

BAB II

LANDASAN TEORI

A. Reverse Geocoding

Menurut Safaat (2014), *reverse geocoding* adalah konversi kordinat menjadi sebuah alamat yang memungkinkan dapat menemukan sebuah alamat. Pada penelitian ini, kordinat yang dimaksud ialah titik *latitude* dan *longtitude* pada sebuah perangkat *mobile* atau dalam penelitian ini bisa disebut dengan *android*. Dengan memanfaatkan fitur bawaan yang ada di *android* diharapkan aplikasi ini nanti dapat memperoleh titik koordinat dari perangkat *android* untuk selanjutnya di proses untuk mengetahui nama lokasi dan tempat dari karyawan yang melakukan absensi.

Fitur bawaan dari perangkat *android* yang dimaksud untuk mendapatkan data *latitude* dan *longtitude* adalah *A - GPS* sedangkan alat untuk memproses data *latitude* dan *longtitude* agar mendapatkan nama lokasi dan tempat dari data *A - GPS* yang telah didapatkan adalah *API nominatim.org*.

1. *Assisted – GPS (Global Positioning System)*

A – GPS merupakan hasil pengembangan dari teknologi *system GPS* generasi sebelumnya. Dimana *A – GPS* di yakini mampu lebih baik dalam menangani penentu lokasi dengan mengadopsi *system TTFF (Time To First Fix)* atau kecepatan menentukan posisi pertama kali. Jika *GPS* sebelumnya hanya menerima sinyal dari satelit yang meng orbit di bumi untuk menentukan sebuah posisi dari sebuah *GPS receiver (android)*, Beda dengan *A – GPS* yang

menentukan lokasi selain dari satelit yang mengorbit di bumi *A – GPS* juga menentukan lokasi dengan *server* bantuan. *Server* bantuan tersebut akan memberikan informasi tambahan ke perangkat *receiver* yang dapat membantu dalam perhitungan lokasi. Hal tersebut sangat berguna di bandingkan dengan teknologi *GPS* lama yang mungkin akan mengalami kesulitan dalam mendapatkan sinyal satelit secara langsung.

Jaringan operator seluler disini bertindak sebagai *server* bantuan untuk *A-GPS* karena seringkali menara *BTS* memiliki unit penerima *GPS* dan unit penerima *GPS* tersebut secara terus menerus akan mengunduh informasi dari satelit yang mengorbit bumi . Data yang telah di sediakan oleh menara *BTS* kemudian bisa diberikan oleh perangkat seluler *android* untuk mengidentifikasi titik lokasi dengan bentuk *latitude* dan *longitude*.

2. *API nominatim.org*

API (Application Programming Interface) atau secara bahasa bisa diartikan antarmuka pemograman aplikasi. Yang maksudnya adalah seperangkat antarmuka (baik itu berbentuk fungsi, *method*, atau bisa sebuah *URL endpoint*) yang bisa kita gunakan untuk mengembangkan aplikasi, baik dalam satu *platform* atau multi *platform*.

Dengan digunakannya *API* dalam pembuatan sebuah *project* aplikasi, *project* aplikasi dapat cepat di kerjakan oleh *programmer* atau bisa disebut disini adalah sebagai penulis. Karena dengan digunakannya *API* penulis tidak perlu membuat fungsi, *method* atau *URL endpoint* dari nol.

API nominatim.org adalah salah satu layanan *API* gratis yang disediakan di internet. *API nominatim.org* sendiri adalah *API* yang menangani data *latitude* dan *longitude* menjadi data nama lokasi dan wilayah sesuai dengan penjelasan yang telah di sediakan oleh pihak *nominatim.org* di dalam dokumentasi penggunaannya.

Meskipun banyak *API* yang menangani data *latitude* dan *longitude* menjadi data nama lokasi atau bisa disebut dengan *reverse geocoding*, tetapi alasan penulis menggunakan *API* dari *nominatim.org* adalah sebagai berikut

a. Gratis tanpa syarat

Maksud dari gratis tanpa syarat adalah *API nominatim.org* benar – benar gratis tanpa adanya batasan penggunaan. Tidak seperti layanan *API* yang lain dengan menawarkan fasilitas yang hampir sama yaitu *reverse geocoding* mereka memiliki syarat Batasan dalam penggunaan. Jika kita melebihi dari batas penggunaan layanan gratis yang telah ditentukan maka kita tidak bisa menggunakan layanan *API reverse geocoding*. Dan tentunya pasti akan berakibat fatal pada proses kelancaran aplikasi absensi yang tengah berjalan

b. Mudah digunakan

Maksud dari mudah di gunakan adalah *API nominatim.org* dapat digunakan tanpa perlu registrasi terlebih dahulu. Tanpa perlu memberikan data diri kita untuk memakai layanan tersebut.

Dengan 2 alasan yang telah dijabarkan maka penulis memilih untuk menggunakan layanan *API* dari *nominatim.org* dalam memproses data *latitude* dan *longitude*. Untuk penggunaan dari *API* tersebut adalah dengan menggunakan

sebuah *URL* endpoint yang telah disediakan. Dan kemudian diantara *URL* yang telah disediakan di sisipkan dengan parameter data *latitude* dan *longtitude*. Dengan mengakses *URL endpoint* yang telah disisipkan data *latitude* dan *longtitude* maka data lokasi yang di inginkan akan muncul, dan data inilah yang nanti bakal di proses lebih lanjut.

<https://nominatim.openstreetmap.org/reverse?format=geojson&lat=44.50155&lon=11.33989>

Gambar 2.1 API URL Endpoint

```
{
  "type": "FeatureCollection",
  "licence": "Data © OpenStreetMap contributors, ODbL 1.0. https://osm.org/copyright",
  "features": [
    {
      "type": "Feature",
      "properties": {
        "place_id": "18512203",
        "osm_type": "node",
        "osm_id": "1704756187",
        "place_rank": "30",
        "category": "place",
        "type": "house",
        "importance": "0",
        "address_type": "place",
        "name": null,
        "display_name": "71, Via Guglielmo Marconi, Saragozza-Porto, Bologna, BO, Emilia-Romagna, 40122, Italy",
        "address": {
          "house_number": "71",
          "road": "Via Guglielmo Marconi",
          "suburb": "Saragozza-Porto",
          "city": "Bologna",
          "county": "BO",
          "state": "Emilia-Romagna",
          "postcode": "40122",
          "country": "Italy",
          "country_code": "it"
        }
      }
    }
  ]
}
```

Gambar 2.2 Data API Nominatim.org

B. Android

Menurut Sifaat *android* adalah sistem operasi untuk telepon seluler yang berbasis *Linux*. *Android* awalnya dikembangkan oleh *Android, Inc.*, dengan dukungan finansial dari *Google*, dan kemudian *Google* membeli *android* pada

tahun 2005. Umumnya antarmuka yang diusung *android* adalah sebuah layar yang menampilkan beberapa menu dan aplikasi yang bisa disentuh langsung untuk pengoperasiannya.

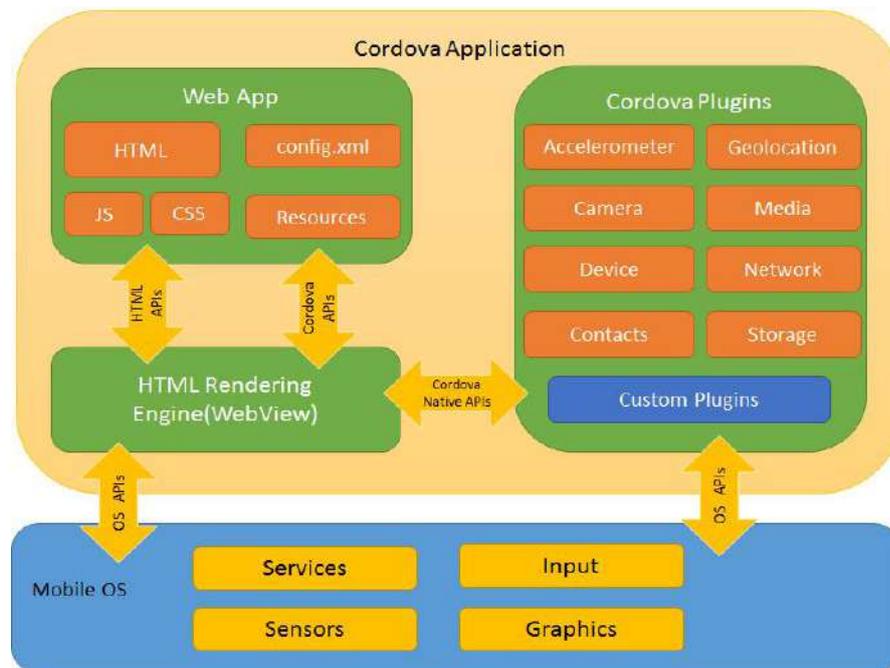
Selain dengan antarmuka yang bisa dianggap futuristik *android* juga memiliki layanan *store* yang memiliki banyak aplikasi yang bisa di unduh secara gratis. Dan *android* juga menjadi *system* operasi seluler terpopuler saat ini.

Untuk melakukan pembuatan aplikasi berbasis *android* diperlukan *tools* untuk proses pengembangannya. Salah satunya adalah *Android Software Development Kit (Android SDK)*.

Di dalam *Android SDK* terdapat beberapa *tools* antara lain *debugger*, *software libraries*, *emulator*, dokumentasi, *sample code* dan tutorial untuk mempermudah proses *developing*. Untuk Bahasa pemrograman yang sering digunakan dalam proses pengembangan aplikasi *android* adalah *Java*. Selain Bahasa pemrograman *Java*, *android* juga bisa dikembangkan dengan beberapa Bahasa pemrograman seperti, *C++*, *Go*, *Kotlin*, dan baru baru ini *android* bisa dikembangkan dengan Bahasa pemrograman *Dart*.

1. *Cordova*

Cordova adalah salah satu *tools* untuk pengembangan aplikasi *android* yang sifatnya *hybrid*. Artinya aplikasi yang dibuat dengan *cordova* bisa di convert ke *platform* lain selain *android*. *Cordova* sendiri sebenarnya adalah alat untuk mengubah aplikasi *website* yang di bangun dengan *HTML*, *CSS*, dan Bahasa pemrograman *JavaScript* agar bisa berjalan di *platform android* layaknya aplikasi *native*.



Gambar 2.3 Arsitektur *Cordova*

Untuk memulai pengembangan aplikasi *android* berbasis *Cordova* diperlukan *tools* sebagai berikut :

- Node Js* berfungsi untuk mengunduh dan untuk instalasi tool *Cordova*
- Java JDK* yang berfungsi sebagai bahasa pemrograman untuk menjalankan *android SDK* dan untuk membuat plugins *cordova* berfungsi sebagai mana mestinya
- Android SDK* sebagai *tools debugger* dan untuk melakukan *build* aplikasi ke dalam ekstensi *.apk* agar aplikasi berbasis *cordova* dapat di install dan berjalan di *system* operasi *android*.

Setelah *tools* diatas sudah lengkap kita bisa memulai pembuatan aplikasi *android* berbasis *Cordova*.

Cordova sendiri memiliki *command line* untuk proses pengembangan aplikasi *hybrid* diantaranya sebagai berikut :

Tabel 2.1 *Comand line Cordova*

<i>npm install – cordova</i>	Perintah disamping digunakan untuk melakukan instalasi <i>Cordova</i> ke dalam <i>system operasi windows</i>
<i>cordova create hello com.example.hello HelloWorld</i>	Perintah disamping digunakan untuk membuat <i>project</i> baru dengan nama <i>project</i> aplikasi bernama hello
<i>cordova platform add ios</i>	Perintah disamping digunakan jika <i>project cordova</i> akan di gunakan dan di <i>install</i> untuk <i>platform ios</i> atau <i>apple</i>
<i>cordova platform add android</i>	Perintah disamping digunakan jika <i>project cordova</i> akan di gunakan dan di <i>install</i> untuk <i>platform android</i>
<i>cordova build</i>	Perintah yang digunakan untuk melakukan build aplikasi <i>cordova</i> agar berbentuk .apk untuk <i>android</i> dan .ipa untuk aplikasi <i>ios</i> . Sesuai dengan <i>platform</i> yang ditambahkan dengan di <i>project</i> .
<i>cordova build android</i>	Perintah yang di gunakan untuk melakukan build aplikasi tapi dengan <i>platform</i> yang lebih spesifik.
<i>npm info cordova version</i>	Untuk melakukan <i>check</i> versi <i>cordova</i> yang digunakan
<i>cordova plugin add</i>	Untuk menambahkan plugin untuk membantu pengerjakan <i>project cordova</i>
<i>cordova plugin ls</i>	Untuk menampilkan daftar plugin yang digunakan di dalam <i>project cordova</i>

C. Pemograman *Website*

1. *Website*

Bekti (2015:35), *Website* merupakan kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman.

Seperti yang dijelaskan barusan *website* memiliki 2 sifat, yaitu *website* statis dan *website* dinamis. Yang dimaksud *website* statis adalah sebuah *website* dengan isi atau konten yang terkandung di dalamnya memiliki nilai yang tetap (statis) tidak bisa berubah – ubah sesuai keinginan *user*. Contoh dari *website* statis adalah sebuah *website profile* perusahaan yang dimana *website* tersebut hanya menampilkan informasi – informasi yang tetap dan tidak berubah ubah. Sedangkan yang dimaksud dengan *website* dinamis adalah sebuah *website* dengan isi atau konten yang terkandung didalamnya dapat berubah – ubah (dinamis) yang dapat berinteraksi dengan input yang dimasukkan oleh *user* langsung. Contoh dari *website* dinamis adalah sebuah *website toko online* dimana konten yang terdapat di dalam *website* tersebut dapat diubah ubah nilainya sesuai kehendak *user*. Dan nilai yang telah di ubah biasanya di simpan di dalam database yang terhubung dengan *website* tersebut.

Website yang akan menampilkan data lokasi karyawan yang akan di akses oleh *HRD* adalah termasuk *website* yang dinamis. Karena nilai yang terdapat didalamnya bersifat berubah – ubah tergantung lokasi karyawan melakukan

absensi. Untuk membuat sebuah *website* dinamis diperlukan bahasa pemrograman yang sesuai. Di penelitian ini penulis menggunakan 2 bahasa pemrograman untuk membangun sebuah *website* dinamis, yaitu bahasa pemrograman *PHP* dan bahasa pemrograman *Javascript*.

2. Bahasa pemrograman *PHP*

Menurut Anhar (2010:23), *PHP* (*Hypertext Preprocessor*) adalah bahasa pemrograman *web* berupa *script* yang dapat diintegrasikan dengan *HTML*.

Sedangkan *HTML* sendiri menurut Nugroho (2006c:48) *HTML* adalah bahasa pemformatan teks untuk dokumen-dokumen pada jaringan komputer yang sering disebut sebagai *world wide web*.

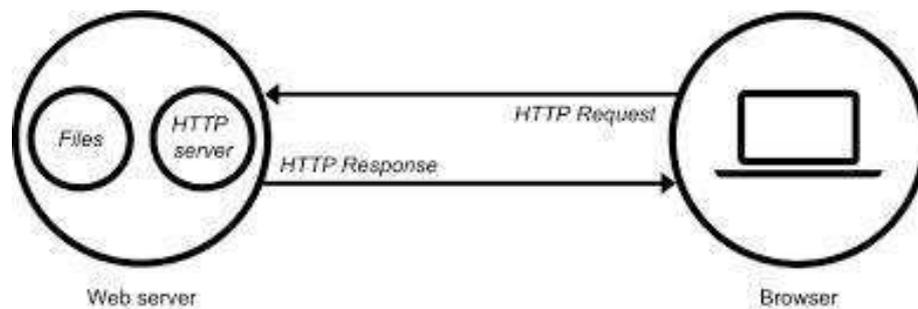
PHP juga sering disebut juga dengan bahasa pemrograman *server side* karena bahasa pemrograman *PHP* berfungsi sebagai pengolah data dari sebuah *database engine* agar bisa ditampilkan ke sebuah halaman *website* dinamis agar data bisa dibaca oleh pengguna tanpa perlu membuka *database* nya langsung.

Untuk penulisan dari bahasa pemrograman *PHP* sendiri memiliki aturan harus diawali dengan script `<?php` dan diakhiri dengan `?>` dan disimpan dalam bentuk file ber ekstensi `.php`. Untuk penulisan program *PHP* bisa di sebuah *tool text editor* di *system* operasi yang dipakai. Penulis menggunakan *text editor Visual Studio Code* sebagai *tool* untuk menulis kode *PHP* yang dapat di unduh gratis di dalam Internet.

Agar program *PHP* dapat berjalan sebagai mestinya diperlukan sebuah *Web Server*. Pengertian dari *Web server* itu sendiri adalah sebuah *software* yang

memberi layanan data yang berfungsi untuk menerima permintaan dari *HTTP* ataupun *HTTPS* dari sebuah *browser*.

Umumnya sebuah *web server* sudah tersedia modul untuk menganani bahasa pemograman *PHP* dan modul – modul lainnya yang dapat menangani proses permintaan *HTTP* ataupun *HTTPS*. Contoh dari *web server* yang sering di gunakan di internet adalah *Web server Apache*.



Gambar 2.4 *Web Server*

Dalam pengembangan *website* dinamis menggunakan bahasa pemograman *PHP*, penulis di mudahkan dengan adanya *framework PHP*. *PHP* memiliki banyak sekali jenis *framework*, salah satunya adalah *framework PHP Laravel*. Yang digunakan penulis dalam pembuatan aplikasi *monitoring* absensi, yang nantinya akan diakses oleh bagian *HRD*.

Menurut Syam (2009), *Framework* adalah sekumpulan *library* yang diorganisasikan pada sebuah rancangan arsitektur untuk memberikan kecepatan, ketepatan, kemudahan dan konsistensi di dalam pengembangan aplikasi.

Laravel adalah sebuah sebutan dari salah satu *framework PHP* yang memiliki konsep *MVC* dalam pengembangan aplikasinya. Konsep *MVC* yang dimaksud adalah konsep yang memisahkan logika program (*Controller*), *query database (Model)*, dan tampilan aplikasi *website (View)* yang ada dalam *folder*

project supaya *file project* lebih tertata dan rapi serta mudah untuk proses pengkodean. Kelebihan menggunakan *framework Laravel* dalam pengerjaan sebuah aplikasi *website* antara lain :

- a. *Framework Laravel* gratis, untuk menggunakannya pengguna bisa mengunduhnya lewat aplikasi *composer*.
- b. Kaya akan fitur, *Laravel* menyediakan fitur yang sudah siap pakai jadi pengguna tidak perlu membuatnya dari awal. Contohnya adalah *system login*, secara *default Laravel* sudah menyiapkan fitur tersebut dan pengguna tinggal memodifikasi sesuai akan kebutuhan.
- c. Dokumentasi yang menarik, selain dokumentasi yang di suguhkan lengkap, dokumentasi juga ditampilkan didalam sebuah *website* yang tampilannya menarik dan *responsive*.
- d. Terdapat *Comand Line*, Uniknya *framework Laravel* memiliki perintah *commad line* yang bisa di jalankan di *terminal / CMD* untuk mempercepat pengerjakan *project aplikasi*.
- e. Komunitas yang aktif, dengan adanya komunitas memudahkan pengguna untuk melakukan diskusi program jika pada saat penggunaan *framework Laravel* memiliki masalah.
- f. Banyak *Library*, dengan disediakan *library* membuat pengerjakan *project* akan lebih cepat dan tidak memakan waktu

3. Bahasa pemograman *Javascript*

“*Javascript* adalah bahasa pemrograman yang digunakan untuk membuat *web* lebih dinamis dan interaktif” Adhi (2012 :300) .

Menurut Sidik (2007), terdapat beberapa hal khusus yang terdapat dalam *Javascript* sebagai berikut :

- a. Menggunakan blok awal “{” dan blok akhir “}”.
- b. *Automatic conversion* dalam pengoperasian tipe data yang berbeda.
- c. *Case sensitive*, sehingga programmer harus berhati-hati dalam menggunakan variabel, fungsi, dan lain-lain.
- d. *File extension* yang umumnya digunakan adalah “*.js”.
- e. Setiap statement dapat diakhiri dengan “;” tetapi dapat juga tidak.
- f. Jika tidak didukung oleh *browser* tipe lama, scriptnya dapat disembunyikan di antara tag “<!--” dan “-->”.
- g. Jika program dalam satu baris terlalu panjang, dapat disambung ke baris berikutnya dengan karakter “\” di awal baris tersebut.

Dengan menggunakan *Javascript* dapat membantu kita dalam pembuatan suatu proses yang dibuat dengan *PHP* tampak begitu cepat karena tanpa memalui *request* ke *web server*. Sebagai contoh saat kita mengisi *form* registrasi untuk pendaftaran pada sebuah situs *web*, lalu men-klik tombol submit, *website* akan memproses isian tersebut dalam beberapa detik ke *web server* dan saat proses selesai ternyata mendapati bahwa ada isian *form* yang belum terisi yang membuat error program.

Dengan digunakannya *Javascript* proses untuk mengecek isian *form* yang perlu di isi bisa langsung diproses di *web browser* tanpa melalui *web server* terlebih dahulu. Ini akan memudahkan *user* agar tampak *website* yang dibangun se akan akan lebih cepat dan *responsive*. *Javascript* nantinya akan digunakan untuk

menangani proses penampilan data lokasi karyawan dengan bantuan *library leaflet.js*. *Leaflet.js* sendiri adalah *library javascript* yang bisa digunakan untuk menampilkan halaman peta layaknya *Google Maps*.

D. Database

Database atau basis data merupakan suatu kumpulan data terhubung yang disimpan secara bersama-sama pada suatu media, yang diorganisasikan berdasarkan sebuah skema atau struktur tertentu, dan dengan *software* untuk melakukan manipulasi untuk kegunaan tertentu. Basis data bisa diartikan juga sebagai sekumpulan data yang disusun dalam bentuk beberapa *table* yang saling memiliki relasi maupun berdiri sendiri. (Pamungkas, 2017:2).

PostgreSQL adalah salah satu *Relational Database Manajemen System (RDBMS)* yang bersifat *open source* artinya *PostgreSQL* di kembangkan oleh tim yang ada di seluruh dunia yang dapat digunakan secara bebas. *PostgreSQL* tidak di kontrol oleh perusahaan atau badan swasta lainnya sehingga *source code* (kode sumber) yang tersedia bisa di dapatkan secara gratis.

Adapun kelebihan menggunakan antara lain sebagai berikut :

1. *PostgreSQL* memiliki arsitektur multiproses (*forking*) yang berarti memiliki stabilitas yang lebih tinggi, sebab satu proses anak yang mati tidak akan menyebabkan seluruh daemon mati meskipun pada kenyataannya, dulu ini sering terjadi.
2. Dalam kondisi load tinggi (jumlah koneksi simultan besar), kecepatan *PostgreSQL* sering mengalahkan *MySQL* untuk *query* dengan klausa *JOIN*

yang kompleks, hal ini dikarenakan *PostgreSQL* mendukung *locking* di level yang lebih rendah, yaitu *row*.

3. *PostgreSQL* memiliki fitur OO seperti pewarisan tabel dan tipe data, atau tipe data *array* yang kadang praktis untuk menyimpan banyak item data di dalam satu *record*. Dengan adanya kemampuan OO ini maka di *PostgreSQL*, kita dapat mendefinisikan sebuah tabel yang mewarisi definisi tabel lain.
4. *PostgreSQL* menyediakan hampir seluruh fitur-fitur *database* seperti yang terdapat dalam produk *database* komersial pada umumnya.
5. *PostgreSQL* pun memiliki tipe data geometri (seperti titik, garis, lingkaran, poligon) yang mungkin berguna bagi aplikasi ilmiah tertentu
6. *PostgreSQL* memberikan kita kemampuan mendefinisikan sebuah *field* sebagai *array*.
7. *PostgreSQL* memiliki hampir semua fasilitas standar yang biasanya diinginkan: *view* (tabel virtual), *trigger*, *subselect*, *stored procedure* (dalam beberapa bahasa), dan *foreign key constraint*
8. *PostgreSQL* juga memiliki apa yang disebut *rule*, yaitu tindakan *custom* yang bisa kita definisikan dieksekusi saat sebuah tabel di-*INSERT*, *UPDATE*, atau *DELETE*
9. *PostgreSQL* juga mempunyai kemampuan untuk membuat konektivitas dengan database lain seperti *pgdump*, *Interbase*, *pgaccess* dan hampir semua *database* pada Linux.
10. Kemampuannya menampung data spasial, sehingga ia bisa digunakan dalam pembuatan situs yang berbasis **WebGIS** untuk pemetaan dan sebagainya.

11. *PostgreSQL* memiliki lisensi *GPL (General Public License)* dan oleh karena itu *PostgreSQL* dapat digunakan, dimodifikasi dan didistribusikan oleh setiap orang tanpa perlu membayar lisensi (*free of charge*) baik untuk keperluan pribadi, pendidikan maupun komersil.
12. *PostgreSQL* mendukung banyak jenis bahasa pemrograman, antara lain: *SQL, C, C++, Java, PHP, etc.*
13. *PostgreSQL* juga merupakan *alternative* untuk sistem database *open-source* lainnya seperti *MySQL* dan *Firebird*, terutama sistem propietari seperti *Oracle, Sybase, IBM's DB2* dan *Microsoft SQL Server*