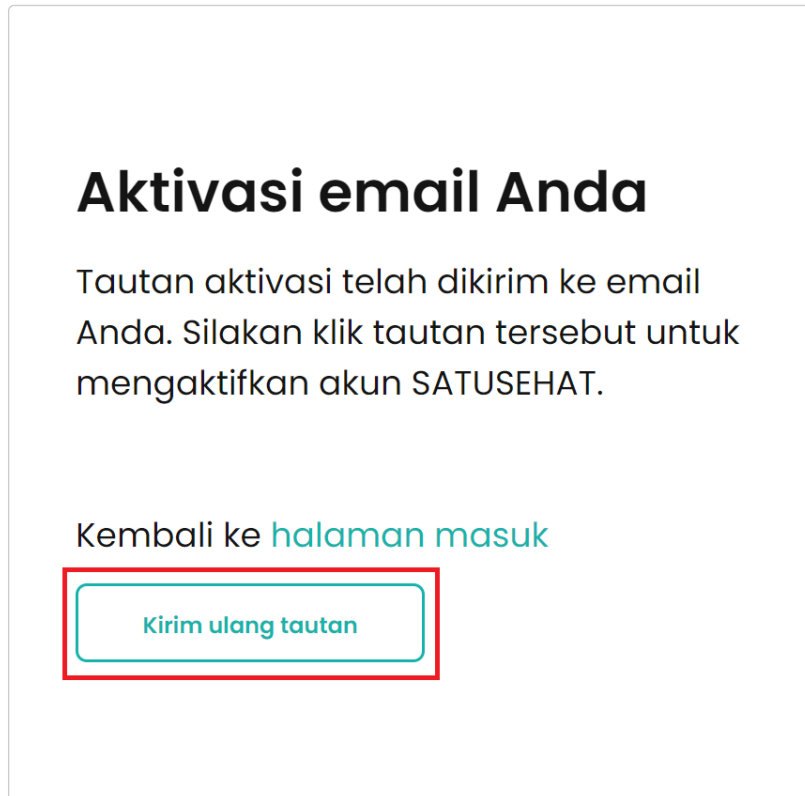
Konfirmasi kata sandi

Minimal 8 karakter termasuk huruf kapital (A-Z), huruf kecil (a-z) dan angka (0-9)

Buat akun

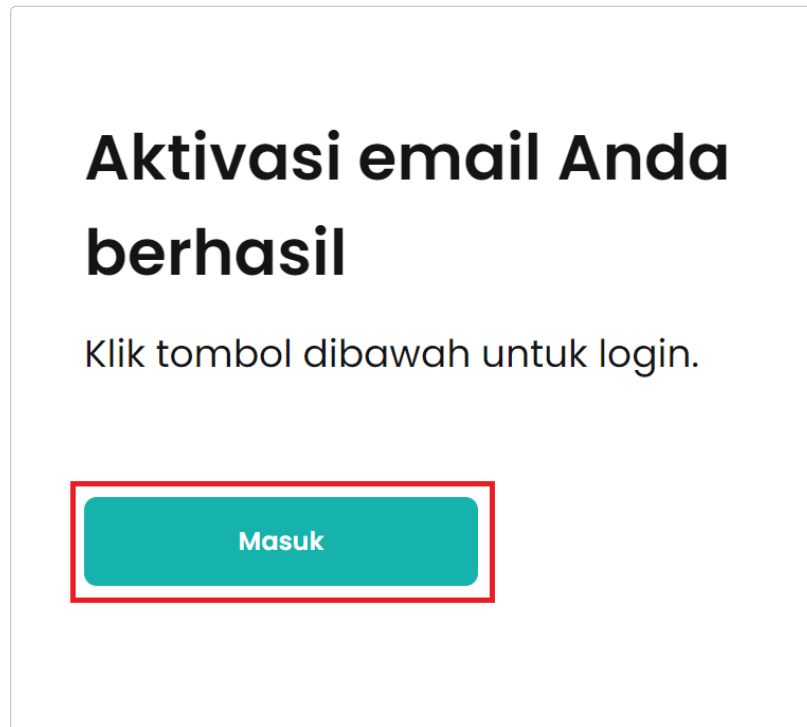
4. Selanjutnya akan muncul laman **Aktivasi e-mail Anda**, apabila aktivasi akun SATUSEHAT tidak terkirim pada *e-mail* yang telah didaftarkan, klik **Kirim ulang tautan** .



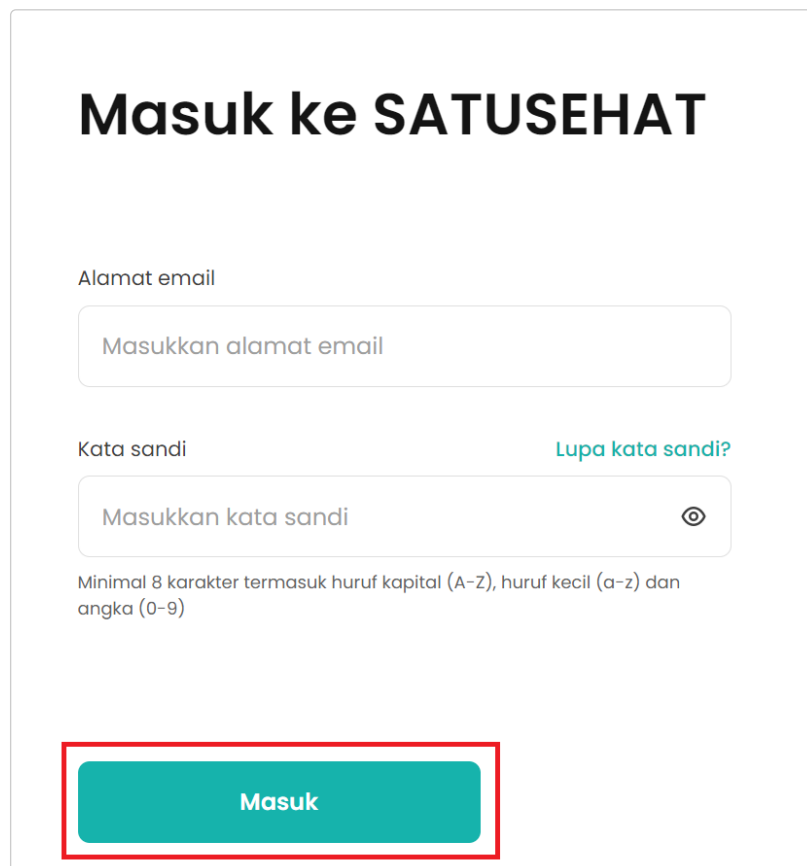
5. Buka *e-mail* aktivasi akun SATUSEHAT, pada *detail*/badan *e-mail* klik **Aktifkan akun** atau dengan menyalin tautan pada browser Anda.



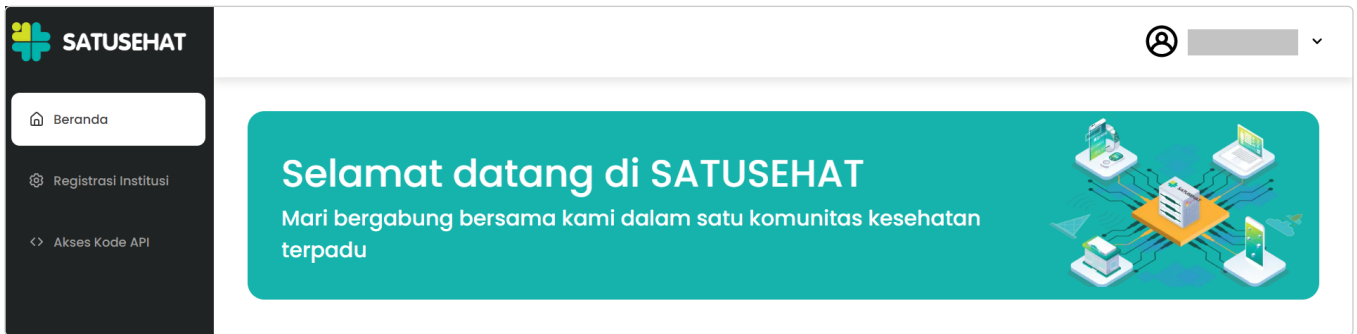
6. Aktivasi akun berhasil, Anda dapat melanjutkan Integrasi Faskes dengan klik **Masuk**.



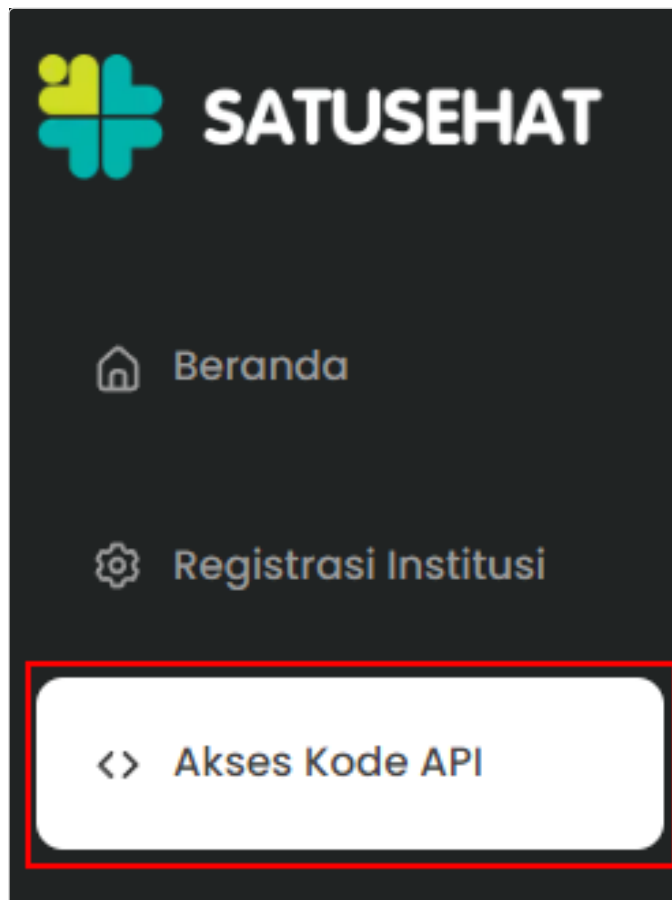
7. Pada laman *login* masukan alamat *e-mail* dan kata sandi yang telah terdaftar, kemudian klik **Masuk**.



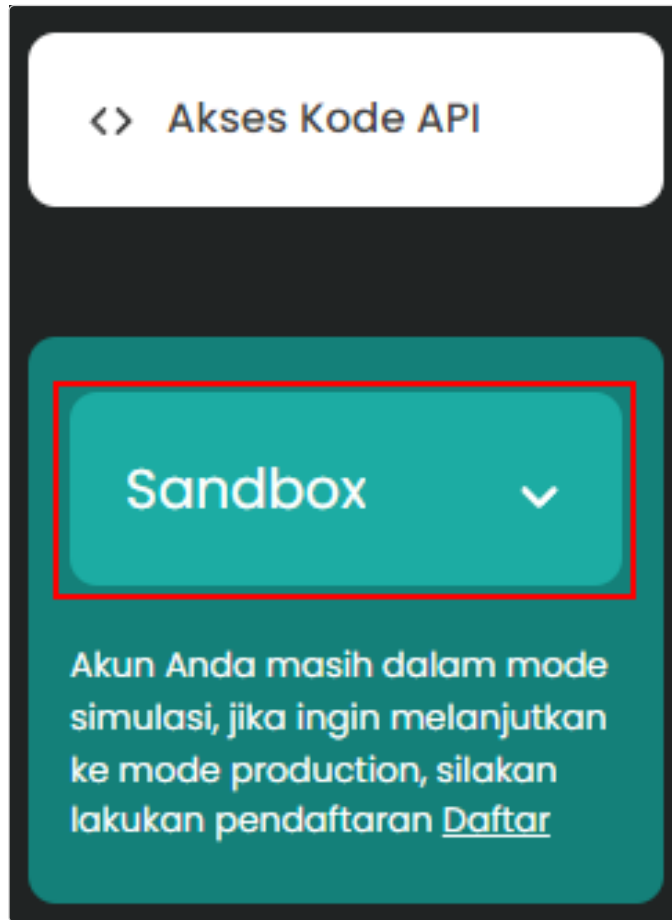
8. Setelah *login*, Anda akan masuk ke laman beranda.



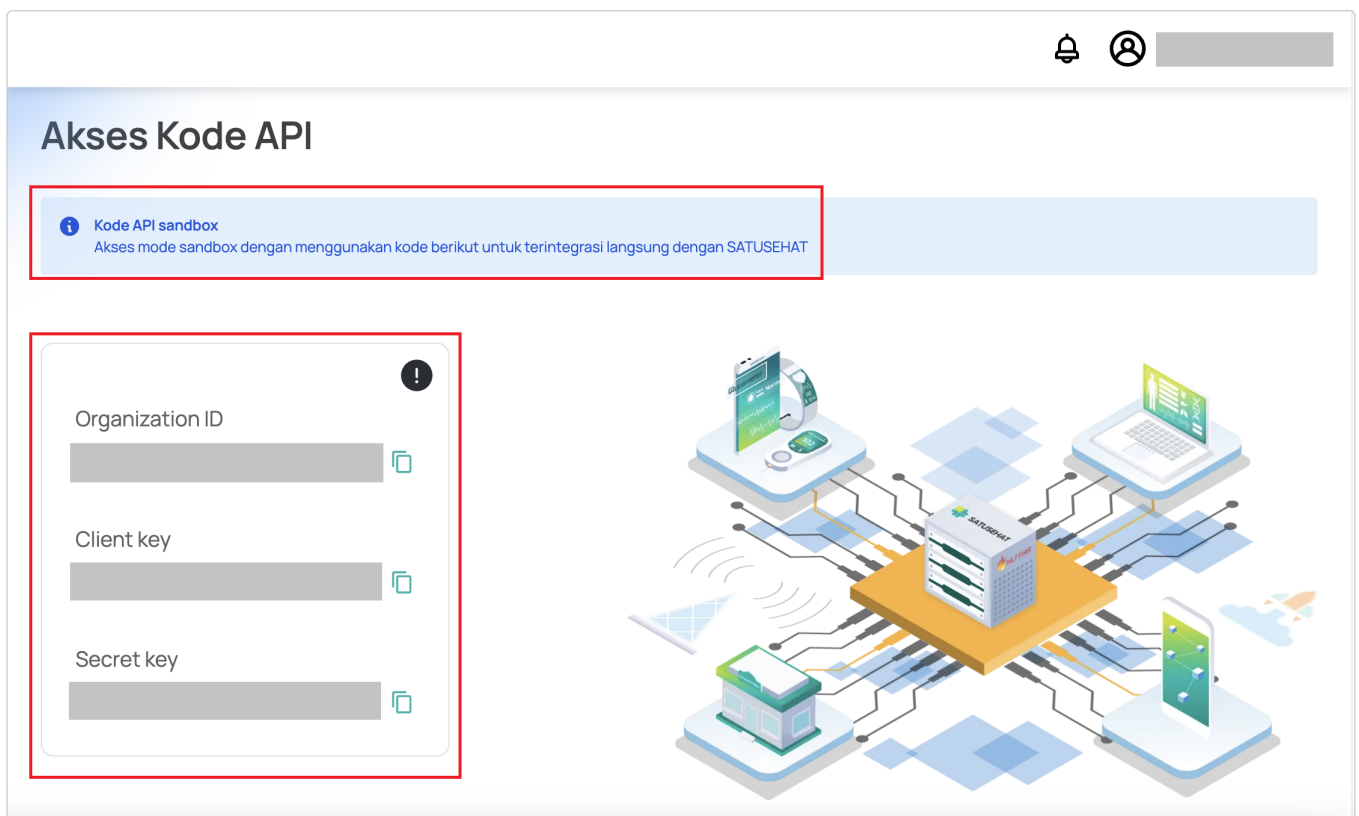
9. Pada *sidebar* menu, klik menu <> **Akses Kode API**.



10. Pada *sidebar* menu, dibawah menu <> **Akses Kode API**. Pilih *endpoint* API yang diinginkan, diantaranya **Sandbox(Development)**, **Staging**, atau **Production**.



11. Anda dapat melihat **Organization ID**, **Client Key**, dan **Secret Key** yang dapat digunakan pada API SATUSEHAT sesuai dengan *endpoint* API yang diinginkan.



4. Autentikasi - Mendapatkan Token

Melakukan proses autentikasi untuk mendapatkan akses token yang akan dipakai pada setiap *request* ReST API Kamus Farmasi dan Alat Kesehatan (KFA) selanjutnya.



Setiap terdapat simbol asterik * sebelum nama variabel atau parameter yang disebutkan, maka variabel atau parameter tersebut bersifat **WAJIB**, **harus ada**, atau **pasti selalu ada**, contoh: **variabel*.

4.1. Request

URL

`https://api-satusehat.kemkes.go.id/oauth2/v1/accesstoken`

HTTP Verb/Method

POST

Header

Nama Parameter	Tipe Data	Keterangan
<i>*Content-Type</i>	string	Mime type dari <i>payload</i> data yang akan dikirimkan di dalam <i>body</i> dalam bentuk <i>URL Encoded</i> , WAJIB diisi dengan <code>application/x-www-form-urlencoded</code>

Query String

Nama Parameter	Tipe Data	Keterangan
<i>*grant_type</i>	string	Tipe permintaan akses (<i>grant</i>) Oauth2, WAJIB diisi dengan <code>client_credentials</code> .

Body (`application/x-www-form-urlencoded`)

Nama Parameter	Tipe Data	Keterangan
<i>*client_id</i>	string	Nilai <i>client ID</i> yang telah didapatkan dari Pusat Data dan Teknologi Informasi - Digital Transformation Office (DTO) Kementerian Kesehatan Republik Indonesia setelah melakukan Pengajuan dan Verifikasi Akses pada website https://satusehat.kemkes.go.id/platform , WAJIB diisi. Nilai ini bisa disamakan seperti <i>username</i> yang akan digunakan untuk akses aplikasi.

Nama Parameter	Tipe Data	Keterangan
* <code>client_secret</code>	<code>string</code>	Nilai <i>client secret</i> yang telah didapatkan dari Pusat Data dan Teknologi Informasi - Digital Transformation Office (DTO) Kementerian Kesehatan Republik Indonesia setelah melakukan Pengajuan dan Verifikasi Akses pada website https://satusehat.kemkes.go.id/platform , WAJIB diisi. Nilai ini bisa disamakan seperti kata sandi (<i>password</i>) yang akan digunakan untuk akses aplikasi.

Contoh Data



Setiap nilai yang dicontohkan atau ditampilkan di dokumentasi ini adalah nilai yang tidak sebenarnya dan tidak dapat dipakai. Nilai-nilai tersebut hanya untuk keperluan contoh saja, tidak untuk dipakai.

```
client_id: <client-id>
client_secret: <client-secret>
```

4.2. Response

Hasil *response*, dengan HTTP *Status Code* berpola `2xx` atau `4xx`, yang dikembalikan dari server mempunyai parameter *Content-Type* dengan nilai `application/json` di salah satu parameter *header*-nya.

2xx Success

Dari hasil *response* ini, **PERLU** disimpan nilai akses token yang didapat dari properti `access_token`, dimana tipe token (lihat properti `token_type`) tersebut adalah `BearerToken`. Nilai akses token tersebut **WAJIB** selalu digunakan sebagai nilai dari *header* `Authorization: Bearer <access_token>` saat melakukan *request* lainnya dari ReST API SATUSEHAT.

Contoh Data



Setiap nilai yang dicontohkan atau ditampilkan di dokumentasi ini adalah nilai yang tidak sebenarnya dan tidak dapat dipakai. Nilai-nilai tersebut hanya untuk keperluan contoh saja, tidak untuk dipakai.

```
{
  "refresh_token_expires_in": "0",
  "api_product_list": "[api-sandbox]",
  "api_product_list_json": [
    "api-sandbox"
  ],
  "organization_name": "ihs-prod-1",
  "developer.email": "<developer-email>",
  "token_type": "BearerToken",
  "issued_at": "1685039759453",
  "client_id": "<client-id>",
  "access_token": "<access-token>",
  "application_name": "22b6ee41-518d-430a-8245-702d7995a337",
  "scope": ""
```

```
"expires_in": "3599",  
"refresh_count": "0",  
"status": "approved"  
}
```

4xx Client Error

Sistem akan mengembalikan pesan *error* bila *client* belum melakukan autentikasi, tidak memiliki akses, menggunakan HTTP *method* yang tidak tepat, atau mengirimkan data dengan format atau ketentuan yang tidak sesuai.

Contoh Data

```
{  
  "resourceType": "OperationOutcome",  
  //data.terkait.resource.OperationOutcome  
}
```

5xx Server Error (Content-Type: text/plain)

Sistem akan mengembalikan pesan *error* bila terjadi kesalahan pada sisi server saat memproses data yang telah dikirimkan.

Contoh Data

Gateway Timeout

4.3. Contoh Penggunaan/Kode



Setiap nilai yang dicontohkan atau ditampilkan di dokumentasi ini adalah nilai yang tidak sebenarnya dan tidak dapat dipakai. Nilai-nilai tersebut hanya untuk keperluan contoh saja, tidak untuk dipakai.

cURL (Windows)

```
curl --insecure --location \  
  --header "Content-Type: application/x-www-form-urlencoded" ^  
  --data-urlencode "client_id=<client-id>" ^  
  --data-urlencode "client_secret=<client-secret>" ^  
  --request POST ^  
  "https://api-satusehat.kemkes.go.id/oauth2/v1/accesstoken?grant_type=client_credentials"
```

cURL (Linux)

```
curl --insecure --location \  
  --header 'Content-Type: application/x-www-form-urlencoded' \  
  --data-urlencode 'client_id=<client-id>' \  
  --data-urlencode 'client_secret=<client-secret>' \  
  --request POST \  
  'https://api-satusehat.kemkes.go.id/oauth2/v1/accesstoken?grant_type=client_credentials'
```

Postman

1. Buat *request* baru menggunakan **New** > **HTTP Request**, atau klik untuk buat tab *request* baru.
2. Masukkan *request* URL

```
https://api-satusehat.kemkes.go.id/oauth2/v1/accesstoken
```

3. Lalu pilih *request method* **POST**.
4. Pada tab **Params**, di bagian **Query Params**:
 - a. masukan nilai **grant_type** pada kotak masukan pada kolom **KEY**,
 - b. lalu masukan nilai **client_credentials`** pada kotak masukan pada kolom **VALUE**.
5. Pada tab **Body**:
 - a. pilih **x-www-form-urlencoded**,
 - b. masukan nilai **client_id** pada kotak masukan pada kolom **KEY**,
 - c. lalu masukan nilai *client ID* yang sudah didapatkan dari **Pusat Data dan Teknologi Informasi - Digital Transformation Office (DTO) Kementerian Kesehatan Republik Indonesia** pada kotak masukan pada kolom **VALUE**,
 - d. selanjutnya masukan nilai **client_secret** pada kotak masukan pada kolom **KEY**,
 - e. terakhir masukan nilai *client secret* yang sudah didapatkan dari **Pusat Data dan Teknologi Informasi - Digital Transformation Office (DTO) Kementerian Kesehatan Republik Indonesia** pada kotak masukan pada kolom **VALUE**.
6. Klik tombol .
7. Hasil *response* akan ditampilkan di bagian **Response**.

C. PRICE

Untuk melakukan transaksi data dari Kamus Farmasi dan Alat Kesehatan (KFA), perlu dilakukan proses autentikasi terlebih dahulu agar mendapatkan akses yang tersedia. Autentikasi yang digunakan oleh Kamus Farmasi dan Alat Kesehatan (KFA) mengikuti standar protokol OAuth 2 dengan tipe pemberian akses (*grant type*) adalah `client_credentials`.

Autentikasi menggunakan *grant type* `client_credentials` adalah proses autentikasi yang dilakukan antara *server to server*, sehingga tidak ada proses registrasi atau *log in* di sini. Autentikasi dengan tipe tersebut hanya memerlukan data berupa `client_id` dan `client_secret`, dimana nilai tersebut didapatkan ketika pihak yang ingin menggunakan atau mengakses Kamus Farmasi dan Alat Kesehatan (KFA) ini telah melakukan pengajuan, terdaftar, serta mendapatkan persetujuan dari **Pusat Data dan Teknologi Informasi - Digital Transformation Office (DTO) Kementerian Kesehatan Republik Indonesia**.

Cara Mendapatkan Nilai dari `client_id` dan `client_secret`

Silakan terlebih dahulu melakukan **Pengajuan dan Verifikasi Akses** pada website <https://satusehat.kemkes.go.id/platform> melalui web *browser* Anda. Untuk informasi lebih lanjut dapat dilihat pada [Pengajuan dan Verifikasi Akses](#)



Setiap teks yang berwarna **biru muda**, dapat diklik untuk melompat ke bagian yang direferensikan.

Pada bagian ini akan dijelaskan spesifikasi untuk **ReST API Kamus Farmasi dan Alat Kesehatan (KFA) Terkait Price**, yang mempunyai *endpoint* berdasarkan jenis lingkungan pengembangannya (*development environment*) yaitu:

- *staging* <https://api-satusehat-stg.dto.kemkes.go.id/kfa>
- *production* <https://api-satusehat.kemkes.go.id/kfa>



Semua penerapan, penjelasan, dan contoh yang akan dibahas akan menggunakan *endpoint production*.

1. Integration - Mendapatkan Harga Produk JKN



Setiap terdapat simbol asterik * sebelum nama variabel atau parameter yang disebutkan, maka variabel atau parameter tersebut bersifat **WAJIB**, **harus ada**, atau **pasti selalu ada**, contoh: **variabel*.

1.1. Request

URL

```
https://api-satusehat.kemkes.go.id/kfa/products/farmalkes-price-jkn
```

HTTP Verb/Method


```
GET
```

Header

Nama Parameter	Tipe Data	Keterangan
<i>*Accept</i>	<i>string</i>	<i>Mime type</i> dari hasil <i>response</i> yang harus dikirimkan oleh server di dalam <i>body</i> dalam format JSON, WAJIB diisi dengan <i>application/json</i> .
<i>*Authorization</i>	<i>string</i>	<i>Header</i> ini WAJIB diisi dengan nilai sesuai format: <i>Bearer <access_token></i> . Nilai dari variabel <i><access_token></i> didapatkan dari properti <i>access_token</i> pada <i>object</i> dari hasil <i>response</i> JSON setelah proses autentikasi.

Query String

Nama Parameter	Tipe Data	Keterangan
<i>*page</i>	<i>number</i>	Isi dengan nomor halaman (<i>page</i>) yang diinginkan. Contoh: <i>1</i> .
<i>*limit</i>	<i>number</i>	Isi dengan banyaknya data yang ingin ditampilkan dalam satu halaman (<i>page</i>). Contoh: <i>50</i> .
<i>*kfa_code</i>	<i>string</i>	Isi dengan kode produk KFA yang diinginkan. Contoh: <i>93004418</i> .

Nama Parameter	Tipe Data	Keterangan
<code>region_code</code>	<code>string</code>	<p>Isi dengan kode regional JKN yang diinginkan.</p> <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;">  <ul style="list-style-type: none"> • regional1: Bali, Banten, Jawa Barat, Jawa Timur, Jakarta, Jawa Tengah, Lampung, Yogyakarta • regional2: Bangka Belitung, Bengkulu, Jambi, Nusa Tenggara Barat, Riau, Sumatra Barat, Sumatra Selatan, Sumatra Utara • regional3: Aceh, Kalimantan Barat, Kalimantan Timur, Kepulauan Riau, Kalimantan Tengah, Sulawesi Utara, Sulawesi Selatan, Sulawesi Tengah • regional4: Gorontalo, Kalimantan Tengah, Kalimantan Utara, Sulawesi Tenggara, Sulawesi Barat • regional5: Maluku, Maluku Utara, Nusa Tenggara Timur, Papua Barat • regional6: Papua Pegunungan, Papua Selatan, Papua Tengah </div> <p>Contoh: <code>regional1</code></p>
<code>document_ref</code>	<code>string</code>	Isi dengan kode dokumen referensi harga JKN.

1.2. Response

Hasil *response*, dengan HTTP *Status Code* berpola `2xx` atau `4xx`, yang dikembalikan dari server mempunyai parameter *Content-Type* dengan nilai `application/json` di salah satu parameter *header*-nya.

2xx Success

Contoh Data



Setiap nilai yang dicontohkan atau ditampilkan di dokumentasi ini adalah nilai yang tidak sebenarnya dan tidak dapat dipakai. Nilai-nilai tersebut hanya untuk keperluan contoh saja, tidak untuk dipakai.

```
{
  "total": 6,
  "page": 1,
  "limit": 10,
  "items": {
    "data": [
      {
        "product_template_name": "Acarbose 100 mg Tablet",
        "kfa_code": "92000372",
        "document_ref": "HK.01.07/MENKES/1905/2023",
        "active": true,
      }
    ]
  }
}
```

```
"region_name": "Regional 1",
"region_code": "regional1",
"start_date": "2023-08-23",
"end_date": null,
"price_unit": 848.0,
"uom_name": "Tablet",
"updated_at": "2023-08-29 04:11:02",
"uom_pack": [
  "Blister",
  "Strip"
],
"province": [
  {
    "province_code": "51",
    "province_name": "Bali"
  },
  {
    "province_code": "36",
    "province_name": "Banten"
  },
  {
    "province_code": "32",
    "province_name": "Jawa Barat"
  },
  {
    "province_code": "35",
    "province_name": "Jawa Timur"
  },
  {
    "province_code": "31",
    "province_name": "DKI Jakarta"
  },
  {
    "province_code": "33",
    "province_name": "Jawa Tengah"
  },
  {
    "province_code": "18",
    "province_name": "Lampung"
  },
  {
    "province_code": "34",
    "province_name": "Yogyakarta"
  }
]
},
/* lompat beberapa data */
{
  "product_template_name": "Acarbose 100 mg Tablet",
  "kfa_code": "92000372",
  "document_ref": "HK.01.07/MENKES/1905/2023",
  "active": true,
  "region_name": "Regional 6",
  "region_code": "regional6",
  "start_date": "2023-08-23",
  "end_date": null,
  "price_unit": 1060.0,
```



```
"uom_name": "Tablet",
"updated_at": "2023-08-29 04:11:02",
"uom_pack": [
  "Blister",
  "Strip"
],
"province": [
  {
    "province_code": "95",
    "province_name": "Papua Pegunungan"
  },
  {
    "province_code": "93",
    "province_name": "Papua Selatan"
  },
  {
    "province_code": "94",
    "province_name": "Papua Tengah"
  }
]
}
}
```

4xx Client Error

Sistem akan mengembalikan pesan *error* bila *client* belum melakukan autentikasi, tidak memiliki akses, menggunakan HTTP *method* yang tidak tepat, atau mengirimkan data dengan format atau ketentuan yang tidak sesuai.

Contoh Data

```
{
  "detail": [{
    "loc": [
      "query",
      "code"
    ],
    "msg": "field required",
    "type": "value_error.missing"
  }
]
}
```

5xx Server Error (Content-Type: text/plain)

Sistem akan mengembalikan pesan *error* bila terjadi kesalahan pada sisi server saat memproses data yang telah dikirimkan.

Contoh Data

```
Gateway Timeout
```

1.3. Contoh Penggunaan/Kode



Setiap nilai yang dicontohkan atau ditampilkan di dokumentasi ini adalah nilai yang tidak sebenarnya dan tidak dapat dipakai. Nilai-nilai tersebut hanya untuk keperluan contoh saja, tidak untuk dipakai.

cURL (Windows)

```
curl --insecure --location ^  
  --header "Authorization: Bearer lWnwCNpXQqdGpyOTvJpyT1iTJeyP" ^  
  --header "Accept: application/json" ^  
  --request GET ^  
  "https://api-satusehat.kemkes.go.id/kfa/products/farmalkes-price-jkn?{path-code}"
```


cURL (Linux)

```
curl --insecure --location \  
  --header 'Authorization: Bearer lWnwCNpXQqdGpyOTvJpyT1iTJeyP' \  
  --header 'Accept: application/json' \  
  --request GET \  
  'https://api-satusehat.kemkes.go.id/kfa/products/farmalkes-price-jkn?{path-code}'
```

Postman

1. Buat *request* baru menggunakan **New** > **HTTP Request**, atau klik  untuk buat tab *request* baru.
2. Masukkan *request* URL

```
https://api-satusehat.kemkes.go.id/kfa/products/farmalkes-price-jkn
```

3. Lalu pilih *request method* **GET**.
4. Pada tab **Auth**:
 - a. pada pilihan **Type**, pilih **Bearer Token**,
 - b. lalu masukan nilai akses token yang sudah didapatkan pada saat autentikasi pada kotak inputan **Token**.
5. Pada tab **Headers**:
 - a. Pada kolom **KEY** isi **Accept**, lalu pada kolom **VALUE** isi dengan **application/json**.
6. Pada tab **Params**, di bagian **Query Params**:
 - a. silakan masukan parameter untuk melakukan pencarian sesuai dengan yang sudah dijelaskan pada bagian *query string* terkait ReST API Kamus Farmasi dan Alat Kesehatan (KFA) ini,
 - b. masukan satu atau lebih nama dari parameter tersebut pada kotak masukan pada kolom **KEY**,
 - c. sedangkan untuk nilainya, masukan pada kotak masukan pada kolom **VALUE**.
7. Klik tombol .
8. Hasil *response* akan ditampilkan di bagian **Response**.

D. PRODUCTS VERSION 1

Untuk melakukan transaksi data dari Kamus Farmasi dan Alat Kesehatan (KFA), perlu dilakukan proses autentikasi terlebih dahulu agar mendapatkan akses yang tersedia. Autentikasi yang digunakan oleh Kamus Farmasi dan Alat Kesehatan (KFA) mengikuti standar protokol OAuth 2 dengan tipe pemberian akses (*grant type*) adalah `client_credentials`.

Autentikasi menggunakan *grant type* `client_credentials` adalah proses autentikasi yang dilakukan antara *server to server*, sehingga tidak ada proses registrasi atau *log in* di sini. Autentikasi dengan tipe tersebut hanya memerlukan data berupa `client_id` dan `client_server`, dimana nilai tersebut didapatkan ketika pihak yang ingin menggunakan atau mengakses Kamus Farmasi dan Alat Kesehatan (KFA) ini telah melakukan pengajuan, terdaftar, serta mendapatkan persetujuan dari **Pusat Data dan Teknologi Informasi - Digital Transformation Office (DTO) Kementerian Kesehatan Republik Indonesia**.

Cara Mendapatkan Nilai dari `client_id` dan `client_secret`

Silakan terlebih dahulu melakukan **Pengajuan dan Verifikasi Akses** pada website <https://satusehat.kemkes.go.id/platform> melalui web *browser* Anda. Untuk informasi lebih lanjut dapat dilihat pada [Pengajuan dan Verifikasi Akses](#)



Setiap teks yang berwarna **biru muda**, dapat diklik untuk melompat ke bagian yang direferensikan.

Pada bagian ini akan dijelaskan spesifikasi untuk **ReST API Kamus Farmasi dan Alat Kesehatan (KFA) Terkait Produk**, yang mempunyai *endpoint* berdasarkan jenis lingkungan pengembangannya (*development environment*) yaitu:

- *staging* <https://api-satusehat-stg.dto.kemkes.go.id/kfa>
- *production* <https://api-satusehat.kemkes.go.id/kfa>



Semua penerapan, penjelasan, dan contoh yang akan dibahas akan menggunakan *endpoint production*.

1. Integration - Pencarian Produk Sesuai ATC dengan Paginasi



Setiap terdapat simbol asterik * sebelum nama variabel atau parameter yang disebutkan, maka variabel atau parameter tersebut bersifat **WAJIB**, **harus ada**, atau **pasti selalu ada**, contoh: `*variabel`.

1.1. Request

URL

```
https://api-satusehat.kemkes.go.id/kfa/products/atc
```

HTTP Verb/Method



```
GET
```

Header

Nama Parameter	Tipe Data	Keterangan
<code>*Accept</code>	<code>string</code>	<i>Mime type</i> dari hasil <i>response</i> yang harus dikirimkan oleh server di dalam <i>body</i> dalam format JSON, WAJIB diisi dengan <code>application/json</code> .
<code>*Authorization</code>	<code>string</code>	<i>Header</i> ini WAJIB diisi dengan nilai sesuai format: <code>Bearer <access_token></code> . Nilai dari variabel <code><access_token></code> didapatkan dari properti <code>access_token</code> pada <i>object</i> dari hasil <i>response</i> JSON setelah proses autentikasi.

Query String

Nama Parameter	Tipe Data	Keterangan
<code>*page</code>	<code>number</code>	Isi dengan nomor halaman (<i>page</i>) yang diinginkan. Contoh: <code>1</code> .
<code>*size</code>	<code>number</code>	Isi dengan banyaknya data yang ingin ditampilkan dalam satu halaman (<i>page</i>). Contoh: <code>50</code> .

Nama Parameter	Tipe Data	Keterangan
*atc_code	string	<p>Isi dengan kode Anatomical Therapeutic Chemical (ATC). Kode unik yang ditetapkan untuk obat menurut organ atau sistem tempat obat bekerja dan cara kerjanya. Sistem klasifikasi dikelola oleh Organisasi Kesehatan Dunia (WHO).</p> <div data-bbox="632 383 1489 1173" style="border: 1px solid #ccc; padding: 10px;">  <ol style="list-style-type: none"> 1. Saluran pencernaan (A) 2. Darah dan organ pembentuk darah (B) 3. Sistem kardiovaskular (C) 4. Kulit (D) 5. Sistem reproduksi (G) 6. Sistem endokrin (H) 7. Anti-infeksi untuk penggunaan sistemik (J) 8. Penyakit ganas dan sistem imun (L) 9. Otot, tulang, dan sendi (M) 10. Otak dan sistem saraf (N) 11. Produk antiparasitik, insektisida, dan repelan (P) 12. Sistem pernafasan (R) 13. Organ sensorik (S) 14. Aneka ragam (V) <p>referensi: Daftar Kode ATC [atc-code]</p> </div> <p>Contoh: L.</p>
*level	number	<p>Dalam sistem klasifikasi Anatomical Therapeutic Chemical (ATC), zat aktif dibagi ke dalam kelompok yang berbeda sesuai dengan organ atau sistem tempat mereka bekerja dan sifat terapeutik, farmakologis, dan kimianya. Obat diklasifikasikan dalam kelompok pada lima tingkat yang berbeda.</p> <div data-bbox="632 1507 1489 1951" style="border: 1px solid #ccc; padding: 10px;">  <ol style="list-style-type: none"> 1. ATC tingkat 1: Sistem ini memiliki empat belas kelompok anatomis atau farmakologis utama (tingkat 1). 2. ATC tingkat 2: Subkelompok Farmakologis atau Terapi 3. ATC tingkat 3 & 4: Subkelompok Kimia, Farmakologi atau Terapi 4. ATC tingkat 5: Substansi kimia <p>referensi: Daftar Level ATC [atc-level]</p> </div> <p>Contoh: 1.</p>

1.2. Response

Hasil *response*, dengan HTTP *Status Code* berpola **2xx** atau **4xx**, yang dikembalikan dari server mempunyai parameter **Content-Type** dengan nilai **application/json** di salah satu parameter *header*-nya.

2xx Success

Contoh Data



Setiap nilai yang dicontohkan atau ditampilkan di dokumentasi ini adalah nilai yang tidak sebenarnya dan tidak dapat dipakai. Nilai-nilai tersebut hanya untuk keperluan contoh saja, tidak untuk dipakai.

```
{
  "atc_code": "L",
  "total": 978,
  "page": 1,
  "size": 50,
  "items": {
    "data": [
      {
        "name": "Leuprorelin Acetate 1,88 mg Serbuk Injeksi (TAPROS, VIAL + AMPUL PELARUT)",
        "kfa_code": "/",
        "active": true,
        "state": "draft",
        "image": null,
        "updated_at": "2022-12-16 07:13:01",
        "farmalkes_type": {
          "code": "medicine",
          "name": "Obat",
          "group": "farmasi"
        },
        "produksi_buatan": "import",
        "nie": "DKI0870700244C1",
        "nama_dagang": "TAPROS",
        "manufacturer": "TAKEDA PHARMACEUTICAL COMPANY",
        "registrar": "TAKEDA INDONESIA",
        "generik": null,
        "rxterm": "leuprolide",
        "dose_per_unit": 1,
        "fix_price": 578067.0,
        "het_price": null,
        "farmalkes_hscore": null,
        "tayang_lkpp": true,
        "kode_lkpp": "48347184",
        "net_weight": null,
        "net_weight_uom_name": "g",
        "volume": null,
        "volume_uom_name": "mL",
        "uom": {
          "name": "Prefilled Syringe"
        }
      },
      {
        "product_template": {
          "kfa_code": "92000732",
          "name": "Leuprorelin Acetate 1,88 mg Serbuk Injeksi",

```

```

        "state": "valid",
        "active": true,
        "display_name": "Leuprorelin Acetate 1,88 mg Serbuk Injeksi",
        "updated_at": "2023-08-29 04:33:05"
    },
    "replacement": {
        "product": null,
        "template": null
    }
},
/*lompat beberapa data*/
{
    "name": "Epirubicin Hydrochloride 2 mg/mL Larutan Injeksi (SANBE FARMA, 25 mL)",
    "kfa_code": "93000220",
    "active": true,
    "state": "valid",
    "image": null,
    "updated_at": "2023-09-01 03:56:41",
    "farmalkes_type": {
        "code": "medicine",
        "name": "Obat",
        "group": "farmasi"
    },
    "produksi_buatan": "lokal",
    "nie": "GKL1222251043A1",
    "nama_dagang": "EPIRUBICIN HCL",
    "manufacturer": "SANBE FARMA (UNIT IV)",
    "registrar": "SANBE FARMA",
    "generik": true,
    "rxterm": "epirubicin",
    "dose_per_unit": 1,
    "fix_price": null,
    "het_price": 1096360.0,
    "farmalkes_hscore": "30049089",
    "tayang_lkpp": null,
    "kode_lkpp": null,
    "net_weight": null,
    "net_weight_uom_name": "g",
    "volume": 25.0,
    "volume_uom_name": "mL",
    "uom": {
        "name": "Vial"
    },
    "product_template": {
        "kfa_code": "92000231",
        "name": "Epirubicin Hydrochloride 2 mg/mL Larutan Injeksi",
        "state": "valid",
        "active": true,
        "display_name": "Epirubicin Hydrochloride 2 mg/mL Larutan Injeksi",
        "updated_at": "2023-08-29 00:49:25"
    },
    "replacement": {
        "product": null,
        "template": null
    }
}
]

```

```
}  
}
```

4xx Client Error

Sistem akan mengembalikan pesan *error* bila *client* belum melakukan autentikasi, tidak memiliki akses, menggunakan HTTP *method* yang tidak tepat, atau mengirimkan data dengan format atau ketentuan yang tidak sesuai.

Contoh Data

```
{  
  "detail": [  
    {  
      "loc": [  
        "Ut nisi amet",  
        "velit nulla quis minim"  
      ],  
      "msg": "cupidatat Excepteur enim in",  
      "type": "voluptate laborum reprehenderit velit"  
    },  
    {  
      "loc": [  
        "id ad",  
        "ad"  
      ],  
      "msg": "cillum",  
      "type": "esse sit"  
    }  
  ]  
}
```

5xx Server Error (Content-Type: text/plain)

Sistem akan mengembalikan pesan *error* bila terjadi kesalahan pada sisi server saat memproses data yang telah dikirimkan.

Contoh Data

Gateway Timeout

1.3. Contoh Penggunaan/Kode



Setiap nilai yang dicontohkan atau ditampilkan di dokumentasi ini adalah nilai yang tidak sebenarnya dan tidak dapat dipakai. Nilai-nilai tersebut hanya untuk keperluan contoh saja, tidak untuk dipakai.

cURL (Windows)

```
curl --insecure --location ^  
  --header "Authorization: Bearer lWnwCNpXQqdGpyOTvJpyT1iTJeyp" ^  
  --header "Accept: application/json" ^
```



```
--request GET ^  
"https://api-satusehat.kemkes.go.id/kfa/products/atc?page=1&size=50&atc_code=L&level=1"
```

cURL (Linux)

```
curl --insecure --location \  
--header 'Authorization: Bearer lWnwCNpXQqdGpyOTvJpyT1iTJey' \  
--header 'Accept: application/json' \  
--request GET \  
'https://api-satusehat.kemkes.go.id/kfa/products/atc?page=1&size=50&atc_code=L&level=1'
```

Postman

1. Buat *request* baru menggunakan **New** > **HTTP Request**, atau klik untuk buat tab *request* baru.
2. Masukkan *request* URL

```
https://api-satusehat.kemkes.go.id/kfa/products/atc
```

3. Lalu pilih *request method* **GET**.
4. Pada tab **Auth**:
 - a. pada pilihan **Type**, pilih **Bearer Token**,
 - b. lalu masukan nilai akses token yang sudah didapatkan pada saat autentikasi pada kotak inputan **Token**.
5. Pada tab **Headers**:
 - a. Pada kolom **KEY** isi **Accept**, lalu pada kolom **VALUE** isi dengan **application/json**.
6. Pada tab **Params**, di bagian **Query Params**:
 - a. silakan masukan parameter untuk melakukan pencarian sesuai dengan yang sudah dijelaskan pada bagian *query string* terkait ReST API Kamus Farmasi dan Alat Kesehatan (KFA) ini,
 - b. masukan satu atau lebih nama dari parameter tersebut pada kotak masukan pada kolom **KEY**,
 - c. sedangkan untuk nilainya, masukan pada kotak masukan pada kolom **VALUE**.
7. Klik tombol .
8. Hasil *response* akan ditampilkan di bagian **Response**.

2. Integration - Mendapatkan ATC Metadata



Setiap terdapat simbol asterik * sebelum nama variabel atau parameter yang disebutkan, maka variabel atau parameter tersebut bersifat **WAJIB**, **harus ada**, atau **pasti selalu ada**, contoh: **variabel*.

2.1. Request

URL

```
https://api-satusehat.kemkes.go.id/kfa/products/atc
```

HTTP Verb/Method



```
GET
```


Header

Nama Parameter	Tipe Data	Keterangan
<i>*Accept</i>	<i>string</i>	<i>Mime type</i> dari hasil <i>response</i> yang harus dikirimkan oleh server di dalam <i>body</i> dalam format JSON, WAJIB diisi dengan <i>application/json</i> .
<i>*Authorization</i>	<i>string</i>	<i>Header</i> ini WAJIB diisi dengan nilai sesuai format: <i>Bearer <access_token></i> . Nilai dari variabel <i><access_token></i> didapatkan dari properti <i>access_token</i> pada <i>object</i> dari hasil <i>response</i> JSON setelah proses autentikasi.

Query String

Nama Parameter	Tipe Data	Keterangan
<i>*page</i>	<i>number</i>	Isi dengan nomor halaman (<i>page</i>) yang diinginkan. Contoh: <i>1</i> .
<i>*size</i>	<i>number</i>	Isi dengan banyaknya data yang ingin ditampilkan dalam satu halaman (<i>page</i>). Contoh: <i>50</i> .

Nama Parameter	Tipe Data	Keterangan
*tag_code	string	<p>Isi dengan kode tag produk yang diinginkan.</p> <div data-bbox="632 255 1489 555" style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;">  <ul style="list-style-type: none"> • ginjal • jantung • kanker • stroke • uronefrologi • leukemia </div> <p>Contoh: kanker.</p>
*level	number	<p>Dalam sistem klasifikasi Anatomical Therapeutic Chemical (ATC), zat aktif dibagi ke dalam kelompok yang berbeda sesuai dengan organ atau sistem tempat mereka bekerja dan sifat terapeutik, farmakologis, dan kimianya. Obat diklasifikasikan dalam kelompok pada lima tingkat yang berbeda.</p> <div data-bbox="632 887 1489 1335" style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;">  <ol style="list-style-type: none"> 1. ATC tingkat 1: Sistem ini memiliki empat belas kelompok anatomis atau farmakologis utama (tingkat 1). 2. ATC tingkat 2: Subkelompok Farmakologis atau Terapi 3. ATC tingkat 3 & 4: Subkelompok Kimia, Farmakologi atau Terapi 4. ATC tingkat 5: Substansi kimia <p>referensi: Daftar Level ATC [atc-level]</p> </div> <p>Contoh: 1.</p>

Nama Parameter	Tipe Data	Keterangan
*parent_code	string	<p>Isi dengan kode parent produk yang diinginkan.</p> <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;">  <ol style="list-style-type: none"> 1. Saluran pencernaan (A) 2. Darah dan organ pembentuk darah (B) 3. Sistem kardiovaskular (C) 4. Kulit (D) 5. Sistem reproduksi (G) 6. Sistem endokrin (H) 7. Anti-infeksi untuk penggunaan sistemik (J) 8. Penyakit ganas dan sistem imun (L) 9. Otot, tulang, dan sendi (M) 10. Otak dan sistem saraf (N) 11. Produk antiparasitik, insektisida, dan repelan (P) 12. Sistem pernafasan (R) 13. Organ sensorik (S) 14. Aneka ragam (V) <p>referensi: Daftar Kode ATC [atc-code]</p> </div> <p>Contoh: A.</p>

2.2. Response

Hasil *response*, dengan HTTP *Status Code* berpola **2xx** atau **4xx**, yang dikembalikan dari server mempunyai parameter **Content-Type** dengan nilai **application/json** di salah satu parameter *header*-nya.

2xx Success

Contoh Data



Setiap nilai yang dicontohkan atau ditampilkan di dokumentasi ini adalah nilai yang tidak sebenarnya dan tidak dapat dipakai. Nilai-nilai tersebut hanya untuk keperluan contoh saja, tidak untuk dipakai.

```
{
  "result": [
    {
      "code": "A",
      "name": "ALIMENTARY TRACT AND METABOLISM",
      "level": "1",
      "schedule_code": "3013",
      "parent": null
    },
    /*lompat beberapa data*/
    {
      "code": false,
```

```
"name": false,  
"level": false,  
"schedule_code": null,  
"parent": null  
}  
]  
}
```

4xx Client Error

Sistem akan mengembalikan pesan *error* bila *client* belum melakukan autentikasi, tidak memiliki akses, menggunakan HTTP *method* yang tidak tepat, atau mengirimkan data dengan format atau ketentuan yang tidak sesuai.

Contoh Data

```
{  
  "detail": [  
    {  
      "loc": [  
        "Ut nisi amet",  
        "velit nulla quis minim"  
      ],  
      "msg": "cupidatat Excepteur enim in",  
      "type": "voluptate laborum reprehenderit velit"  
    },  
    {  
      "loc": [  
        "id ad",  
        "ad"  
      ],  
      "msg": "cillum",  
      "type": "esse sit"  
    }  
  ]  
}
```

5xx Server Error (Content-Type: text/plain)

Sistem akan mengembalikan pesan *error* bila terjadi kesalahan pada sisi server saat memproses data yang telah dikirimkan.

Contoh Data

Gateway Timeout

2.3. Contoh Penggunaan/Kode



Setiap nilai yang dicontohkan atau ditampilkan di dokumentasi ini adalah nilai yang tidak sebenarnya dan tidak dapat dipakai. Nilai-nilai tersebut hanya untuk keperluan contoh saja, tidak untuk dipakai.

cURL (Windows)

```
curl --insecure --location ^  
  --header "Authorization: Bearer lWnwCNpXQqdGpyOTvJpyT1iTJeyP" ^  
  --header "Accept: application/json" ^  
  --request GET ^  
  "https://api-satusehat.kemkes.go.id/kfa/products/atc?{path-code}"
```

cURL (Linux)

```
curl --insecure --location \  
  --header 'Authorization: Bearer lWnwCNpXQqdGpyOTvJpyT1iTJeyP' \  
  --header 'Accept: application/json' \  
  --request GET \  
  'https://api-satusehat.kemkes.go.id/kfa/products/atc?{path-code}'
```

Postman

1. Buat *request* baru menggunakan **New** > **HTTP Request**, atau klik untuk buat tab *request* baru.
2. Masukkan *request* URL

```
https://api-satusehat.kemkes.go.id/kfa/products/atc
```

3. Lalu pilih *request method* **GET**.
4. Pada tab **Auth**:
 - a. pada pilihan **Type**, pilih **Bearer Token**,
 - b. lalu masukan nilai akses token yang sudah didapatkan pada saat autentikasi pada kotak inputan **Token**.
5. Pada tab **Headers**:
 - a. Pada kolom **KEY** isi **Accept**, lalu pada kolom **VALUE** isi dengan **application/json**.
6. Pada tab **Params**, di bagian **Query Params**:
 - a. silakan masukan parameter untuk melakukan pencarian sesuai dengan yang sudah dijelaskan pada bagian *query string* terkait ReST API Kamus Farmasi dan Alat Kesehatan (KFA) ini,
 - b. masukan satu atau lebih nama dari parameter tersebut pada kotak masukan pada kolom **KEY**,
 - c. sedangkan untuk nilainya, masukan pada kotak masukan pada kolom **VALUE**.
7. Klik tombol .
8. Hasil *response* akan ditampilkan di bagian **Response**.

3. Integration - Pencarian Produk Sesuai Tag dengan Paginasi



Setiap terdapat simbol asterik * sebelum nama variabel atau parameter yang disebutkan, maka variabel atau parameter tersebut bersifat **WAJIB**, **harus ada**, atau **pasti selalu ada**, contoh: **variabel*.

3.1. Request

URL

`https://api-satusehat.kemkes.go.id/kfa/products/tag`

HTTP Verb/Method



GET

Header

Nama Parameter	Tipe Data	Keterangan
<i>*Accept</i>	string	Mime type dari hasil <i>response</i> yang harus dikirimkan oleh server di dalam <i>body</i> dalam format JSON, WAJIB diisi dengan <i>application/json</i> .
<i>*Authorization</i>	string	Header ini WAJIB diisi dengan nilai sesuai format: <i>Bearer <access_token></i> . Nilai dari variabel <i><access_token></i> didapatkan dari properti <i>access_token</i> pada <i>object</i> dari hasil <i>response</i> JSON setelah proses autentikasi.

Query String

Nama Parameter	Tipe Data	Keterangan
<i>*page</i>	number	Isi dengan nomor halaman (<i>page</i>) yang diinginkan. Contoh: <i>1</i> .
<i>*size</i>	number	Isi dengan banyaknya data yang ingin ditampilkan dalam satu halaman (<i>page</i>). Contoh: <i>50</i> .

Nama Parameter	Tipe Data	Keterangan
*tag_code	string	<p>Isi dengan kode tag produk yang diinginkan.</p> <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;">  <ol style="list-style-type: none"> 1. Saluran pencernaan (A) 2. Darah dan organ pembentuk darah (B) 3. Sistem kardiovaskular (C) 4. Kulit (D) 5. Sistem reproduksi (G) 6. Sistem endokrin (H) 7. Anti-infeksi untuk penggunaan sistemik (J) 8. Penyakit ganas dan sistem imun (L) 9. Otot, tulang, dan sendi (M) 10. Otak dan sistem saraf (N) 11. Produk antiparasitik, insektisida, dan repelan (P) 12. Sistem pernafasan (R) 13. Organ sensorik (S) 14. Aneka ragam (V) <p>referensi: Daftar Kode ATC [atc-code]</p> </div> <p>Contoh: {qs-code-tag}.</p>
*level	number	<p>Dalam sistem klasifikasi Anatomical Therapeutic Chemical (ATC), zat aktif dibagi ke dalam kelompok yang berbeda sesuai dengan organ atau sistem tempat mereka bekerja dan sifat terapeutik, farmakologis, dan kimianya. Obat diklasifikasikan dalam kelompok pada lima tingkat yang berbeda.</p> <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;">  <ol style="list-style-type: none"> 1. ATC tingkat 1: Sistem ini memiliki empat belas kelompok anatomis atau farmakologis utama (tingkat 1). 2. ATC tingkat 2: Subkelompok Farmakologis atau Terapi 3. ATC tingkat 3 & 4: Subkelompok Kimia, Farmakologi atau Terapi 4. ATC tingkat 5: Substansi kimia <p>referensi: Daftar Level ATC [atc-level]</p> </div> <p>Contoh: kanker.</p>

3.2. Response

Hasil *response*, dengan HTTP *Status Code* berpola **2xx** atau **4xx**, yang dikembalikan dari server mempunyai parameter **Content-Type** dengan nilai **application/json** di salah satu parameter *header*-nya.

2xx Success

Contoh Data



Setiap nilai yang dicontohkan atau ditampilkan di dokumentasi ini adalah nilai yang tidak sebenarnya dan tidak dapat dipakai. Nilai-nilai tersebut hanya untuk keperluan contoh saja, tidak untuk dipakai.

```
{
  "tag_code": "kanker",
  "total": 1324,
  "page": 1,
  "size": 50,
  "items": {
    "data": [
      {
        "name": "Foscarnet 24 mg/mL Injeksi (Umum)",
        "kfa_code": "/",
        "active": true,
        "state": "draft",
        "image": null,
        "updated_at": "2023-03-14 13:30:02",
        "farmalkes_type": {
          "code": "medicine",
          "name": "Obat",
          "group": "farmasi"
        },
        "produksi_buatan": null,
        "nie": null,
        "nama_dagang": null,
        "manufacturer": null,
        "registrar": null,
        "generik": null,
        "rxterm": "foscarnet",
        "dose_per_unit": 1,
        "fix_price": 1.0,
        "het_price": null,
        "farmalkes_hscore": null,
        "tayang_lkpp": null,
        "kode_lkpp": null,
        "net_weight": null,
        "net_weight_uom_name": "g",
        "volume": null,
        "volume_uom_name": "mL",
        "uom": {
          "name": "Botol"
        },
        "product_template": {
          "kfa_code": "92004216",
          "name": "Foscarnet 24 mg/mL Injeksi",
          "state": "valid",
          "active": true,
          "display_name": "Foscarnet 24 mg/mL Injeksi",
          "updated_at": "2023-03-14 13:30:17"
        }
      }
    ]
  }
}
```

```
"replacement": {
  "product": null,
  "template": null
},
/*lompat beberapa data*/
{
  "name": "ECG 12 Channel (BTL, Wireless, Trolley + Spiro upgrade)",
  "kfa_code": "83004842",
  "active": true,
  "state": "valid",
  "image": null,
  "updated_at": "2022-11-28 03:14:41",
  "farmalkes_type": {
    "code": "device",
    "name": "Alat Kesehatan",
    "group": "alkes"
  },
  "produksi_buatan": "lokal",
  "nie": "AKD 20502220148",
  "nama_dagang": "BTL-08 MT Plus",
  "manufacturer": "PT. BOLD TECHNOLOGIES LEADING INDONESIA",
  "registrar": "PT. BOLD TECHNOLOGIES LEADING INDONESIA",
  "generik": null,
  "rxterm": null,
  "dose_per_unit": 1,
  "fix_price": 101500000.0,
  "het_price": null,
  "farmalkes_hrcode": null,
  "tayang_lkpp": true,
  "kode_lkpp": "1437299",
  "net_weight": null,
  "net_weight_uom_name": "g",
  "volume": null,
  "volume_uom_name": "mL",
  "uom": {
    "name": "Units"
  },
  "product_template": {
    "kfa_code": "82000160",
    "name": "ECG 12 Channel",
    "state": "valid",
    "active": true,
    "display_name": "ECG 12 Channel",
    "updated_at": "2023-08-29 00:49:25"
  },
  "replacement": {
    "product": null,
    "template": null
  }
}
]
```

4xx Client Error

Sistem akan mengembalikan pesan *error* bila *client* belum melakukan autentikasi, tidak memiliki akses, menggunakan HTTP *method* yang tidak tepat, atau mengirimkan data dengan format atau ketentuan yang tidak sesuai.

Contoh Data

```
{
  "detail": [
    {
      "loc": [
        "Ut nisi amet",
        "velit nulla quis minim"
      ],
      "msg": "cupidatat Excepteur enim in",
      "type": "voluptate laborum reprehenderit velit"
    },
    {
      "loc": [
        "id ad",
        "ad"
      ],
      "msg": "cillum",
      "type": "esse sit"
    }
  ]
}
```

5xx Server Error (Content-Type: text/plain)

Sistem akan mengembalikan pesan *error* bila terjadi kesalahan pada sisi server saat memproses data yang telah dikirimkan.

Contoh Data

Gateway Timeout

3.3. Contoh Penggunaan/Kode



Setiap nilai yang dicontohkan atau ditampilkan di dokumentasi ini adalah nilai yang tidak sebenarnya dan tidak dapat dipakai. Nilai-nilai tersebut hanya untuk keperluan contoh saja, tidak untuk dipakai.

cURL (Windows)

```
curl --insecure --location ^
--header "Authorization: Bearer lWnwCNpXQqdGpyOTvJpyT1iTJey" ^
--header "Accept: application/json" ^
--request GET ^
"https://api-satusehat.kemkes.go.id/kfa/products/tag?{path-code}"
```

cURL (Linux)

```
curl --insecure --location \  
  --header 'Authorization: Bearer lWnwCNpXQqdGpy0TvJpyT1iTJey' \  
  --header 'Accept: application/json' \  
  --request GET \  
  'https://api-satusehat.kemkes.go.id/kfa/products/tag?{path-code}'
```

Postman

1. Buat *request* baru menggunakan **New** > **HTTP Request**, atau klik untuk buat tab *request* baru.
2. Masukkan *request* URL

```
https://api-satusehat.kemkes.go.id/kfa/products/tag
```

3. Lalu pilih *request method* **GET**.
4. Pada tab **Auth**:
 - a. pada pilihan **Type**, pilih **Bearer Token**,
 - b. lalu masukan nilai akses token yang sudah didapatkan pada saat autentikasi pada kotak inputan **Token**.
5. Pada tab **Headers**:
 - a. Pada kolom **KEY** isi **Accept**, lalu pada kolom **VALUE** isi dengan **application/json**.
6. Pada tab **Params**, di bagian **Query Params**:
 - a. silakan masukan parameter untuk melakukan pencarian sesuai dengan yang sudah dijelaskan pada bagian *query string* terkait ReST API Kamus Farmasi dan Alat Kesehatan (KFA) ini,
 - b. masukan satu atau lebih nama dari parameter tersebut pada kotak masukan pada kolom **KEY**,
 - c. sedangkan untuk nilainya, masukan pada kotak masukan pada kolom **VALUE**.
7. Klik tombol .
8. Hasil *response* akan ditampilkan di bagian **Response**.

4. Integration - Mendapatkan TAG Metadata



Setiap terdapat simbol asterik * sebelum nama variabel atau parameter yang disebutkan, maka variabel atau parameter tersebut bersifat **WAJIB**, **harus ada**, atau **pasti selalu ada**, contoh: **variabel*.

4.1. Request

URL

```
https://api-satusehat.kemkes.go.id/kfa/products/tag
```

HTTP Verb/Method



```
GET
```

Header

Nama Parameter	Tipe Data	Keterangan
<i>*Accept</i>	<i>string</i>	<i>Mime type</i> dari hasil <i>response</i> yang harus dikirimkan oleh server di dalam <i>body</i> dalam format JSON, WAJIB diisi dengan <i>application/json</i> .
<i>*Authorization</i>	<i>string</i>	<i>Header</i> ini WAJIB diisi dengan nilai sesuai format: <i>Bearer <access_token></i> . Nilai dari variabel <i><access_token></i> didapatkan dari properti <i>access_token</i> pada <i>object</i> dari hasil <i>response</i> JSON setelah proses autentikasi.

Query String

Nama Parameter	Tipe Data	Keterangan
<i>*page</i>	<i>number</i>	Isi dengan nomor halaman (<i>page</i>) yang diinginkan. Contoh: <i>1</i> .
<i>*size</i>	<i>number</i>	Isi dengan banyaknya data yang ingin ditampilkan dalam satu halaman (<i>page</i>). Contoh: <i>50</i> .

Nama Parameter	Tipe Data	Keterangan
*tag_code	string	<p>Isi dengan kode tag produk yang diinginkan.</p> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;">  <ul style="list-style-type: none"> • ginjal • jantung • kanker • stroke • uronefrologi • leukemia </div> <p>Contoh: kanker.</p>
*level	number	<p>Dalam sistem klasifikasi Anatomical Therapeutic Chemical (ATC), zat aktif dibagi ke dalam kelompok yang berbeda sesuai dengan organ atau sistem tempat mereka bekerja dan sifat terapeutik, farmakologis, dan kimianya. Obat diklasifikasikan dalam kelompok pada lima tingkat yang berbeda.</p> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;">  <ol style="list-style-type: none"> 1. ATC tingkat 1: Sistem ini memiliki empat belas kelompok anatomis atau farmakologis utama (tingkat 1). 2. ATC tingkat 2: Subkelompok Farmakologis atau Terapi 3. ATC tingkat 3 & 4: Subkelompok Kimia, Farmakologi atau Terapi 4. ATC tingkat 5: Substansi kimia <p>referensi: Daftar Level ATC [atc-level]</p> </div> <p>Contoh: 1.</p>
*parent_code	string	<p>Isi dengan kode parent dari produk yang diinginkan.</p> <p>Contoh: kanker.</p>

4.2. Response

Hasil *response*, dengan HTTP *Status Code* berpola **2xx** atau **4xx**, yang dikembalikan dari server mempunyai parameter **Content-Type** dengan nilai **application/json** di salah satu parameter *header*-nya.

2xx Success

Contoh Data



Setiap nilai yang dicontohkan atau ditampilkan di dokumentasi ini adalah nilai yang tidak sebenarnya dan tidak dapat dipakai. Nilai-nilai tersebut hanya untuk keperluan contoh saja, tidak untuk dipakai.

```
{
  "result": [
    {
      "code": "ginjal",
      "name": "Ginjal",
      "level": "1",
      "description": false,
      "parent": null
    },
    /*lompat beberapa data*/
    {
      "code": "leukemia",
      "name": "Leukemia",
      "level": "2",
      "description": false,
      "parent": {
        "code": "kanker",
        "name": "Kanker",
        "level": "1"
      }
    }
  ]
}
```

4xx Client Error

Sistem akan mengembalikan pesan *error* bila *client* belum melakukan autentikasi, tidak memiliki akses, menggunakan HTTP *method* yang tidak tepat, atau mengirimkan data dengan format atau ketentuan yang tidak sesuai.

Contoh Data

```
{
  "detail": [
    {
      "loc": [
        "Ut nisi amet",
        "velit nulla quis minim"
      ],
      "msg": "cupidatat Excepteur enim in",
      "type": "voluptate laborum reprehenderit velit"
    },
    {
      "loc": [
        "id ad",
        "ad"
      ],
      "msg": "cillum",
      "type": "esse sit"
    }
  ]
}
```

5xx Server Error (Content-Type: text/plain)

Sistem akan mengembalikan pesan *error* bila terjadi kesalahan pada sisi server saat memproses data yang telah dikirimkan.

Contoh Data

Gateway Timeout

4.3. Contoh Penggunaan/Kode



Setiap nilai yang dicontohkan atau ditampilkan di dokumentasi ini adalah nilai yang tidak sebenarnya dan tidak dapat dipakai. Nilai-nilai tersebut hanya untuk keperluan contoh saja, tidak untuk dipakai.

cURL (Windows)

```
curl --insecure --location ^  
  --header "Authorization: Bearer lWnwCNpXQqdGpyOTvJpyT1iTJeyP" ^  
  --header "Accept: application/json" ^  
  --request GET ^  
  "https://api-satusehat.kemkes.go.id/kfa/products/tag?{path-code}"
```

cURL (Linux)

```
curl --insecure --location \  
  --header 'Authorization: Bearer lWnwCNpXQqdGpyOTvJpyT1iTJeyP' \  
  --header 'Accept: application/json' \  
  --request GET \  
  'https://api-satusehat.kemkes.go.id/kfa/products/tag?{path-code}'
```

Postman

1. Buat *request* baru menggunakan **New** > **HTTP Request**, atau klik  untuk buat tab *request* baru.
2. Masukkan *request* URL

```
https://api-satusehat.kemkes.go.id/kfa/products/tag
```

3. Lalu pilih *request method* **GET**.
4. Pada tab **Auth**:
 - a. pada pilihan **Type**, pilih **Bearer Token**,
 - b. lalu masukan nilai akses token yang sudah didapatkan pada saat autentikasi pada kotak inputan **Token**.
5. Pada tab **Headers**:
 - a. Pada kolom **KEY** isi **Accept**, lalu pada kolom **VALUE** isi dengan **application/json**.
6. Pada tab **Params**, di bagian **Query Params**:
 - a. silakan masukan parameter untuk melakukan pencarian sesuai dengan yang sudah dijelaskan pada bagian *query string* terkait ReST API Kamus Farmasi dan Alat Kesehatan (KFA) ini,

- b. masukan satu atau lebih nama dari parameter tersebut pada kotak masukan pada kolom **KEY**,
 - c. sedangkan untuk nilainya, masukan pada kotak masukan pada kolom **VALUE**.
7. Klik tombol .
 8. Hasil *response* akan ditampilkan di bagian **Response**.

E. PRODUCTS VERSION 2

Untuk melakukan transaksi data dari Kamus Farmasi dan Alat Kesehatan (KFA), perlu dilakukan proses autentikasi terlebih dahulu agar mendapatkan akses yang tersedia. Autentikasi yang digunakan oleh Kamus Farmasi dan Alat Kesehatan (KFA) mengikuti standar protokol OAuth 2 dengan tipe pemberian akses (*grant type*) adalah `client_credentials`.

Autentikasi menggunakan *grant type* `client_credentials` adalah proses autentikasi yang dilakukan antara *server to server*, sehingga tidak ada proses registrasi atau *log in* di sini. Autentikasi dengan tipe tersebut hanya memerlukan data berupa `client_id` dan `client_server`, dimana nilai tersebut didapatkan ketika pihak yang ingin menggunakan atau mengakses Kamus Farmasi dan Alat Kesehatan (KFA) ini telah melakukan pengajuan, terdaftar, serta mendapatkan persetujuan dari **Pusat Data dan Teknologi Informasi - Digital Transformation Office (DTO) Kementerian Kesehatan Republik Indonesia**.

Cara Mendapatkan Nilai dari `client_id` dan `client_secret`

Silakan terlebih dahulu melakukan **Pengajuan dan Verifikasi Akses** pada website <https://satusehat.kemkes.go.id/platform> melalui web *browser* Anda. Untuk informasi lebih lanjut dapat dilihat pada [Pengajuan dan Verifikasi Akses](#)



Setiap teks yang berwarna **biru muda**, dapat diklik untuk melompat ke bagian yang direferensikan.

Pada bagian ini akan dijelaskan spesifikasi untuk **ReST API Kamus Farmasi dan Alat Kesehatan (KFA) Terkait Produk**, yang mempunyai *endpoint* berdasarkan jenis lingkungan pengembangannya (*development environment*) yaitu:

- *staging* <https://api-satusehat-stg.dto.kemkes.go.id/kfa-v2>
- *production* <https://api-satusehat.kemkes.go.id/kfa-v2>



Semua penerapan, penjelasan, dan contoh yang akan dibahas akan menggunakan *endpoint production*.

1. Integration - Mendapatkan Detail Produk



Setiap terdapat simbol asterik * sebelum nama variabel atau parameter yang disebutkan, maka variabel atau parameter tersebut bersifat **WAJIB**, **harus ada**, atau **pasti selalu ada**, contoh: **variabel*.

1.1. Request

URL

`https://api-satusehat.kemkes.go.id/kfa-v2/products`


HTTP Verb/Method

GET

Header

Nama Parameter	Tipe Data	Keterangan
<i>*Accept</i>	string	Mime type dari hasil <i>response</i> yang harus dikirimkan oleh server di dalam <i>body</i> dalam format JSON, WAJIB diisi dengan <i>application/json</i> .
<i>*Authorization</i>	string	Header ini WAJIB diisi dengan nilai sesuai format: <i>Bearer <access_token></i> . Nilai dari variabel <i><access_token></i> didapatkan dari properti <i>access_token</i> pada <i>object</i> dari hasil <i>response</i> JSON setelah proses autentikasi.

Query String

Nama Parameter	Tipe Data	Keterangan
<i>*identifier</i>	string	<p>Isi sumber data yang ingin digunakan, seperti: <i>kfa</i>, <i>nie</i>, atau <i>lkpp</i>.</p> <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;">  <ul style="list-style-type: none"> nie: Data Nomor Izin Edar (NIE) yang bersumber dari BPOM. lkpp: Data inventaris, distribusi, pengelolaan, dan harga obat yang beredar bersumber dari Lembaga Kebijakan Pengadaan Barang/Jasa Pemerintah (LKPP). kfa: Data kode unik produk farmasi dan alat kesehatan yang bersumber pada Kamus Farmasi dan Alat Kesehatan (KFA). </div> <p>Contoh: <i>kfa</i>.</p>

Nama Parameter	Tipe Data	Keterangan
*code	string	Isi kode dari produk yang akan dicari. Contoh: 93004418.

1.2. Response

Hasil *response*, dengan HTTP *Status Code* berpola *2xx* atau *4xx*, yang dikembalikan dari server mempunyai parameter *Content-Type* dengan nilai *application/json* di salah satu parameter *header*-nya.

2xx Success

Contoh Data



Setiap nilai yang dicontohkan atau ditampilkan di dokumentasi ini adalah nilai yang tidak sebenarnya dan tidak dapat dipakai. Nilai-nilai tersebut hanya untuk keperluan contoh saja, tidak untuk dipakai.

```
{
  "search_code": "93015993",
  "search_identifier": "kfa",
  "result": {
    "name": "Abacavir Sulfate 300 mg Tablet Salut Selaput (KIMIA FARMA)",
    "kfa_code": "93015993",
    "active": true,
    "state": "valid",
    "image": null,
    "updated_at": "2023-09-21 07:17:24",
    "farmalkes_type": {
      "code": "medicine",
      "name": "Obat",
      "group": "farmasi"
    },
    "ucum": {
      "cs_code": "mg",
      "name": "milligram"
    },
    "dosage_form": {
      "code": "BS077",
      "name": "Tablet Salut Selaput"
    },
    "controlled_drug": {
      "code": "3",
      "name": "Obat Keras"
    },
    "rute_pemberian": {
      "code": "0",
      "name": "Oral"
    },
    "uom": {
      "name": "Tablet"
    },
    "produksi_buatan": "lokal",
    "nie": "GKL2012431917A1",
    "nama_dagang": "ABACAVIR SULFATE",
    "manufacturer": "KIMIA FARMA TBK",
    "registrasi": "KIMIA FARMA Tbk.",
    "generik": true,
    "rxterm": "abacavir",
    "dose_per_unit": 1,
  }
}
```

```
"fix_price": 7215.0,  
"het_price": 13297.0,  
"farmalkes_hscore": null,  
"tayang_lkpp": true,  
"kode_lkpp": "45463910",  
"score_tkdn": null,  
"score_bmp": null,  
"score_tkdn_bmp": null,  
"med_dev_jenis": null,  
"med_dev_subkategori": null,  
"med_dev_kategori": null,  
"med_dev_kelas_risiko": null,  
"klasifikasi_izin": null,  
"net_weight": null,  
"net_weight_uom_name": "g",  
"volume": null,  
"volume_uom_name": "mL",  
"atc_ddd": {  
  "name": "0.6 g - 0"  
},  
"atc_l1": {  
  "name": "ANTIINFECTIVES FOR SYSTEMIC USE",  
  "code": "J",  
  "level": "1",  
  "parent_code": false,  
  "comment": null  
},  
"atc_l2": {  
  "name": "ANTIVIRALS FOR SYSTEMIC USE",  
  "code": "J05",  
  "level": "2",  
  "parent_code": "J",  
  "comment": null  
},  
"atc_l3": {  
  "name": "DIRECT ACTING ANTIVIRALS",  
  "code": "J05A",  
  "level": "3",  
  "parent_code": "J05",  
  "comment": null  
},  
"atc_l4": {  
  "name": "Nucleoside and nucleotide reverse transcriptase inhibitors",  
  "code": "J05AF",  
  "level": "4",  
  "parent_code": "J05A",  
  "comment": null  
},  
"atc_l5": {  
  "name": "Abacavir",  
  "code": "J05AF06",  
  "level": "5",  
  "parent_code": "J05AF",  
  "comment": null  
},  
"description": "<p>Abacavir secara kompetitif menghambat reverse transcriptase retrovirus, mengganggu DNA polimerase yang bergantung pada RNA virus HIV yang mengakibatkan penghambatan replikasi virus.<br></p>",  
"indication": "<p>Infeksi HIV</p><p>Dewasa: Dikombinasikan dengan antiretroviral lain: 300 mg dua kali sehari atau 600 mg sekali sehari.</p><p>Anak: 3 bln, berat badan 14 kg sampai <20 kg: 150 mg dua kali sehari atau 300 mg sekali sehari; 20 kg sampai <25 kg: 150 mg di pagi hari dan 300 mg di malam hari atau 450 mg sekali sehari; >25 kg: Sama seperti dosis dewasa.</p>",  
"warning": "<p>Pasien dengan faktor risiko penyakit hati (misalnya obesitas) dan mereka yang memiliki faktor risiko penyakit jantung koroner (misalnya hipertensi, DM, merokok). Gangguan ginjal atau hati ringan. Kehamilan.<br></p>",  
"side_effect": "<p>Demam, ruam, batuk, sesak, lesu, malaise, sakit kepala, mialgia, gangguan GI, terutama mual, muntah, diare dan sakit perut; pankreatitis dan peningkatan nilai enzim hati, osteonekrosis, sindrom pemulihan kekebalan, MI, sindrom lipodistrofi. Jarang, eritema multiforme, sindrom Stevens-Johnson, nekrolisis epidermal toksik.</p><p>Berpotensi Fatal: Reaksi hipersensitivitas yang serius dan fatal dengan keterlibatan beberapa organ, asidosis laktat, dan hepatomegali berat dengan steatosis.</p>",  
"identifier_ids": [
```

```
{
  {
    "name": "ABACAVIR SULFATE",
    "code": "GKL2012431917A1",
    "source_name": "NIE BPOM",
    "url": null
  },
  {
    "name": "Abacavir Sulfate 300 mg Tablet Salut Selaput",
    "code": "93015993",
    "source_name": "Kamus Farmalkes (KFA - IHS)",
    "url": null
  }
],
"packaging_ids": [
  {
    "name": "Dus isi 60",
    "kfa_code": "94021264",
    "pack_price": 0.0,
    "uom_id": "Tablet",
    "qty": 60.0
  }
],
"product_template": {
  "kfa_code": "92000888",
  "name": "Abacavir Sulfate 300 mg Tablet Salut Selaput",
  "state": "valid",
  "active": true,
  "display_name": "Abacavir Sulfate 300 mg Tablet Salut Selaput",
  "updated_at": "2023-08-29 00:49:25"
},
"active_ingredients": [
  {
    "kfa_code": "91000651",
    "active": true,
    "state": "valid",
    "zat_aktif": "Abacavir",
    "kekuatan_zat_aktif": "300 mg",
    "updated_at": "2022-11-17 10:56:16"
  }
],
"dosage_usage": [],
"cvx_info": {},
"replacement": {
  "product": null,
  "template": null
},
"tags": []
}
```

4xx Client Error

Sistem akan mengembalikan pesan *error* bila *client* belum melakukan autentikasi, tidak memiliki akses, menggunakan HTTP *method* yang tidak tepat, atau mengirimkan data dengan format atau ketentuan yang tidak sesuai.

Contoh Data

```
{
  "detail": [{
    "loc": [
      "query",
      "identifier"
    ],
  ]
}
```

```
"msg": "field required",  
"type": "value_error.missing"  
}, {  
  "loc": [  
    "query",  
    "code"  
  ],  
  "msg": "field required",  
  "type": "value_error.missing"  
}  
]  
}
```

5xx Server Error (Content-Type: text/plain)

Sistem akan mengembalikan pesan *error* bila terjadi kesalahan pada sisi server saat memproses data yang telah dikirimkan.

Contoh Data

Gateway Timeout

1.3. Contoh Penggunaan/Kode



Setiap nilai yang dicontohkan atau ditampilkan di dokumentasi ini adalah nilai yang tidak sebenarnya dan tidak dapat dipakai. Nilai-nilai tersebut hanya untuk keperluan contoh saja, tidak untuk dipakai.

cURL (Windows)

```
curl --insecure --location ^  
  --header "Authorization: Bearer lWnwCNpXQqdGpyOTvJpyT1iTJeyP" ^  
  --header "Accept: application/json" ^  
  --request GET ^  
  "https://api-satusehat.kemkes.go.id/kfa-v2/products?identifier=kfa&code=93004418"
```

cURL (Linux)

```
curl --insecure --location \  
  --header 'Authorization: Bearer lWnwCNpXQqdGpyOTvJpyT1iTJeyP' \  
  --header 'Accept: application/json' \  
  --request GET \  
  'https://api-satusehat.kemkes.go.id/kfa-v2/products?identifier=kfa&code=93004418'
```

Postman

1. Buat *request* baru menggunakan **New** > **HTTP Request**, atau klik  untuk buat tab *request* baru.
2. Masukkan *request* URL

https://api-satusehat.kemkes.go.id/kfa-v2/products

3. Lalu pilih *request method* **GET**.
4. Pada tab **Auth**:
 - a. pada pilihan **Type**, pilih **Bearer Token**,
 - b. lalu masukan nilai akses token yang sudah didapatkan pada saat autentikasi pada kotak inputan **Token**.
5. Pada tab **Headers**:
 - a. Pada kolom **KEY** isi **Accept**, lalu pada kolom **VALUE** isi dengan **application/json**.
6. Pada tab **Params**, di bagian **Query Params**:
 - a. silakan masukan parameter untuk melakukan pencarian sesuai dengan yang sudah dijelaskan pada bagian *query string* terkait ReST API Kamus Farmasi dan Alat Kesehatan (KFA) ini,
 - b. masukan satu atau lebih nama dari parameter tersebut pada kotak masukan pada kolom **KEY**,
 - c. sedangkan untuk nilainya, masukan pada kotak masukan pada kolom **VALUE**.
7. Klik tombol **Send**.
8. Hasil *response* akan ditampilkan di bagian **Response**.

2. Integration - Pencarian Produk dengan Paginasi



Setiap terdapat simbol asterik * sebelum nama variabel atau parameter yang disebutkan, maka variabel atau parameter tersebut bersifat **WAJIB**, **harus ada**, atau **pasti selalu ada**, contoh: **variabel*.

2.1. Request

URL

```
https://api-satusehat.kemkes.go.id/kfa-v2/products/all
```

HTTP Verb/Method

```
GET
```

Header

Nama Parameter	Tipe Data	Keterangan
<i>*Accept</i>	<i>string</i>	<i>Mime type</i> dari hasil <i>response</i> yang harus dikirimkan oleh server di dalam <i>body</i> dalam format JSON, WAJIB diisi dengan <i>application/json</i> .
<i>*Authorization</i>	<i>string</i>	<i>Header</i> ini WAJIB diisi dengan nilai sesuai format: <i>Bearer <access_token></i> . Nilai dari variabel <i><access_token></i> didapatkan dari properti <i>access_token</i> pada <i>object</i> dari hasil <i>response</i> JSON setelah proses autentikasi.

Query String

Nama Parameter	Tipe Data	Keterangan
<i>*page</i>	<i>number</i>	Isi dengan nomor halaman (<i>page</i>) yang diinginkan. Contoh: <i>1</i> .
<i>*size</i>	<i>number</i>	Isi dengan banyaknya data yang ingin ditampilkan dalam satu halaman (<i>page</i>). Contoh: <i>100</i> .
<i>*product_type</i>	<i>string</i>	Isi dengan kategori/jenis produk yang diinginkan. Contoh: <i>farmasi</i> .
<i>from_date</i>	<i>string</i>	Isi dengan waktu mulai dengan format <i>YYYY-MM-DD</i> Contoh: <i>2023-06-26</i> .
<i>to_date</i>	<i>string</i>	Isi dengan waktu selesai dengan format <i>YYYY-MM-DD</i> Contoh: <i>2023-06-26</i> .

Nama Parameter	Tipe Data	Keterangan
<code>farmalkes_type</code>	<code>string</code>	Isi dengan kategori/jenis farmalkes yang diinginkan. Contoh: <code>vaccine</code> .
<code>keyboard</code>	<code>string</code>	Isi dengan kategori/jenis produk yang diinginkan. Contoh: <code>glove</code> .
<code>template_code</code>	<code>string</code>	Isi dengan kode produk virtual/template (PAV) KFA yang diinginkan. Contoh: <code>92xxxxxx</code> untuk farmasi atau <code>82xxxxxx</code> untuk alkes.
<code>packaging_code</code>	<code>string</code>	Isi dengan kode kemasan (PAK) KFA yang diinginkan. Contoh: <code>94xxxxxx</code> untuk farmasi atau <code>84xxxxxx</code> untuk alkes.

2.2. Response

Hasil *response*, dengan HTTP *Status Code* berpola `2xx` atau `4xx`, yang dikembalikan dari server mempunyai parameter *Content-Type* dengan nilai `application/json` di salah satu parameter *header*-nya.

2xx Success

Contoh Data



Setiap nilai yang dicontohkan atau ditampilkan di dokumentasi ini adalah nilai yang tidak sebenarnya dan tidak dapat dipakai. Nilai-nilai tersebut hanya untuk keperluan contoh saja, tidak untuk dipakai.

```
{
  "total": 66227,
  "page": 1,
  "size": 10,
  "items": {
    "data": [
      {
        "name": "IV Catheter With Vialon Material (VIECARE, Non Injection Port Non Wing / 18 G, 20 G, 22 G, 24 G, 18 G, 20 G, 22 G, 24 G)",
        "kfa_code": "/",
        "active": true,
        "state": "valid",
        "image": null,
        "updated_at": "2023-07-11 08:46:13",
        "farmalkes_type": {
          "code": "device",
          "name": "Alat Kesehatan",
          "group": "alkes"
        },
        "produksi_buatan": "lokal",
        "nie": null,
        "nama_dagang": "VieCare IV Catheter",
        "manufacturer": null,
        "registrar": null,
        "generik": null,
        "rxterm": null,
        "dose_per_unit": 1,
        "fix_price": 8000.0,

```

```
"het_price": null,  
"farmalkes_hscore": null,  
"tayang_lkpp": true,  
"kode_lkpp": null,  
"net_weight": null,  
"net_weight_uom_name": "g",  
"volume": null,  
"volume_uom_name": "mL",  
"uom": {  
  "name": "Units"  
},  
"dosage_form": {  
  "code": false,  
  "name": false  
},  
"product_template": {  
  "kfa_code": "82002082",  
  "name": "IV Catheter With Vialon Material",  
  "state": "valid",  
  "active": true,  
  "display_name": "IV Catheter With Vialon Material",  
  "updated_at": "2023-07-11 08:39:06"  
},  
"active_ingredients": [],  
"replacement": {  
  "product": null,  
  "template": null  
},  
"tags": []  
},  
/*Lompat beberapa data*/  
{  
  "name": "ECG Monitor (Umum)",  
  "kfa_code": "/",  
  "active": true,  
  "state": "valid",  
  "image": null,  
  "updated_at": "2023-05-11 05:00:56",  
  "farmalkes_type": {  
    "code": "device",  
    "name": "Alat Kesehatan",  
    "group": "alkes"  
  },  
  "produksi_buatan": "lokal",  
  "nie": null,  
  "nama_dagang": "MIKI Electrocardiograph CSN-1212A",  
  "manufacturer": null,  
  "registrar": null,  
  "generik": true,  
  "rxterm": null,  
  "dose_per_unit": 1,  
  "fix_price": 40499000.0,  
  "het_price": null,  
  "farmalkes_hscore": null,  
  "tayang_lkpp": true,  
  "kode_lkpp": null,  
  "net_weight": null,  
  "net_weight_uom_name": "g",  
  "volume": null,  
  "volume_uom_name": "mL",  
  "uom": {  
    "name": "Units"  
  },  
  "dosage_form": {  
    "code": false,  
    "name": false  
  },  
  "product_template": {
```

```
    "kfa_code": "82000161",
    "name": "ECG Monitor",
    "state": "valid",
    "active": true,
    "display_name": "ECG Monitor",
    "updated_at": "2023-08-29 00:49:25"
  },
  "active_ingredients": [],
  "replacement": {
    "product": null,
    "template": null
  },
  "tags": []
}
]
```

4xx Client Error

Sistem akan mengembalikan pesan *error* bila *client* belum melakukan autentikasi, tidak memiliki akses, menggunakan HTTP *method* yang tidak tepat, atau mengirimkan data dengan format atau ketentuan yang tidak sesuai.

Contoh Data

```
{
  "detail": [
    {
      "loc": [
        "Ut nisi amet",
        "velit nulla quis minim"
      ],
      "msg": "cupidatat Excepteur enim in",
      "type": "voluptate laborum reprehenderit velit"
    },
    {
      "loc": [
        "id ad",
        "ad"
      ],
      "msg": "cillum",
      "type": "esse sit"
    }
  ]
}
```

5xx Server Error (Content-Type: text/plain)

Sistem akan mengembalikan pesan *error* bila terjadi kesalahan pada sisi server saat memproses data yang telah dikirimkan.

Contoh Data

```
Gateway Timeout
```

2.3. Contoh Penggunaan/Kode



Setiap nilai yang dicontohkan atau ditampilkan di dokumentasi ini adalah nilai yang tidak sebenarnya dan tidak dapat dipakai. Nilai-nilai tersebut hanya untuk keperluan contoh saja, tidak untuk dipakai.

cURL (Windows)

```
curl --insecure --location ^
--header "Authorization: Bearer lWnwCNpXQqdGpyOTvJpyT1iTJey" ^
--header "Accept: application/json" ^
--request GET ^
"https://api-satusehat.kemkes.go.id/kfa-v2/products/all?page=1&size=100&product_type=farmasi"
```

cURL (Linux)

```
curl --insecure --location \
--header 'Authorization: Bearer lWnwCNpXQqdGpyOTvJpyT1iTJey' \
--header 'Accept: application/json' \
--request GET \
'https://api-satusehat.kemkes.go.id/kfa-v2/products/all?page=1&size=100&product_type=farmasi'
```

Postman

1. Buat *request* baru menggunakan **New > HTTP Request**, atau klik **+** untuk buat tab *request* baru.
2. Masukkan *request* URL

```
https://api-satusehat.kemkes.go.id/kfa-v2/products/all
```

3. Lalu pilih *request method* **GET**.
4. Pada tab **Auth**:
 - a. pada pilihan **Type**, pilih **Bearer Token**,
 - b. lalu masukan nilai akses token yang sudah didapatkan pada saat autentikasi pada kotak inputan **Token**.
5. Pada tab **Headers**:
 - a. Pada kolom **KEY** isi **Accept**, lalu pada kolom **VALUE** isi dengan **application/json**.
6. Pada tab **Params**, di bagian **Query Params**:
 - a. silakan masukan parameter untuk melakukan pencarian sesuai dengan yang sudah dijelaskan pada bagian *query string* terkait ReST API Kamus Farmasi dan Alat Kesehatan (KFA) ini,
 - b. masukan satu atau lebih nama dari parameter tersebut pada kotak masukan pada kolom **KEY**,
 - c. sedangkan untuk nilainya, masukan pada kotak masukan pada kolom **VALUE**.
7. Klik tombol **Send**.
8. Hasil *response* akan ditampilkan di bagian **Response**.

DAFTAR ISTILAH

API

Singkatan dari *Application Programming Interface*, yaitu kumpulan definisi dan protokol terkait koneksi antara komputer atau aplikasi yang memungkinkan untuk melakukan pertukaran data.

JWT

Singkatan dari *JSON Web Token*, yaitu suatu standar untuk pertukaran data yang diperlukan dalam proses autentikasi.

KFA

Kamus Farmalkes (Farmasi & Alat Kesehatan) atau KFA merupakan kamus standar untuk data farmasi dan alat kesehatan yang digunakan pada *platform* SATUSEHAT dan mengacu pada model **NHS dm+d1**

KFA management tools

Merupakan tools yang digunakan untuk tata kelola isi data dari Kamus Farmasi dan Alat Kesehatan. Akses pada tools manajemen Kamus Farmasi dan Alat Kesehatan terbatas.

KFA Browser

Merupakan browser pada website untuk menampilkan data farmasi dan alat kesehatan yang sudah terstandarisasi. Kamus Browser ini bersifat publik yang dapat diakses di <https://dto.kemkes.go.id/kfa-browser>.

REFERENSI

- [restfulapi.net] "ReST API Tutorial". <https://restfulapi.net>.
- [wikipedia-rest] "Representational State Transfer". https://en.wikipedia.org/wiki/Representational_state_transfer.
- [wikipedia-http-method] "Hypertext Transfer Protocol, Request_methods". https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol#Request_methods
- [wikipedia-http-status-code] "List of HTTP status codes". https://en.wikipedia.org/wiki/List_of_HTTP_status_codes
- [wikipedia-iso8601] "ISO 8601". https://en.wikipedia.org/wiki/ISO_8601
- [wikipedia-api] "Application Programming Interface". <https://en.wikipedia.org/wiki/API>
- [wikipedia-whitespace] "Whitespace character". https://en.wikipedia.org/wiki/Whitespace_character
- [wikipedia-fhir] "Fast Healthcare Interoperability Resources", <https://en.wikipedia.org/wiki/>
- [hl7-fhir] "Welcome to FHIR. FHIR is a standard for health care data exchange, published by HL7®". <https://www.hl7.org/fhir/index.html>
- [satusehat-fhir] "SATUSEHAT FHIR R4 Implementation Guide". <https://simplifier.net/guide/SATUSEHAT-FHIR-R4-Implementation-Guide/Home?version=current>
- [admonitions] "Admonitions". <https://docs.asciidoctor.org/asciidoc/latest/blocks/admonitions>
- [atc-code] "Daftar Kode ATC" https://id.wikipedia.org/wiki/Kategori:Kode_ATC
- [atc-level] "Daftar Level ATC" <https://www.who.int/tools/atc-ddd-toolkit/atc-classification>