# BAB II LANDASAN TEORI

## **Fuzzy**

### Definisi *Fuzzy*

*Fuzzy* secara bahasa diartikan sebagai kabur atau samar-samar. Suatu nilai dapat bernilai besar atau salah secara bersamaan. Dalam *fuzzy* dikenal derajat keanggotaan yang memiliki rentang nilai 0 (nol) hingga 1 (satu). Berbeda dengan himpunan tegas yang memiliki nilai 1 atau 0 (ya atau tidak). Logika *Fuzzy* merupakan seuatu logika yang memiliki nilai kekaburan atau kesamaran (*fuzzyness*) antara benar atau salah. Dalam teori logika *fuzzy* suatu nilai bias bernilai benar atau salah secara bersama. Namun berapa besar keberadaan dan kesalahan suatu tergantung pada bobot keanggotaan yang dimilikinya. Logika *fuzzy* memiliki derajat keanggotaan dalam rentang 0 hingga 1. Berbeda dengan logika digital yang hanya memiliki dua nilai 1 atau 0. Logika *fuzzy* digunakan untuk menterjemahkan suatu besaran yang diekspresikan menggunakan bahasa (*linguistic*), misalkan besaran kecepatan laju kendaraan yang diekspresikan dengan pelan, agak cepat, cepat, dan sangat cepat. Logika *fuzzy* merupakan modifikasi dari teori himpunan dimana setiap anggotanya memiliki derajat keanggotaan yang bernilai kontinu antara 0 sampai 1. Sejak ditemukan pertama kali oleh Lotfi A. Zadeh pada tahun 1965, logika fuzzy telah digunakan pada lingkup domain permasalahan yang cukup luas, seperti kendali proses, klasifikasi dan pencocokan pola, manajemen dan pengambil keputusan, riset operasi, ekonomi dan lain lain. Sejak tahun 1985, terjadi perkembangan yang sangat pesat pada logika *fuzzy*, terutama dalam hubungan yang bersifat *non-linear*, *ill-defined*, *time-varying* dan situasi-situasi yang sangat kompleks. Logika *fuzzy* menunjukan sejauh mana suatu nilai itu benar dan sejauh mana suatu nilai itu salah. Tidak seperti logika klasik (*crisp*)/ tegas, suatu nilai hanya mempunyai 2 kemungkinan yaitu merupakan suatu anggota himpunan atau tidak. Derajat keanggotaan 0 (nol) artinya nilai bukan merupakan anggota himpunan dan 1 (satu) berarti nilai tersebut adalah anggota himpunan. Logika *fuzzy* adalah suatu cara yang tepat untuk memetakan suatu ruang *input* ke dalam suatu ruang *output*, mempunyai nilai kontinyu. *Fuzzy* dinyatakan dalam derajat dari suatu keanggotaan dan derajat dari kebenaran. Oleh sebab itu sesuatu dapat dikatakan sebagian benar dan sebagian salah pada waktu yang sama. Logika *Fuzzy* memungkinkan nilai keanggotaan antara 0 dan 1, tingkat keabuan dan juga hitam dan putih, dan dalam bentuk linguistik, konsep tidak pasti seperti "sedikit", "lumayan" dan "sangat". Kelebihan dari teori logika *fuzzy* adalah kemampuan dalam proses penalaran secara bahasa (*linguistic reasoning*). Sehingga dalam perancangannya tidak memerlukan persamaan matematik dari objek yang akan dikendalikan. Logika *fuzzy* merupakan teknik untuk mengolah istilah *linguistik*. Teknik ini memperluas ide logika lebih dari sekedar benar atau salah untuk memungkinkan kebenaran parsial (bahkan kontinu). Pengetahuan yang tidak pasti dan pertimbangan yang tidak persisi adalah aspek penting keahlian dalam menerapkan akal sehat dalam situasi pengambilan keputusan. Dalam logika *fuzzy* nilai benar atau salah digantikan dengan derajat himpunan keanggotaan (Husni, Arifin, & Yuliyanti, 2013).

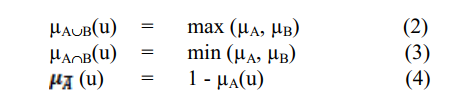
### Himpunan *Fuzzy*

Misalkan U adalah himpunan klasik yang terdiri dari objek-objek sebagai anggota himpunan, yang disebut sebagai semesta pembicaraan dan u adalah anggota himpunan U. Himpunan fuzzy F dalam semesta pembicaraan U ditandai dengan fungsi keanggotaan

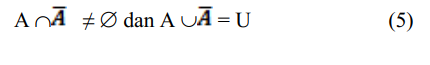


Dimana µf (U) untuk setiap u € U menunjukkan derajat keanggotaan u dalam himpunan fuzzy F (Raju and Majumdar, 1988).

Penggunaan operasi teori himpunan (seperti *union, intersection, dan complementation* dan sebagainya) juga berlaku untuk himpunan fuzzy. Ambil A dan B adalah dua himpunan bagian dari semesta himpunan fuzzy dengan fungsi keanggotaan µa dan µb. Fungsi keanggotaan dari A∪B, A∩B dan **A** (kompelemen A) diberikan sebagai berikut :



Berdasarkan definisi tersebut, sebagian besar sifat yang berlaku pada operasi himpunan klasik, seperti hokum de Morgan, telah terbukti berlaku untuk himpunan fuzzy. Satu-satunya hukum dari teori himpunan biasa yang tidak berlaku adalah hokum kompelemen yaitu :



Dimana Q adalah himpunan kosong yaitu µQ (U) = 0 untuk setiap u € U.

Himpunan fuzzy memungkinkan suatu item x dapat masuk menjadi anggota pada lebih dari 1 himpunan (Kusumadewi, 2010). Himpunan fuzzy memiliki dua atribut yaitu :

1. linguistik, yaitu penamaan suatu grup yang mewakili suatu keadaan atau kondisi tertentu yang menggunakan Bahasa alami.
2. Numeris, yaitu suatu nilai angka yang menunjukkan nilai suatu variabel.

Komponen-komponen dalam system fuzzy:

1. Variabel fuzzy : merupakan variabel yang hendak dibahas dalam suatu system fuzzy.
2. Himpunan fuzzy : suatu grup yang mewakili suatu kondisi atau keadaan tertentu dalam suatu variabel fuzzy
3. Semesta pembicaraan : keseluruhan nilai yang diperbolehkan untuk dioperasikan dalam suatu variabel fuzzy dan merupakan suatu himpunan bilangan real yang senantiasa naik (bertambah) secara monoton dari kiri ke kanan. Nilai semesta pembicaraan dapat berupa bilangan positif maupun negative, dan ada kalanya tidak dibatasi batas atasnya.
4. Domain : keseluruhan nilai yang di izinkan dalam semesta pembicaraan dan boleh dioperasikan dalam suatu himpunan fuzzy.

Pada dasarnya tidak ada ketentuan khusus dalam menentukan suatu fungsi keanggotaan yang akan digunakan dalam suatu proses *fuzzyfikasi,* karena masing-masing system bisa memiliki tingkat kesesuaian yang berbeda-beda.

Ada dua keadaan himpunan fuzzy yang linear. Pertama, kenaikan himpunan dimulai pada nilai domain yang memiliki derajat keanggotaan nol [0] bergerak ke kanan menuju ke nilai domain yang memiliki derajat keanggotaan lebih tinggi. Representasi kurva segitiga yang pada dasarnya merupakan gabungan antara 2 garis (linear). Representasi kurva trapezium yang pada dasarnya seperti bentuk segitiga, hanya saja ada beberapa titik yang memiliki nilai keanggotaan 1. Representasi kurva bentuk bahu, daerah yang terletak di tengah-tengah suatu variabel yang direpresentasikan dalam bentuk segitiga, pada sisi kanan dan kirinya akan naik dan turun, Tetapi terkadang salah satu sisi dari variabel tersebut tidak mengalami perubahan. Himpunan *fuzzy* ‘bahu’, bukan segitiga, digunakan untuk mengakhiri variabel suatu daerah *fuzzy*. Bahu kiri bergerak dari benar ke salah, demikian juga bahu kanan bergerak dari salah ke benar.

Koordinat keanggotaan himpunan logika *fuzzy* berisi urutan pasangan berurutan yang berisi nilai domain dan kebenaran nilai keanggotaannya dalam bentuk: Skalar(i) / Derajat(i). ‘Skalar’ adalah suatu nilai yang digambar dari domain himpunan *fuzzy*, sedangkan ‘Derajat’ skalar merupakan derajat keanggotaan himpunan *fuzzy*nya. Seperti halnya himpunan konvensional, ada beberapa operasi yang didefinisikan secara khusus untuk mengkombinasi dan memodifikasi himpunan *fuzzy*. Nilai keanggotaan sebagai hasil dari operasi 2 himpunan sering dikenal dengan nama *fire strength* atau α–predikat (Efendi, Ernawati, & Hidayati, 2014)

### Operator Dasar Zadeh Untuk Operasi Himpunan *Fuzzy*

Ada 3 operator dasar yang diciptakan oleh Zadeh, yaitu: Operator AND, Operator ini berhubungan dengan operasi interseksi pada himpunan. α– predikat sebagai hasil operasi dengan operator AND diperoleh dengan mengambil nilai keanggotaan terkecil antar elemen pada himpunan-himpunan yang bersangkutan. A∩B = min(­A[x], ­B[y]).

Operator OR, Operator ini berhubungan dengan operasi union pada himpunan. α– predikat sebagai hasil operasi dengan operator OR diperoleh dengan mengambil nilai keanggotaan terbesar antar elemen pada himpunan-himpunan yang bersangkutan. A­B = max(­A[x], ­B[y]).

Operator NOT, Operator ini berhubungan dengan operasi komplemen pada himpunan. α–predikat sebagai hasil operasi dengan operator NOT diperoleh denganmengurangkan nilai keanggotaan elemen pada himpunan yang bersangkutan dari 1. A’ = 1-­A[x] (Efendi, Ernawati, & Hidayati, 2014).

### Himpunan Keanggotaan

Pada himpunan tegas (*crisp*), nilai keanggotaan suatu item x dalam suatu himpunan A, yang sering ditulis dengan µA[x], memiliki 2 kemungkinan (Kusumadewi S, Purnomo H, 2010) yaitu:

* Satu (1), yang berarti bahwa suatu item menjadi anggota dalam suatu himpunan,atau
* Nol (0), yang berarti bahwa suatu item tidak menjadi anggota dalam suatu himpunan.

Terkadang kemiripan antara keanggotaan *fuzzy* dengan probabilitas menimbulkan kerancuan. Keduanya memiliki nilai pada interval [0.1], namun interpretasi nilainya sangat berbeda antara kedua kasus tersebut. Keanggotaan *fuzzy* memberikan suatu ukuran terhadap pendapat atau keputusan, sedangkan probabilitas mengindikasikan proporsi terhadap keseringan suatu hasil bernilai benar dalam jangka panjang. Misalnya, jika nilai keanggotaan bernilai suatu himpunan *fuzzy* USIA adalah 0.9; maka tidak perlu dipermasalahkan berapa seringnya nilai itu diulang secara individual untuk mengharapkan suatu hasil yang hampir pasti muda. Di lain pihak, nilai probabilitas 0.9 usia berarti 10% dari himpunan tersebut diharapkan tidak muda.

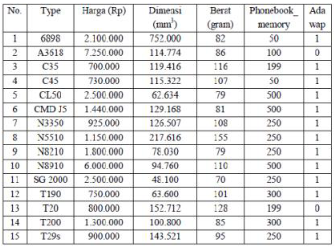
Himpunan *fuzzy* memiliki 2 atribut, yaitu:

1. Linguistik, yaitu penamaan suatu grup yang mewakili suatu keadaan atau kondisi tertentu dengan menggunakan bahasa alami, seperti: MUDA, PAROBAYA,TUA
2. Numeris, yaitu suatu nilai (angka) yang menunjukkan ukuran dari suatu variable seperti: 40, 25, 50,dsb.

### Fungsi Keanggotaan

Sebagian besar basis data standar diklasifikasikanberdasarka bagaimana data tersebut dipandangoleh user. Basis datafuzzymodel Tahani masihtetap menggunakan relasi standar, hanya sajamodel ini menggunakan teori himpunanfuzzyuntuk mendapatkan informasi padaquery-nya.Jadi, data awal yang diproses adalah data yangmemiliki nilaicrisp(pasti/ jelas keberadaannya),dan ketika hendakmelakukan proses pencariandata yang bersifat samar maka proses tersebut yangdinamakan prosesfuzzy querymelaluifuzzy databasemodel Tahani.Ide dari sistemfuzzy databasemodel Tahaniadalah mendefinisikan konsep dari relasifuzzydalam sebuahdatabasesistem denganmenggunakan derajat keanggotaan. Contoh kasusyang dapat diselesaikan denganfuzzy databasemodel Tahani adalah aplikasifuzzy database systemmodelTahani untuk memberikan rekomendasi pilihan HP bagi konsumen berdasarkan kriteria-kriteria yang diinginkan oleh konsumen. Pada kasus tersebut, outlet HP memiliki data HP yang tersimpan pada tabel. Atribut yang ada pada setiap tipe HP berupa: harga, dimensi,nberat, dan phone book\_memory; serta ada tidaknya fasilitas WAP seperti pada Tabel 1 DT\_HP pada Outlet Penjualan HP.

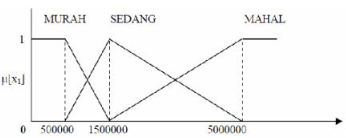
Tabel 1 Contoh Tabel DT\_HP pada Outlet Penjualan HP



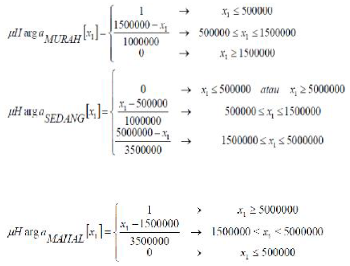
Jika dengan menggunakan basisdata standar, kita dapat mencari data-data HP dengan spesifikasi ertentu dengan menggunakan query. Misal kitaingin mendapatkan informasi tentang tipe HP yang harga nya kurang dari Rp. 1.000.000,- tetapi beratnya kurang dari atau sama dengan 100 gram, maka kita bisa menuliskan suatu query: *SELECT TYPE FROM DT\_HP WHERE (Harga <1000000) and (Berat≤ 100)*. Sehingga muncul tipe HP T29s, sedangkan tipe HP T190 yang harganya kurang dari Rp.1.000.000,- tidak terpilih padahal beda beratnya selisih 1 gram dari krtieria yang ditentukan. Disini berarti pada basisdata standar, adanya perubahan kecil saja mengakibatkan perbedaan kategori yang cukup signifikan antara HP yang terpilih dan tidak terpilih. Agar konsumen lebih mudah dalam memberikan kriteria yang diinginkan, maka penyampaian informasi tentang kriteria suatu atribut direpresentasikan menggunakan bahasa sehari-hari/variabel lingusitik. Pada kasus penjualan HP, seorang konsumen ingin mencari informasi mengenai tipe HP yang sesuai dengan kriteria yang diinginkannya, adapun datanya menggunakan bahasa sehari-hari (bersifat samar). Pada basisdata fuzzy Tahani, awalnya dibentuk suatu himpunan fuzzy dengan fungsi keanggotaannya. Untuk menilai tipe HP yang diinginkan maka dibentuklah beberapa kategori yang perlu dipertimbangkan dengan himpunan yang dimiliki pada masing masing kriteria. Kriteria-kriteria tersebut adalah sebagai berikut:

1. Harga HP (MURAH, SEDANG, dan MAHAL)
2. Dimensi (KECIL, SEDANG, dan BESAR)
3. berat (RINGAN, SEDANG, dan BERAT)
4. phonebook memory (SEDIKIT, SEDANG, dan BANYAK)

Misalkan kita mengkategorikan harga HP ke dalam himpunan: MURAH, SEDANG, dan MAHAL. Himpunan MURAH dan MAHAL menggunakan pendekatan fungsi keanggotaan berbentuk bahu, sedangkan himpunan SEDANG menggunakan pendekatan fungsi keanggotaan berbentuk segitiga. Fungsi keanggotaan harag Hp terlihat pada gambar 1.

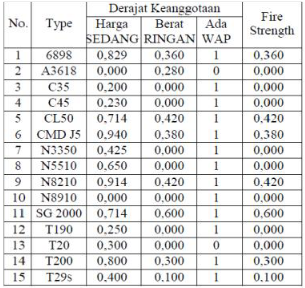


Gambar 1 Fungsi Keanggotaan pada Variabel Harga

Fungsi keanggotaan variabel harga dirumuskan sebagai berikut:

Misalnya ingin diketahui tipe HP apa saja yang harganya SEDANG, beratnya RINGAN, dan ada WAP. Maka, Structured Query Laguage (SQL) yang dibentuk adalah: SELECT type FROM dt\_hp WHERE (harga = ”SEDANG”) AND (berat = ”RINGAN”) AND (WAP = TRUE) Karena variabel WAP bersifat non fuzzy, maka hanya ada 2 kemungkinan derajat keanggotaan, yaitu 1 apabila ada WAP, dan 0 apabila tidak ada WAP. Tabel 2 memperlihatkan fire strength sebagai hasil operasi dari (Harga SEDANG) AND (Berat RINGAN) AND (Ada WAP) sebagai: μHargaSEDANGᴖBeratRINGANᴖAdaWAP=min(μHargaSEDANG[x1],μBeratRINGAN[x3],μAdaWAP[x5]).

Tabel 2. Fire strength untuk query Harga SEDANG, Berat RINGAN, Ada WAP



Tabel 2 memperlihatkan hasil pencarian terhadap Query 2. Ada 7 HP yang terekomendasi, dengan rekomendasi tertinggi adalah HP tipe SG 2000, dan rekomendasi terendah adalah HP tipe T29s. Sedangkan 8 tipe lainnya tidak direkomendasikan (Efendi, Ernawati, & Hidayati, 2014).

### *Fuzzy* Tahani

Basis data klasik hanya menangani data-data yang bersifat pasif dan tegas. Sedangkan pada kenyataannya manusia seringkali berkomunkasi dengan Bahasa yang tidak jelas batasannya. Untuk menangani hal tersebut maka dibangunlah sebuah basis data dengan pendekatan logika *fuzzy*. Basis data yang menggunakan pendekatan *fuzzy* tidak hanya menyimpan dan memanipulasi fakta-fakta yang pasti tetapi juga pendapat-pendapat subjectif, keputusan dan nilai-nilai yang dapat djabarkan dalam istilah lnguistik.

Pada umumnya, ada du acara untuk memasukkan unsur kekaburan (*fuzziness*) kedalam sebuah basis data yaitu:

1. *Fuzzy Database*

*Fuzzy Database* adalah basis data yang mempunyai kemampuan untuk menyimpan dan memanipulasi data-data yang mengandung ketidakpastian secara langsung. Artinya, pengguna memasukkan informasi-informasi yang mengandung unsur kekaburan kedalam pangkalan data. Basis data jenis ini juga didukung oleh *query* yang bersifat *fuzzy* untuk memperoleh informasi.

1. *Fuzzy Query Database*

*Fuzzy Query Database* adalah membuat *fuzzy query* terhadap basis data klasik. Pengguna membuat suatu aplikasi yang bisa menangani suatu *query* dimana dalam *query* tersebut terdapat variabel-variabel yang bernilai *fuzzy* atau dengan kata lain *query* tersebut memilik variabel-variabel linguistik. Sedangkan data pada basis data yang akan diakses merupakan data yang bersifat pasti. Pangkalan data yang diusulkan oleh Tahani adalah bentuk dari *Fuzzy Query* *Database*. Basis data Tahani masih tetap menggunakan relasi standar, hanya model ini menggunakan teori himpunan *fuzzy* untuk mendapatkan informasi pada *query-*nya.

Tahani mengembangkan sebuah kerangka kerja pada level konseptual tingkat tinggi untuk memproses *query fuzzy* pada lingkungan basis data konvensional atau *non-fuzzy*. Tahani merumuskan sebuah arsitektur dan pendekatan formal untuk menangani basis data dengan *fuzzy query* yang sederhana. Bahasa *query* yang digunakan berdasarkan SQL.

Program aplikasi basis data *fuzzy* adalah program untuk melakukan pencarian data dengan metode pencarian linguistik. Program ini merupakan penerapan dari teori tentang basis data *fuzzy*. Input awal dari program ini adalah kriteria user, selanjutnya penentuan variabel *fuzzy* dan penentuan himpunan *fuzzy* yang akan digunakan dalam pencarian. Setelah data dimasukkan dan pengguna telah memilih kriteria pencarian pada table pencarian linguistik, langkah selanjutnya yaitu menghitung derajat keanggotaan suatu data disetiap himpunan pada suatu variabel berdasarkan fungsi keanggotaan yang telah dipilih sebelumnya. Selanjutnya proses *fire strength* berdasarkan kriteria yang dipilih. Data yang memiliki nilai *fire strength* tertinggi menunjukkan bahwa data tersebut yang paling mendekati kriteria pencarian. Sebaliknya, data yang memiliki nilai fire strength yang terkecil menunjukkan bahwa data tersebut semakin menjauhi kriteria pencarian. Hasil yang ditampilkan pada tab pencarian linguistic merupakan data yang mendekati kriteria pencarian dengan urutan nilai fire strength terbesar sampai nilai fire strength terkecil yang menunjukan urutan data yang paling mendekati kriteria pencarian sampai yang paling jauh dari kriteria pencarian. Berikut ini beberapa tahapan atau langkah-langkah dalam metode *Fuzzy Logic Database* Tahani yaitu:

1. Menentukan variabel fuzzy
2. Menentukan himpunan fuzzy
3. Menentukan domain

Disini diasumsikan sebuah konvensional (*non-fuzzy*) DBMS,dan mecoba mengembangkan serta mengimplementasikan sebuah system logika *fuzzy query.* Dalam system logika fuzzy query ini berupaya mencapai sebuah kelenturan (*flexibility*) dari sebuah *database* yang mana mempunyai aspek-aspek variasi seperti koreksi kesalahan secara otomatis, pencarian flexible, kemampuan menghindari respon kosong, kemungkinan dari ketepatan, istilah ucapan atau sebutan dalam sebuah *query*. Pendekatan pertama dalam *fuzzy query* ke database adalah Tahani. Ide dari system basis data *fuzzy* model Tahani mendefinisikan konsep dari relasi *fuzzy* dalam sebuah DBMS dengan derajat keanggotaan.

## **Rumah**

### Definisi Rumah

Rumah adalah suatu bangunan yang berfungsi untuk tempat tinggal penghuninya, rumah biasanya dibuat senyaman mungkin dan seindah mungkin supaya si penghuni rumah tersebut merasa nyaman saat menempatinya. Rumah terdiri dari berbagai komponen bahan, diantaranya yaitu, rumah kayu, rumah dari anyaman bambu, rumah dari susunan material batu bata, bahkan di jaman sekarang sudah banyak rumah rumah yang menggunakan beton precast . Perbedaan penggunaan komponen bahan tersebut tergantung dari beberapa faktor diantaranya adalah kondisi wilayah/ iklim, suhu, dll. Tapi hal yang paling mendasar adalah kondisi ekonomi . untuk ekonomi rendah biasanya menggunakan rumah dari anyaman kayu sebagai alternatifnya. Rumah ada beberapa tingkatan berdasarkan ekonomi pemilik rumah tersebut, yaitu rumah sangat sederhana atau sering disebut dengan RSS, rumah sederhana atau RS, rumah modern, dan juga rumah mewah . sebenarnya secara fungsi rumah memiliki fungsi yang sama. akan tetapi banyak orang yang menilai rumah sebagai salah satu alat ukur ekonomi seseorang.

Rumah terdiri dari beberapa komponen yang disusun dan mempunyai fungsi sendiri – sendiri . adapun susunan – susunan komponen tersebut sebagai berikut :

* Pondasi, pondasi adalah komponen struktur paling dasar atau paling bawah yang berfungsi sebagai tumpuan dari bangunan, dan fungsi pondasi yang utama adalah menyalurkan beban dari komponen – komponen bangunan di atasnya.
* Lantai, sebagai dasar pada suatu rumah . berposisi diatas tanah.
* Langit – langit atau plafon, plafon berfungsi sebagai pemisah antara ruangan dalam dengan atap.
* Balok sloof, balok sloof berada pada posisi atas pondasi
* Kolom, kolom berfungsi sebagai struktur pada sebuah rumah
* Dinding, dinding berfungsi sebagai partisi atau pembatas antara ruang satu dengan ruang yang lain .
* Pintu dan jendela, berfungsi sebagai akses keluar masuk orang dan juga akses sirkulasi udara dan juga cahaya .
* Ring balk, ring balk berfungsi sebagai bpengikat dinding dan juga kolom
* Atap, atap adalah penutup dati sebuah rumah, dan posisinya berada paling atas pada sebuah bangunan . atap berfungsi untuk melindungi penghuni rumah dari sinar matahari dan juga dari hujan .

Selain susunan pokok tersebut, ada juga komponen komponen pelengkap bangunan yang juga harus di perhatikan . seperti listrik sebagai penerangan, sumur atau PDAM sebagai sumber air, dan juga saluran air kotor serta tampungan air kotor seperti septictank, sumur resapan air kotor, dan juga sumur resapan air hujan . selain itu ada yang namanya pekerjaan finishing rumah, seperti plesteran, acian, pengecatan, dll (Ahadi, 2016).

## **Perangkat Analisa Sistem**

### *Flowchart*

*Flowchart* adalah bagan-bagan yang mempunyai arus menggambarkan langkah-langkah suatu masalah. *Flowchart* merupakan cara penyajian dari algoritma. (Ladjamudin, 2005).

Menurut (Ladjamudin, 2005) ada dua macam *flowchart* yang menggambarkan proses dengan komputer, yaitu:

1. *System flowchart*

Bagan yang memperlihatkan urutan proses dalam sistem dengan menunjukkan alat media input, output, serta jenis media penyimpanan dalam proses pengolahan data.

1. Program *flowchart*

Bagan yang memperlihatkan intruksi yang digambarkan dengan symbol tertenru untuk memecahkan masalah dalam suatu program.

Tabel 2.3 Simbol Flowchart

|  |  |
| --- | --- |
| Simbol | Keterangan |
|  | **Arus / flow**  Menyatakan jalannya arus suatu proses |
|  | **Communication Link**  Menyatakan bahwa adanya transisi suatu data/informasi dari suatu lokasi ke lokasi lainnya. |
|  | **Connector**  Menyatakan sambungan dari satu proses ke proses lainnya dalam halaman/lembar yang sama. |
|  | Offline Connector  Menyatakan sambungan dari satu proses ke proses lainnya dalam halaman/lembar yang berbeda. |
|  | **Manual**  Menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer |
|  | **Decision/Logika**  Untuk menunjukan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban ya atau tidak |
|  | **Predefined Proses**  Untuk menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal |

### *Data Flow Diagram (DFD)*

*Data Flow Diagram* merupakan model dari system untuk menggambarkan pembagian system ke modul yang lebih kecil. Salah satu keuntungan mengunakan DFD adalah memudahkan pemakai atau user yang kurang menguasasi computer untuk mengerti system yang akan dikerjakan. DFD terdiri dari tiga bagian yaitu (Ladjamudin, 2005).

1. Diagram Konteks

Diagram konteks adalah diagram yang terdiri dari suatu proses dan menggambarkan luang lingkup suatu system. Diagram konteks merupakan level tertinggi dari DFD yang menggambarkan seluruh input ke system ataupun output ke system, ia akan memberikan gambaran tentang keseluruhan system. System dibatasi oleh boundry (dapat digambarkan dengan garis putus). Dalam diagram konteks harus ada satu proses tidak boleh ada ada store dalam diagram konteks.

1. Diagram Nol/Zero (Overview Diagram)

Diagram nol adalah diagram yang menggambarkan proses dan data flow diagram. Diagram nol memberikan pandangan secara menyeluruh mengenai system yang ditangani, mengenai tentang fungsi-fungsi utama atau proses yang ada, aliran data, dan eksternal entity.

1. Diagram Rinci (Level Diagram)

Diagram rinci adalah diagram yang menguraikan proses apa yang ada di dalam diagram zero atau ada diagram level diatasnya.

Tabel elemen dasar data flow diagram versi yourdan, De Marco.

Tabel 2.4 Simbol Data Flow Diagram

|  |  |
| --- | --- |
| Simbol | Keterangan |
|  | **Entitas luar**  Menggambarkan suatu system yang ada di luar system, tetapi ia memberikan data ke dalam system atau mendapat data dari system. |
|  | **Arus Data**  Menggambarkan arus data yang berupa masukan untuk system atau hasil dari system. |
|  | **Proses**  Menggambarkan apa yang sedang dikerjakan oleh system. |
| Data Store | **Penyimpanan Data**  Menggambarkan tempat penyimpanan data yang ada dalam system |

### *Entity Relational Diagram (ERD)*

Entity Relational Diagram (ERD) adalah notasi yang digunakan untuk melakukan aktifitas pemodelan data. Attribute dari masing-masing objek data yang ditulis pada ERD dapat digambarkan dengan menggunakan deskripsi objek data (Pressman, 2002).

Model E-R terdiri dari beberapa komponen dasar yaitu sebagai berikut:

1. Entitas

Entitas adalah sesuatu atau objek yang di dunia nyata yang dapat dibedakan dari sesuatu atau objek yang lainnya. Sebagai contoh setiap mahasiswa dalam suatu universitas adalah suatu entitas. Dapat dikatakan bahwa entitas bersifat konseptual/abstrak atau nyta hadir di dunia nyata.

1. Attribute

Attribute adalah property deskriptif yang dimiliki oleh setiap anggota dari himpunan entitas. Sebagai contoh entitas mahasiswa attribute-attribut yang dimiliki adalah nim, nama, alamat dan lain-lain.

1. Hubungan antar relasi (Relationship)

Hubungan antar relasi adalah hubungan antar suatu himpunan entitas dengan himpunan entitas lainnya. Misalnya entitas mahasiswa memiliki hubungan tertentu dengan entitas matakuliah (mahasiswa mengambil matakuliah). Pada penggambaran model E-R, relasi adalah perekat yang menghubungkan suatu entitas dangan entitas lainnya.

1. Kardinalitas atau derajat relasi

Kardinalitas relasi menunjukkan jumlah maksimum entitas yang dapat berelasi dengan entitas pada himpunan entitas yang lainnya. Sebagai contoh entitas-entitas pada himpunan entitas mahasiswa dapat berelasi dengan satu entitas, banyak entitas atau tidak satupun entitas dari himpunan entitas matakuliah.

Kardinalitas relasi yang terjadi diantara dua himpunan entitas dapat berupa:

1. Satu ke satu (One To One)

Setiap entitas pada himpunan entitas A berhubungan dengan satu entitas pada himpunan entitas B, begitu pun sebaliknya.

1. Satu ke Banyak (One To Many)

Setiap entitas pada himpuna entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, tetapi tidak sebaliknya, dimana setiap entitas pada himpunan entitas B berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas A.

1. Banyak ke Satu (Many To One)

Setiap entitas pada himpunan entitas A dapat berhubungan dengan paling banyak entitas pada himpunan entitas B, tetapi tidak sebaliknya pada enntitas B.

1. Banyak ke Banyak (Many to Many)

Setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, dan demikian sebaliknya, dimana setiap entitas pada himpunan entitas dapat berhubungan dengan banyak entitas pada himpunan entitas A.

### Normalisasi

Normalisasi adalah proses pengelompokan data kedalam bentuk table atau relasi atau file untuk menyatakan entitas dan hubungan mereka sehingga terwujud satu bentuk database yang mudah untuk dimodifikasi.

Ada beberapa bentuk normalisasi antara lain yaitu (Ladjamudin, 2005):

1. Bentuk normal I (First Normal Form/1-NF)

Pada tahap ini dilakukan penghilangan beberapa group yang berulang agar menjadi satu harga tunggal yang berintraksi diantara setiap baris pada suatu table dan setiap attribute harus mempunyai nilai data yang atomic (bersifat atomic value).

Syarat-syarat untuk normal ke satu adalah:

1. Setiap data dibentuk dalam flat file, data dibentuk dalam satu record demi satu record nilai dari field berupa “*atomic value*”.
2. Tidak ada set attribute yang berulang atau bernilai ganda.
3. Telah dibentuknya primary key untuk table atau relasi tersebut.
4. Tiap attribute hanya memiliki satu pengertian
5. Bentuk normal II (Second Normal Form/2-NF)

Suatu relasi memenuhi 2-NF jika dan hanya jika:

1. Bentuk data telah memenuhi kriteria bentuk normal ke satu
2. Attribute bukan kunci haruslah memenuhi ketergantungan fungsi utama pada kunci sepenuhnya.
3. Bentuk normal III (Third Normal Form/3.NF)

Suatu relasi memenuhi bentuk 3-NF jika dan hanya jika:

1. Bentuk data telah memenuhi kriteria dari bentuk normal kedua
2. Attribute bukan kunci haruslah tidak memiliki ketergantungan transitif, seluruh attribute bukan kunci pada suatu relasi hanya memiliki ketergantungan fungsional pada primary key direlasi itu saja.

### Kamus Data

Kamus data berfungsi pelaku sistem untuk mengartikan aplikasi secara detail dan mengorganisasi semua elemen data yang digunakan dalam sistem secara persis sehingga pemakai dan penganalisa sistem mempunyai dasar pengertian yang sama tentang masukan, keluaran, penyimpanan dan proses (Ladjamudin, 2005).

### Waterfall

Model waterfall sering disebut model sekuensial linier (Sequential Linier) atau alur hidup klasik (Classic Life Cicle). Model waterfall menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengodean, pengujian dan tahap pendukung (pressman 2002).

1. Analisis Kebutuhan Perangkat Lunak

Proses pengumpulan kebutuhan di intensifkan dan di fokuskan khususnya pada perangkat lunak untuk memahami sifat program yang dibangun, perekayasa perangkat lunak (Analisis) harus memahami domain informasi, tingkah laku, unjuk karya dan antarmuka (interface) yang diperlukan.

1. Design

Design perangkat lunak sebenarnya adalah proses multi langkah yang berfokus pada empat attribute sebuah program yang berbeda struktur data, arsitektur perangkat lunak, representasi perangkat lunak yang dapat diperkirakan demi kualitas sebelum dimulai pengkodean.

1. Generasi kode

Desain harus diterjemahkan kedalam bentuk mesin yang bisa dibaca, langkah pembutan kode melakukan tugas ini jika desain dilakukan secara lengkap, pembuatan kode dapat dilakukan secara mekanis.

1. Test atau Pengujian

Sekali kode dibuat pengujian program dimulai. Proses pengujian berfokus pada logika internal perangkat lunak memastikan bahwa semua pernyataan telah diuji, dan pada eksternal fungsional yaitu mengarahkan pengujian untuk menemukan kesalahan-kesalahan dan memastikan bahwa input dibatasi dan akan memberikan hasil actual yang sesuai dengan hasil yang dibutuhkan.

1. Pendukung atau Pemeliharaan

Tidak menutup kemungkinan bahwa setiap perangkat lunak mengalami perubahan ketika sudah dikirimkan ke user. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung atau perawatan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tetapi tidak untuk membuat perangkat lunak baru.

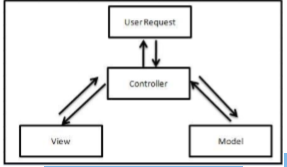
## **CodeIgniter**

Menurut (Destiningrum & Adrian, 2017) CodeIgniter adalah Sebuah framework php yang bersifat open source dan menggunakan metode MVC (Model, View, Controller) untuk memudahkan developer atau programmer dalam membangun sebuah aplikasi berbasis web tanpa harus membuatnya dari awal”. Dalam situs resmi codeigniter, (Official Website CodeIgniter,2002) menyebutkan bahwa codeigniter merupakan framework PHP yang kuat dan sedikit bug. Codeigniter ini dibangun untuk para pengembang dengan bahasa pemrogram PHP yang membutuhkan alat untuk membuat web dengan fitur lengkap. Framework Codeigniter dikembangkan oleh Rick Ellis, CEO Ellislab, Inc. kelebihan dari framework codeigniter jika dibandingkan dengan framework lain adalah sebagai berikut:

1. Gratis (Open-Source) Kerangka kerja Codeigniter memiliki lisensi dibawah Apache/BSD open-source sehingga bersifat bebas atau gratis.

2. Berukuran kecil Ukuran yang kecil merupakan keunggulan tersendiri jika dibandingkan framework lain yang berukuran besar dan membutuhkan resource yang besar dan juga dalam eksekusi maupun penyimpanannya.

3. Menggunakan konsep M-V-C Codeigniter merupakan konsep M-V-C (ModelView-Controller) yang memungkinkan pemisahan antara layer application-logic dan presentation. Dengan konsep ini kode PHP, query Mysql, Javascript dan CSS dapat saling dipisah-pisahkan sehingga ukuran file menjadi lebih kecil dan lebih mudah dalam perbaikan kedepannya atau maintenance.



Gambar 2.1 Konsep Aliran M-V-C

### Framework

Menurut (Destiningrum & Adrian, 2017) Framework adalah kumpulan intruksi-intruksi yang dikumpulkan dalam class dan function-function dengan fungsi masingmasing untuk memudahkan developer dalam memanggilnya tanpa harus menuliskan syntax program yang sama berulang-ulang serta dapat menghemat waktu.

### Desain Patern MVC (Model, View, Controller)

Model View Controller merupakan suatu konsep yang cukup populer dalam pembangunan aplikasi web, berawal pada bahasa pemrograman Small Talk, MVC memisahkan pengembangan aplikasi berdasarkan komponen utama yang membangun sebuah aplikasi seperti manipulasi data, user interface, dan bagian yang menjadi kontrol aplikasi. Terdapat 3 jenis komponen yang membangun suatu MVC pattern dalam suatu [aplikasi](https://id.wikipedia.org/wiki/Aplikasi) yaitu :

1. View, merupakan bagian yang menangani presentation logic. Pada suatu aplikasi web bagian ini biasanya berupa file template HTML, yang diatur oleh controller. View berfungsi untuk menerima dan merepresentasikan data kepada user. Bagian ini tidak memiliki akses langsung terhadap bagian model.
2. Model, biasanya berhubungan langsung dengan database untuk memanipulasi data (insert, update, delete, search), menangani validasi dari bagian controller, namun tidak dapat berhubungan langsung dengan bagian view.
3. Controller, merupakan bagian yang mengatur hubungan antara bagian model dan bagian view, controller berfungsi untuk menerima request dan data dari user kemudian menentukan apa yang akan diproses oleh aplikasi.

Dengan menggunakan prinsip MVC suatu aplikasi dapat dikembangkan sesuai dengan kemampuan developernya, yaitu programmer yang menangani bagian model dan controller, sedangkan designer yang menangani bagian view, sehingga penggunaan arsitektur MVC dapat meningkatkan maintanability dan organisasi kode. Walaupun demikian dibutuhkan komunikasi yang baik antara programmer dan designer dalam menangani variabel-variabel yang akan ditampilkan.

Ada beberapa kelebihan CodeIgniter (CI) dibandingkan dengan Framework PHP lain,

* Performa sangat cepat : salah satu alasan tidak menggunakan framework adalah karena eksekusinya yang lebih lambat daripada PHP from the scracth, tapi Codeigniter sangat cepat bahkan mungkin bisa dibilang codeigniter merupakan framework yang paling cepat dibanding framework yang lain.
* Konfigurasi yang sangat minim *(nearly zero configuration)* : tentu saja untuk menyesuaikan dengan database dan keleluasaan routing tetap diizinkan melakukan konfigurasi dengan mengubah beberapa file konfigurasi seperti database.php atau autoload.php, namun untuk menggunakan codeigniter dengan setting standard, anda hanya perlu mengubah sedikit saja file pada [folder](https://id.wikipedia.org/w/index.php?title=Folder&action=edit&redlink=1) config.
* Banyak komunitas: dengan banyaknya komunitas CI ini, memudahkan kita untuk berinteraksi dengan yang lain, baik itu bertanya atau teknologi terbaru.

Dokumentasi yang sangat lengkap : Setiap paket instalasi codeigniter sudah disertai [user guide](https://id.wikipedia.org/w/index.php?title=Panduan&action=edit&redlink=1) yang sangat bagus dan lengkap untuk dijadikan permulaan, bahasanya pun mudah dipahami.

* Dan banyak lagi yang lainnya.

## ***PHP***

*PHP* adalah singkatan dari *Hypertext Preprosesor* yaitu bahasa pemrograman *web server-side* yang bersifat *open source*. *PHP* merupakan *script* yang terintegrasi dengan *HTML* dan berada pada server. *PHP* adalah *script* yang digunakan untuk membuat halaman *website* yang dinamis. Dinamis berarti halaman yang akan ditampilkan di bentuk saat client melakukan pengaksesan halaman tersebut.

Mekanisme ini menyebabkan informasi yang diterima *client* selalu *up to date*. Semua *script PHP* dieksekusi pada *server* dimana *script* tersebut dijalankan. *PHP* mempunyai keunggulan di bandingkan dengan bahasa pemrograman lain selain sebagai sebuah bahasa *script* yang tidak melakukan sebuah kompilasi dalam penggunaanya, juga merupakan *scripting* yang paling mudah dipelajari karena memiliki referensi yang cukup banyak serta bersifat *open source* yang dapat digunakan di berbagai sistem operasi (Pahlefi, 2015)

## ***MySQL***

Menurut (Destiningrum & Adrian, 2017) *MySQL* (*My Structured Query Language*) adalah suatu sistem basis data relation atau Relational Database managemnt System (RDBMS) yang mampu bekerja secara cepat dan mudah digunakan MySQL juga merupakan program pengakses database yang bersifat jaringan, sehingga sapat digunakan untuk aplikasi multi user (banyak pengguna). MySQL didistribusikan gratis dibawah lisensi GPL (General Public License). Dimana setiap program bebas menggunakan MySQL namun tidak bisa dijadikan produk turunan yang dijadikan closed source atau komersial.

## **Xampp**

Xampp adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai server yang berdiri sendiri (localhost), yang terdiri atas program Apache HTTP Server, MySQL database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMPP merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP dan Perl. Program ini tersedia dalam GNU General Public License dan bebas, merupakan web server yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis. Untuk mendapatkanya dapat mendownload langsung dari web resminya (Palit, Rindengan, & Lumenta, 2015)

## ***Website***

### Definisi *Website*

*Website* adalah sering juga disebut *Web*, dapat diartikan suatu kumpulan-kumpulan halaman yang menampilkan berbagai macam informasi teks, data, gambar diam ataupun bergerak, data animasi, suara, video maupun gabungan dari semuanya, baik itu yang bersifat statis maupun yang dinamis, yang dimana membentuk satu rangkaian bangunan yang saling berkaitan dimana masing-masing dihubungkan dengan jaringan halaman atau hyperlink.

Definisi *website* adalah kumpulan dari berbagai macam halaman situs, yang terangkum didalam sebuah domain atau juga subdomain, yang lebih tempatnya berada di dalam *WWW (World Wide Web)* yang tentunya terdapat di dalam Internet. Halaman website biasanya berupa dokumen yang ditulis dalam format *Hyper Text Markup Language (HTML),* yang bisa diakses melalui *HTTP, HTTP* adalah suatu protokol yang menyampaikan berbagai informasi dari *server website* untuk ditampilkan kepada para user atau pemakai melalui *web browser.*

Jenis-jenis *website* ada 3 (tiga) macam diantaranya, bisa dibaca dibawah ini:

1. *Website* *Statis* adalah suatu *website* yang mempunyai halaman yang tidak berubah. Yang artinya adalah untuk melakukan sebah perubahan pada suatu halaman hanya bisa dilakukan secara manual yitu dengan cara mengedit kode-kode yang menjadi struktur dari *website* itu sendiri.
2. *Website Dinamis* adalah merupakan suatu *website* yang secara strukturnya diperuntukan untuk *update* sesering mungkin. Biasanya selain dimana utamanya yang bisa diakses oleh para pengguna (*user*) pada umumnya, juga telah disediakan halaman *backend* yaitu untuk mengedit kontent dari *website* tersebut. Contoh dari *website dinamis* seperti web berita yang didalamnya terdapat fasilitas berita, dsb.
3. *Website Interaktif* adalah suatu website yang memang pada saat ini memang terkenal. Contohnya website interaktif seperti forum dan blog. Di *website* ini para pengguna bisa berinteraksi dan juga beradu argument mengenai apa yang menjadi pemikiran mereka (Santoso, 2018)

### Sejarah dan Perkembangan *Website*

*Website* pertama kali ditemukan oleh Sir Timothy John, Tim Berners-Lee. Pada tahun 1991 *website* terhubung dengan jaringan.Tujuan dari dibuatnya *website* pada saat itu yakni untuk mempermudah tukar menukar dan memperbaharui informasi kepada sesama peneliti di tempat mereka bekerja. Dengan demikian pengertian *website* saat itu masih sebatas tukar menukar informasi, bukan pengertian *website* secara *terminolog*i.

*Website* dipublikasikan ke publik setelah adanya pengumuman dari CERN pada tanggal 30 april 1993. CERN menyatakan bahwa *website* dapat digunakan secara gratis oleh semua orang. Pada saat ini pengertian *website* sudah masuk ke dalam ranah publik karena sudah bisa digunakan oleh semua orang dimanapun dan kapanpun (Adi, 2017)