BAB 2
LANDASAN TEORI

1.1 Surveilans Berbasis Masyarakat

Menurut WHO, surveilans berbasis masyarakat adalah pengumpulan, analisis, dan analisis data secara terus-menerus dan sistematis yang kemudian disebarluaskan kepada pihak-pihak yang bertanggung jawab dalam pencegahan penyakit dan masalah kesehatan lainnya. Surveilans memantau terus-menerus kejadian dan kecenderungan penyakit, mendeteksi dan memprediksi outbreak pada populasi, mengamati faktor-faktor yang mempengaruhi kejadian penyakit, seperti perubahan-perubahan biologis pada agen,vector, dan reservoir. Selanjutnya surveilans menghubungkan informasi tersebut kepada pembuat keputusan agar dapat dilakukan langkah-langkah pencegahan dan pengendalian penyakit. Kadang digunakan istilah surveilans epidemiologi. Baik surveilans berbasis masyarakat maupun surveilans epidemiologi hakikatnya sama saja, sebab menggunakan metode yang sama, dan tujuan epidemiologi adalah untuk mengendalikan masalah kesehatan masyarakat, sehingga epidemiologi dikenal sebagai sains inti kesehatan masyarakat (*core science of public health*) [2].

Surveilans memungkinkan pengambil keputusan untuk memimpin dan mengelola dengan efektif. Surveilans berbasis masyarakat memberikan informasi kewaspadaan dini bagi pengambil keputusan dan manajer tentang masalah-masalah kesehatan yang perlu diperhatikan pada suatu populasi. Surveilans merupakan instrument penting untuk mencegah *outbreak* penyakit dan mengembangkan respon segera ketika penyakit mulai menyebar. Informasi dari surveilans juga penting bagi kementerian kesehatan, kementrian keuangan, dan donor untuk memonitor sejauh mana populasi telah terlayani dengan baik. Surveilans berbeda dengan pemantauan biasa. Surveilans dilakukan secara terus menerus tanpa terputus, sedang pemantauan dilakukan intermiten atau episodik dengan mengamati secara terus-menerus dan sistematis maka perubahan-

perubahan kecenderungan penyakit dan faktor yang mempengaruhi dapat diamati atau diantisipasi, sehingga dapat dilakukan langkah-langkah investigasi dan pengendalian penyakit yang tepat [2].

1.1.1 Ruang Lingkup Sistem Surveilans Epidemiologi Kesehatan.

Masalah kesehatan dapat disebabkan oleh berbagai sebab, oleh karena itu secara operasional masalah-masalah kesehatan tidak dapat diselesaikan oleh sektor kesehatan sendiri, diperlukan tatalaksana terintegrasi dan komprehensif dengan kerjasama yang harmonis antara sektor dan antar program, sehingga perlu dikembangkan subsistem survailans kesehatan yang terdiri dari surveilans penyakit menular, surveilans penyakit tidak menular, surveilans kesehatan lingkungan dan perilaku, surveilans masalah kesehatan, dan surveilans kesehatan matra[1].

1. Surveilans Epidemiologi Penyakit Menular

Merupakan analisis terus menerus dan sistematis terhadap penyakit menular dan faktor risiko untuk mendukung upaya pemberantasan penyakit menular.

2. Surveilans Epidemiologi Penyakit Tidak Menular

Merupakan analisis terus menerus dan sistematis terhadap penyakit tidak menular dan faktor risiko untuk mendukung upaya pemberantasan penyakit tidak menular.

3. Surveilans Epidemiologi Kesehatan Lingkungan dan Perilaku

Merupakan analisis terus menerus dan sistematis terhadap penyakit dan faktor risiko untuk mendukung program penyehatan lingkungan.

4. Surveilans Epidemiologi Masalah Kesehatan

Merupakan analisis terus menerus dan sistematis terhadap masalah kesehatan dan faktor risiko untuk mendukung program-program kesehatan tertentu.

5. Surveilans Epidemiologi Kesehatan Matra

Merupakan analisis terus menerus dan sistematis terhadap masalah kesehatan dan faktor risiko untuk upaya mendukung program kesehatan matra.

1.1.2 Penyelenggaraan Berdasarkan Metode Pelaksanaan

Pelaksanaan surveilans epidemiologi kesehatan dapat menggunakan satu cara atau kombinasi dari beberapa cara penylenggaraan surveilans epidemiologi.

1. Surveilans Epidemiologi Rutin Terpadu**,** adalah penyelenggaraan surveilans epidemiologi terhadap beberapa kejadian, permasalahan, dan atau faktor risiko kesehatan.
2. Surveilans Epidemiologi Khusus**,** adalah penyelenggaraan surveilans epidemiologi terhadap suatu kejadian, permasalahan, faktor risiko atau situasi khusus kesehatan.
3. Surveilans Sentinel**,** adalah penyelenggaraan surveilans epidemiologi pada populasi dan wilayah terbatas untuk mendapatkan signal adanya masalah kesehatan pada suatu populasi atau wilayah lebih luas.
4. Surveilans Epidemiologi**,** penyelenggaraan surveilans pada periode tertentu serta populasi dan wilayah tertentu untuk mengetahui lebih mendalam gambaran epidemiologi penyakit, permasalahan atau faktor risiko kesehatan.

1.1.3 Sasaran Penyelenggaraan

Sasaran penyelenggaraan sistem surveilans epidemiologi kesehatan meliputi masalah-masalah yang berkaitan dengan program kesehatan yang ditetapkan berdasarkan prioritas nasional, bilateral, regional dan global, penyakit ptensial wabah, bencana dan komitmen litas sekor serta sasaran spesifik lokal atau daerah [1]. Secara rinci sasaran penyelenggaraan sistem surveilans epidemiologi kesehatan adalah sebagai berikut :

1. Surveilans Epidemiologi Penyakit Menular

Prioritas sasaran penyelenggaraan surveilans epidemiologi penyakit menular adalah :

1. Surveilans penyakit yang dapat dicegah dengan imunisasi.

2. Surveilas AFP.
3. Surveilans penyakit potensial wabah atau kejadian luar biasa penyakit menular dan keracunan.
4. Surveilans penyakit demam berdarah dan demam berdarah dengue.
5. Surveilans malaria.
6. Surveilans penyakit-penyakti zoonis, antraks, rabies, leptospirosis dan sebagainya.
7. Surveilans penyakit filariasis.
8. Surveilans penyakit tuberculosis.
9. Surveilans penyakit diare, tipus perut, kecacingan dan penyakit perut lainnya.
10. Surveilans penyakit kusta.
11. Surveilans penyakit kusta.
12. Surveilans penakit frambosia.
13. Surveilans penyakit HIV/AIDS.
14. Surveilans penyakit menular seksual.
15. Surveilans penyakit pneumonia, termasuk penyakit pneumonia akut berat.

2. Surveilans epidemiologi tidak menular

Prioritas sasaran penyelenggaaan surveilans epidemiologi penyakit tidak menukar adalah :

1. Surveilans hipertensi, stroke dan penyakit jantung coroner, Surveilans jantung coroner.
2. Surveilans diabetes mellitus
3. Surveilans neoplasma
4. Surveilans penyakit paru obstuksi kronis
5. Surveilans gangguan mental
6. Surveilans kesehatan akibat kecelakaan

3. Surveilans Epidemiologi Kesehatan Lingkungan dan Perilaku

Prioritas sasaran penyelenggaraan surveilans epidemiologi kesehatan lingkungan dan perilkaku adalah :

1. Surveilans sarana air bersih
2. Surveilans tempat-temat umum
3. Surveilans pemukiman dan lingkungan perumahan
4. Surveilans limba industri, rumah sakit dan kegiatan lainnya.
5. Surveilans vektor penyakit
6. Surveilans kesehatan dan keselamatan kerja
7. Surveilans rumah sakit dan sarana pelayanan kesehatan lainnya, termasuk infeksi nosokomial.

4. Surveilans Epidemiologi Masalah Kesehatan

Prioritas sasaran penyelenggaraan surveilans epidemiologi masalah kesehatan adalah :

1. Surveilans gizi dan sistem kewaspadaan pangan dan gizi (SKPG).
2. Surveilans gizi mikro kurang yodium, anemia gizi besi, kekurangan vitamin A.
3. Surveilans gizi lebih.
4. Surveilans kesehatan ibu dan anak termasuk reproduksi.
5. Surveilans kesehatan lanjut usia.
6. Survei penyalahgunaan obat, narkotika, psikotropika, zat adiktif dan bahan berbahaya.
7. Surveilans penggunaan sediaan farmasi, obat, obat tradisionil, bahan kosmetika, serta peralatan.
8. Surveilans kualitas makanan dan bahan tambahan makanan

5. Surveilans Epidemiologi Kesehatan Matra

Prioritas sasaran penyelenggaraan surveilans epidemiologi kesehatan matra adalah :

1. Surveilans kesehatan haji.
2. Surveilans kesehatan pelabuhan dan lintas batas perbatasan.

3. Surveilans bencana dan masalah sosial.
4. Surveilans kesehatan matra laut dan udara.
5. Surveilans pada kejadian luar biasa penyakit dan keracunan.

1.2 ISPA

ISPA adalah dari singkatan Infeksi Saluran Pernafasa Atas.ISPA bisa disebabkan oleh beberapa faktor dan disebarkan dari satu orang ke orang lainnya melalui udara pada saat penderita bersin atau batuk. Atau penularan juga bisa terjadi bila ada kontak lewat makanan yang terkontaminasi tangan penderita. ISPA bisa disebabkan oleh beberapa penyebab seperti reaksi alergi, virus, bakteri dan jamur.[3]

Gejala gejala ISPA bervariasi seperti hidung berair, hidung mampet, bersin, batuk, sakit tenggorokan, demam dan sakit badan sehingga penurunan berat badan [3].

1.3 Rhinitis akut

Rhinitis akut adalah peradangan pada mukosa hidung yang berlangsung akut (<12 minggu). Hal ini dapat disebabkan oleh infkesi virus, bakteri, ataupun iritan. Radang sering ditemukan karena manifestasi dari rhinitis simpleks (*common cold*), influenza, penyakit eksantem (seperti morbili, variola, varicella,pertusis), penyakit spesifik, serta sekundaer dari iritasi local atau trauma. Rhinitis akut merupakan penyebab morbiditas yang signifikan walaupun sering dianggap sepele oleh para praktisi. Gejala-gejala rhinitis secara signifikan mempengaruhi kualitas hidup pasien karena gejala-gejala sistemik yang menyertainya seperti *fatigue* dan sakit kepala [4].

Berdasarkan hasil anamnesis dari dokter, pasien datang dengan keluhan keluar ingus dari hidung (rinorea), hidung tersbumbat disertai rasa panas dan gatal pada hidung.

1. Rhinitis simpleks : Gejala berupa panas di daerah belakang hidung pada awalnya, lalu segera diikuti dengan hidung tersumbat,rinorea, dan bersin

yang berulang-ulang. Pasien merasa dingin, dan terdapat demam diikuti juga dengan gejala sistemik seperti demam, malaise dan sakit kepala.

2. Rhinitis influenza : gejala sistemik umumnya lebih berat disertai sakit pada otot.
3. Rhinitis eksantematous : gejala terjadi sebelum tanda karakteristik atau ruam muncul.
4. Rhinitis iritan : gejala berupa ingus yang sangat banyak dan bersin.
5. Rhinitis difteria : gejala berupa demam, toksemia, terdapat limfadenitis, dan mungki ada paralisis otot pernafasan.

1.4 Faringitis

Faringitis merupakan peradangan dinding faring yang disebabkan oleh virus (40-60%), bakteri (5-40%), alergi trauma, iritan, dan lain-lain. Setiap tahunnya ± 40 juta orang mengunjungi pusat pelayanan kesehatan karena faringitis. Anak-anak dan orang dewasa umunya mengalami 3-5 kali infeksi virus pada saluran pernafasan atas termasuk faringitis. Secara global di dunia ini vira faringitis merupakan penybab utama seseorang absen bekerja atau sekolah [4].



Gambar 2. 1 Contoh pasien yang terdampak faringitis

Berdasarkan hasil anamnesis dari dokter, pasien datang dengan keluhan nyeri tenggorokan, sakit jika menelan dan batuk. Gejala dan tanda yang ditimbulkan faringitis tergantung pada mikroorganisme yang menginfkesi. Secara garis besar faringitis menunjukan tanda dan gejala umum seperti leas, anorexia,

demam, suara serak, kaku dan sakit pada otot leher. Gejala khas berdasarkan jenisnya, yaitu :

1. Faringitis viral (umumnya oleh Rhinovirus) : diawali dengan gejala rhinitis dan beberapa hari kemudia timbul faringitis. Gejala lain demam disertai rinorea dan mual.
2. Faringitis bacterial : nyeri kepada hebat, muntah, kadang disertai demam dengan suhu yang tinggi, jarang disertai batuk.
3. Faringitis fungal : terutama nyeri tenggorok dan nyeri menelan.
4. Faringitis kronik hiperplastik : mula-mula tenggorok kering, gatal dan akhirnya batuk dan berdahak.
5. Faringitis kronik atrofi : umumnya tenggorokan kering dan tebal serta mulut berbau.
6. Faringitis tuberkulosis : nyeri hebat pada faring dan tidak berespon dengan pengobatan bacterial non spesifik.
7. Bila dicurigai faringitis gonorea atau faringitis luetika, ditanyakan riwayat hubungan seksual.

1.5 Tonsilitis

Tonsilitis adalah peradangan tonsil palatine yang merupakan bagian dari cincin Waldeyer. Cincin Waldeyer teridiri atas susunan kelenjar limfa yang terdapat di dalam rongga mulut yaitu : tonsil faringeal (adenoid), tonsil palatine (tonsil faucial), tonsil lingual (tonsil pangkal lidah), tonsil tuba Eustachius (lateral band dinding faring / *Gerlach`s tonsil*). Penyakit ini banyak diderita oleh anak-anak berusia 3 sampai 10 tahun dan anak reaja berusia 15 hingga 25 tahun [4].



Gambar 2. 2 Contoh pasien yang terdampak tonsilitis

Berdasarkan hasil anamnesis dari dokter, pasien datang dengan keluhan nyeri pada tenggorokan. Gejala lainnya tergantung penyebab tonsillitis.

1. Penderita tonsillitis akut awalnya mengeluh rasa kering di tenggorokan, kemudia berubah menjadi rasa nyeri di tenggorokan dan nyeri saat menelan. Rasa nyeri semakin lama semakin bertambah sehingga anak menjadi tidak mau makan. Nyeri hebat ini menyebar sebagai *referred pain* ke sendi-sendi dan telinga. Nyeri pada telinga (otalgia) tersebut tersebar melalu nervus glossofaringeus (IX). Keluhan lainnya berupa demam yang dapat sangat tinggi sampai menimbulkan kejang pada bayi dan anak-anak. Rasa nyeri kepala, badan lesu dan nafsu makan berkurang sering menyertai pasien tonsillitis akut. Suara pasien terdengan seperti orang yang mulutnya penuh terisi makanan panas. Keadaan ini disebut *plummy voice* / *hot potato voice*. Mulut berbau (*foetorex ore*) dan ludah menumpuk dalam kavum oris akibat nyeri telan yang hebat (ptialismus). Tonsilitis viral lebih menyerupai *common cold* yang disertai rasa nyeri tenggorokan.
2. Pada tonsillitis kronik, pasien mengelu ada penghalang/ mengganjal di tenggorokan, tenggorokan terasa kering dan pernafasan berbau (halitosis).

3. Pada *Angina Plaut Vincent* (Stomatitis ulseromembranosa) gejala yang timbul adalah demam tinggi (39 C), nyeri dimulut, gigi dan kepala, sakit tenggorokan, badan lemah, gusi mudah berdarah dan hipersalivasi.

1.6 Varisela

Infeksi akut primer oleh virus *Varicellazoster* yang menyerang kulit dan mukosa, klinis terdapat gejala konstitusi, kelainan kulit *polimorf*, terutama berlokasi di bagian sentral tubuh. Masa inkubasi 14-21 hari. Penularan melalui udara dan kontak langsung [4].



Gambar 2. 3 Contoh pasien yang terdampak varisela

Berdasarkan hasil anamnesis dari dokter, pasien datang dengan keluhan demam,*malaise*, dan nyeri kepala. Kemudia disusul timbulnya lesi kulit berupa papul eritem yang dalam waktu beberapa jam berubah menjadi vesikel. Biasanya disertai rasa gatal. Faktor risiko terjadinya varisela terjadi pada anak-anak, riwayat kontak dengan penderita varisela dan keadaan *imunodefisiensi*. Pada kulit terjadinya erupsi berupa *papul eritematosa* yang dalam waktu beberapa jam berubah menjadi vesikel. Bentuk vesikel ini khas berupa tetesan embun. Vesikel akan menjadi keruh dan kemudian menjadi krusta. Sementara proses ini berlangsung, timbul lagi vesikel-vesikel baru yang menimbulkan gambaran *polimorfik* khas untuk varisela. Penyebaran terjadi sentrifugal, serta dapat menyerang selaput lender, mata, mulut, dan saluran napas atas [4].

1.7 Dermatitis Atopik

Dermatitis Atopik adalah peradangan kulit berulang dan kronis dengan disertai gatal. Pada umumnya terjadi selama masa bayi dan anak-anak dan sering berhubungan dengan peningkatan kadar IgE dalam serum serta riwayat atopi pada keluarga atau penderita [4].

Sinonim dari penyakit ini adalah *eczema* atopic, *eczema* konstitusional, *eczema* fleksural, *neurodermatitis diseminata, prurigo* Besnier.



Gambar 2. 4 Contoh pasien yang terdampak dermatitis atopik

Berdasarkan hasil anamnesis dari dokter, pasien datang dengan keluhan gatal yang bervariasi lokasinya tergantung pada jenisnya. Gejala utama DA adalah pruritus (gatal), dapat hilang timbul sepanjang hari, tetapi umumnya lebih hebat pada malam hari. Akibatnya penderita akan menggaruk. Pasien biasanya mempunyai riwayat juga sering merasa cemas, egois, frustasi, agresit, atau merasa tertekan. Pada kulit penderita terlihat kulit kering, pucat, terdapat papul, likenifikasi, eritema, erosi, eksoriasi, eksudasi dan krusta pada lokasi predileksi [4].

Predileksi pada DA akan berbeda berdasarkan umur.

a. Tipe Bayi (infantil)
    1. Dahi, pipi, kulit kepala, pergelangan tangan dan tungkai, serta lutut (pada anak yang mulai merangkak).

2. Lesi berupa eritmeta, papul vesikel halus, eksudatif, krusta.

b. Tipe anak
   1. Lipat siku, lipat lutut, pergelangan tangan bagian dalam, kelopak mata, leher, kadang-kadang di wajah.
   2. Lesi berupa papul, sedikit eksudatif, sedikit skuama, likenifikasi, erosi. Kadang-kadang disertai pustule.

c. Tipe remaja dan dewasa
   1. Lipat siku, lipat lutut, samping leher, dahi, sekitar mata, tangan dan pergelangan tangan, kadang-kadang ditemukan setempat misalnya bibir mulut, bibir kelamin putting susu, atau kulit kepala.
   2. Lesi berupa plak popular eritematosa, skuama, likenifikasi, kadang-kadang erosi dan eksudasi, terjadi hiperpigmentasi.

1.8 Pneumonia aspirasi

Pneumonia aspirasi adalah pneumonia yang disebabkan oleh terbawanya bahan yang ada diorofaring pada saat respirasi ke saluran napas bawah dan dapat menimbulkan keruasakan parenkim paru [4].

Berdasarkan hasil anamnesis dari dokter, pasien mendadak batuk dan sesak napas sesudah makan atau minum. Umumnya pasien datang 1-2 minggu sesudah aspirasi, dengan keluhan demam mengigil, nyeri pleuritik, batuk, dan dahak purulent berbau. Pasien tampak sesak napas. Dapat terderngar bising gesek pleura [4].

1.9 Konjungtivitis

Konjungtivitis adalah radang konjungtiva yang dapat disebabkan oleh mikroorganisme (virus, bakteri), iritasi atau reaksi alergi. Konjungtivitis ditularkan melalui kontak langsung dengan sumber infeksi. Penyakit ini dapat menyerang semua umur [4].

Berdasarkan hasil anamnesis dari dokter, pasien datang dengan keluhan mata merah, rasa mengganjal, gatal dan berair, kadang disertai secret. Umumnya tanpa disertai penurunan tajam penglihatan dan daya tahan tubuh menurun [3].

1.10 Kecerdasan Buatan

Para ahli mendefinisikan AI secara berbeda-bea tergantung pada sudut pandang mereka masing-masing, ada yang focus pada logika berpikir manusia saja,tetapi ada jga yang mendefinisikan AI secara lebih luas pada tingkah aku manusia [7]. Pada [RUS95], Stuart Russet dan Peter Norvig mengelompokan definisi AI, yang diperoleh dari bebrapa *textbook* berbeda, ke dalam empat kategori, yaitu :

1. *Thinking humanly* **:** *the cognitive modeling approach*

   Pendekatan ini dilakukan dengan dua cara sebagai berikut :

   1. Melalui introspeksi.
   2. Melalui eksperimen-eksperiman psikologi

2. Acting humanly **:** *the turing test approach*

   Pada tahun 1950, Alan Turing merancang suatu ujian bagi computer berintelejensi manusia untuk menguji apakah komputer tersebut mampu mengelabui seorang manusia yang menginterogassinya melalui *teletype* (komunikasi berbasis teks jarak jauh). Jika *interrogator* tidak dapat membedakan yang diinterogasi adalah manusia atau komputer, maka komputer berintelejensi tersebut lolos dari *turing test*. Komputer tersebut perlu memiliki kemampuan *Natural Language Processing, Knowledge Representation, Automated Reasoning, Machine Learning, Computer Vision, Robotics*. *Turing test* sengaja menghindari interaksi fisik antara *interrogator* dan komputer karena simulasi fiisk manusia tidak memerlukan intelijensia [7].

3. *Thinking rationally* **:** *thelaws of thought approach*

   Terdapat dua masalah dalam pendekatan ini, yaitu :

   1. Tidak mudah untuk membaut pengetahuan informal dan menyatakan pengetahuan tersebut ke dalam *formal term* yang diperlukan oleh notasi logika, khsususnya ketika pengetahuan tersebut memiliki kepastian kurang dari 100%

2. Terdapat perbedaan besar antara dapat memecahkan masalah “dalam prinsip” dan memecahkannya “dalam dunia nyata”.

4. *Actiong rationally* **:***The rational agent approach*

Membuat inferensi yang logis merupakan bagian dari suatu *rational agent.* Hal ini disebabkan satu-satunya cara untuk melakukan aksi secara rasional adalah dengan menalar secara logis. Dengan menalar secara logis, maka bisa didapatkan kesimpulan bahwa aksi yang diberikan akan mencapai tujuan atau tidak. Jika mencapai tujuan, maka *agent* dapat melakukan aksi berdasarkan kesimpulan tersebut [7].

1.11 Sistem Pakar

Sistem Pakar adalah bagian kecerdasan buatan yang berisi kondisi pemahaman teoritis tentang suatu persoalan dan sekumpulan aturan pemecahan persoalan heuristic yang dikembangkan oleh manusia untuk dapat memecahkan problema pada suatu domain yang spesifik.[5]

Sistem pakar merupakan perangkat lunak yang digunakan untuk memecahkan masalah yang biasanya diselesaikan oleh seorang pakar. Aturan-aturan di dalamnya tersimpan. Berdasarkan itu program memberikan solusi-solusi atau bantuan pengambilan keptutusan permasalahan tertentu, mirip dengan saran seorang pakar.[5]

1.11.1 Faktor Kepastian (*Certainty Factor*)

Faktor kepastian (*certainty factor*) diperkenalkan oleh Shortliffe Buchanan dalam pembeuatan MYCIN (Wesley,1984). *Cerntaity factor* (CF) merupakan nilai parameter klinis yang diberikan MYCIN untuk menunjukan besarnya kepercayaan [5]. *Certainty factor* didefinisikan sebagai berikut :

$$CF(H,E) = MB(H,E) – MD(H,E)$$

CF(H,E) : *certainty factor* dari hipotesis H yang diperngaruhi oleh gejala (*evidence*) E. Besarnya CF berkisar antara -1 sampai dengan 1. Nilai -1

menunjukan ketidakpercayaan mutlak, sedangkan nilai 1 menunjukan kepercayaan mutlak.

MB(H,E) : ukuran kenaikan kepercayaan (*measure of increased belief*) terhadap hipotesis H yang dipengaruhi oleh gejala E antara (0 sampai 1).

MD(H,E) : ukuran kenaikan ketidakpercayaan (*measure of increased disbelief*) terhadap hipotesis H yang dipengaruhi oleh gejala E antara (0 sampai 1).

1.11.1.1 CF Paralel

CF parallel merupakan CF yang diperoleh dari beberapa premis pada sebuah aturan. Besarnya CF sequensial dipengaruhi oleh CF user untuk masing-masing premis dan operator dari premis [5]. Rumus untuk masing-masing operator adalah :

CF(x dan y) = Min(CF(x),CF(y))

1.11.1.2 CF Sequensial

CF Sequencsial diperoleh dari hasil perhitungan CF parallel dari semua premis dalam satu aturan dengan CF aturan yang diberikan pakar [11]. Berikut rumus untuk melakukan perhitungan CF Sequensial :

CF(x,y) = CF(x) * CF(y)

dengan :

CF(x,y) : CF sequensial

CF(x) = CF paralel dari semua premis

CF(y) = CF pakar

1.12 Algoritma Fuzzy Evolusi

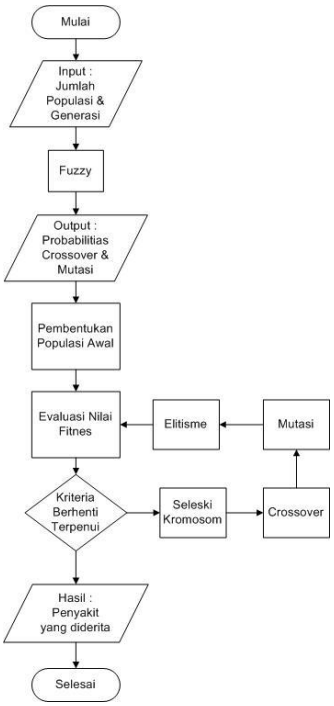
Algoritma *fuzzy* evolusi adalah sebuah teknik komputasi gabungan antara algoritma genetika dan sistem *fuzzy*. Metode ini hampir sama dengan metode algoritma genentika, namun parameter-parameter yang dipakai dihasilkan melalui

sistem *fuzzy*. Dalam menentukan nilai yang dihasilkan melalui sistem *fuzzy* perlu dibuat aturan-aturan *fuzzy* yang digunakan untuk penentuan hasil [8].

Beberapa tahap yang digunakan dalam algoritma *fuzzy* evolusi sama seperti dengan tahapan pada algoritma genetika, yaitu :

1. Representasi kromosom
2. Inisialisasi populasi
3. Fungsi evaluasi
4. Seleksi
5. Operator genentika, meliputi *crossover* dan mutasi
6. Penentuan parameter, yaitu paramenter control algoritma genetika, yaitu ukuran populias, peluang *crossover* dan peluan mutasi.

Untuk lebih jelas dapat dilihat pada gambar 2.5 .



Gambar 2. 5 Proses Evolusi *Fuzzy*

Dalam penentuan parameter ini dilakukan proses sistem *fuzzy* untuk mendapatkan nilai yang akan digunakan sebagai parameter. Dikarenakan cara kerja *fuzzy* evolusi menyerupai algortima genetika, maka kasus yang

menggunakan algoritma *fuzzy* evolusi hampir sama dengan kasus yang digunakan oleh algoritma genetika seperti permasalahan *learning*, peramalan, pemrograman otomatis, dan sebagainya [6].

Berdasarkan penelitian Tedi Sefuro yang berjudul "ANALISIS PERBANDINGAN ALGORITMA GENETIKA DAN ALGORITMA FUZZY EVOLUSI DALAM PENYELESAIAN TRAVELING SALESMAS PROBLEM" mendapatkan hasil bahwa algoritma fuzzy evolusi lebih unggul dalam pencarian solusi optimal dalam hal pencarial solusi optimal dibandingkan dengan algoritma genetika pada penyelesaian Traveling Salesmasn Problem [6].

1.12.1 Algoritma *fuzzy* evolusi model Xu

Pada algoritma *fuzzy* evolusi terdapat berbagai model yang digunakan dalam perpaduan antara algoritma genetika dan sistem *fuzzy*. Model yang dikemukakan oleh Xu, yaitu algoritma *fuzzy* evolusi yang menggunakan sistem *fuzzy* dalam penentuan paramenternya adalah menggunakan sistem inferensi *fuzzy* metode mamdani. Metode mamdani ini juga dikenal dengan metode Max-Min. Metode ini dikenalkan oleh Ebrahim Mamdani (1975). Metode mamdani yang digunakan oleh Xu untuk algoritma *fuzzy* evolusi adalah menggunakan dua buah masukan dan dua buah keluaran [8]. Dua buah masukan yang digunakan adalah :

1. Jumlah populasi yang digunakan.
2. Jumlah generasi yang akan diproses

Sedangkan dua buah keluaran yang akan dihasilkan adalah :

1. Nilai probabilitias *crossover*
2. Nilai probabilitas mutasi

Dalam menentukan nilai yang dihasilkan melalui sistem *fuzzy* perlu dibuat aturan-aturan *fuzzy* yang digunakan untuk penentuan hasil. Dalam model Xu aturan yang digunakan didasarkan dari masukan jumlah populasi yang diinginkan serta jumlah maksimum generasi [8]. Dari dua masukan tersebut akan menghasilkan nilai

untuk probabilitias *crossover* dan probabilitas mutasi. Aturan yang ditentukan oleh Xu dapat dilihat dalam tabel 2.1 dan 2.2.

Tabel 2. 1 Aturan untuk nilai probabilitas *crossover*

Pc	**Population Size**		
Generation	Small	Medium	Large
Short	Medium	Small	Small
Medium	Large	Large	Medium
Long	Very Large	Very Large	Large

Tabel 2. 2 Aturan untuk nilai probabilitas mutasi

Pc	**Population Size**		
Generation	Small	Medium	Large
Short	Large	Medium	Small
Medium	Medium	Small	Very Small
Long	Small	Very Small	Very Small

Dengan aturan diatas, jumlah populasi dan maksimum generasi yang dimasukan akan dianalisa dengan metode mamdani, sehingga didaptkan nilai probabilitas *crossover* dan probabilitas mutasi yang mana akan dipakai dalam iterasi.

Dalama algoritma *fuzzy* evolusi, aturan-aturan *fuzzy* yang telah dibuat harus sudah diimplementasikan terlebih dahulu sebelum proses iterasi dilakukan. Dari model Xu yang digunakan sesuai tabel 2.1 dan tabel 2.2 didapatkan sebanyak 9 aturan, yaitu :

1. IF (Populasi is SMALL) AND (Generasi is SHORT) THEN (ProbCrossover is MEDIUM) AND (ProbMutasi is LARGE).
2. IF (Populasi is MEDIUM) AND (Generasi is SHORT) THEN (ProbCrossover is SMALL) AND (ProbMutasi is MEDIUM).
3. IF (Populasi is LARGE) AND (Generasi is SHORT) THEN (ProbCrossover is SMALL) AND (ProbMutasi is SMALL).

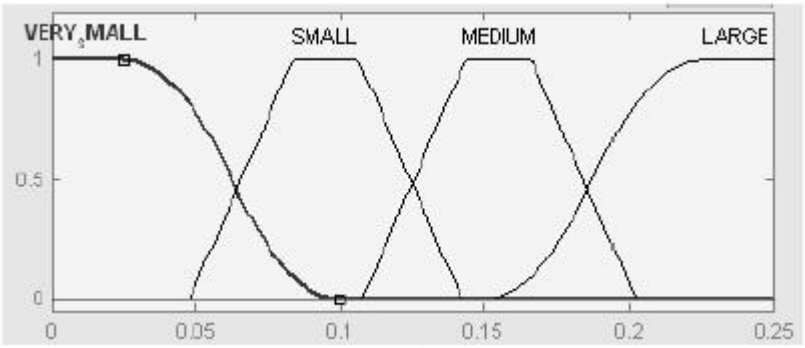
4. IF (Populasi is SMALL) AND (Generasi is MEDIUM) THEN (ProbCrossover is MEDIUM) AND (ProbMutasi is MEDIUM).
5. IF (Populasi is MEDIUM) AND (Generasi is MEDIUM) THEN (ProbCrossover is LARGE) AND (ProbMutasi is SMALL).
6. IF (Populasi is LARGE) AND (Generasi is MEDIUM) THEN (ProbCrossover is MEDIUM) AND (ProbMutasi is VERYSMALL).
7. IF (Populasi is SMALL) AND (Generasi is LONG) THEN (ProbCrossover is VERYLARGE) AND (ProbMutasi is SMALL).
8. IF (Populasi is MEDIUM) AND (Generasi is LONG) THEN (Probcrossover is VERYLARGE) AND (ProbMutasi is VERYSMALL).
9. IF (Populasi is LARGE) AND (Generasi is LONG) THEN (ProbCrossover is LARGE) AND (ProbMutasi is VERYSMALL).

Aturan-aturan yang dikembangkan oleh XU diimplementasikan dalam sistem *fuzzy* mamdani, tetapi perlu diperhatikan supaya sistem *fuzzy* mamdani dapat menghasilkan hasil, tentunya diperlukan semesta pembicaraan dan domain yang memberikan nilai batas untuk setiap himpunan yang ada pada setiap variabel [8].

Adapun semesta pembicaraan yang digunakan seperti jumlah populasi, generasi, probabilitas *crossover* dan mutasi dalam model Xu, ditentukan oleh peneliti karena hal ini belum ditemukan studi literature yang menjelaskan tentang ini. Fungsi keanggotaan yang biasa digunakan adalah fungsi keanggotaan segitiga dan trapesium. Dalam semesta pembicaraan jumlah populasi dan generasi nilai minimum yang digunakan tidak bernilai 0 (nol), dikarenakan pada saat jumlah populasi dan generasi bernilai 0 (nol) maka dalam proses pembentukan populasi tidak akan ada pembentukan disebabkan inisialisasi awal jumlah populasi yang digunakan adalah 0 (nol), begitu juga dengan jumlah generasi [8].

Pada umumnya probabilitas untuk *crossover* adalah antara 0.6 sampai 0.9. Sehingga semesta pembicaraanya diantara 0.6 sampai 0.9, aturan nilai yang digunakan dimisalkan adalah sebagai berikut :

1. Semesta pembicaraan : [0.6 0.9]
2. Domain SMALL : [0.625 0.7]
3. Domain MEDIUM : [0.63 0.7 0.72 0.78]
4. Domain LARGE : [0.72 0.78 0.8 0.87]
5. Domain VERY LARGE : [0.8 0.875]

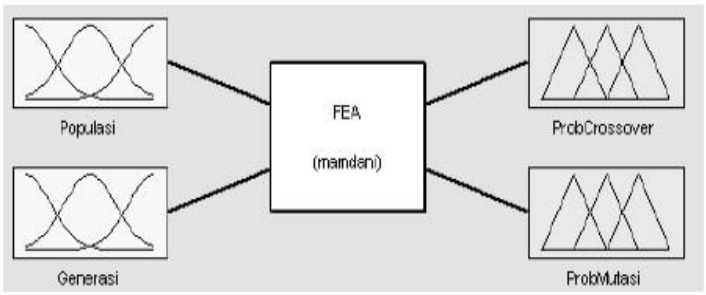


Gambar 2. 6 Semesta pembicaraan dan domain probabilitas mutasi

Sedangkan pada probabilitas mutasi pada umumnya sangat kecil, sekitar 1 dibagi dengan jumlah gen yang digunakan. Artinya peluang mutasi hanya terjadi pada sekitar satu gen saja pada tiap individu. Atau dapat dikatakan probabilitas mutasi mendekati nilai 0. Sehingga pada semesta pembicaraan dan domain untuk nilai probabilitas mutasi, aturan nilai yang digunakan dimisalkan sebagai berikut :

1. Semesta pembicaraan [0 0.25]
2. Domain VERY SMALL [0.025 0.1]
3. Domain SMALL [0.47 0.83 0.1 0.14]
4. Domain MEDIUM [0.1 0.14 0.167 0.2]
5. Domain LARGE [0.15 0.225]

Dibawah ini adalah gambaran proses sistem *fuzzy* mamdani yang digunakan pada penentuan nilai *fuzzy* untuk parameter probabilitas *crossover* dan mutasi pada algoritma *fuzzy* evolusi.

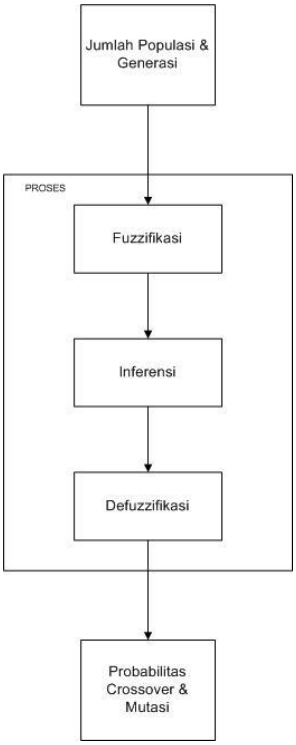


Gambar 2. 7 Alur proses sistem *fuzzy* mamdani

1.13 Algoritma Logika *Fuzzy*

Fuzzy secara bahasa diartika sebagai kabur atau samar-samar, SUatu nilai dapat bernilai besar atau salah secara bersamaan. Dalam *fuzzy* dikenal derajat keanggotaan yang memiliki rentang nilai 0 hinga 1. Berbeda dengan himpunan tegas yang memiliki nilai 1 atau 0 [7].

Algoritma Logika Fuzzy biasa digunakan dalam permasalahan yang mengandung ketidakpastian, beberapa kalangan menerjemahkan istilah ini sebagai logika samar. Namun berapa besar kesalahan suatu tergantung pada bobot keanggotaan yang dimilikinya. Berbeda dengan logika digital yang hanya memiliki dua nilai 1 atau 0. Algoritma *fuzzy* digunakan untuk menterjemahkan suatu besaran yang diekspresikan menggunakan bahasa, misalkan besaran kecepatan laju kendaraan yang diekspresikan dengan pelan, agak cepat, cepat, dan sangat cepat. Algortima *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 [7].



Gambar 2. 8 Alur proses algoritma logika *fuzzy*

1.13.1 Konsep Dasar Logika *Fuzzy*

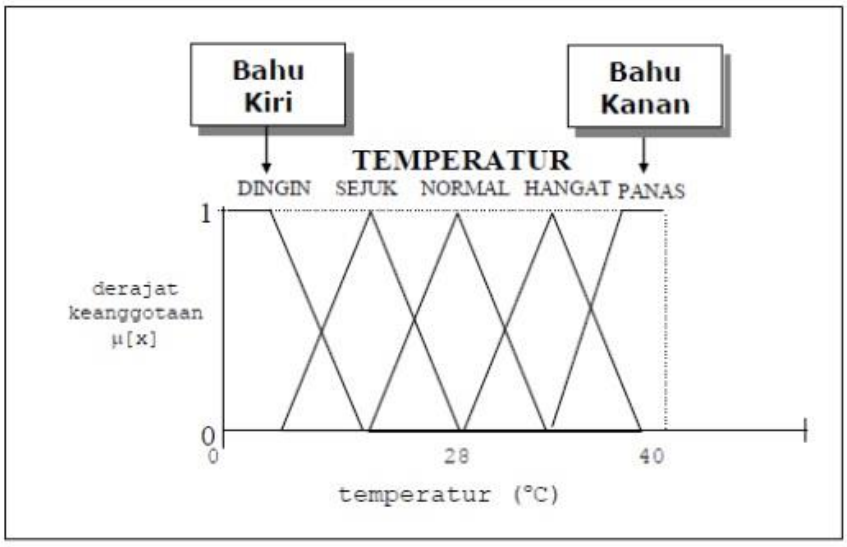
Teori tentang himpunan *fuzzy* atau himpunan samar pertama kali dikemukakan oleh Lotfi Zadeh sekitar tahun 1965.Dengan teori *fuzzy*, kita dapat merepresentasikan dan menangani masalah ketidakpastian yang dalam hal ini berarti keraguan,ketidakpastian,kekurangan informasi, dan kebenaran yang bersifat sebagian. Dalam logika *fuzzy* dikenal berhingga keandaan dari nilai 0 sampai nilai 1 [7]. Logika *fuzzy* tidak hanya mengenal dua keadaan tetapi juga mengenal sejumlah keadaan yang berkisar dari keadaan salah sampai keadaan benar. Ada beberapa hal yang perlu diketahui dalam memahami sistem *fuzzy* ini, yaitu :

1. Variabel *fuzzy*

   Variabel *fuzzy* merupakan variabel yang hendak dibahas dalam suatu sistem *fuzzy*.
2. Himpunan *fuzzy*

Himpunan *fuzzy* adalah himpunan yang tiap elemennya mempunyai derajat keanggotaan tertentu terhadap himpunannya. Himpunan *fuzzy* memiliki 2 atribut, yaitu :

1. Linguistik, yaitu penamaan suatu grup yang mewakili suatu keadaan atau kondisi tertentu dengan menggunakan bahasa alami, seperti : kecil, sedang , dan besar.



Gambar 2. 9 Contoh himpunan *fuzzy*

2. Numerik, yaitu suatu nilai (angka) yang menunjukan ukuran dari suatu variabel.

Pada himpunan *crisp*, nilai keanggotaan suatu item x dalam suatu himpunan A, yang sering ditulis dengan μA[x], memiliki 2 kemungkinan, yaitu :

1) Satu (1), yang berarti bahwa suatu item menjadi anggota dalam suatu himpunan.
2) No (0), yang berarti bahwa suatu item tidak menjadi anggota dalam suatu himpunan.

3. Semesta pembicaraan

Semesta pembicaraan adalah suatu keseluruhan nilai yang diperbolehkan dalam suatu variabel *fuzzy*. Nilai semesta pembicaraan dapat berupa bilangan positif atau bilangan negative.

4. Domain

Domain himpunan *fuzzy* adalah keseluruhan nilai yang diijinkan dalam semesta pembicaraan dan boleh dioperasikan dalam suatu himpunan. *Fuzzy* nilai *domain* dapat berupa bilangan positif maupun bilangan negatif.

1.13.2 Fuzzyfication

Fuzzyfication adalah pengubahan seluruh variabel *input/output* ke bentuk himpunan *fuzzy*. Rentang nilai variabel *input* dikelopokan menjadi beberapa himpunan *fuzzy* dan tiap himpunan mempunyai derajat keanggotaan tertentu.

1.13.3 Fungsi Keanggotaan

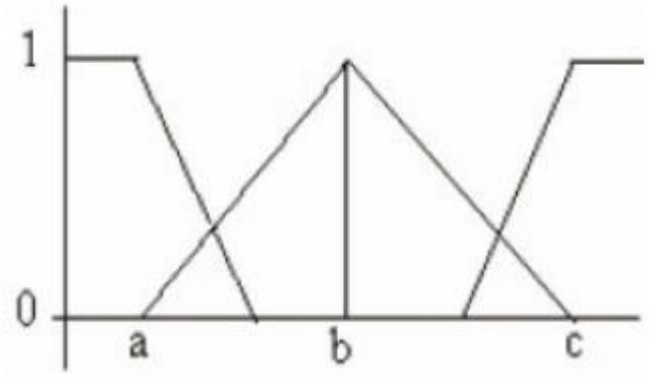
Basis data fungsi keanggotaan adalah suatu kurva yang menunjukan pemetaan titik-titik data ke adalam nilai keanggotaanya atau derajat keanggotaan yang memiliki interval 0 sampai 1. Salah satu cara yang dapat digunakan mendapatkan nilai keanggotaan adalah dengan melalui pendekatan fungsi. Ada beberapa fungsi yang digunakan fungsi keanggotaan datau himpunan fuzzy[7]. Dapat ditentukan dengan fungsi segitiga,trapezium, sigmoid dan linier

1. Fungsi Segitiga

Fungsi segitiga hanya terdapat satu nilai x yang memiliki derajat keanggotaan sama dengan 1, yaitu x = b. Tetapi, nilai-nilai disekitar b memiliki derajat keanggotaan yang turun menjauhi nilai 1. Persamaan fungsi keanggotaan segitiga adalah :

$$\mu(x;a,b,c) = \begin{cases} 0 & x < a \\ \dfrac{x-a}{b-a} & a \le x \le b \\ \dfrac{c-x}{c-b} & b < x \le c \\ 1 & x > c \end{cases}$$

Persamaan tersebut drepresentasikan dalam pada gambar 2.10 sebagai berikut :



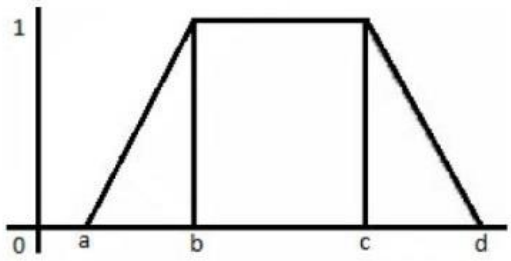
Gambar 2. 10 grafik fungsi keanggotaan segitiga

2. Fungsi Trapesium

Fungsi trapesium ini terdapat beberapa nilai x yang memiliki derajat keanggotaan sama dengan 1, yaitu ketika b ≤ x ≤ c. Tetapi derajat keanggotaan untuk a < x < b dan c < x < d memiliki karakteristik yang sama dengan fungsi segitiga. Persamaan fungsi keanggotaan trapesium adalah :

$$\mu(x;a,b,c,d) = \begin{cases} 0 & x < a \\ \dfrac{x-a}{b-a} & a \le x < b \\ 1 & b \le x \le c \\ \dfrac{d-x}{d-c} & c < x \le d \\ 0 & x > d \end{cases}$$

Persamaan tersebut direpresentasikan pada gambar 2.11 sebagai berikut :



Gambar 2. 11 grafik fungsi keanggotaan trapesium

1.13.4 Inferensi

Sistem kendali logika fuzzy disebut juga sistem inferensi fuzzy (*Fuzzy Inference System*) adalah sistem yang dapat melakukan penalaran dengan prinsip serupa seperti manusia melakukan penalaran dengan nalurinya. Terdapat beberapa jenis FIS yang dikenal yaitu Mamdani, Sugeno dan Tsukamoto. FIS yang paling mudah dimengerti, karena paling sesuai dengan naluri manusia adalah FIS mamdani. Dalam FIS Mamdani digunakan operasi MIN-MAX, pada proses implikasi digunakan operasi MIN dan operasi MAX dalam menentukan komposisi antar *rule* untuk mendapatkan nilai fuzzy baru yang akan difuzzifikasi [7].

1.13.5 *Deffuzyfication*

Deffuzyfication merupakan kebalikan *fuzzyfication*, yaitu pemetaan dari himpunan fuzzy ke himpunan tegas. Input dari proses *deffuzyfication* adalah suatu himpunan fuzzy yang diperoleh dari komposisi aturan-aturan fuzzy. Terdapat

beberapa metode *deffuzyfication*, salah satunya adalah metode *centroid*. Metode ini disebut juga sebagai *center of area* atau *center of gravity* [7]. Metode ini merupakan metode yang paling penting dan menarik diantara semua metode yang ada. Metode ini menghitung nilai *crisp* menggunakan rumus :

$$y^* = \frac{\int y\mu(y)dy}{\int \mu(y)\,dy}$$

1.14 Algoritma Genetika (AG)

Algoritma Genetika (AG) adalah algoritma pencarian yang didasarkan pada mekanisme seleksi alamiah dan genetika alamiah. Pada awalnya AG memang digunakan sebagai algoritma pencarian parameter-parameter optimal,tetapi dalam perkembangannya AG bisa diaplikasikan untuk berbagai masalah lain, seperti *learning*, peramalan, pemrograman otomatis, dan sebagainya. Pada bidang *soft computing*, AG banyak digunakan untuk mendapatkan nilai-nilai parameter yang optimal pada JST maupun sistem *fuzzy* [7].

Dalam Algoritma Genetika terdapat teori evolusi dimana sekumpulan solusi awal (individu) yang disebut populasi. Populasi awal akan ber-evolusi menjadi populasi baru melalui serangkaian iterasi. Pada akhir iterasi, AG mengembalikan satu anggota populasi yang terbaik sebagai solusi untuk masalah yang dihadapi. Pada setiap iterasi, proses evolusi yang terjadi adalah sebagai berikut :

1. Dua individu dipilih sebagai orang tua (*parent*) berdasarkan mekanisme tertentu. Kedua *parent* ini kemudian dikawinkan melalui operator *crossover* untuk mengasilkan dua individu anak atau *offspring*.
2. Dengan probabilitas tertentu, dua invidiu anak ini mungkin mengalami perubahan gen melalui operator *mutation*.
3. Suatu skema penggantian (*replacement scheme*) tertentu diterapkan sehingga mengahilkan populasi baru.
4. Proses initerus berulang sampai kondisi berhenti tertentu. Kondisi berhenti bisa berupa jumlah iterasi tertentu, waktu tertentu, atau ketika variansi

individu-individu dalam populasi tersebut sudah lebih kecil dari suatu nilai tertentu yang diinginkan.

Terdapat dua variasi algoritma genetika, yaitu *Steady State* dan *generational Replacement*. Pada AG yang berjenis Steady War,proses replacement dilakukan setiap kali dihasilkan dua *offspring* hasil *crosswover*. Offspring menggantikan kromosom yang nilai fitness-nya paling kecil. Dengan demikian,populasi baru yang dihasilkan selalu memiliki individu-individu yang lebih baik dibandingkan populasi lama [7].

1.14.1 Komponen-Komponen AG

AG terdiri dari delapan komponen, yaitu skema pengkodean,nilai *fitness*, seleksi *parent*, pindah silang (*crossover*), mutasi, elitism (untuk AG) berjenis *generational replacement*), penggantian populasi, dan kriteria penghentian. Banyak metode yang sangat bervariasi yang telah diusulkan untuk mengimplementasikan masing-masing komponen dasar tersebut. Masing-masing metode mempunyai kelebihan dan kekurangan. Suatu metode yang bagus menyelesaikan suatu masalah, belum tentu bagus untuk masalah yang lain, atau bahkan tidak bisa digunakan untuk masalah yang lain lagi [7].

1.14.2 Skema Pengkodean

Untuk dapat diproses menggunakan AG, suatu permasalahan harus dikonversi dulu ke dalam bentuk individu yang diawali oleh satu atau lebih kromosom dengan kode tertentu. Berbeda dengan teori genetika di dunia nyata yang merepresentasikan gen (buatan), secara umum, sebagai bilangan real, decimal, atau biner, yaitu :

1. *Real-number encoding*. Pada skema ini, nilai gen berada dalam interval [0,R], dimana R adalah bilangan real positif dan biasanya R = 1.
2. *Discrete decimal encoding*. Pada skema ini, nilai gen bisa berupa deretan bilangan bulat dalam interval [0,9].
3. *Binary encoding*. Setiap gen bisa berupa deretan nilai 0 atau 1.

Selain tiga macam skema pengkodean tersebut, sebenarnya masih banyak skema pengkodean lain yang bersifat khusus untuk suatu masalah tertentu.

1.14.3 Nilai *Fitness*

Pada AG, suatu individu dievaluasi berdasarkan suatu fungsi tertentu sebagai ukuran nilai *fitness*-nya. Pada masalah optimasi, jika solusi yang dicari adalah memaksimalkan sebuah fungsi h (dikenal sebagai masalah maksimasi), maka nilai *fitness* yang digunakan adalah nilai dari fungsi h tersebut, yakni f = h (dimana f adalah nilai *fitness*). Tetapi jika masalahnya adalah meminimilkan fungsi h (masalah minimasi), maka fungsi h tidak bisa digunakan secara langsung. Hal ini disebabkan AG menggunakan suatu aturan bahwa individu yang memiliki nilai *fitness* lebih tinggi akan memiliki kemampuan bertahar hidup lebih tinggi dari pada individu yang bernilai *fitness* rendah. Oleh karena itu, nilai *fitness* ntuk masalah minimasi adalah f = 1/h, yang artinya semakin kecil nilai h semakin besar nilai f. Tetapi, fungsi ini akan bermasalah jika h bernilai 0, yang mengakibatkan f bisa bernilai tak hingga. Untuk mengatasi masalah tersebut, h perlu ditambah dengan sebuah bilangan yang dianggap sangat kecil, sehingga formula fungsi *fitness*-nya menjadi $= \frac{1}{(h+a)}$. Dimana a adalah bilangan yang dianggap sangat kecil dan bervariasi sesuai dengan masalah yang akan diselesaikan [7].

1.14.4 Seleksi Individu

Seleksi bertujuan memberikan kesempatan reproduksi yang lebih besar bagi anggota populasi yang paling *fit*. Masing-masing kromosom dalam suatu wadah seleksi akan menerima probabilitas reproduksi yang tergantu pada nilai objektif dirinya sendiri terhadap nilai objektif dari semua kromosom dalam wadh seleksi tersebut [7]. Nilai *fitness* inilah yang nantinya akan digunakan pada tahap seleksi berikutnya,

Proses pemilihan dua individu sebagai *parent* biasanya dilakukan secara proporsional berdasarkan nilai-nilai *fitness*-nya. Salah satu metode seleksi yang umum digunakan adalah *roulette-wheel*. Cara kerja metode ini adalah sebagai berikut :

1. Dihitung nilai *invers* fitness (Q) dari masing-masing kromosom dengan persamaan :

$$Q[i] = \frac{1}{fitness}$$

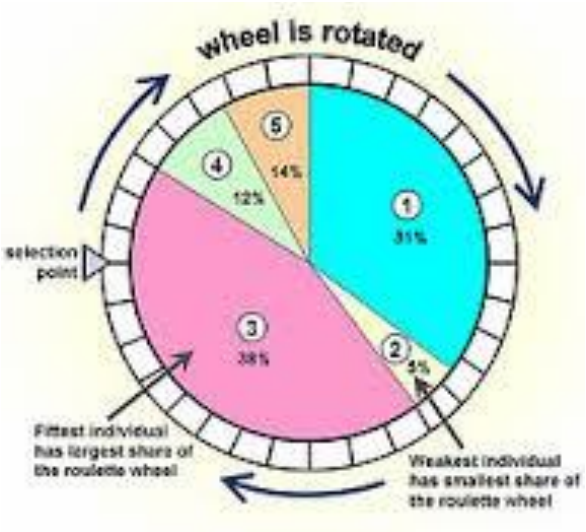
2. Dihitung nilai total dengan menjumlahkan semua nilai *inverse* fitness yang telah dicari.
3. Dihitung probabilitas (P) dari masing-masing individu dengan menggunakan persamaan :

$$p[i] = \frac{Q[i]}{Qtotal}$$

4. Dari probabilitas tersebut, akan mulai terlihat kromosom yang mempunyai fitness paling kecil mempunyai probabilitas untuk terpilih pada generasi selanjutnya lebih besar dari individu yang lainnya.
5. Sebelum dilakukan proses seleksi menggunakan *roulette-wheel* terlebih dahulu di cari nilai kumulatif probabilitas (C)

$$C[i] = C[n] + P[n+1]$$

6. Setelah didapatkan nilai kumulatif probabilitas (C) dimana masing-masing individu menempati potongan lingkaran berdasarkan nilai kumulatif probabilitasnya sesuai dengan gambar 2.18.



Gambar 2. 12 Contoh Seleksi *parent* menggunakan metode *roulette-wheel*.

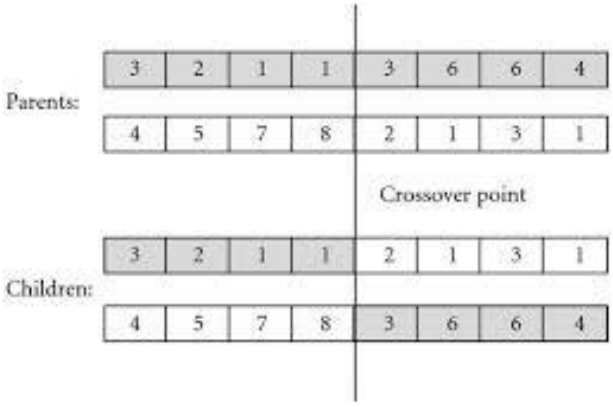
1.14.5 Crossover

Pada proses *crossover* terjadi kombinasi pewarisan gen-gen dari induknya, gen-gen dari kedua induk dapat bercampur sehingga dihasilkan susunan kromosom yang baru. Dari roses tersebut dihasilkan variasi genetika.

Dengan suatu skema tertentu, dua individu dipilih sebagai *parent*. Setelah didapatkan dua individu *parent*, selanjutnya ditenetukan titik pindah silah secara acak. Jika diasumsikan L adalah panjang kromosom, maka titik *crossover* berada diantara 1 sampai L-1. Kemudia beberapa bagian dari dua kromosom ditukar pada titik *crossover* yang dipilih. Titik *crossover* adalah titik terjadinya pertukaran gen antara dua individu *parent*. Pertukaran tersebut akan menghasilkan dua individu anak [7]. Bagaimanapun, operasi *crossover* tidak selamanya berhasil. Peluang keberhasilan operasi *crossover* dinyatakan dengan probabilitas *crossover* atau p_c. Terdapat tiga skema *crossover* yang biasa digunakan, yaitu :

1. *Single-point crossover*

   crossover ini merupakan skema *crossover* yang paling sederhana. Titik *crossover* hanya satu dengan posisi yang dibangkitkan secara acak.

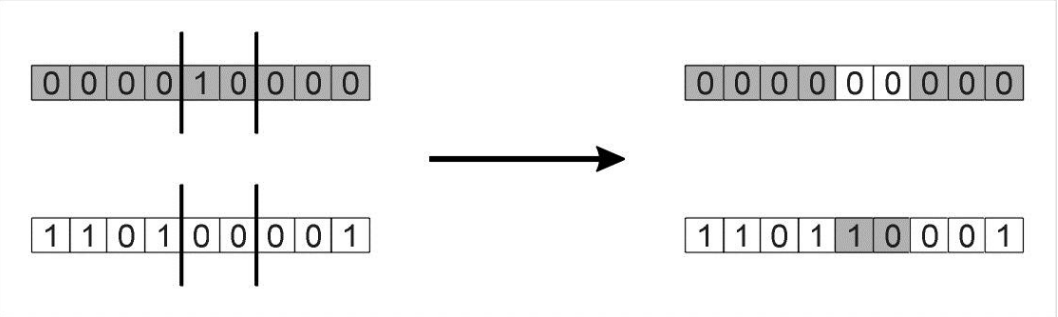


Gambar 2. 13 Single-point Crossover

2. M*ulti-point crossover*

   Pada suatu masalah tertentu dimana suatu individu terdiri dari sangat banyak gen (misalkan 10000 gen), mungkin kita memerlukan lebih dari

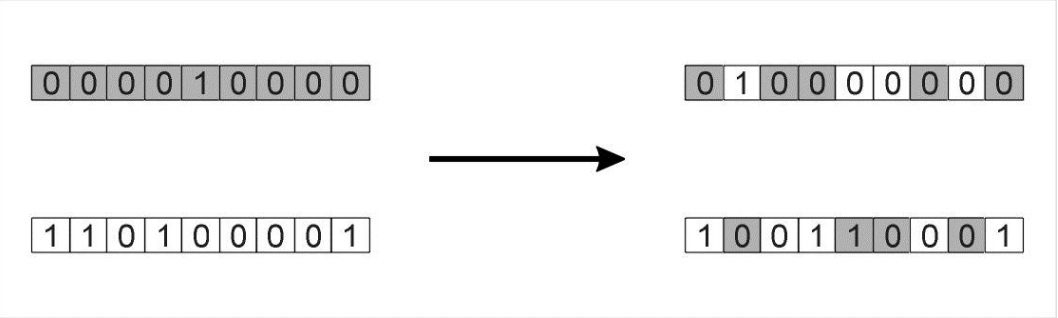
satu titik *crossover*. Banyak titik *crossover* ini akan mempengaruhi pola pertukaran gen-gen antar individu *parent*.



Gambar 2. 14 Multi-point Crossover

3. *Uniform crossover*

   Dengan operasi *crossover* pola seragam maka komposisi gen-gen tertentu pada suatu individu dapat dipertahankan. Hal ini akam memudahkan proses pencarian solusi.



Gambar 2. 15 Uniform Crossover

1.14.6 Mutasi

Mutasi diperlukan untum mengembalikan informasi bit yang hilang akibat *crossover*. Mutasi diterapkan dengan probabilitas yang sangat kecil. Jika mutasi dilakukan terlalu sering, maka akan menghasilkan individu yang lemah karena konfigurasi gen pada individu yang unggul akan dirusak [7]. Berdasarkan bagian yang termutasi, proses mutasi dapat dibedakan atas tiga bagian :

1. Mutasi pada tingkat kromosom : semua gen dalam kromosom berubah
2. Mutasi pada tingkat gen : semua bit dalam satu gen akan berubah.
3. Mutas pada tingkat hanya satu bit yang berubah.

Prosedur mutasi sangatlah sederhana. Untuk semua gen yang ada, jika bilangan random yang dibangkitkan kurang dari probabilitas mutasi yang ditentukan maka ubah gen tersebut menjadi nilai kebalikannya (dalam *binary encoding*, 0 diubah 1, dan 1 diubah 0). Biasanya probabilitas mutasi diset sebagai 1/n, dimana n adalah jumlah gen dalam kromosom. Dengan probabilitas mutasi sebesar ini berarti mutasi hanya terjadi pada sekitar satu gen saja. Nilai probabilitas mutasi yang besar kaan tidak menguntukan dan cenderung merusak kromosom yang sudah bagus [7] .

1.14.7 Elitisme

Karena seleksi dilakukan secara acak, maka tidak ada jaminan bahwa suatu individu bernilai *fitness* tertinggi akan selalu terpilih. Kalaupun individu bernilai *fitness* tertinggi terpilih, mungkin saja individu tersebut akan rusak karena proses *crossover*. Untuk menjaga agar individu bernilai *fintess* tertinggi tersebut tidak hilang selama evolusi, perlu dibuat satu atau dua yang bernilai *fitness* yang sama dengan *fitness* tertinggi. Prosedur ini dikenal sebagai elitism. Prosedur ini hanya digunakan pada AG berjenis *generational replacement* [7].

1.14.8 Penggantian Populasi

Pada AG berjenis *generational replacement*, N individu pada suatu generasi digantikan sekaligus oleh N individu baru hasil *crossover* dan mutasi. Untuk mempertahankan individu terbaik, diperlukan skema elitisme seperti dijelaskan diatas.

Bagaimana skema penggantian populasi pada AG berjenis *steady-state* ? Banyak prosedur yang bisa digunakan , diantaranya adalah

1. Selalu mengganti individu yang memiliki nilai *fitness* terkecil.
2. Selalu mengganti individu yang paling tua.
3. Membandingkan anak dengan kedua *parent*. Apabila anak memiliki nilai *fitness* yang lebih baik daripada salah satu kedua *parent*, maka anak menggantikan *parent* yang memiliki nilai *fitness* terendah. Skema ini dapat menjaga keanekaragaman yang lebih baik dibanding skema sebelumnya.

Dengan mengganti *parent* yang memiliki *Hamming Distance* (beda bit) yang lebih sedikit, penggantian tersebut dapat diharapkan tidak cepat menghilangkan keragaman.

1.14.9 Kriteria Penghentian

Terdapat berbagai macam kriteria penghentian yang bisa digunakan, tiga diantaranya adalah :

1. Memberi batasan jumlah iterasi. Apabila batas iterasi tersebut dicapai, iterasi dihentikan dan laporkan individu bernilai *fitness* tertinggi sebagai solusi terbaik.
2. Memberikan batasan waktu proses AG. Kriteria ini digunakan pada sistem waktu nyata, dimana solusi harus ditemukan paling lama, misalkan 3 menit. Dengan demikian AG bisa dihentikan ketika proses sudah berlangsung selama hamper 3 menit.
3. Menghitung kegagalan penggantian anggota populasi yang terjadi berurutan sampai jumlah tertenu.

1.15 Basis Data

Basis data adalah kumpulan informasi yang disimpan dalam komputer secara sistematik sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut. Perangkat lunak yang digunakan untuk mengelola dan memanggil *query* basis data disebut sistem manajemen basis data [12].

Konsep dasar dari basis data adalah kumpulan dari catatan-catatan, atau potongan dari pengetahuan Sebuah basis data memiliki penjelasan terstruktur dari jenis fakta yang tersimpan di dalamnya. Skema menggambarkan obyek yang diwakili suatu basis data, dan hubungan diantara obyek tersebut. Ada abanyak cara untuk mengorganisasi skema, atau memodelkan struktur basis data ini dikenal sebagai model basis data atau model data. Model yang umum digunakan sekarang adalah model relasional, yang menurut istilah layanan mewakili semua informasi dalam bentuk tabel-tabel yang salin berhubungan dimana setiap tabel

terdiri dari basis dan kolom. Dalam model ini, hubungan antar tabel diwakili dengan menggunakan nilai yang sama antar tabel. Model lain seperti model hierarkis dan model jaringan menggunakan cara yang lebih eksplisit untuk mewakili hubungan antar tabel. Salah satu perangkat lunak basis data yang menggunakan model relasional adalah MySQL [12].

1.15.1 MySQL

MySQL merupakan perangkat lunak yang tergolong sebagai DBMS (*Database Management System*) yang bersifat *Open Source*. *Open Source* menyatakan bahwa perangkat lunak ini dilengkapi dengan *source code*, selain itu tentu saja bentuk *executable*-nya atau kode yang dapat dijalankan secara langsung dalam sistem operasi, dan bisa diperoleh dengan cara mengunduh di internet secara gratis [12].

Sebagai perangkat lunak DBMS, MySQL meiliki sejumlah fitur seperti yang dijelaskan di bawah ini.

4. *Multiplatform*

   MySQL tersedia pada beberapa *platform* (Windows, Linux, Unix, dan lain-lain).
5. Andal, Cepat dan Mudah Digunakan

   MySQL tergolong sebagai *database server* (*server* yang melayani permintaan terhadap *database*) yang andal, dapat menangani *database* yang besar dengan kecepatan tinggi, mendukung banyak sekali fungsi untuk mengakses *database*, dan sekaligus mudah untuk digunakan. Berbagai *tool* pendukung juga tersedia (walaupun dibuat oleh pihak lain). Perlu diketahu, MySQL dapat menagani sebuah tabel berukuran dalam *terabyte* (1 *terabyte* = 1024 *gigabyte*). Namun, ukuran yang sesungguhnya sangat bergantung pada batasan sistem operasi. Sebagai contoh, pada sistem Solaris 9/10, batasan ukuran sebesar 16 *terabyte*.
6. Jaminan keamanan akses

MySQL mendukung pengamanan database dengan berbagai kriteria pengaksesan. Sebagai gambaran, dimungkinkan untuk mengatur user tertentu agar bisa mengakses data yang bersifat rahasia, sedangkan user lain tidak boleh. ySQL, juga mendukung konektivitas ke berbagai perangkat lunak. Sebagai contoh, dengan menggunakan ODBC (Open Database Connectivity), *database* yang ditangani MySQL, dapat diakses melalui program yang dibuat dengan *Visual Basic*. MySQL, juga mendukung program *client* yang berbasis java untuk berkomunikasi dengan database MySQL melalui JDBC (*Java Database Connectivity*). MySQL juga bisa diakses melalui aplikasi berbasis Web, misalnya dengan menggunakan PHP.

7. Dukungan SQL

   Seperti tersirat dalam namanya, MySQL endukung perintah SSQL. (*Structured Query Language*). Sebgaiana diketahui, SQL merupakan standar dalam pengaksesan database relasional. Pengetahuan akan SQL. Akan memudahkan siapa pun untuk menggunakan MySQL.

1.16 UML

UML adalah kependekatan dari *Unified Modeling Language* yang merupakan suatu cara untuk menyelesaikan suatu cara masalah dengan mendeskripsikannya yang telah menjadi standar dalam dunia industry untuk memvisualisasikan, merancang dan mendokumentasikan sistem perangkat lunak. Dengan menggunakan UML kita dapat membangun model untuk segala bentuk dan jenis aplikasi perangkat lunak, yang mana aplikasi yang dibangun dapat berjalan pada perangkat lunak, yang mana aplikasi yang dibangun dapat berjalan pada perangkat lunak dengan sistem operasi dan jaringan apapun. Selain itu, dengan menggunakan UML, *programmer* dapat mengerti, memahami dengan jelas maksud, tujuan dan ara rancangan sistem, sehingga sistem perangkat lunak yang dirancang dapat dibangun dengan bahasa pemrograman apapun. Namun karena UML merupakan suatu model pengembangan sistem perangkat lunak yang berbasis orientasi objek sehingga menggunakan bentu *class* dan *operation* dalam

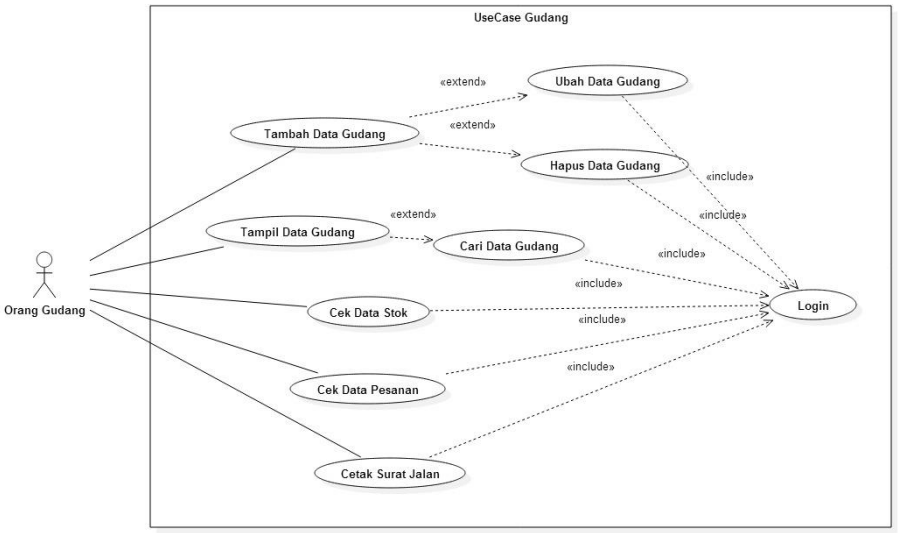
konsep dasarnya, maka lebih cocok dalam bahasa pemrograman yang digunakan berbasis OOP (*Object Oriented Programming*) seperti C,C++, Java, VB dan lain sebagainya [11].

Secara lebih mendalam, UML lebih dari sekedar sebuah *standard* dan penemuan dari suatu notasi-notasi yang disatukan, tetapi juga berisi konsep-konsep baru yang menarik yang tidak ditemukan sceara umum dalam komunitas *object oriented*. Dalam membangun suatu model perangkat lunak dengan UML, digunakan bentuk-bentuk diagram atau simbol yang merepresentasikan elemen-elemen dalam sistem. Bentuk diagram yang digunakan untuk merepresentasikannya adalah sebagai berikut :

1. *Use-case Diagram*

Use-case diagram merupakan suatu bentuk diagram yang menggambarkan fungsi-fungsi yang diharapkan dari sebuah sistem yang dirancang. Sebuah *use-case* akan merepesentasikan sebuah interaksi antara pelaku dengan sistem [11].

Use-case diagram yang digunakan dalam merancang suatu sistem dapat sangat membantu pada saat menyusun *requirement* sebuah sistem, mengomunikasikannya dengan pengguna, dan meracangan pengujian untuk semua fitur yang terdapat dalam sistem. Dalam suatu sistem aplikasi *database*, *use-case diagram* sangat membantu *requirement* apa saja yang diperlukan [11].



Gambar 2. 16 Contoh *use case diagram*

2. *Class Diagram*

Sebuah *Class diagram* menunjukan struktur yang statis dari beberapa *class* dalam suatu sistem. Beberapa *Class* merepresentasikan suatu keadaan (atribut/properti) dan yang akan dikerjakan oleh sistem [11]. *Class* memiliki tiga area pokok :

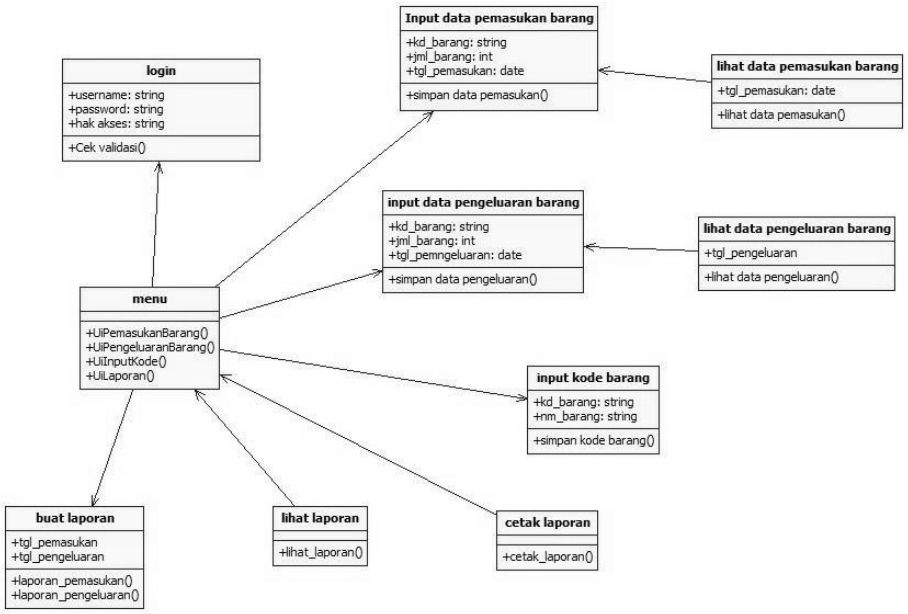
1. Nama
2. Atribut
3. Metode

Atribut dan metode dalam *class diagram* dapat memiliki salah satu sifat sepert berikut di bawah ini :

1. *Private*, hanya dapat diakses oleh *class* itu sendiri.
2. *Protected*, hanya dapat diakses oleh *class* itu sendiri dan turunannya.
3. *Public*, dapat diakses oleh *class* lain meski tidak bersangkutan dengan *class* tersebut.

Class dapat direpresentasikan dalam sebuah *interface* atau sebaliknya merupaka implementasi dari sebuah *interface* yang berupa *class* abstrak yang

hanya tidak memiliki *attribute* dan hanya memiliki metode [11]. Berikut merupaka bentuk *class diagram* secara umum .

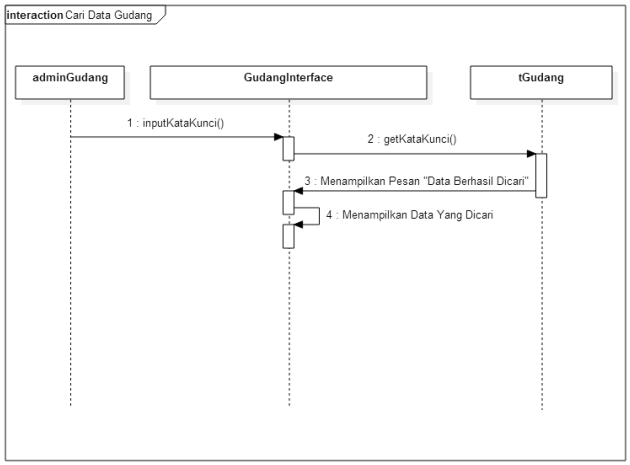


Gambar 2. 17 Contoh *class diagram*

3. *Sequence Diagram*

Sequence diagram merupakan diagram yang menggambarkan kolaborasi yang dinamis antara objek satu dengan yang lain. Kolaborasi ini ditunjukan dengan adanya interaksi antar objek di dalam dan di sekita sistem yang berupa pesan atau instruksi yang berurutan [11].

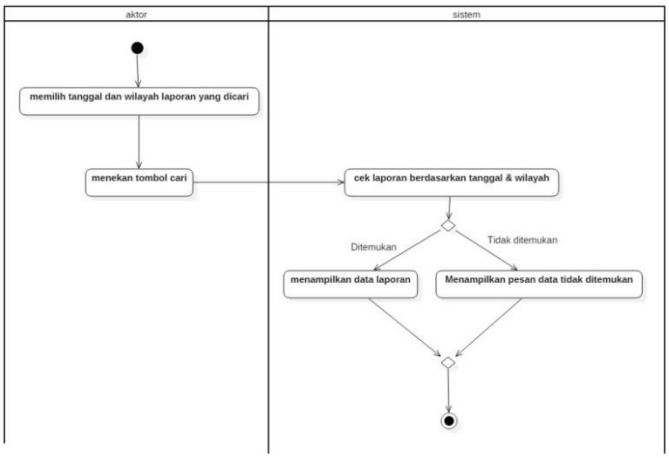
Sequence diagram umumnya digunakan untuk menggambarkan suatu scenario atau urutan langkah-langkah yang dilakukan oleh pengguna maupun sistem yang merupakan respon dari sebuah kejadian untuk mendapatkan hasil [11].



Gambar 2. 18 Contoh *Sequence diagram*

4. *Activiy Diagram*

Sebuah *Activity diagram* menunjukan suatu alur kegiatan secara berurutan. *Activiy diagram* digunakan untuk mendeskripsikan kegiatan-kegiatan dalam sebuah operasi meskipun juga dapat digunakan untuk mendeskripsikan alur kegiatan yang lainnya seperti *use-case* atau suatu interaksi [11].



Gambar 2. 19 Contoh *Activity diagram*

1.17 Metode Pengujian Black Box

Metode pengujian adalah cara atua teknik untuk menguji perangkat lunak, mempunyai mekanisme untuk menentukan data uji yang dapat menguji perangkat lunak secara lengkap dan mempunyai kemungkinan tinggi untuk menemukan

kesalahan. Pengujian *black box* merupakan pendekatan komplementer dari teknik *white box*, karena pengujian *black box* diharapkan mampu mengungkap kelas kelas kesalahan yang lebih luas dibandingkan teknik *white box*. Pengujian *black box* berfokus pada pengujian persyaratan fungsional perangkat lunak, untuk mendapatkan serangkaian kondisi *input* yang sesuai dengan persyaratan fungional suatu program[13].

Pengujian *black box* adalah pengujian aspek fundamental sistem tanpa memperhatikan struktur logika internal perangkat lunak. Metode ini digunakan untuk mengetahui apakah perangkat lunak berfungsi dengan benar. Pengujian *black box* merupakan metode perancangan data uji yang didasarkan pada spesifikasi perangkat lunak. Data uji dibangkintkan, dieksekusi pada perangkat lunak dan kemudian keluaran dari perangkat lunak dicek apakah telah sesuai dengan yang diharapkan[13].

Pengujian *black box* berusaha menemukan kesalahan dalam kategori :

1. Fungi-fungsi yang tidak benar atau hilang.
2. Kesalahan *interface*.
3. Kesalahan dalam struktur data atu akses database eksternal
4. Kesalahan kinerja
5. Inisialisasi dan kesalahan terminasi.