

BAB II

LANDASAN TEORI

A. Sistem Informasi

Sistem informasi adalah suatu sistem yang menyediakan informasi yang berguna bagi pemakainya (Aprillia, Astuti, & Dewantara, 2017). Sistem didefinisikan secara teknis sebagai satuan yang berhubungan satu sama lain mengumpulkan, memproses, menyimpan serta untuk mendistribusikan informasi dalam pengambilan keputusan dan kendali dari sebuah organisasi. Adapun aktivitas yang menghasilkan informasi yang dibutuhkan suatu organisasi diantaranya untuk membuat keputusan, mengendalikan operasi, meneliti permasalahan yang ada, dan menciptakan jasa atau produk baru. Suatu sistem memiliki enam komponen sebagai berikut:

1. Komponen Input merupakan data yang sudah masuk kedalam sistem informasi. Komponen ini digunakan sebagai dasar untuk pengolahan sebuah informasi.
2. Komponen Output yaitu informasi yang berguna bagi pemakainya. Hasil output sistem informasi dibuat menggunakan data yang sudah disiapkan di basis data, kemudian akan diproses dengan menggunakan proses tertentu.
3. Komponen Basis Data adalah kumpulan data yang saling berhubungan satu sama lain. Lalu akan tersimpan di perangkat keras yang kemudian digunakan perangkat lunak untuk memanipulasinya.
4. Komponen Model merupakan informasi yang dihasilkan dari sebuah data. Dan data tersebut diambil dari sebuah basis data kemudian diolah dengan menggunakan model tertentu yaitu model logika yang menunjukkan proses perhitungan matematika.
5. Komponen Teknologi adalah komponen yang mempercepat laju sistem informasi dalam pengolahan data.
6. Komponen Kontrol yaitu suatu komponen yang digunakan sebagai jaminan bahwa informasi yang dihasilkan oleh sistem informasi adalah informasi yang tepat dan akurat.

B. Data

Sumber informasi adalah data. Data merupakan fakta kasar atau sebuah gambaran yang dikumpulkan dari suatu keadaan tertentu (Hastanti, Purnama, & Wardati, 2015). Data adalah hal yang merujuk pada fakta-fakta baik berupa angka, teks, dokumen, gambar, badan, suara yang mewakili deskriptif verbal ataupun kode-kode tertentu. Jadi, data merupakan bentuk yang masih mentah sehingga perlu dilakukan pengolahan lebih lanjut. Dalam penelitian ini data yang digunakan adalah data sekunder. Adapun yang dinamakan data sekunder yaitu data tersebut tidak diperoleh secara langsung dari narasumber. Data yang digunakan dalam penelitian ini berasal dari website BMKG Semarang.

C. Cuaca

Cuaca merupakan fenomena yang saat ini sedang terjadi di atmosfer bumi (Ulinuha & Farida, 2018). Aktivitas manusia dapat dipengaruhi oleh suatu kondisi cuaca, seperti pertanian, bidang perkebunan, dan masih banyak lagi. Penentuan cuaca merupakan proses penerapan ilmu pengetahuan dan teknologi untuk menentukan keadaan atmosfer bumi. Proses penentuan cuaca menggunakan metode yang menghasilkan cuaca penentuan keluaran. Penggunaan metode yang baik, baru-baru ini menjadi suatu kegiatan yang sering dilakukan oleh peneliti cuaca ataupun atmosfer. Untuk mendapatkan hasil yang lebih baik dan akurat, peneliti didorong untuk terus meneliti atmosfer yang berhubungan dengan cuaca dan mengembangkan metode untuk terus memenuhi kebutuhan informasi tentang cuaca atau kondisi atmosfer.

Berikut ini unsur-unsur cuaca, diantaranya sebagai berikut:

1. Suhu/temperatur

Suhu udara/temperatur udara di suatu tempat akan berbeda dengan tempat yang lainnya, disebabkan oleh beberapa faktor diantaranya:

- a. Ketinggian suatu tempat, semakin datar suatu tempat maka panas yang diterima juga semakin besar.

- b. Lama penyinaran matahari ke bumi adalah intensitas penyinaran matahari dibelahan bumi yang sangat bervariasi tergantung oleh letak lintang.
- c. Keadaan tumbuhan dipermukaan bumi.
- d. Keadaan awan, jika di atmosfer banyak terdapat awan maka panas yang diterima bumi akan lebih kecil karena terserap oleh awan.
- e. Sudut penyinaran matahari adalah suatu sudut yang terbentuk oleh sinar matahari pada bidang permukaan di bumi.

2. Tekanan Udara

Yang dimaksud dengan tekanan udara yaitu suatu gaya yang timbul karena adanya berat dari lapisan udara, besarnya tekanan udara disuatu tempat dapat berubah-ubah. Atau kerapatan massa udara dalam satuan wilayah tertentu.

3. Kelembaban Udara

Kelembaban udara yaitu banyak sedikitnya uap air yang terkandung di dalam udara pada saat tertentu.

4. Angin

Angin merupakan udara yang bergerak dari daerah bertekanan tinggi ke daerah tekanan rendah.

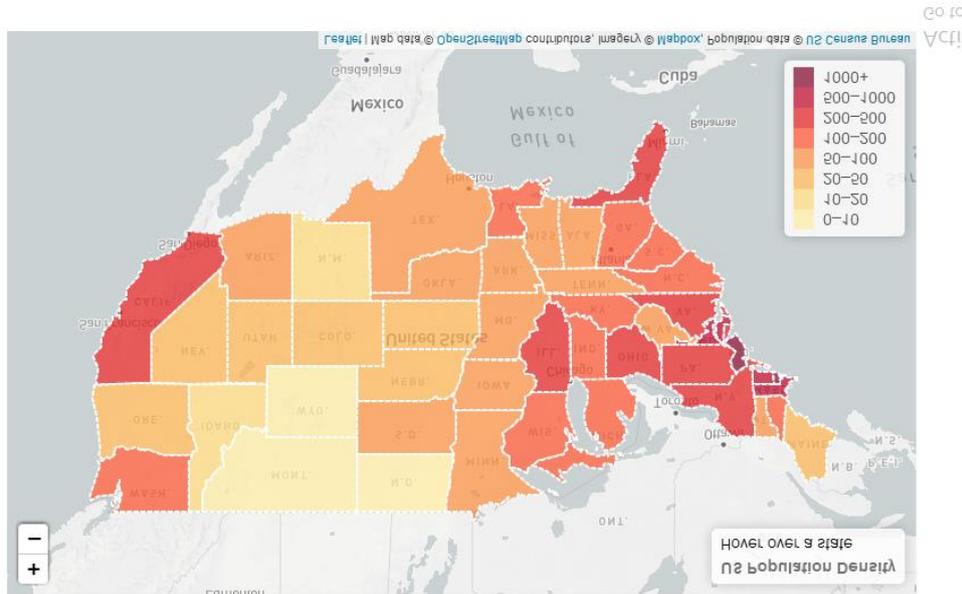
5. Hujan

Hujan adalah jumlah air hujan yang turun pada suatu wilayah dalam waktu tertentu.

D. Leaflet

Leaflet merupakan pemetaan interaktif yang digunakan untuk pengembangan sebuah *library JavaScript* (Ganesha, Hayuhardhika, & Rahayudi, 2020). Ukuran *file JavaScript leaflet* ini sangat ringan yaitu sekitar 38 KB. Untuk kebutuhan pengembangan, pemetaan *library* ini mempunyai fitur yang lengkap. Rancangan dari *leaflet* dibuat dengan sangat sederhana dan bisa beroperasi secara efisien pada semua jenis *platform* baik itu *desktop* maupun *smartphone*. *Leaflet* memiliki keunggulan antara lain dapat diperluas dengan

menggunakan banyak *plugin* dan mempunyai dokumentasi yang baik. Adapun contoh pemanfaatan leaflet sebagai berikut:



Sumber: *Leaflet* (2020)

Gambar 1 Contoh pemanfaatan *leaflet*

Mapbox merupakan salah satu peta terbesar pada situs-situs ternama seperti *Pinterest*, *Foursquare* dan *Evernote* sejak tahun 2010 (Ganesha, Hayuhardhika, & Rahayudi, 2020). *Mapbox* memperbanyak pilihan peta kustomnya yang berfungsi untuk mengisi keterbatasan yang dimiliki penyedia peta seperti *google maps*. *Mapbox* yaitu pencipta sejumlah pustaka dan aplikasi bebas terkenal, meliputi spesifikasi *MBTiles*, Kartografi *TileMill IDE*, pustaka *Java Script Leaflet*, dengan bahasa gaya dan parser peta *CartoCSS*, dan pustaka *Java Script Mapbox.js*. Adapun data *mapbox* seperti *Open Street Map* dan *NAS* diambil dari data-data terbuka dan sumber data berbayar seperti *Digital Globe*. Dibangun dengan teknologi *Node.js*, *CouchDB*, *Mapnik*, *GDAL* dan *Leafletjs*. *Mapbox* merupakan dukungan terhadap *GeoJSON-encoding* obyek geografis dalam bentuk *JSON*. Turunan dari *Open Street Map* salah satunya *Mapbox*.

E. PHP (*Hypertext Preprocessor*)

PHP adalah suatu bahasa dalam bentuk *script* yang ditempatkan didalam *server* yang kemudian diproses (Rubiati, 2018). Selanjutnya hasil pemrosesan tersebut dikirim kepada website browser klien. Adanya bahasa pemrograman ini dirancang secara khusus untuk membuat web dinamis (Informatika et al., 2018). Artinya dalam pemrograman bahasa PHP ini bisa membentuk sebuah tampilan berdasarkan permintaan terkini. Ketika dipanggil dalam *web browser*, PHP yang dituliskan akan di-*parsing* di dalam *web server* oleh *interpreter* PHP, kemudian akan diterjemahkan ke dalam dokumen HTML. Selanjutnya akan ditampilkan kembali ke *web browser*. Karena pemrosesan program PHP dilakukan dilingkungan *web server*, PHP dikatakan sebagai bahasa sisi server (*server-side*). Oleh karena, kode PHP tidak akan terlihat pada saat *user* memilih perintah “View Source” pada *web browser* yang mereka gunakan. PHP mendukung delapan tipe nilai atau tipe data yang terdiri dari empat nilai skalar (*single-value*): *integer*, *floating point*, *string*, dan *boolean*; dua nilai koleksi: *array* dan objek. Sisanya adalah tipe khusus: *resource* dan NULL. Adapun penjelasan dari 8 tipe data ini sebagai berikut:

1. *Integer*

Integer adalah tipe data yang merepresentasikan bilangan utuh atau bulat, seperti 12, 14, 2009, dan sebagainya. Rentang yang diperbolehkan adalah dari -2, 147, 483, 648 sampai +2, 147, 483, 647. Nilai literal dari tipe *integer* dapat ditulis dalam bentuk desimal (basis 10), oktal (basis 8), maupun heksadesimal (basis 16).

2. *Floating-point*

Floating-point number (sering juga dinamakan dengan bilangan *real*) adalah tipe data yang merepresentasikan nilai-nilai numerik dalam bentuk pecahan atau mengandung angka desimal dibelakang koma. *Floating-point* dalam PHP selaras dengan tipe *double* dalam bahasa C, biasanya berada dalam rentang 1.7E-308 dan 1.7E+308 dengan 15 digit akurasi. Bilangan *floating-point* dapat ditulis dalam notasi normal dan notasi sains (*scientific*). *Floating-point* merupakan representasi “kira-kira” dari suatu

bilangan. Sebagai contoh, 3.5 secara aktual direpresentasikan dengan nilai 3.4999999999.

3. *String*

PHP mendukung dua tipe *string* yaitu yang diapit oleh petik ganda dan tunggal. Dalam pembuatan *web*, *string* merupakan tipe data yang kerap kali dijumpai dalam berbagai kasus. *String* yang diapit oleh petik ganda juga dapat mendukung *escape sequence*, yaitu beberapa karakter yang dianggap sebagai satu karakter, yang digunakan untuk merepresentasikan kedudukan karakter tertentu. Adapun tabel keterangan dari *escape sequence* sebagai berikut:

Tabel 1 Keterangan *escape sequence*

| Escape Sequence | Karakter yang direpresentasikan |
|---|--|
| <code>\”</code> | <i>Double quotes</i> |
| <code>\n</code> | <i>Newline</i> |
| <code>\r</code> | <i>Carriage Return</i> |
| <code>\t</code> | <i>Tab</i> |
| <code>\\</code> | <i>Backslash</i> |
| <code>\\$</code> | <i>Dollar sign</i> |
| <code>\{</code> | <i>Left brace</i> |
| <code>\}</code> | <i>Right brace</i> |
| <code>\[</code> | <i>Left bracket</i> |
| <code>\]</code> | <i>Right bracket</i> |
| <code>\0</code> sampai <code>\777</code> | Karakter ASCII yang direpresentasikan dengan nilai oktal |
| <code>\x0</code> sampai <code>\xFF</code> | Karakter ASCII yang direpresentasikan dengan nilai |

Sumber: Rubiati (2018)

4. *Boolean*

Tipe *boolean* digunakan untuk merepresentasikan nilai kebenaran (*truth value*). Nilai yang mungkin dimasukan kedalam variabel *boolean* adalah *true* dan *false*. Tipe ini biasanya dijumpai pada pendefinisian kondisi, baik dalam struktur pemilihan maupun pengulangan. Dalam PHP, berikut ini adalah nilai-nilai *false* sebagai berikut:

- a. *Keyword false*
- b. *Integer 0*
- c. *Floating-point 0.0*
- d. *String* kosong (“”) dan *string* “0”
- e. *Array* dengan nol elemen
- f. Objek yang tidak memiliki nilai atau fungsi
- g. Nilai NULL

5. *Array*

Array adalah variabel yang menyimpan sekelompok nilai, yang dapat didefinisikan berdasarkan posisinya (nomor, dengan 0 sebagai indeks pertamanya) atau dengan nama (bertipe *string*) yang telah didefinisikan sebelumnya. Terdapat beberapa cara untuk mengakses nilai *array* melalui blok pengulangan, namun yang paling lazim adalah dengan menggunakan struktur pengulangan *foreach*.

6. *Objek*

PHP mendukung *Objek-Oriented Programming* (OOP), yang dapat memudahkan dalam proses *debugging*, *maintenance*, dan penggunaan ulang kode (*reuseable code*). Dalam OOP, kelas merupakan unit terpenting. Kelas adalah suatu definisi dari struktur yang memiliki properti dan metode (fungsi). Dalam PHP, kelas didefinisikan menggunakan kata kunci *class*.

Adapun operator dalam PHP dapat dikategorikan menjadi beberapa kelompok, antara lain:

1. Operator *Assignment* (pengisian)

Operator *assignment* merupakan operator yang digunakan untuk mengisi nilai ke dalam suatu variabel. PHP menggunakan operator sama dengan (=) untuk melakukan hal ini, seperti layaknya bahasa C/C++ dan Java. Sebagai bahan perbandingan, dalam bahasa Pascal/Delphi operator yang digunakan sebagai operator *assignment* adalah :=.

2. Operator Aritmetika

PHP memiliki dukungan terhadap dukungan operasi perhitungan standar dengan menyediakan operator-operator aritmetika.

Daftar operator yang termasuk ke dalam kelompok ini sebagai berikut:

Tabel 2 Keterangan simbol operator aritmetika

| Operator | Keterangan |
|-----------------|---|
| + | Operator untuk penjumlahan |
| - | Operator untuk pengurangan |
| * | Operator untuk perkalian |
| / | Operator untuk pembagian |
| % | Operator untuk mengetahui sisa dari pembagian (modulus) |
| ++ | Operator untuk menurunkan nilai variabel sebesar 1 (<i>increment</i>) |
| -- | Operator untuk menurunkan nilai variabel sebesar 1 (<i>decrement</i>) |

Sumber: Rubiati (2018)

3. Operator Relasional

Operator ini dapat digunakan untuk membandingkan dua buah nilai. Hasil operasi perbandingan ini selalu menghasilkan nilai *boolean* (true / false). Dalam PHP, yang termasuk ke dalam operator relasional adalah sebagai berikut :

Tabel 3 Keterangan simbol operator relasional

| Operator | Keterangan |
|-----------------|-------------------|
|-----------------|-------------------|

| | |
|--------------|---|
| $\$a == \b | Memeriksa apakah \$a sama dengan \$b |
| $\$a != \b | Memeriksa apakah \$a tidak sama dengan \$b |
| $\$a < \b | Memeriksa apakah \$a lebih kecil dari \$b |
| $\$a > \b | Memeriksa apakah \$a lebih besar dari \$b |
| $\$a <= \b | Memeriksa apakah \$a lebih kecil atau sama dengan \$b |
| $\$a >= \b | Memeriksa apakah \$a lebih besar atau sama dengan \$b |

Sumber: Rubiati (2018)

4. Operator Logika

Operator ini dapat digunakan untuk mengoperasikan dua buah operasi yang bertipe *boolean*. Hasil dari operasi ini akan bertipe *boolean*. Tabel dibawah ini menunjukkan operator logika dalam PHP sebagai berikut:

Tabel 4 Keterangan simbol operator logika

| Operator | Keterangan |
|-----------------|-------------------|
| && | Logika AND |
| | Logika OR |
| ! | Logika NOT |

Sumber: Rubiati (2018)

Operator && akan menghasilkan nilai true jika kedua operasi bernilai true. Selain operasi ini maka menghasilkan nilai false. Operator || akan menghasilkan nilai false jika kedua operasi nilainya false. Selain kondisi tersebut, operator || akan menghasilkan nilai true. Sedangkan operator ! digunakan untuk melakukan negasi terhadap suatu nilai boolean. Jika operator ini bernilai true, maka operator ini menghasilkan nilai false. Jika sebaliknya, maka operator ini menghasilkan nilai false.

F. MariaDB

MariaDB adalah implementasi dari sebuah sistem manajemen *database* relasional (RDBMS) dengan menggunakan lisensi dari GPL (*General Public License*) yang akan didistribusikan secara gratis (Kaban, Yunita, Fajrillah, & Faradillah, 2019). Pada setiap pengguna bisa bebas memakai MariaDB ini, akan tetapi dengan menggunakan batasan dari perangkat lunak tersebut tidak boleh dijadikan sebagai produk turunan yang bersifat secara komersial. Adapun MariaDB ini merupakan versi dari pengembangan MySQL sejak diakuisisinya MySQL oleh Oracle pada bulan September 2010. Semua perintah dari SQL dalam MySQL bisa diterapkan kedalam MariaDB.

G. Website

Website merupakan komponen atau kumpulan komponen yang terdiri dari teks, gambar, suara animasi sehingga lebih merupakan media informasi yang menarik untuk dikunjungi (Hanafi & Karsam, 2012). Website adalah halaman informasi yang disediakan melalui jalur internet sehingga bisa diakses diseluruh dunia selama terkoneksi dengan jaringan internet.

Secara garis besar, website bisa digolongkan menjadi dua bagian yaitu:

1. Website Statis

Website statis adalah web yang mempunyai halaman tidak berubah. Artinya untuk melakukan perubahan pada suatu halaman website dilakukan secara manual dengan mengedit source code yang menjadi struktur website tersebut.

2. Website Dinamis

Website dinamis merupakan website yang secara struktur diperuntukan untuk update sesering mungkin. Website dinamis terdiri dari halaman frontend yang bisa diakses oleh user pada umumnya, juga disediakan halaman backend untuk mengedit konten dari website dinamis adalah web berita atau web portal yang didalamnya terdapat fasilitas berita, polling dan sebagainya.

H. *Web Services*

Web services merupakan aplikasi yang dibuat agar bisa dipanggil atau diakses oleh aplikasi lain melalui jaringan internet dengan menggunakan format pengiriman pesan (Christian, Rindengan, & Manembu, 2016). Format pertukaran data yang dapat dilakukan dengan menggunakan format data xml dan json. Sebuah *web services* dapat dipanggil dengan menggunakan protokol lain seperti SMTP (*Simple Mail Transfer Protocol*), akan tetapi yang paling umum digunakan http . Karena *web services* menggunakan protokol http, tentu PHP sebagai bahasa pemrograman web menjadi salah satu kekuatan bahasa pemrograman yang mengelola *web services*.

I. *Rest*

Rest adalah gaya arsitektur perangkat lunak yang bisa digunakan saat merancang sistem perangkat lunak (Christian, Rindengan, & Manembu, 2016). *Rest* merupakan desain yang ideal untuk perangkat lunak layanan berbasis web. Adapun prinsip-prinsip dalam *rest* sebagai berikut :

1. Menyediakan tiap-tiap sumber daya dengan menggunakan ID yang unik, misalnya URI.
2. Sumber daya URI link dengan satu sama lain, membangun hubungan antara sumber daya.
3. Menggunakan metode standar (http, jenis media dan xml).
4. Sumber Daya dapat memiliki beberapa representasi yang mencerminkan negara sistem yang berbeda.
5. Komunikasi harus menggunakan http

Jika kita menggunakan *rest*, maka pendekatan *client* dan *server* digunakan untuk user terpisah dari penyimpanan data. Adapun interaksi antara *client* dan *server* dengan menggunakan antarmuka seragam. Salah satu kunci element dalam arsitektur yaitu konsep *resource* dimana setiap informasi dapat menjadi *resource*. Setiap informasi mempunyai URI sendiri yang merupakan suatu prinsip utama dari *web semantic* yang berkembang dari www dimana semantik informasi serta layanan di web didefinisikan. Sebuah *resource* juga dapat memiliki meta data yang terkait seperti media-

jenis dan bertahan saat dimodifikasi. Ketika kita mengakses sumber daya dengan menggunakan http, pengenalan sumber daya ditentukan bersama dengan tindakan yang akan dilakukan pada sumber daya itu. URI mengidentifikasi sumber daya tindakan yang akan dilakukan didefinisikan menggunakan kata kerja http. Ada satu set kata kerja http dan setiap kata kerja dapat memiliki semantik terkait yang membantu untuk mengidentifikasi tindakan yang akan dilakukan pada sumber daya. Berikut ini tabel yang merangkum kata kerja http dan bagaimana mereka berlaku saat menggunakan rest sebagai berikut :

Tabel 5 Penjelasan verb di http

| Verb | Deskripsi |
|---------------|--|
| <i>Get</i> | Mengambil sumber daya yang diidentifikasi oleh URI. |
| <i>Post</i> | Mengirim sumber daya ke server. Update sumber daya di lokasi yang diidentifikasi oleh URI. |
| <i>Put</i> | Mengirimkan sumber daya ke server, untuk disimpan di lokasi yang diidentifikasi oleh URI. |
| <i>Delete</i> | Menghapus sumber daya diidentifikasi oleh URI. |
| <i>Head</i> | Mengambil metadata dari sumber daya diidentifikasi oleh URI |

Sumber: Christian, Rindengan & Manembu (2016)

J. Profil Perusahaan

1. Sejarah Perusahaan



Gambar 2 Kantor BMKG Cilacap

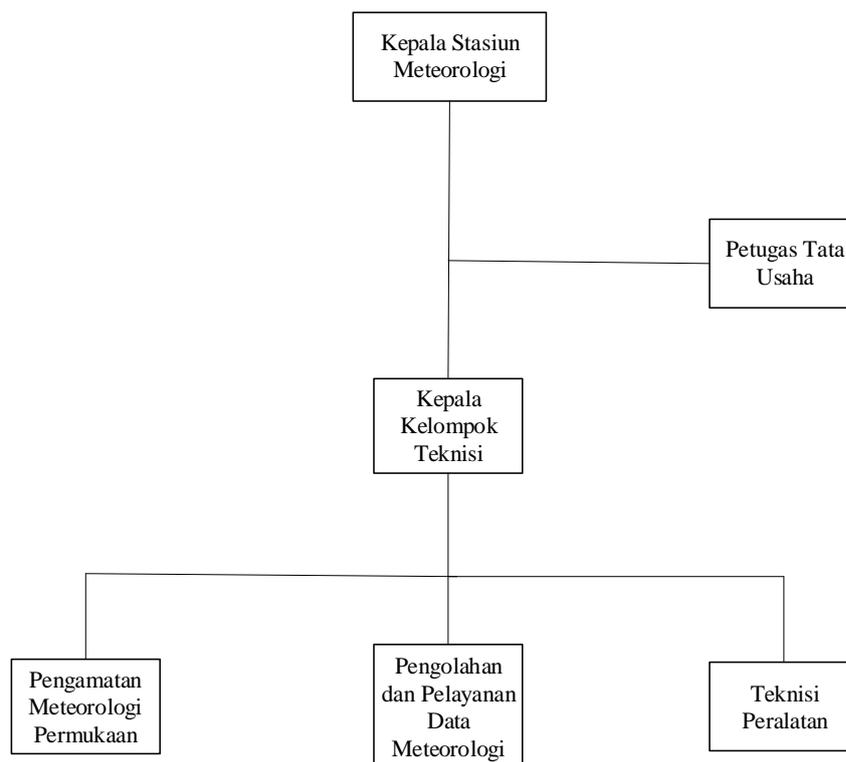
Badan *Meteorologi, Klimatologi, dan Geofisika* (BMKG), sebelumnya bernama Badan *Meteorologi, dan Geofisika* (BMG) adalah lembaga pemerintah non departemen Indonesia yang mempunyai tugas melaksanakan tugas pemerintahan di bidang *meteorologi, klimatologi, dan geofisika*. Stasiun *Meteorologi* Cilacap adalah salah satu Lembaga Pemerintah Non Departemen Indonesia yang berada di Jl. Gatot Subroto No.20, Tambaksari, Sidanegara, Cilacap Tengah, Kabupaten Cilacap, Jawa Tengah 53223 atau berada di Balai Besar Wilayah II. BMKG Cilacap disebut juga dengan Kantor Angin.

Stasiun *Meteorologi* Cilacap merupakan salah satu Unit Pelaksana Teknis (UPT) BMKG bertugas melaksanakan pengamatan, pengumpulan, dan pengiriman data unsur-unsur *Meteorologi* yang secara *Geografis* terletak pada ketinggian 6 meter diatas permukaan laut (dpl) dengan koordinat UTM *Easting*(X) : 281256.216875875, dan *Northing* (Y) :9144672.768245727.

Stasiun *Meteorologi* Cilacap pada awalnya dikelola oleh pemerintahan jepang kemudian setelah Indonesia merdeka, dikelola oleh TNI-AU Jogjakarta tahun 1945-1953. Pada 1 Maret 1953 hak kelola

diserahkan ke lembaga *Meteorologi*, dan *Geofisika* yang sekarang menjadi Badan *Meteorologi, Klimatologi*, dan *Geofisika*. Sejak dikelola oleh TNI-AU Stasiun *Meteorologi* Cilacap mengalami beberapa pergantian kepala Stasiun *Meteorologi* Cilacap yang sekarang dipimpin oleh kepala bagian Bapak Taruna Mona Rachman, SE.

2. Struktur Organisasi



Sumber: BMKG Cilacap

Gambar 3 Struktur organisasi

Adapun tugas dari karyawan Stasiun *Meteorologi* Cilacap adalah sebagai berikut:

- a. Kepala stasiun meteorologi
- b. Kepala kelompok teknisi
- c. Petugas tata usaha
- d. Petugas pengolahan data
- e. Prakirawan
- f. Pengamat

- g. Teknisi
- h. Tugas belajar
- i. Satpam

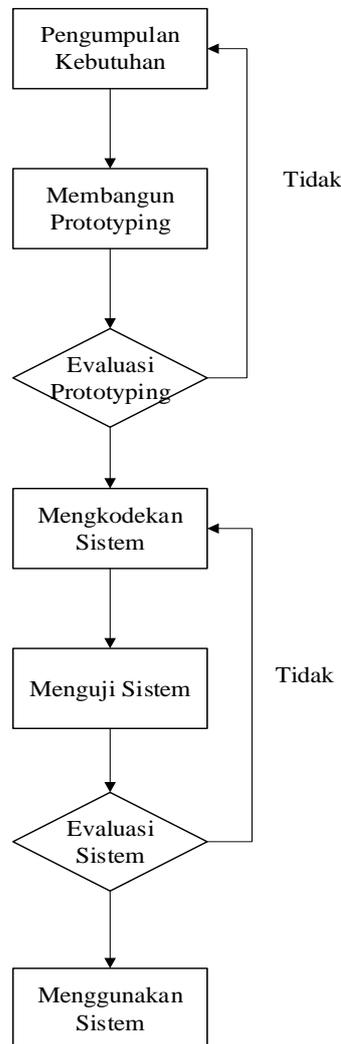
K. Metode *Prototype*

Prototype merupakan sistem dari potensial yang disediakan untuk pengembang dan pengguna yang bisa memberikan suatu gambaran tentang bagaimana sistem tersebut akan berfungsi yang sudah disusun dalam bentuk lengkap (Trisanto & Heriyanto, 2018). *Prototyping* merupakan sebuah proses dalam memproduksi suatu *prototype*. Tujuan dari *prototyping* tersebut adalah menghasilkan suatu *prototype* secepat mungkin. Dalam proses ini dapat diulang beberapa kali sehingga bisa menghasilkan *prototype* yang dianggap sempurna.

Metode model *prototyping* adalah metode untuk pengembangan suatu sistem yang bisa memberikan ide atau gambaran bagi seseorang untuk menganalisis sistem yang dapat menyajikan gambaran secara lengkap mengenai sistem. Adapun alasan digunakannya dalam model ini adalah sebagai berikut:

1. Dalam perancangan sistem ini usulan *raw material acceptance* hanya pada tahap perancangan.
2. Jika dilihat dari kebutuhan penggunanya, menggunakan model ini tentu lebih mudah untuk digunakan dibandingkan dengan model yang lainnya, dikarenakan dengan model *prototype* ini akan lebih mudah untuk melakukan perbaikan sistem jika terjadi perubahan kebutuhan penggunanya.
3. Dengan menggunakan metode ini bisa dilihat model sistem tersebut apakah baik dari sisi tampilan maupun dari prosedur yang akan dibangun.

Adapun urutan dari metode *prototype* ini dapat dilihat pada gambar 4 sebagai berikut:



Sumber: McLeod & Schell (2007)

Gambar 4 Tahapan dari metode prototyping

L. Parsing

Dalam proses parsing tidak hanya bisa dilakukan oleh NLP namun juga pada bidang yang lainnya contoh pada pembuatan *compiler* (Suciadi, 2001). Adapun metode-metode parsing yang dibahas berikut khusus digunakan dalam NLP. Arti dari istilah *constituent* merupakan unsur pembentuk kalimat yang bisa berdiri sendiri, misalnya *noun phrase*, *verb phrase* dan lain-lain. Sedangkan istilah dari *parser* merupakan program yang melakukan dalam proses parsing.

1. *Top-down Parsing*

Dalam *top-down parser* bekerja dengan menggunakan cara menguraikan sebuah kalimat yang dimulai dari *constituent* terbesar sampai menjadi *constituent* terkecil dilakukan secara terus menerus hingga semua komponen yang dihasilkan merupakan *constituent* terkecil pada kalimat merupakan kata. Misalnya dengan menggunakan *grammar* diatas, bisa dilakukan pada proses *top-down parsing* untuk kalimat “*the dog chased the cat*”. *Top-down parser* menelusuri tiap-tiap node pada *parse tree* secara *pre-order*. Metode *parsing* yang bekerja dengan *top-down* sebagai berikut:

- a. *Top-down parser* biasa
- b. *Recursive-descent parser*
- c. *Transition-network parser*
- d. *Chart parser*

Top-down parser bisa diimplementasikan dalam berbagai bahasa pemrograman, tetapi akan lebih baik digunakan sebagai *declarative language* misal Prolog atau LISP. Hal tersebut disebabkan karena pada dasarnya proses *parsing* merupakan proses *searching* yang dapat dilakukan secara *backtracking* dan rekursif. Selanjutnya *parser* yang ditulis pada Prolog akan menjadi jauh lebih sederhana daripada *parser* yang dibuat dengan bahasa prosedural contoh Pascal, bahasa C dan lain-lain.

2. *Bottom-Up Parsing*

Cara kerja dari *bottom-up parser* dengan mengambil satu demi satu kalimat yang akan diberikan kemudian dirangkai menjadi *constituent* yang lebih besar. Dan hal ini dilakukan secara terus-menerus *constituent* yang akan terbentuk adalah *sentence* atau kalimat. Dari cara kerja metode *bottom-up* merupakan cara yang terbalik dari *top-down*. *Parser* ini tidak membedakan antara *word* (lexicon) dengan *rule* (grammar) sehingga dapat bekerja dengan sangat sederhana tetapi sangat bodoh dikarenakan akan terus mengulang kesalahan yang sama. Kesederhaan tersebut

terletak pada predikat untuk *parsing* yaitu *parse* yang hanya memiliki sebuah argumen. Kalimat yang ada dalam argumen akan *diparse* dalam bentuk list pada simbol.

3. Gabungan *Top-Down* dan *Bottom-Up Parsing*

Baik dari *top-down parsing* maupun *bottom-up parsing* mempunyai kekurangan maupun kelebihan masing-masing. Metode *parsing* yang terbaik merupakan gabungan dari *top-down* dan *top-bottom parsing*. Adapun metode yang menggabungkan keduanya ialah *left-corner parsing* dan *earley's parsing*. Kemudian cara kerja dari *left-corner parsing* adalah dengan menerima sebuah kata dan menentukan jenis *constituent* apa yang dimulai dengan menggunakan jenis kata tersebut, selanjutnya akan melakukan proses *parsing* terhadap sisa *constituent* secara *top-down*.

M. Penelitian Terkait

Penelitian terkait dapat menjadi acuan bagi penulis dalam melakukan penelitian untuk memperkaya teori yang akan digunakan untuk mengkaji penelitian yang dilakukan. Adapun penelitian terkait ini berupa beberapa jurnal yang terkait dengan penelitian yang dilakukan penulis. Penelitian terkait dapat dilihat pada tabel 6.

Tabel 6 Penelitian terkait

| No | Judul | Penulis dan Tahun | Hasil |
|----|--|---------------------------|---|
| 1. | Studi analisis metode-metode parsing dan interpretasi semantik pada natural language processing. Jurnal informatika | James Suciadi 2001 | <i>Parsing</i> dibagi menjadi beberapa macam diantaranya <i>top-down parsing</i> , <i>bottom-up parsing</i> dan gabungan dari <i>top-down</i> dan <i>top-up parsing</i> . |

| No | Judul | Penulis dan Tahun | Hasil |
|----|---|---|--|
| 2. | <p>Prediksi cuaca kota surabaya menggunakan <i>autoregresive integrated moving average</i> (arima) <i>box jenkins</i> dan <i>kalman filter</i>.</p> <p>Jurnal matematika “mantik”</p> | <p>Nurissaidah Ulinuha dan Yuniar Farida 2018</p> | <p>Cuaca merupakan fenomena yang saat ini sedang terjadi di atmosfer bumi. Aktivitas manusia dapat dipengaruhi oleh suatu kondisi cuaca, seperti pertanian, bidang perkebunan, dan masih banyak lagi. Penentuan cuaca merupakan proses penerapan ilmu pengetahuan dan teknologi untuk menentukan keadaan atmosfer bumi.</p> |
| 3. | <p>Pembuatan website tanggap darurat dengan menggunakan teknik <i>framework codeigniter</i>.</p> <p>Jurnal SNASTI 2012, SC</p> | <p>Muhammad Hanafi dan Karsam 2012</p> | <p>Website merupakan komponen atau kumpulan komponen yang terdiri dari teks, gambar, suara animasi sehingga lebih merupakan media informasi yang menarik untuk dikunjungi. Website adalah halaman informasi yang disediakan melalui jalur internet sehingga bisa diakses diseluruh dunia selama terkoneksi dengan jaringan internet.</p> |

| No | Judul | Penulis dan Tahun | Hasil |
|----|---|--|---|
| 4. | <p>Sistem penjualan berbasis web (e-commerce) pada tata distro Kabupaten Pacitan.</p> <p>Jurnal bianglala informatika vol.3 no.2 september 2015</p> | <p>Rulia Puji Hastanti, Bambang Eka Purnama, dan Indah Uly Wardati</p> <p>2015</p> | <p>Sumber informasi adalah data. Data merupakan fakta kasar atau sebuah gambaran yang dikumpulkan dari suatu keadaan tertentu. Data adalah hal yang merujuk pada fakta-fakta baik berupa angka, teks, dokumen, gambar, badan, suara yang mewakili deskriptif verbal ataupun kode-kode tertentu.</p> |
| 5. | <p>Pemanfaatan <i>accuweather API</i> dalam pengembangan sistem informasi geografis curah hujan.</p> <p>Jurnal pengembangan teknologi informasi dan ilmu komputer vol.4, no.3, maret 2020</p> | <p>Made Tri Ganesha, Widhy Hayuhardhika, dan Bayu Rahayudi</p> <p>2020</p> | <p><i>Leaflet</i> merupakan pemetaan interaktif yang digunakan untuk pengembangan sebuah <i>library JavaScript</i>. Ukuran <i>file JavaScript leaflet</i> ini sangat ringan yaitu sekitar 38 KB. Untuk kebutuhan pengembangan, pemetaan <i>library</i> ini mempunyai fitur yang lengkap.</p> |
| 6. | <p>Rancang bangun aplikasi fasilitas umum berbasis <i>web services</i>.</p> | <p>Nora Eligia Christian, Yaulie Deo. Y. Rindengan, Pinrolinvic D. K.</p> | <p><i>Rest</i> adalah gaya arsitektur perangkat lunak yang bisa digunakan saat merancang sistem perangkat lunak.</p> |

| No | Judul | Penulis dan Tahun | Hasil |
|----|---|---|--|
| | <i>E-journal</i> teknik Informatika, volume 8, no.1 (2016), ISSN : 2301-8364 | Manembu 2016 | <i>Rest</i> merupakan desain yang ideal untuk perangkat lunak layanan berbasis web |
| 7. | Aplikasi pemesanan tiket bus berbasis android (studi kasus: PT. ALS terminal pasar x Tanjung Beringin) Jurnal manajemen bisnis, volume 32 no 1, Desember 2019 | Roberto Kaban, Windi Yunita, Fajrillah, Yanty Faradillah 2019 | MariaDB adalah implementasi dari sebuah sistem manajemen <i>database</i> relasional (RDBMS) dengan menggunakan lisensi dari GPL (<i>General Public License</i>) yang akan didistribusikan secara gratis. |
| 8. | Aplikasi informasi pelayanan fitness pada golden fitness center dumai dengan bahasa pemrograman php. Jurnal informatika, manajemen dan komputer, vol.10 no.1, mei 2018 | Nur Rubiati 2018 | PHP adalah suatu bahasa dalam bentuk <i>script</i> yang ditempatkan didalam <i>server</i> yang kemudian diproses. Selanjutnya hasil pemrosesan tersebut dikirim kepada website browser klien. Adanya bahasa pemrograman ini dirancang secara khusus untuk membuat web dinamis. |

| No | Judul | Penulis dan Tahun | Hasil |
|----|---|---|---|
| 9. | Perancangan sistem usulan <i>raw material acceptance</i> menggunakan metodologi pengembangan <i>system prototyping</i> pada PT Denso Indonesia. Jurnal teknologi dan manajemen february vol.16 no.1 tahun 2018 | Dedy Trisanto dan Lucky Heriyanto 2018 | <i>Prototype</i> merupakan sistem dari potensial yang disediakan untuk pengembang dan pengguna yang bisa memberikan suatu gambaran tentang bagaimana sistem tersebut akan berfungsi yang sudah disusun dalam bentuk lengkap. <i>Prototyping</i> merupakan sebuah proses dalam memproduksi suatu <i>prototype</i> . Tujuan dari <i>prototyping</i> tersebut adalah menghasilkan suatu <i>prototype</i> secepat mungkin. Dalam proses ini dapat diulang beberapa kali sehingga bisa menghasilkan <i>prototype</i> yang dianggap sempurna. |

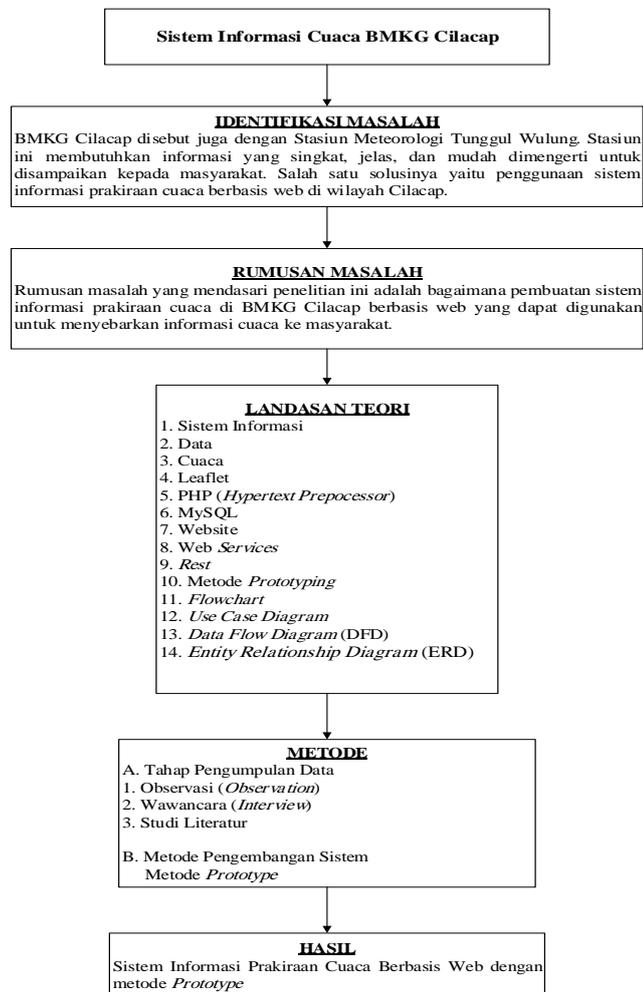
Sumber: Diolah Sendiri (2021)

Dalam penelitian ini penulis mengambil referensi untuk membuat prototipe sistem informasi cuaca di BMKG berbasis web yang dapat digunakan untuk memberikan informasi kepada masyarakat luas mengenai cuaca harian di Kabupaten Cilacap. Sistem ini dibangun dengan menggunakan bahasa pemrograman PHP dan MariaDB. Informasi ini meliputi prakiraan cuaca, suhu, angin, dan kelembaban. Adapun untuk membuat tampilan peta dengan menggunakan bantuan *leaflet* dan untuk membuat geojson Kabupaten Cilacap menggunakan aplikasi QGIS.

Kemudian untuk mengkodekan sistem menggunakan aplikasi visual studio code. Proses yang digunakan untuk pengambilan data menggunakan teknik parsing dengan cara membuat contoh *dummy* web untuk BMKG pusat sebagai web *server* dan selanjutnya data tersebut akan dibaca oleh web BMKG Cilacap sebagai web *client*. Adapun data yang diparsing berupa data json. Sedangkan metode yang digunakan dalam penelitian ini menggunakan metode *prototype* yang meliputi tahap pengumpulan kebutuhan, membangun *prototype*, evaluasi *prototype*, mengkodekan sistem, menguji sistem, evaluasi sistem, dan menggunakan sistem.

N. Kerangka Pemikiran

Berikut ini kerangka pemikiran yang akan dibahas sebagai berikut:



Sumber: Diolah sendiri (2021)

Gambar 5 Kerangka pemikiran