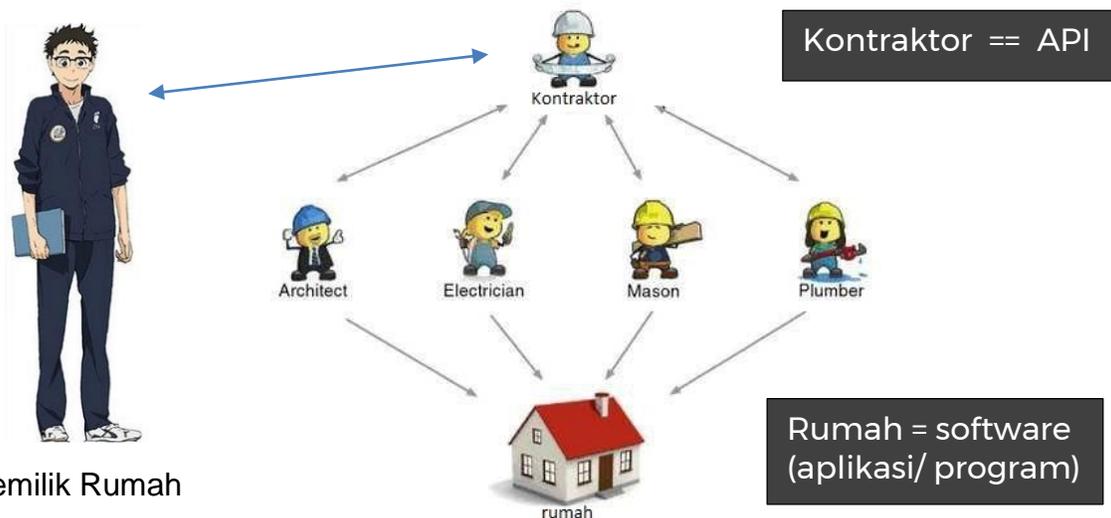


API

(**A**pplication **P**rogramming **I**nterface)

[**A. P. I.**] *a.k.a. Antarmuka Pemrograman Aplikasi* merupakan *protokol* yang terdiri atas kumpulan instruksi yang disimpan dalam bentuk *library* dan menjelaskan(mengatur) bagaimana agar suatu *software* dapat berinteraksi dengan *software* lain. **Jadi**, dengan adanya API, maka terdapat aturan bagaimana cara *software* dapat berinteraksi dengan *software* lain untuk mengakses *resources(data)* yang terdapat di dalam *software* tersebut melalui *interface (fungsi,sintaks,protocol)* yang telah tersedia tanpa perlu mengetahui bagaimana software/aplikasi itu dibuat.

Penjelasan ini dapat dicontohkan dengan analogi apabila kita mau membangun suatu rumah berikut :

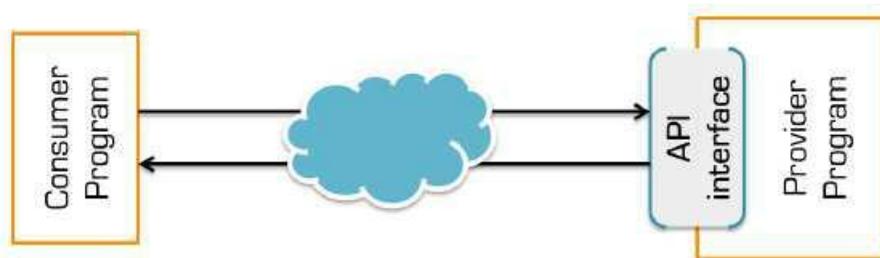


Misalkan, **Bang Ican** ingin membangun sebuah “ **rumah** ”, sebagai pemilik rumah, Abang tentu tidak tau bagaimana prosedur pengerjaan konstruksi rumah abang, yang penting abang punya uang dan abang mau rumah abang selesai sesuai dengan tampilan rumah idaman yang telah abang pilih. Nah, Abang pun menyewa seorang “

kontraktor ” yang akan menangani seluruh prosedur pengerjaan konstruksi rumah itu. Abang tinggal memberikan tugas yang perlu dilakukan kontraktor tanpa harus mengetahui bagaimana cara kontraktor menyelesaikan pekerjaan tersebut (dalam hal ini tugas yang dimaksud adalah kriteria-kriteria rumah idaman abang).

Dari analogi tersebut, rumah merupakan *software* yang akan dibuat, dan kontraktor merupakan API yang mengerjakan bagian tertentu dari *software* tersebut tanpa harus diketahui bagaimana prosedur dalam melakukan pekerjaan tersebut. Artinya Kontraktor (si API) menjadi perantara / media komunikasi pemilik rumah dalam bertukar informasi dalam pengerjaan konstruksi rumahnya hingga selesai.

Struktural API



API merupakan spesifikasi yang terdiri dari berbagai elemen seperti

- *function,*
- *Protocols,*
- *data structure,*
- *objects,*
- *dan tools* lainnya,
- beserta parameter-parameter

yang diperlukan untuk mengakses *resource* dari aplikasi tersebut. **Seluruh spesifikasi tersebut membentuk suatu *interface* yang dimiliki oleh aplikasi untuk berkomunikasi dengan aplikasi lain.** Sehingga memungkinkan *developers* untuk membuat aplikasi dengan lebih mudah.

Klasifikasi API

Kategori API	Deskripsi	Contoh
<i>Operating System</i>	API yang digunakan untuk Fungsi dasar yang dapat dilakukan oleh komputer. Seperti proses I/O,	API for MS Windows

	eksekusi program.	
Programming Languages	API yang digunakan untuk memperluas kapabilitas dalam melakukan eksekusi terhadap suatu bahasa pemrograman.	Java API
Application Services	API yang digunakan untuk mengakses data dan layanan yang disediakan dari suatu aplikasi.	API For mySAP (BAPI/ <i>Business Application Programming Interface</i>)
Infrastructure Services	Digunakan untuk mengakses infrastruktur Dari suatu komputer. Infrastruktur disini adalah komputer beserta <i>peripheral</i> seperti <i>storage</i> , aplikasi, dan lain-lain.	Amazon EC2 (<i>Elastic Compute Cloud</i>) untuk akses Untuk <i>virtual computing</i> dan Amazon S3 (<i>Simple Storage Service</i>) untuk menyimpan data dalam jumlah besar.
Web Services	API yang digunakan untuk mengakses <i>content</i> dan layanan yang disediakan oleh suatu <i>web application</i> .	<i>Facebook Graph API</i> yang Digunakan untuk mengakses informasi yang dapat dibagikan.

Bahasa Pemrograman yang digunakan

API dapat digunakan dengan berbagai bahasa *programming* (pemrograman), ataupun hanya dengan menggunakan URL (*Uniform Resource Locator*) yang telah disediakan oleh suatu *website*. Bahkan, ternyata kita bisa membuat / mendesain API kita sendiri contoh: membuat Web API dengan menggunakan teknologi yang berbeda seperti *PHP*, *Java*, *.NET*, *MySQL*, *XML* dll. Hal ini telah dilakukan oleh Microsoft , dimana mereka mendesain API sendiri dengan (*Visual C++*, *C#*, *Visual Basic*, *F#*, dll.) yang tertanam di Sistem Visual Studio mereka.

Contoh API

- Google MAPS API
- Github API
- Twitter REST API
- Facebook Graph API
- ASPI untuk antarmuka perangkat SCSI
- Karbon dan Kakao untuk Macintosh
- DirectX untuk Microsoft Windows = OS(system operasi) API
- EHLLAPI
- Java API
- ODBC untuk Microsoft Windows
- OpenAL cross-platform sound API
- OpenCL cross-platform API untuk komputasi keperluan umum untuk CPU & GPU
- API grafis cross-platform OpenGL
- OpenMP API yang mendukung pemrograman multiprocessing multi-platform memori bersama di C, C ++ dan Fortran pada banyak arsitektur, termasuk platform Unix dan Microsoft Windows.
- Lapisan DirectMedia Sederhana (SDL)
- Talend mengintegrasikan manajemen datanya dengan BPM dari Bonita Open Solution

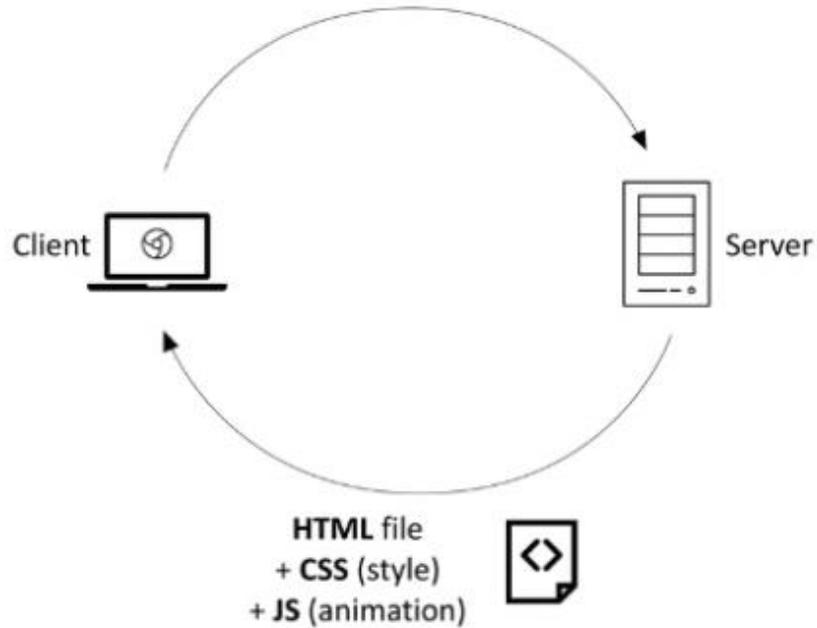
Web API

Ketika digunakan dalam konteks pengembangan web, sebuah API biasanya didefinisikan sebagai sekumpulan pesan permintaan Hypertext Transfer Protocol (HTTP), bersama dengan definisi struktur pesan respons, yang biasanya dalam Extensible Markup Language (XML) atau Format JavaScript Object Notation (JSON). Tren terkini (disebut Web 2.0) telah bergerak menjauh dari layanan berbasis Access Protocol (SOAP) sederhana menuju komunikasi gaya Representational State Transfer (REST) langsung. Web API ini memungkinkan kombinasi beberapa layanan ke dalam satu aplikasi baru yang dikenal sebagai mashup application dan mengambil data / informasi / sumber daya fungsi yang disediakan aplikasi lainnya.

Web-101

HTTP request with URL

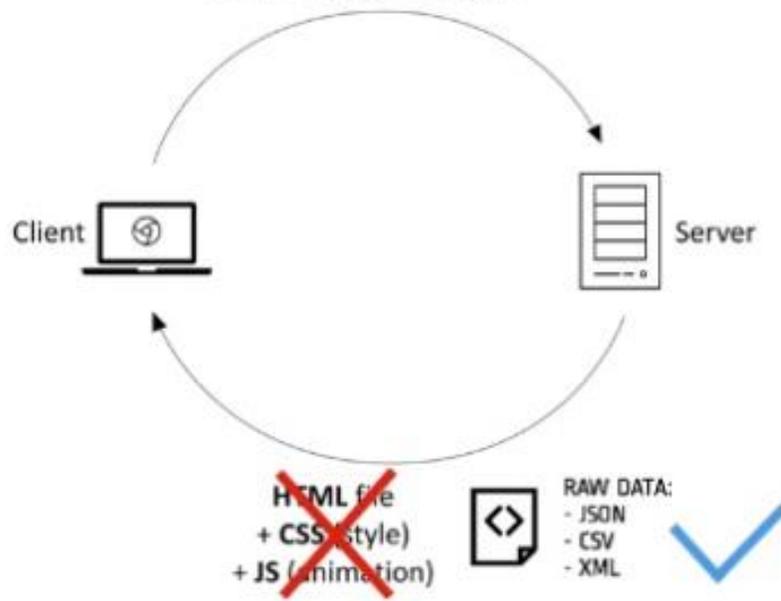
GET "http://lewagon.org/program"



API

HTTP request with URL

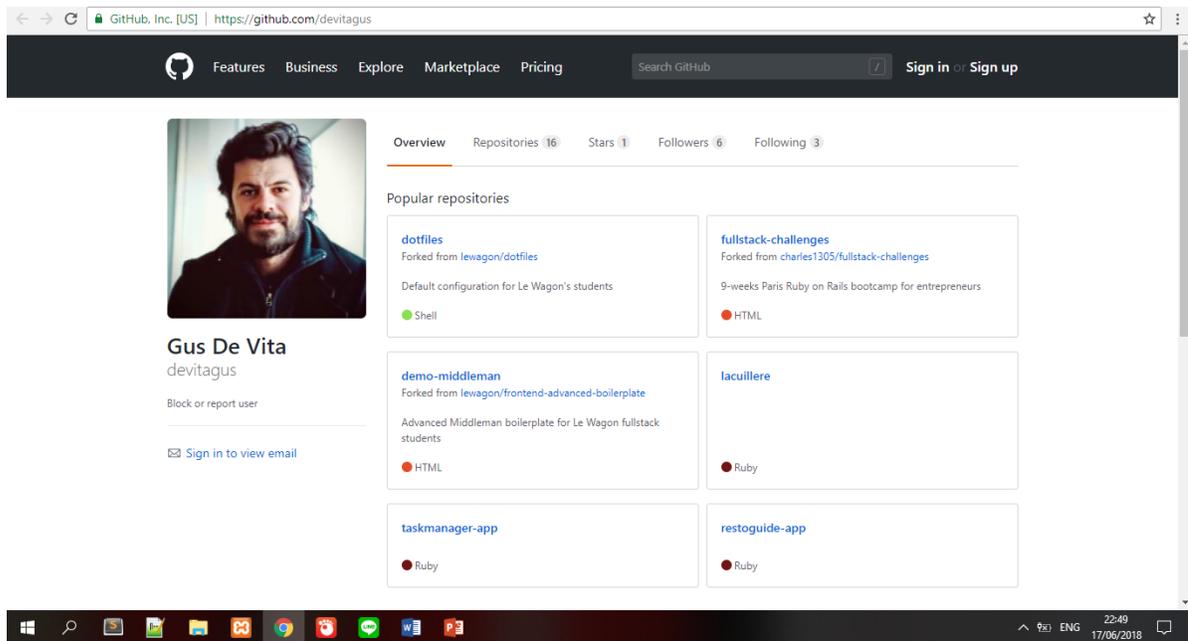
GET "http://lewagon.org/program"



Ayo Bandingkan Web dan Web API dari salah satu user berikut!!!

Website Github

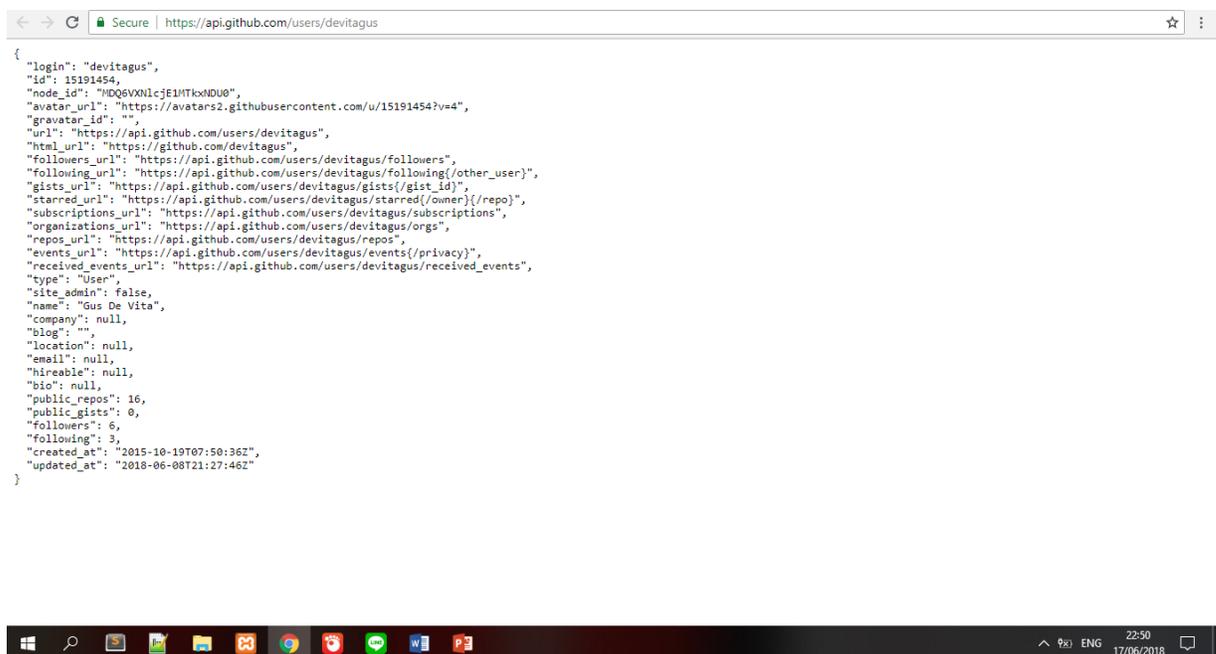
<https://github.com/devitagus>



The screenshot shows the GitHub profile page for user 'devitagus'. The page includes a navigation bar with links for Features, Business, Explore, Marketplace, and Pricing, along with a search bar and 'Sign in or Sign up' buttons. The profile section features a profile picture of Gus De Vita, his name 'Gus De Vita', and the handle 'devitagus'. Below the name, there are options to 'Block or report user' and 'Sign in to view email'. The 'Overview' tab is selected, showing statistics: 16 Repositories, 1 Star, 6 Followers, and 3 Following. A 'Popular repositories' section displays six repositories: 'dotfiles', 'fullstack-challenges', 'demo-middleman', 'lacullere', 'taskmanager-app', and 'restoguide-app', each with a brief description and a language indicator.

API Github

<https://api.github.com/users/devitagus>



The screenshot shows the GitHub API endpoint for user 'devitagus'. The browser address bar displays 'Secure | https://api.github.com/users/devitagus'. The main content area shows a JSON response with the following structure:

```
{
  "login": "devitagus",
  "id": 15191454,
  "node_id": "MDQ6VXNlcjE1MTkxNDU0",
  "avatar_url": "https://avatars2.githubusercontent.com/u/15191454?v=4",
  "gravatar_id": "",
  "url": "https://api.github.com/users/devitagus",
  "html_url": "https://github.com/devitagus",
  "followers_url": "https://api.github.com/users/devitagus/followers",
  "following_url": "https://api.github.com/users/devitagus/following{/other_user}",
  "gists_url": "https://api.github.com/users/devitagus/gists{/gist_id}",
  "starred_url": "https://api.github.com/users/devitagus/starred{/owner}/{repo}",
  "subscriptions_url": "https://api.github.com/users/devitagus/subscriptions",
  "organizations_url": "https://api.github.com/users/devitagus/orgs",
  "repos_url": "https://api.github.com/users/devitagus/repos",
  "events_url": "https://api.github.com/users/devitagus/events{/privacy}",
  "received_events_url": "https://api.github.com/users/devitagus/received_events",
  "type": "User",
  "site_admin": false,
  "name": "Gus De Vita",
  "company": null,
  "blog": "",
  "location": null,
  "email": null,
  "hireable": null,
  "bio": null,
  "public_repos": 16,
  "public_gists": 0,
  "followers": 6,
  "following": 3,
  "created_at": "2015-10-19T07:50:36Z",
  "updated_at": "2018-06-08T21:27:46Z"
}
```

Cara Mengetahui API Suatu Web

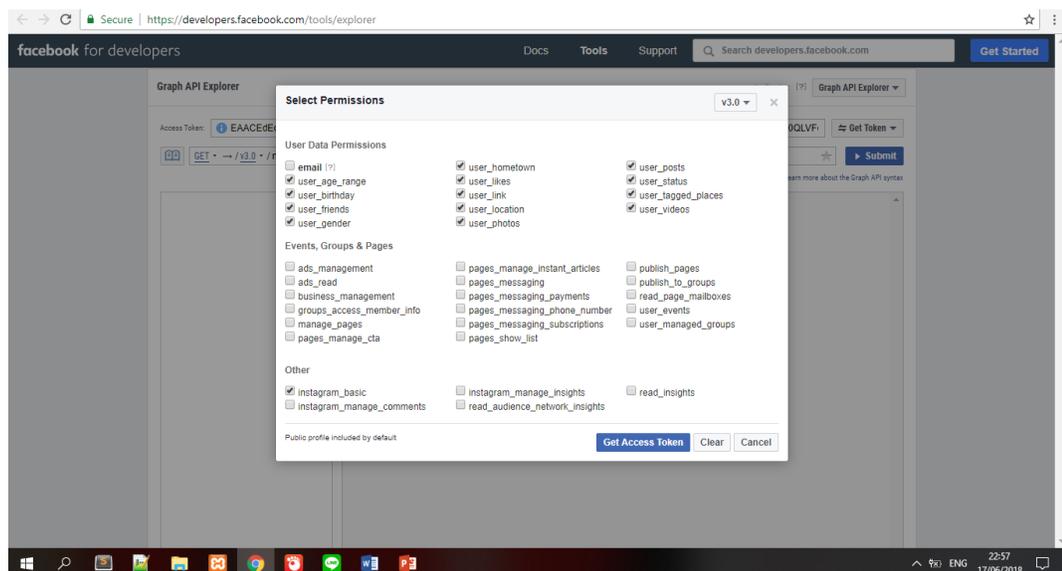
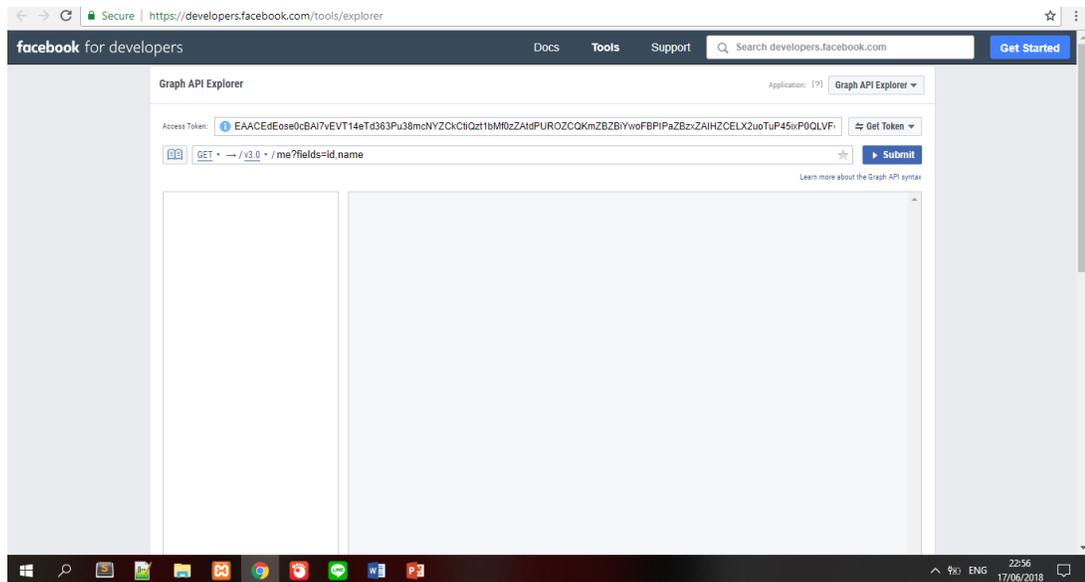
Nah , biasanya untuk mengetahui API dari suatu website kita tinggal mengakses ke link developers dari website itu ,

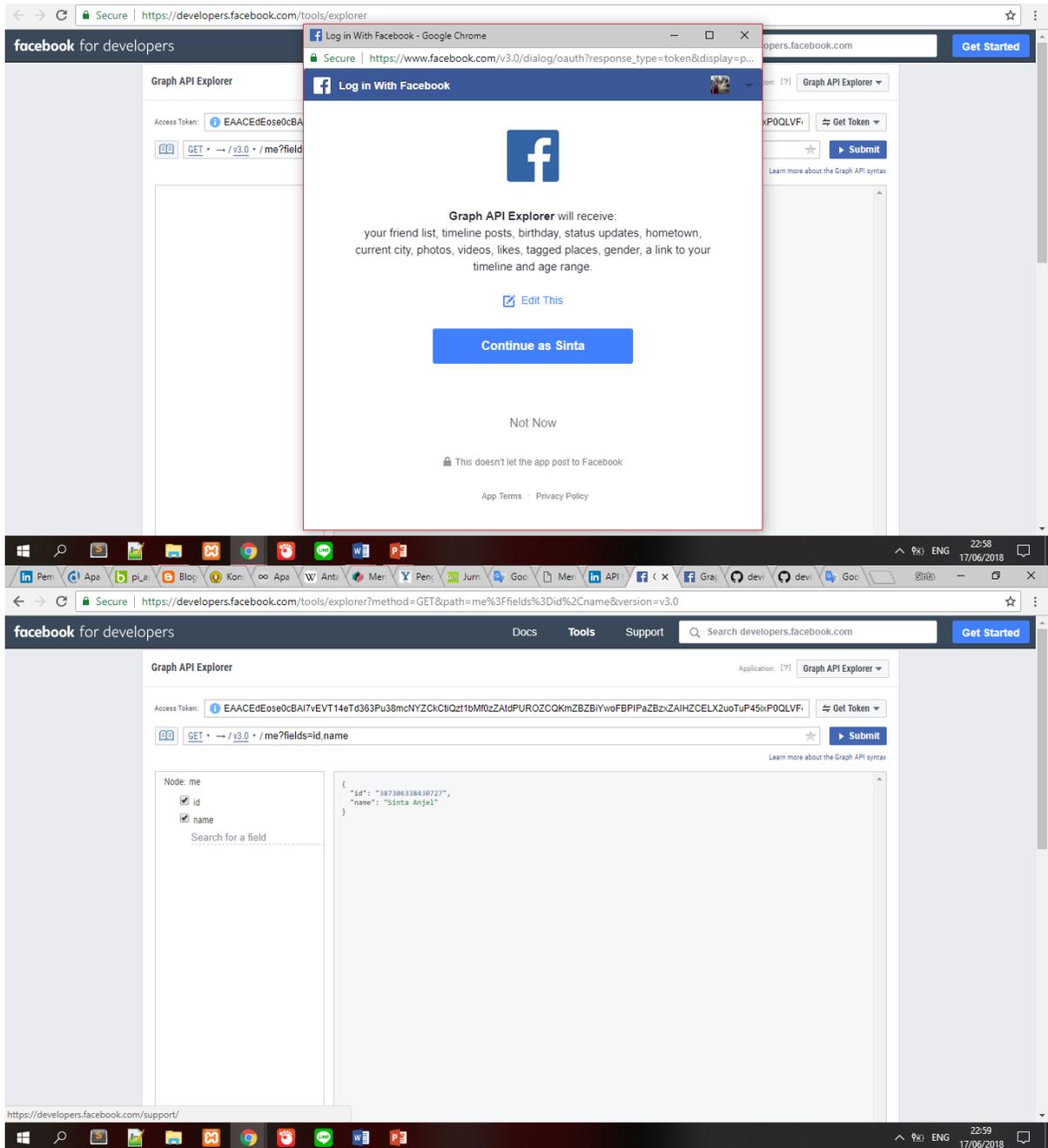
Contoh :

Facebook graph API

<https://developers.facebook.com/tools/explorer>

Nah, disini Saya mencoba mengambil data facebook saya dengan menggunakan API Facebook ini.





Notes : Karena kosong/tidak ada diisi oleh user jadinya cuman keluar id sama nama .

Keuntungan dengan adanya API

Praktik penggunaan API telah memungkinkan komunitas web untuk menciptakan arsitektur terbuka. Maksudnya API memungkinkan untuk berbagi konten dan data antara komunitas dan aplikasi. Seperti namanya **API (Application Programming Interface)** ,sebuah Interface yang dimanfaatkan oleh developer ketika menulis kode Program nya supaya berinteraksi dan berintegrasi dengan Aplikasi lainnya. Dengan

cara ini, konten yang dibuat di satu tempat dapat dipostkan dan diperbarui secara dinamis di beberapa lokasi di web:

1. Foto dapat dibagikan dari situs seperti Flickr dan Photobucket ke situs jejaring sosial seperti Facebook dan MySpace.
2. Konten dapat disematkan, misalnya. embedding presentasi dari SlideShare di profil LinkedIn.
3. Konten dapat dipostkan secara dinamis. Berbagi komentar langsung yang dibuat di Twitter dengan akun Facebook, misalnya, diaktifkan dengan API mereka.
4. Konten video dapat disematkan di situs yang dilayani oleh host lain.
5. Informasi pengguna dapat dibagikan dari komunitas web ke aplikasi luar, memberikan fungsionalitas baru kepada komunitas web yang membagikan data penggunaannya melalui API terbuka. Salah satu contoh terbaik dari ini adalah platform Aplikasi Facebook. Lainnya adalah platform Open Social.
6. Facebook menyediakan API sehingga para pengembang website dapat mengintegrasikan komentar di websitenya langsung melalui komentar akun facebook si pengunjung, atau yang lebih spesifik lagi API facebook juga dapat digunakan untuk membuat fungsi auto post artikel ke facebook saat artikel di website ditambahkan. Semua ini dapat dilakukan karena facebook menyediakan program API untuk dapat mengakses sebagian atau beberapa fungsi dari program facebook. Begitu juga untuk API dari layanan lainnya seperti JNE, POS dst.
7. API memungkinkan sebuah aplikasi berbicara satu sama lain tanpa sepengetahuan pengguna (pemilik informasi). Dan mengambil informasi atau data diluar firewall dari suatu aplikasi jika diketahui api nya . Sehingga, hal ini sering dimanfaatkan dalam aplikasi online shop dalam integrasi website dan system mereka dengan website lain (misalnya : amazon dengan eBay , Twitter , dan Salesforce.com)

Keuntungan Developer memprogram dengan API

- **Portabilitas.**

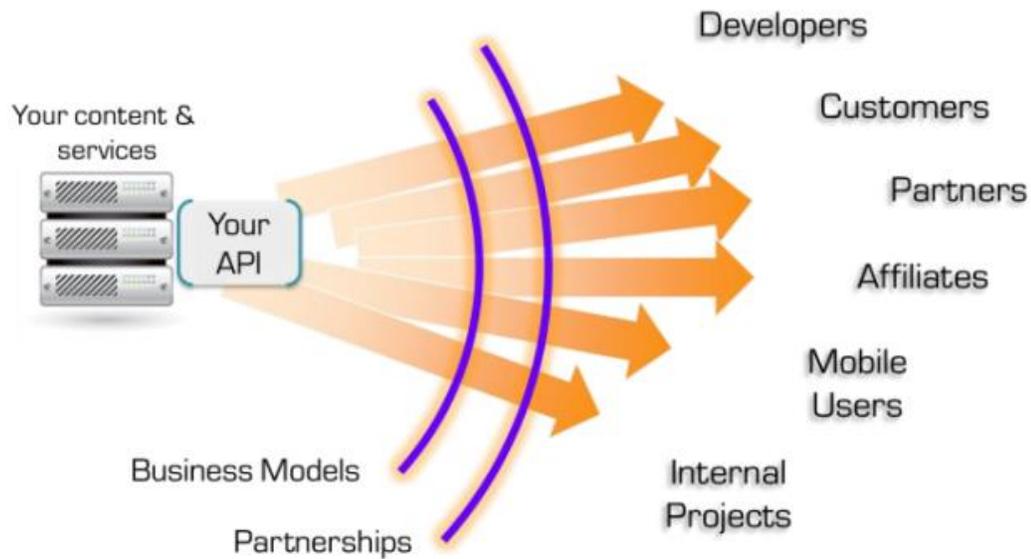
Programmer yang menggunakan API dapat menjalankan programnya dalam sistem operasi mana saja asalkan sudah ter- *install* API tersebut. Sedangkan *system call* berbeda antar sistem operasi, dengan catatan dalam implementasinya mungkin saja berbeda.

- **Lebih Mudah Dimengerti.**

API menggunakan bahasa yang lebih terstruktur dan mudah dimengerti daripada bahasa *system call*. Hal ini sangat penting dalam hal editing dan pengembangan.

Kenapa ?

Berikut sekilas tentang system call :



Gambar : System Call dan API Network

System call interface ini berfungsi sebagai penghubung antara API dan *system call* yang dimengerti oleh sistem operasi. *System call interface* ini akan menerjemahkan perintah dalam API dan kemudian akan memanggil *system calls* yang diperlukan.

Untuk membuka suatu *file* tersebut *user* menggunakan program yang telah dibuat dengan menggunakan bantuan API, maka perintah dari *user* tersebut diterjemahkan dulu oleh program menjadi perintah *open()*.

Perintah *open()* ini merupakan perintah dari API dan bukan perintah yang langsung dimengerti oleh kernel sistem operasi. Oleh karena itu, agar keinginan *user* dapat dimengerti oleh sistem operasi, maka perintah *open()* tadi diterjemahkan ke dalam bentuk *system call* oleh *system call interface*. Implementasi perintah *open()* tadi bisa bermacam-macam tergantung dari sistem operasi yang kita gunakan.

Penjelasan System call diatas sangat membingungkan bukan.. Nah dengan adanya API kita tidak perlu memikirkan hal itu .

- **Mudah Dikembangkan**

Dengan adanya API, memudahkan programmer untuk mengembangkan suatu system.

Contoh fitur yang sering digunakan Web Developer dengan API

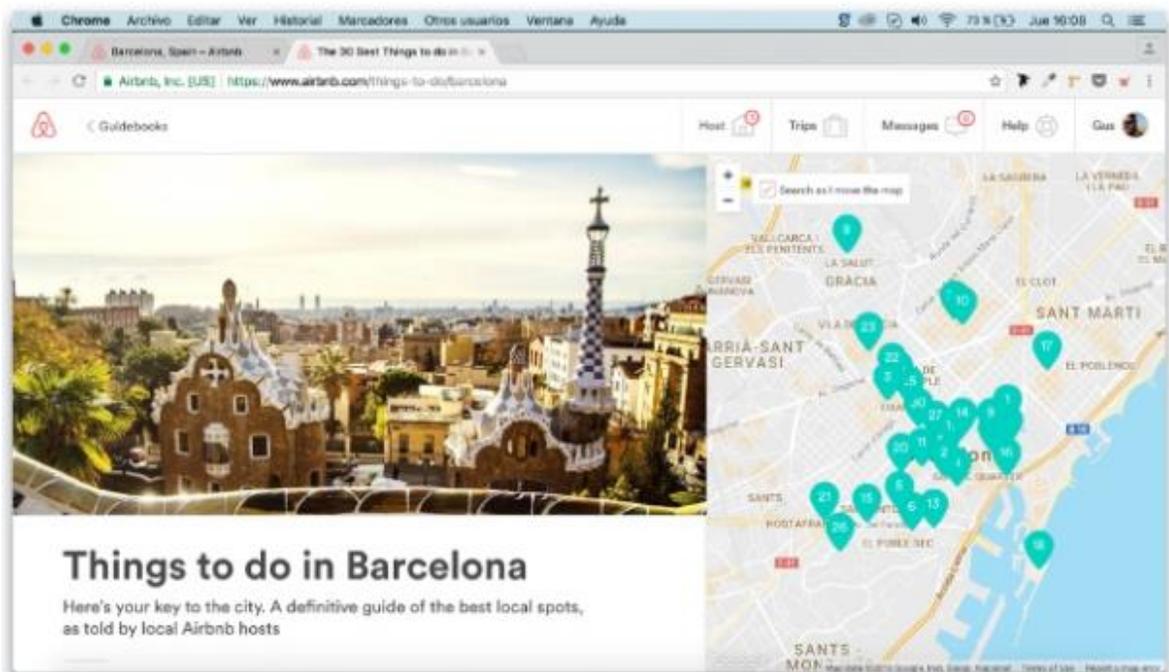
Ada banyak penyedia layanan API, namun penggunaan API paling banyak dan dianggap paling efisien, efektif dan bermanfaat bagi pemrogram yaitu **Info Cuaca**, kita bisa membuat aplikasi tentang cuaca yang selalu update, dan data yang di ambil dari API penyedia layanan

Implementasi API dalam Kehidupan Sehari-hari

Berikut beberapa implementasi API :



Untuk mendapatkan API gratis untuk aplikasi kita



Mengambil data / informasi yang berkaitan dengan geografis peta/ maps/ lokasi tempat-tempat

Google Maps API



API de Google Maps > Para Web > Maps Embed API

Llévate el poder de Google Maps a tu sitio web con HTML simple

Ayuda a tus usuarios a descubrir lugares usando su propio mapa personalizado y aprovechando la completa base de datos de Google de más de 100 millones de lugares en todo el mundo.

- Mapas**
Integra mapas básicos y planos para pisos de interiores.
- Imágenes**
Agrega imágenes de Street View e imágenes satelitales.
- Indicaciones**
Accede a rutas paso a paso, tráfico, rutas de bicicleta y transporte público.

Login

"Gustar" o "pasar" de forma anónima en los perfiles de la gente sugerida

Al continuar, aceptas nuestros Servicios de servicio y Política de privacidad

Inicia sesión con Facebook

No publicamos nada en Facebook.

Flutter

Email

SIGN UP

CONNECT BY LINKEDIN

By continuing, you agree to our Terms and Privacy Policy

Bienvenido

Ahorra hasta un 35% en ofertas de hotel

Organiza, planea y comparte tus viajes

Recibe alertas de estado de vuelo

Sincroniza tu viaje en diferentes dispositivos

Conéctate con Facebook

Conéctate con Google

Inicia sesión con tu email >

Website Apapun bisa login Dengan facebook / google / linked in/ gmail. Jadi ada integrasi data antar website.



Shares of Twilio (NYSE:TWLO) have more than quadrupled from its IPO price of \$15.

Uber menggunakan API Google Maps untuk menampilkan peta/ maps/ lokasi tempat-tempat

Salah Satu Cara Implementasi API

Misalkan kamu sebagai developer website 'Time Out', Kamu yang menanggungjawab semua seluk beluk website 'Time Out' .Lalu Content managernya ingin melihat lokasi pertunjukan di sebuah kota melalui maps . Content managernya memberikan mu lokasi tempatnya tapi dia tentu tidak tau coordinate GPS lokasi tempat itu . jadi kamu bisa memanfaatkan API untuk itu. Yaitu :

Google Geocoding API

<https://developers.google.com/maps/documentation/geocoding/start>
misalkan jalanya : 10 Downing St, London

Bagaimana cara pakainya??

<https://maps.googleapis.com/maps/api/geocode/json?address=10%20Downing%20St,%20London>

Lalu didapat lah, ...

Ini namanya JSON

```
{
  "results" : [
    {
      "address_components" : [
        {
          "long_name" : "10",
          "short_name" : "10",
          "types" : [ "street_number" ]
        },
        {
          "long_name" : "Downing Street",
          "short_name" : "Downing St",
          "types" : [ "route" ]
        },
        {
          "long_name" : "Westminster",
          "short_name" : "Westminster",
          "types" : [ "neighborhood", "political" ]
        },
        {
          "long_name" : "London",
          "short_name" : "London",
          "types" : [ "postal_town" ]
        },
        {
          "long_name" : "Greater London",
          "short_name" : "Greater London",
          "types" : [ "administrative_area_level_2", "political" ]
        },
        {
          "long_name" : "England",
          "short_name" : "England",
          "types" : [ "administrative_area_level_1", "political" ]
        },
        {
          "long_name" : "United Kingdom",
          "short_name" : "GB",
          "types" : [ "country", "political" ]
        },
        {
          "long_name" : "SW1A 2AA",
          "short_name" : "SW1A 2AA",
          "types" : [ "postal_code" ]
        }
      ],
      "formatted_address" : "10 Downing St, Westminster, London SW1A 2AA,
UK",
      "geometry" : {
        "location" : {
          "lat" : 51.5033635,
          "lng" : -0.1276248
        },
        "location_type" : "ROOFTOP",
        "viewport" : {
          "northeast" : {
            "lat" : 51.5047124802915,
```

```

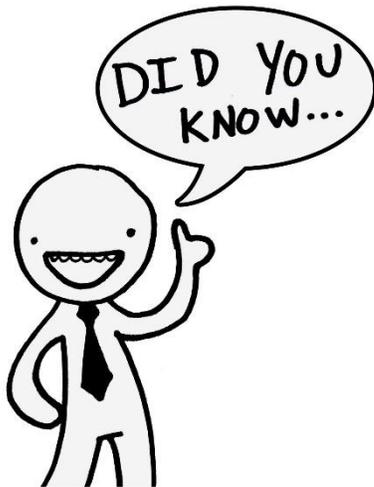
        "lng" : -0.126275819708498
    },
    "southwest" : {
        "lat" : 51.5020145197085,
        "lng" : -0.128973780291502
    }
}
},
"place_id" : "ChIJRxxRQcUEdkgRGVaKyzmkgvg",
"types" : [ "establishment", "point_of_interest" ]
}
],
"status" : "OK"
}

```

Kemudian kita masukkan ke dalam program kita sesuai fungsi Bahasa pemrograman yang kita pakai dalam menyambungkan api.

API Fun Fact

Tahukah Anda ?



API merupakan cara pengambilan data / informasi dari suatu website / software yang diperbolehkan alias **Legal**.

API memang disediakan developer nya sebuah aplikasi / website / software dan diperuntukkan bagi developer lainnya dalam mempermudah pengembangan yang ia lakukan dengan penyediaan data dan pengambilan data dari sebuah aplikasi / website / software milik orang lain.

API awalnya dibuat agar semua informasi dari berbagai aplikasi / website / software bisa saling berintegrasi (bertukar informasi / berkomunikasi) jadi informasi / data didapat dengan mudah , cepat, efisien dan bermanfaat..

Jadi Kalau API yang legal, cara **ilegal**nya apa dong ?

Cara ilegal nya itu dikenal dengan nama **CRAWLING**.

