**API**

adalah antarmuka yang berupa kumpulan fungsi yang dapat di-panggil atau dijalankan oleh program lain.

Penerapannya sangat luas,

* Bahasa Pemograman

Contoh nya koneksi via mysqli dan PDO

* Library & Framework Contohnya

$this->db->get(‘mahasiswa);

* Sistem Operasi, Contohnya kernel.dll, wind32k.sys
* *Web API / Web Service* merupakan sistem perangkat lunak untuk mendukung interoperabilitas antara 2 aplikasi yang berbeda melalui jaringan. Contohnya SOAP (*Simple Object Access Protocol*) dan REST (*Representasional State Transfer*)

**REST API**

Analogikan seorang pelayan adalah API sebagai penghubung antara dapur dan customer, yang melakukan pesan / request dan mengantar kan pesanan / response. kemudian menu adalah REST API(aturan) nya, jadi aturan aturan pemesanannya ada di menu, agar customer tidak sembarangan memesan makanan(JSON) yang tidak ada di menu, walaupun sebenarnya memungkinkan.

**Tentang Rest API**

* Dia adalah gaya perancangan
* Sifat nya *stateless*
* Interaksinya hampir selalu menggunakan HTTP
* Memungkinkan interaksi antar mesin.

URL atau *Uniform Resource Locator* adalah sebuah alamat untuk mengakses sumber daya / *resource* bisa halaman html, gambar, script, css dll. Pasti ujung nya selalu sebuah *resource*



**CARA KERJA BROWSER**

Kita bisa request dengan mengetikan di url [www.google.com](http://www.google.com). Request yang dikirimkan disini menggunakan metode GET



Figure 1 GET

Tapi untuk melakukan request kita juga menggunakan form bedanya request ini disertai data yang diisikan di dalam form.

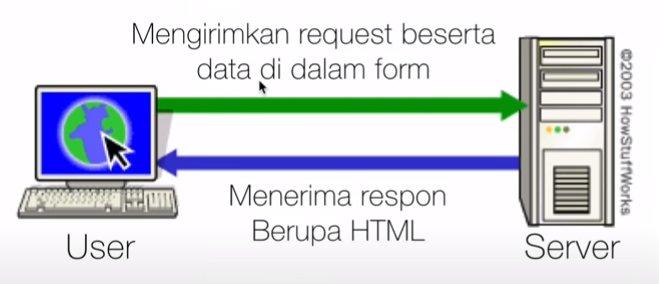


Figure 2 POST

Jadi request itu bisa menggunakan post atau get tergantung operasinya.

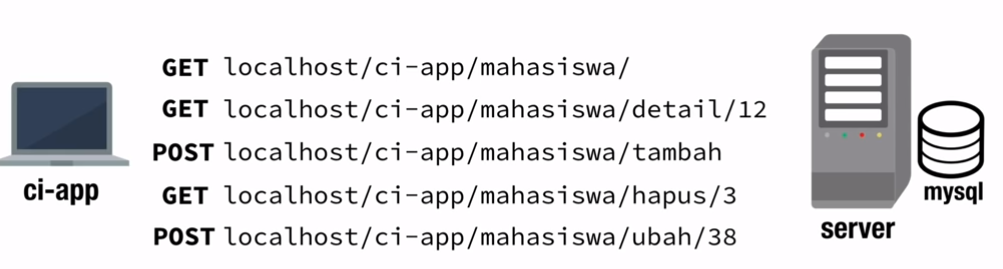
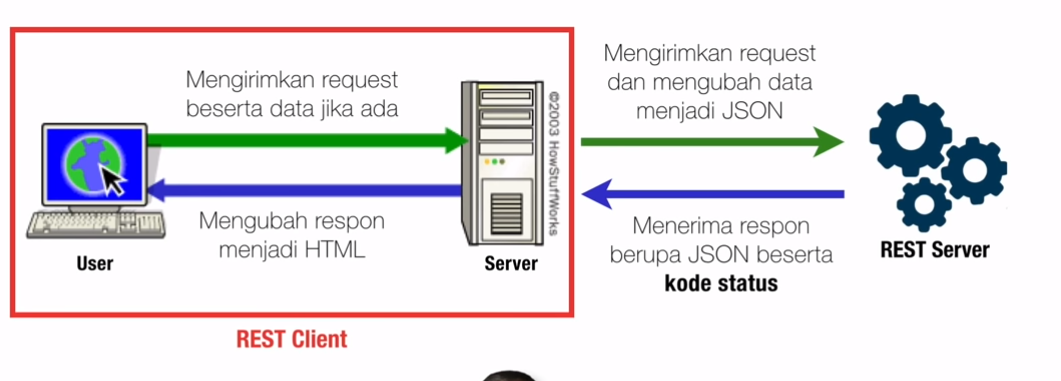


Figure 3 Contoh Request Menggunakan CI

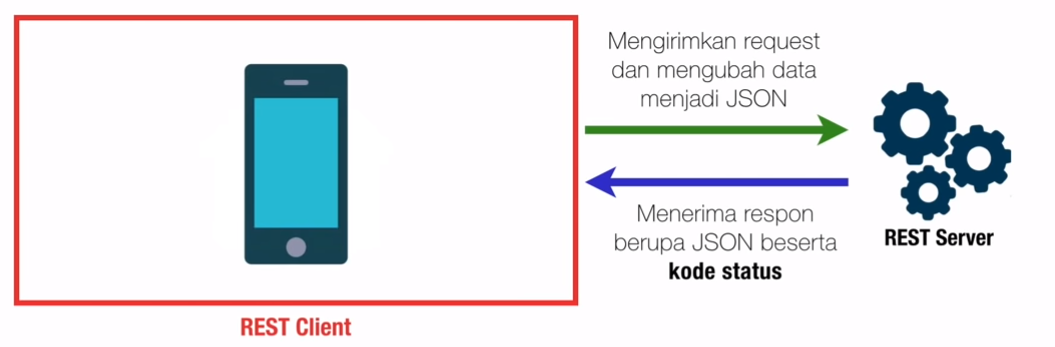
**MEKANISME REST API**

Nah sekarang bagaimana cara mainnya kalao pakai arsitektur REST API.

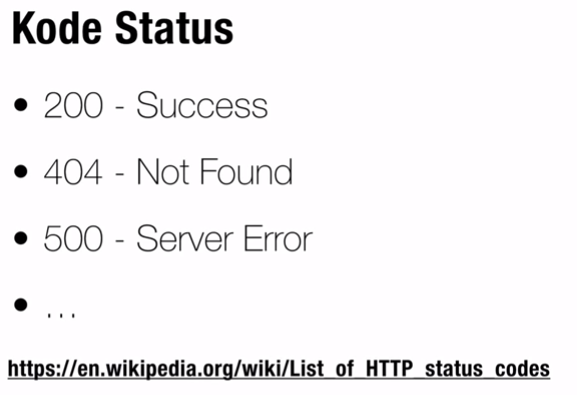


Dari user request data ke server, bisa get atau post kalau ada data penyertanya, trus server request ke Rest Server dan mengubah data penyerta dengan berformat JSON. Kemudian Rest Server merespon dengan data berupa JSON dan beserta status kode nya. Dan oleh server diteruskan lagi ke user berupa HTML.

REST Client adalah semua makhluk yang melakukan request ke REST Server, bisa berupa mobile apps, desktop/pc, smarthome, atau bahkan server.

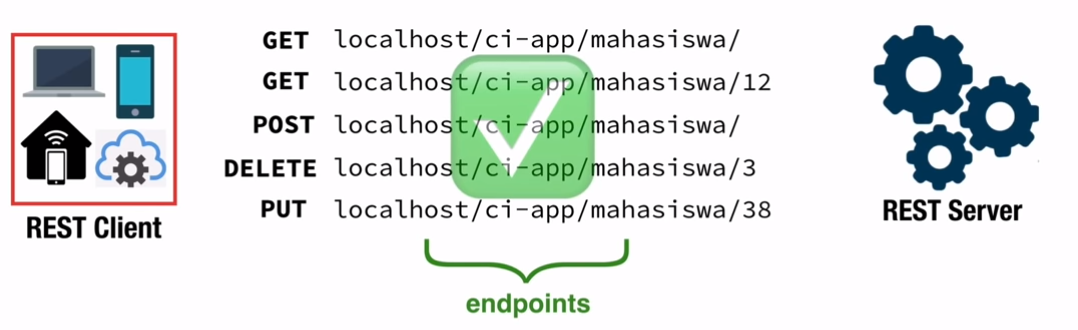


Apa itu kode status ?

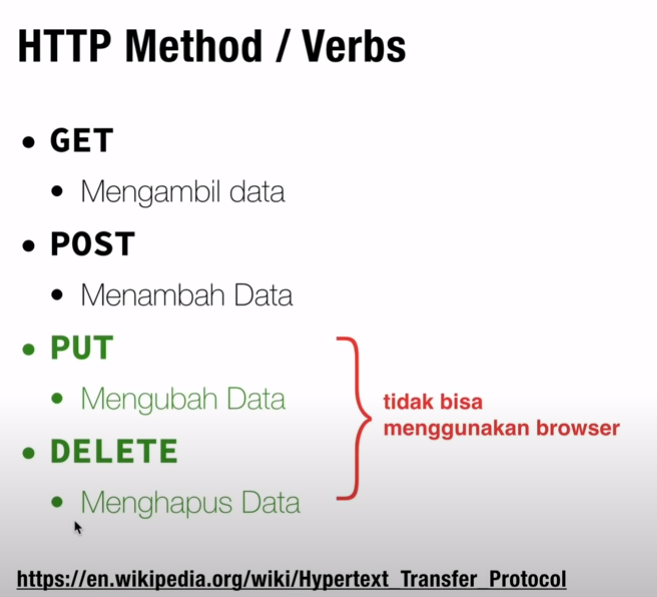


Kenapa butuh kode status ? karena yang ngobrol mesin dengan mesin, intinya biar mesin paham.

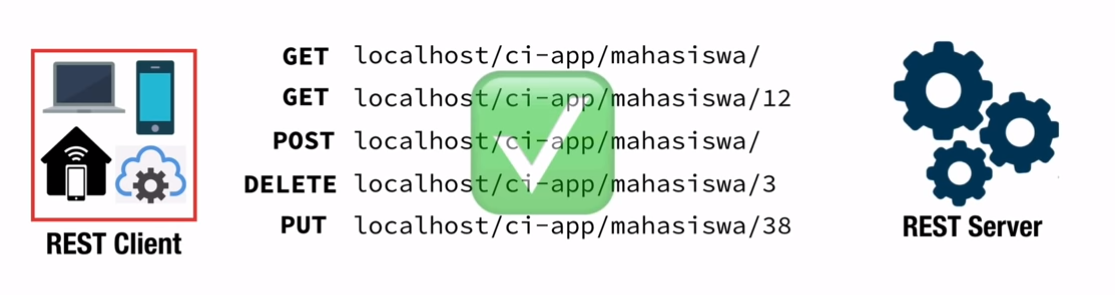
Trus gimana cara ngobrol nya REST Client dengan REST Server ? ya sama aja kayak CI app tadi, bedanya resource yang diakses bukan html lagi, tapi semua nya sudah JSON



Selamat anda sudah berhasil membuat sebuah arsitektur REST. Tapiiii ini belum RESTFUL. Lo apa bedanya ? kalau mau RESTFUL harus menggunakan metode request selain GET dan POST.



Dan masih ada beberapa lagi metode request. Begini penjelasannya, dengan RESTFUL API kita bisa merequest dengan URL yang sama untuk keperluan yang berbeda



Misalnya, untuk mendapatkan data mahasiswa dengan id dua belas kita cukup menggunakan URL localhost/ci-app/mahasiswa/12 dengan metode request GET. Untuk menambah data kita juga bisa menggunakan URL yang sama yaitu localhost/ci-app/mahasiswa/. Untuk menghapus dan mengupdate gunakan URL localhost/ci-app/mahasiswa/12 kita tinggal mengubah metode request nya menjadi DELETE dan PUT. Dan nanti URL di atas di sebut dengan ENDPOINTS.

**STATELESS**

* Setiap request itu dilakukan secara terisolasi, artinya server tidak boleh menyimpan state apapun mengenai sesi dari client, sehingga setiap request yang di jalankan harus berisi semua informasi yang dibutuhkan server, termasuk informasi otentikasi.(*karena stateless tidak menyimpan session*).

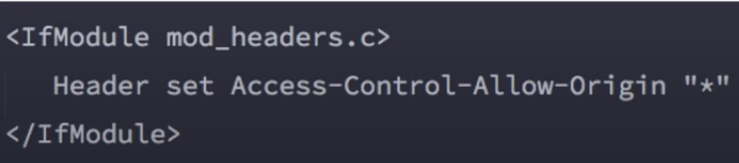
**KESIMPULAN RESTFUL API**

* Untuk menggunakan RESTFUL API, harus menggunakan HTTP method yang benar (POST, GET, PUT, DELETE dll)
* URL/Endpoints nya menggunakan kata benda, tidak boleh menggunakan kata kerja, karena kita tinggal mainkan request method nya saja tergantung operasinya apa.
* Harus Stateless

Jadi intinya nya perbedaan REST dan RESTFUL adalah, dengan RESTFUL kita menggunakan REST secara benar sesuai dengan kaidah kaidah tadi.

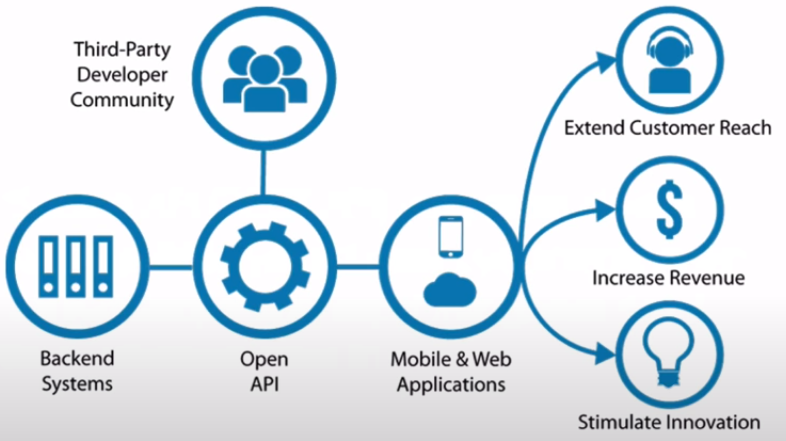
**BEKERJA DENGAN JSON DI LUAR DOMAIN.**

Ada aturan yang dinamakan same-origin policy, yaitu kebijakan browser untuk mengakses dan menampilkan data / script / document dari tempat lain di luar domannya, aturan ini membuat kita tidak bisa mengakses data (.json) dari luar domain / website kita. Jika ingin membuat REST Server kita harus melakukan konfigurasi CORS (Cross Origin Resouce Sharing) pada server kita yaitu mekanisme yang dapat mengakses sebuah resrouce (data, document, script) dari domain lain. Caranya adalah dengan membuat server memperbolehkan adanya akses data dari luar domain. Dengan memodifikas **Access-Control-Allow-Origin** biasanya menggunakan file .htaccess .



**PUBLIC API/OPEN API**

Merupakan REST API yang bisa diakses secara public, yang di dalamnya ada aturan untuk developer agar dapat mengakses data.



Kunjungi OPEN API <https://console.developers.google.com>

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

<https://any-api.com>

<https://github.vom/toddmotton/public-apis>

<https://github.com/farizdotid>

tidak semua API bebas digunakan.

* **Tanpa Otentikasi**
* **Key/Token** – contoh key ini hanya bisa dipakai 1000x dalam sehari. Atau bisa sehari lebih dari 1000 dengan subscription
* **OAuth,** login via facebook, atau via google.

Note : perhatikan juga CORS.