

BAB II

TINJAUAN PUSTAKA DAN DASAR TEORI

Pada penelitian ini literatur diambil dari penelitian yang telah dilakukan sebelumnya yang sejalan dengan analisis dan metode yang akan digunakan. Berikut adalah penelitian sebelumnya tentang analisis *cluster*, metode *k-means* dan evaluasi *cluster* dengan metode *confusion matrix*.

2.1 Tinjauan Pustaka

Penelitian sebelumnya menggunakan dilakukan analisis cluster dengan metode *kmeans clustering* dan evaluasi dengan metode *confusion matrix* untuk mendapatkan intrusi yang efektif dalam mendeteksi dan mencegah gangguan pada lalu lintas jaringan dan hasil dari penelitian tersebut adalah Sistem deteksi intrusi menggunakan algoritma *kmeans clustering* mampu mendeteksi berbagai jenis gangguan, sekaligus mempertahankan rasio positif palsu yang rendah dan kinerja yang sangat baik dengan akurasi lebih dari 90% dalam deteksi intrusi (Nalavade dan Mehsram, 2014).

Penelitian analisis *clustering* pada kasus *stunting* telah dilakukan dengan metode *k-means clustering*. Penelitian bertujuan untuk identifikasi bayi yang menjadi anggota kelompok *severe stunting*, *stunting*, *normal abnormal*. Hasil penelitian tersebut menyatakan ada 48% bayi yang mengalami *severe stunting*, 22% bayi mengalami *stunting*, 28% bayi normal dan 2.5% bayi mengalami abnormal. Proses analisis menghasilkan 70% nilai dari akurasi *k-means clustering* (Sudarmadji dkk., 2019).

Penelitian menggunakan *k-means clustering* berikutnya adalah penelitian yang telah dilakukan untuk mengelompokkan provinsi di Indonesia berdasarkan 4 variabel yaitu kepadatan penduduk, tingkat partisipasi sekolah, indeks pembangunan manusia dan tingkat pengangguran terbuka. Hasil dari penelitian tersebut adalah provinsi di Indonesia dibagi menjadi 5 kelompok yang memiliki 5 pusat *cluster* dan

terletak pada Provinsi Sumatera Selatan, Provinsi Lampung, Provinsi DKI Jakarta, Provinsi Jawa Tengah dan Provinsi Kalimantan Barat (Ahmar dkk., 2018).

Penelitian tentang sistem informasi evaluasi menggunakan *k-means clustering* telah dilakukan dengan menilai kinerja akademik mahasiswa pada salah satu universitas swasta di Nigeria. Penelitian tersebut bertujuan membantu pihak akademik dalam membuat keputusan yang efektif untuk perencanaan akademisi kedepannya. Hasil dari penelitian tersebut adalah didapatkan 9 program studi yang memiliki nilai kinerja akademik mahasiswa dengan interpretasi baik dan total mahasiswa sebanyak 79 orang (Oyelade dkk., 2010).

Penelitian tentang evaluasi dengan *k-means clustering* berikutnya adalah untuk meningkatkan produktivitas kinerja dengan 544 data kinerja tahunan pekerja kontrak. Dalam penelitian ini jumlah cluster yang ditentukan adalah 5 *cluster*. Dari perhitungan euclidean distance, hasil *k-mean clustering* didapat cluster 0 sebanyak 38, cluster 1 sebanyak 473, cluster 2 sebanyak 130, cluster 3 sebanyak 26 dan dari cluster 4 sebanyak 3. Hasil implementasi *k-means clustering* digunakan sebagai pendukung evaluasi kualitas kinerja (Ardiada dkk., 2018).

2.2 Dasar Teori

2.2.1 *Stunting*

Salah satu bentuk kekurangan gizi kronis yang belangsung dalam waktu yang cukup lama akibat kegagalan dalam memenuhi nutrisi pada balita disebut *stunting* (Dake dkk., 2019). Kegagalan pertumbuhan dapat terjadi sedini mungkin sejak kehamilan trimester kedua dan menghasilkan pengurangan proporsional pertumbuhan tulang dan jaringan lunak (Branca dkk., 2002). *Stunting* merupakan salah satu gangguan gizi kronik yang dilihat dari TB/U sebagai indikator untuk mendapatkan hasil ukuran status gizi bayi. Kerangka kerja WHO memberikan gambaran tentang penyebab *stunting* dan mengklasifikasikan mereka menjadi empat faktor proksimal utama seperti faktor rumah tangga dan keluarga, pemberian

makanan pendamping yang tidak memadai, praktik menyusui yang tidak cukup, dan infeksi atau penyakit (Uwiringiyimana dkk., 2019).

Status gizi anak-anak kurang dari lima tahun diukur secara antropometrik. Panjang diukur untuk anak-anak berusia kurang dari dua tahun dan tinggi untuk anak-anak yang berusia di atas dua tahun. Panjangnya diukur menggunakan stadiometer kayu ke 0,1 cm terdekat dan tinggi diukur menggunakan pita *microtoice* ke 0,1 cm terdekat. Status pengukuran tinggi untuk usia adalah dinyatakan dalam satuan standar deviasi (SD) (skor-Z) dari median populasi referensi. *Stunting* telah menggambarkan status pertumbuhan dan perkembangan kurang gizi kronis sejak awal kehidupan. Kondisi ini akan menurunkan perkembangan kemampuan kognitif, anak mudah sakit dan daya saing rendah. *Stunting* dapat terjadi pada waktu yang berbeda dalam perkembangan anak (Ramli dkk., 2009).

Pemerintah Indonesia membentuk kebijakan dan program sebagai upaya dalam menurunkan prevalensi angka *stunting*. Salah satu program yang dibentuk oleh Pemerintah Indonesia untuk percepatan penanganan masalah *stunting* adalah program Intervensi gizi spesifik. Intervensi gizi spesifik adalah praktek penanganan masalah *stunting* yang memiliki dampak secara langsung untuk ibu dan anak. Kegiatan intervensi gizi spesifik umumnya dilakukan pada sektor kesehatan. Intervensi ini juga bersifat jangka pendek dan hasilnya dapat dicatat dalam waktu relatif pendek Intervensi gizi spesifik memiliki target kegiatan yang ditujukan pada 1000 hari pertama kehidupan (HPK) dan berkontribusi sebesar 30% dalam penurunan *stunting*. (TNP2K, 2017).

Rencana strategis (Renstra) kementerian Kesehatan tahun 2016-2018 telah membuat rancangan target untuk 6 indikator yang digunakan dalam beberapa penilaian indikator kinerja kegiatan (IKK) pembinaan gizi masyarakat. Seluruh indikator tersebut merupakan program intervensi gizi spesifik yang dijalankan dalam upaya menyelesaikan masalah gizi masyarakat termasuk kasus *stunting* (Direktorat Gizi Masyarakat, 2019). Indikator tersebut terdiri dari :

- a. Persentase ibu hamil kekurangan energi kronis (KEK) mendapatkan pemberian makanan tambahan (PMT).

Riset Kesehatan dasar daerah tahun 2018 telah menunjukkan bobot ibu hamil KEK (15-49 tahun) sebesar 17,3% terhadap gangguan pertumbuhan dan perkembangan janin yang menjadi faktor terbentuknya generasi *stunted* (Badan Penelitian dan Pengembangan Kesehatan, 2018). Persentase ibu hamil KEK mendapatkan PMT dihitung dengan persamaan (2.1) sebagai berikut.

$$\frac{\sum_i^n A_{(PMT)}}{\sum_i^n A} \times 100\% \quad (2.1)$$

Pada persamaan (2.1) didapatkan bahwa $A_{(PMT)}$ adalah ibu hamil KEK mendapatkan PMT di suatu wilayah pada periode tertentu dan A adalah ibu hamil KEK yang ada suatu wilayah pada periode tertentu.

- b. Persentase ibu hamil mendapatkan tablet tambah darah (TTD)

Masa kehamilan membuat peningkatan pada kebutuhan zat gizi pada wanita hamil sebesar 25% dibandingkan dengan wanita tidak hamil, sehingga untuk mencegah terjadinya anemia dibutuhkan tablet tambah darah (TTD) yang mengandung Fe dan asam folat yang dicanangkan minimal 90 tablet. Rumus untuk menghitung persentase ibu hamil mendapat TTD 90 tablet didapat dengan persamaan (2.2) sebagai berikut.

$$\frac{\sum_i^n B_{(TTD)}}{\sum_i^n B} \times 100\% \quad (2.2)$$

Pada persamaan (2.2) didapatkan bahwa $B_{(TTD)}$ adalah ibu hamil mendapatkan TTD 90 tablet di suatu wilayah pada periode tertentu dan B adalah ibu hamil yang ada di suatu wilayah pada periode tertentu.

c. Persentase bayi usia kurang dari 6 bulan yang mendapatkan ASI eksklusif

Pemberian ASI telah dituangkan dalam peraturan pemerintah nomor 33 tahun 2012 tentang pemberian air susu ibu (ASI) eksklusif. Peraturan tersebut menjelaskan bahwa setiap ibu yang melahirkan harus memberi ASI eksklusif kepada bayi yang dilahirkannya, untuk menjadinkan hak bayi untuk mendapat ASI eksklusif sejak dilahirkan hingga usia 6 bulan. Berikut rumus yang digunakan untuk mencari persentase bayi usia kurang dari 6 bulan yang dapat ASI eksklusif dapat dilihat pada persamaan (2.3).

$$\frac{\sum_i^n C_{(Eks)}}{\sum_i^n C} \times 100\% \quad (2.3)$$

Pada persamaan (2.3) didapatkan bahwa $C_{(Eks)}$ adalah bayi 0-5 bulan 29 hari yang mendapatkan ASI Eksklusif dan tercatat register pencatatan KMS di suatu wilayah pada periode tertentu dan C adalah bayi 0-5 bulan 29 hari yang tercatat register pencatatan KMS di suatu wilayah pada periode tertentu

d. Persentase bayi baru lahir mendapatkan inisiasi menyusui dini (IMD)

Inisiasi menyusui dini adalah proses menyusui dimulai sesaat setelah bayi lahir dengan cara kontak kulit antara bayi dan ibu yang berlangsung

selama minimal satu jam. Persentase bayi baru lahir mendapatkan inisiasi menyusui dini (IMD) dapat dihitung dengan persamaan (2.4) berikut ini.

$$\frac{\sum_i^n D_{(IMD)}}{\sum_i^n D} \times 100\% \quad (2.4)$$

Pada persamaan (2.4) didapatkan bahwa $D_{(IMD)}$ adalah bayi lahir mendapatkan IMD di suatu wilayah pada periode tertentu dan D adalah bayi baru lahir di suatu wilayah pada periode tertentu

e. Persentase balita kurus yang mendapatkan PMT

ASI eksklusif yang diberikan sampai usia 6 bulan akan dilanjutkan dengan makanan pendamping ASI untuk memenuhi kebutuhan zat gizi. Kondisi zat gizi yang tak terpenuhi akan menyebabkan kekurangan gizi yang jika terjadi terus menerus akan mengakibatkan terjadinya gizi kurang bahkan buruk. Persentase balita kurus yang mendapatkan PMT dapat dihitung dengan persamaan (2.5) sebagai berikut.

$$\frac{\sum_i^n E_{(PMT)}}{\sum_i^n E} \times 100\% \quad (2.5)$$

Pada persamaan (2.5) didapatkan bahwa $E_{(PMT)}$ adalah balita kurus mendapatkan PMT di suatu wilayah pada periode tertentu dan E adalah balita kurus yang ada suatu wilayah pada periode tertentu.

f. Persentase Remaja Puteri mendapatkan TTD

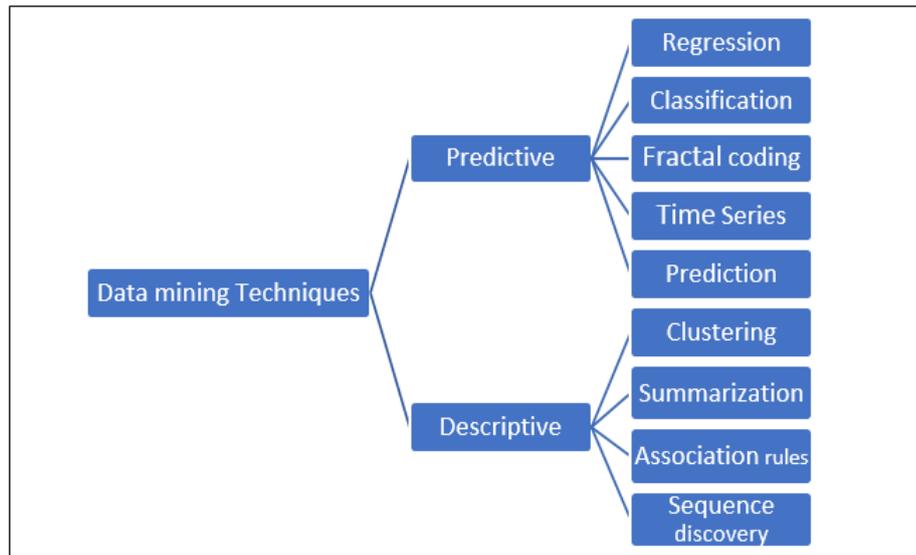
Remaja puteri yang mendapatkan TTD berusia 12-18 tahun dan merupakan siswi yang bersekolah di SMP/SMA atau sederajat. Selanjutnya diberikan secara berkala setiap 1 tablet seminggu. Total selama setahun remaja puteri akan menerima 52 TTD. Adapun menghitung persentase remaja puteri mendapatkan TTD dapat dilakukan dengan persamaan (2.6) berikut ini.

$$\frac{\sum_i^n F_{(TTD)}}{\sum_i^n F} \times 100\% \quad (2.6)$$

Pada persamaan (2.2) didapatkan bahwa $F_{(TTD)}$ adalah remaja puteri mendapatkan TTD di suatu wilayah pada periode tertentu dan F adalah remaja puteri yang ada di suatu wilayah pada periode tertentu.

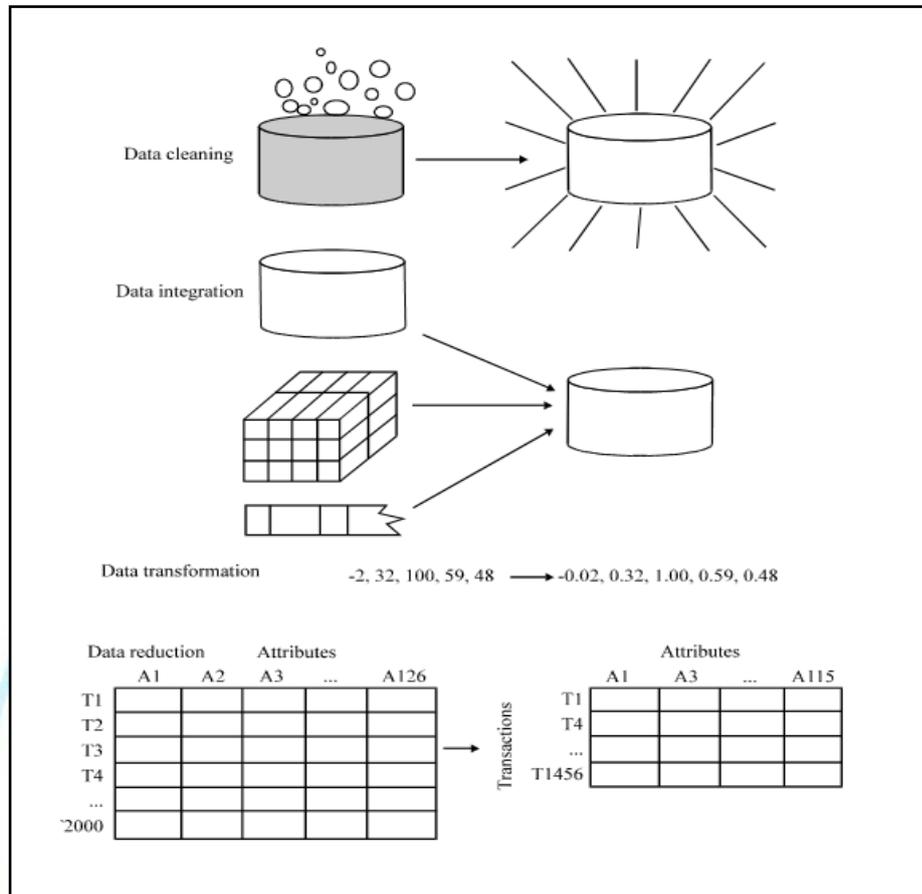
2.2.2 Data Mining

Data mining adalah suatu proses yang dilakukan untuk menemukan pola dan pengetahuan yang terbentuk dari data yang ada. *Data mining* merupakan teknik yang telah banyak diterapkan dalam dunia medis. *Data mining* terdiri dari aturan asosiasi untuk mendapatkan pola, prediksi, klasifikasi dan pengelompokkan. Sampai saat ini ada banyak peneliti tentang *data mining* dalam bidang kesehatan dan sistem terus dikembangkan untuk mendiagnosis berbagai penyakit dengan nilai akurat yang tinggi serta monitoring kesehatan jarak jauh (Lashari dkk., 2017). Struktur teknik *data mining* dapat dilihat pada gambar 2.1.



Gambar 2.1. Teknik *data mining* (Lashari dkk., 2017)

Sebelum melakukan teknik *clustering* pada *data mining* maka akan dilakukan *preprocessing data* dengan tugas yang terdiri dari persiapan dan transformasi menjadi bentuk yang siap untuk dilakukan *clustering*. Tugas yang dilakukan seperti pembersihan data, integrasi, transformasi dan pengurangan. Ada sejumlah alat yang digunakan untuk melakukan *preprocessing data* termasuk pengambilan sampel, memilih subset representatif dari populasi besar, transformasi yang dapat memanipulasi data mentah untuk mendapatkan input dari data mentah, menghilangkan noise dari data, normalisasi yang dilakukan untuk mengatur data agar lebih efisien dan ekstraksi fitur untuk mengeluarkan atau mengurangi data dalam beberapa konteks tertentu (Tamilselvi dkk., 2015). Berikut tahapan *preprocessing data* yang ditunjukkan pada gambar 2.2.



Gambar 2.2 Tahapan *preprocessing data* (Alasadi dkk., 2017)

2.2.3 Asumsi Cluster

Asumsi *cluster* dibutuhkan untuk mengetahui apakah sebuah data layak untuk dilakukan *clustering*. Asumsi *cluster* memiliki 2 teknik (Hair dkk., 2010). Adapun teknik *clustering* pada asumsi analisis *cluster* adalah sebagai berikut :

- a. Sampel dapat mewakili populasi (sampel representatif)

Pada dasarnya tidak ada ketentuan dan aturan jumlah sampel yang representatif, tetapi jumlah sampel yang cukup besar juga diperlukan dalam proses pengelompokan agar dapat berjalan dengan benar. Pengujian sampel dapat dilakukan dengan uji Kaiser Mayer Olkin (KMO). KMO digunakan sebagian syarat

dalam melihat kecukupan suatu sampel dan mencerminkan pentingnya pengambilan sampel variabel yang tepat untuk mendeteksi setiap struktur dasar (Hill, 2011). Uji KMO mempunyai nilai 0 sampai dengan 1. Apabila nilai KMO > 0,5 maka sampel dapat mewakili populasi dan disebut sampel representatif. Berikut perhitungan KMO dilihat persamaan dibawah ini .

$$K = \frac{\sum_i \sum_{ii} r_i^2}{\sum_i \sum_{ii} r_i^2 - \sum_i \sum_{ii} q_i^2} \quad (i \neq 1) \quad (2.7)$$

Pada persamaan (2.7) q_{ii} = adalah element dari baris i dan kolom

b. Multikolinearitas

Data yang digunakan pada proses *cluster* tidak boleh memiliki korelasi atau disebut dengan non multikolinearitas karena analisis *cluster* pada setiap variabel diberi bobot yang sama dalam perhitungan euclidean distance. Jika beberapa variabel memiliki korelasi maka akan menyebabkan pembobotan tidak sama sehingga dapat mempengaruhi hasil analisis. Non Multikolinearitas dapat dihitung dengan nilai *Variance Inflation Factor* (VIF) dengan rumus :

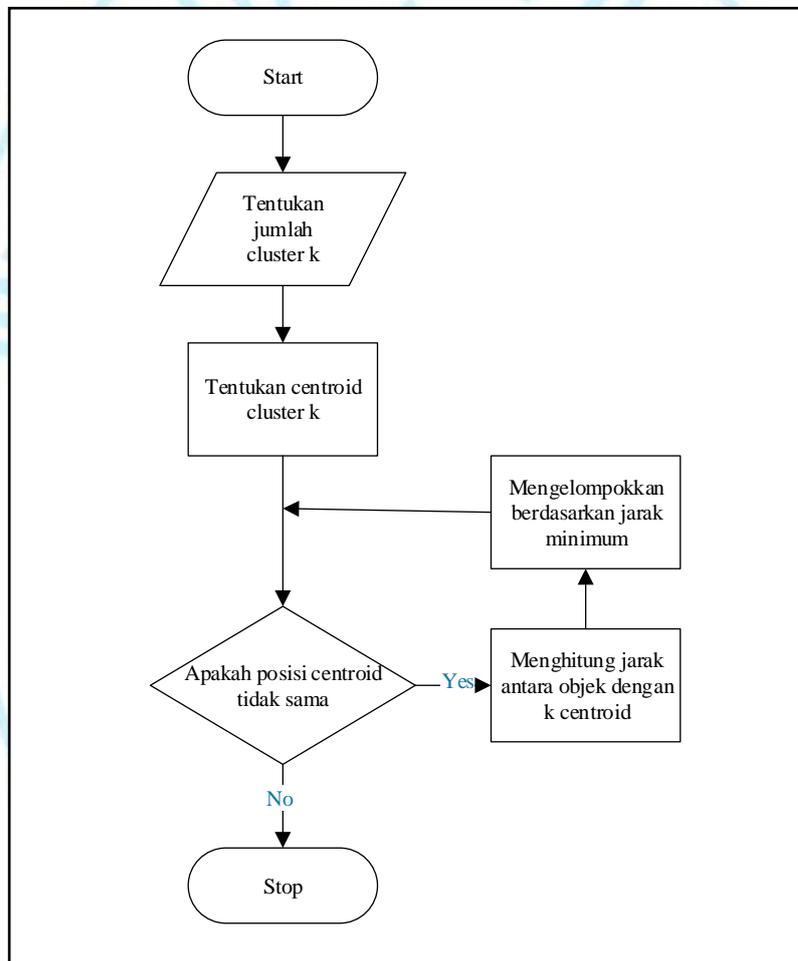
$$VIF_j = \frac{1}{1-R_j^2} \quad (2.8)$$

Pada persamaan (2.8) tersebut R adalah koefisien regresi eksplanator dari J.

2.2.4 K-means Clustering

K-means clustering adalah algoritma yang membutuhkan inputan k yang membagi jumlah objek menjadi k *cluster* dengan tujuan untuk mendapatkan pola

dengan melihat tingkat kesamaan yang besar antar anggota didalam *cluster* dan tingkat perbedaan yang tinggi dengan anggota yang berada diluar *cluster* (Dubey dkk., 2018). Tingkat kesamaan antar anggota didalam *cluster* diukur dengan nilai kedekatan objek pada nilai rata-rata *cluster* yang disebut dengan *cluster centroid* (Sammour dkk., 2016). Berikut adalah algoritma *k-means clustering* yang dapat dilihat pada gambar 2.3.



Gambar 2.3. Algoritma *k-means clustering* (Ahmar dkk., 2018).

Tahap awal algoritma *k-means* jumlah *k cluster* akan ditentukan secara acak sebelum digunakan metode untuk menentukan jumlah *cluster* terbaik. Tahap selanjutnya yaitu menentukan *centroid cluster* pada setiap *k cluster*. *Centroid cluster* bisa ditentukan secara acak dari objek-objek yang berada sebanyak *k* kluster. Selanjutnya hitung *centroid cluster* *i* berikutnya, digunakan rumus sebagai berikut:

$$v = \frac{\sum_i^n x_i}{n}; i = 1,2,3, \dots, n \quad (2.9)$$

Pada persamaan (2.9) *v* adalah *centroid* pada *cluster*, x_i adalah objek ke-*i* dan *n* adalah banyaknya objek/jumlah objek yang menjadi anggota *cluster*. Kemudian menghitung jarak antara objek dengan *centroid* menggunakan *euclidean distance*. Berikut rumus *euclidean distance* yang digunakan pada perhitungan *k-means* :

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}; i = 1,2,3, \dots, n \quad (2.10)$$

Pada persamaan (2.10) diketahui $d(x, y)$ adalah jarak objek *x* dengan masing-masing *centroid* *y*, *x* adalah sebuah objek, *y* adalah sebuah *centroid*, x_i adalah nilai atribut ke- *i* dari objek *x*, y_i adalah nilai atribut ke- *i* dari objek *y* dan *n* adalah banyaknya atribut. Selanjutnya objek dimasukkan kedalam *cluster* dengan melihat jarak terpendek terhadap *centroid*. Jika masih terjadi perpindahan objek maka lakukan iterasi kembali dengan menentukan posisi *centroid* baru dan lakukan perhitungan *euclidean distance* menggunakan persamaan 2.5.

Setelah itu melakukan pengecekan pada *matriks group assignment* pada iterasi sebelumnya dengan *matriks group assignment* pada iterasi saat ini. Jika hasilnya sama maka proses iterasi dapat dihentikan dan objek sudah berada pada

cluster sebenarnya. Namun jika belum maka perlu dilakukan proses iterasi berikutnya.

2.2.5 Confusion Matrix

Confusion matrix merupakan salah satu mekanisme pengujian untuk mengevaluasi kinerja klasifikasi dan *clustering* dengan bantuan empat komponennya, yaitu, *True Positive (TP)*, *False Negative (FN)*, *False Positive (FP)* dan *True Negative (TN)*. Diantaranya, akurasi klasifikasi merupakan parameter utama untuk menilai efisiensi suatu pengklasifikasi (Patro & Patra, 2015). Untuk mengevaluasi hasil deteksi intrusi ukuran berikut umumnya digunakan (Nalavade & Mehram, 2014). Selanjutnya untuk mendapat nilai akurasi dihitung dengan persamaan (2.11) berikut ini.

$$akurasi = \frac{TP+TN}{TP+FP+FN+TN} \quad (2.11)$$

Pada persamaan (2.11) True positive (TP) adalah angka koneksi yang tadi diklasifikasikan dengan benar sebagai intrusi, True Negative (TN) adalah jumlah koneksi salah diklasifikasikan sebagai intrusi, False positive (FP) adalah jumlah koneksi intrusi yang salah diklasifikasikan sebagai normal, FN negatif palsu) adalah jumlah koneksi normal yang salah diklasifikasikan sebagai intrusi.

Nilai akurasi tersebut digambarkan dalam tabel mekanisme pengujian *confusion matrix*. Tabel confusion matrix terdiri dari baris dan kolom yang memiliki nilai TP, TN, FP dan FN seperti yang dapat dilihat pada tabel 2.1 berikut ini.

Tabel 2.1 *Confusion matrix* (Nalavade & Mehsram, 2014)

	Positif		
		Normal	Attack
Negatif	Normal	TP	FN
	Attack	FP	TN

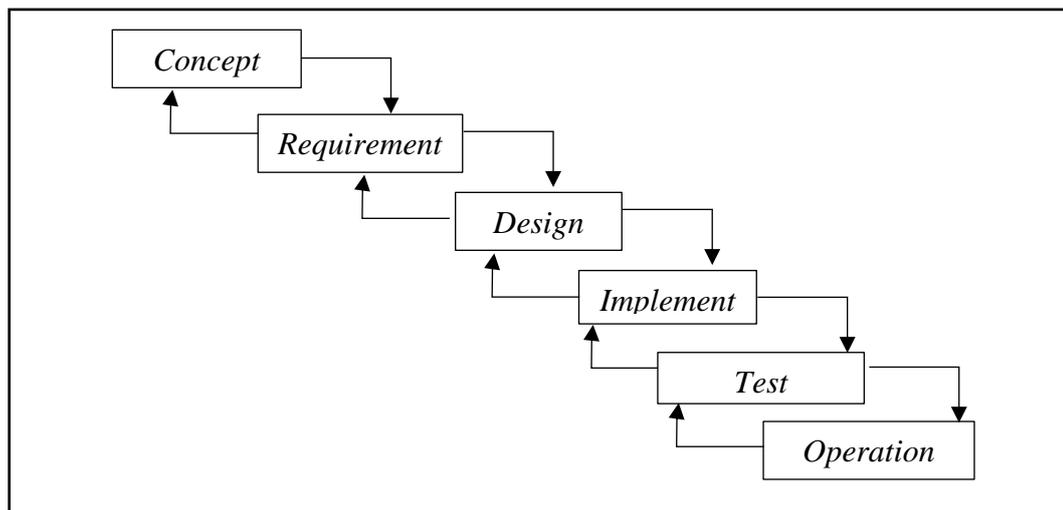
2.2.6 Pengembangan Perangkat Lunak

Proses membangun perangkat lunak komputer dan sistem informasi selalu didukung oleh berbagai metodologi pengembangan. Metodologi pengembangan perangkat lunak digunakan untuk merencanakan, mengelola, dan mengendalikan proses pengembangan sistem informasi (Sommerville dkk., 2011). *Software Development Live Cycle* (SDLC) merupakan siklus pengembangan perangkat lunak yang memiliki tahapan yang berurutan dimulai dari perencanaan hingga pengujian perangkat lunak (Ragunath dkk., 2010).

Model siklus hidup perangkat lunak adalah karakterisasi deskriptif atau preskriptif tentang bagaimana perangkat lunak dikembangkan. Model deskriptif menggambarkan histori bagaimana sistem perangkat lunak tertentu dikembangkan. Model deskriptif dapat digunakan sebagai dasar untuk memahami dan meningkatkan proses pengembangan perangkat lunak atau untuk membangun model preskriptif yang didasarkan secara empiris (Larman dkk., 2003).

Model *waterfall* adalah model SDLC tertua dan paling terkenal. Model ini banyak digunakan dalam proyek-proyek besar. Fitur khusus dari model ini adalah setiap bagian akan diselesaikan secara berurutan. Diawali dengan fase analisis persyaratan, desain, pengkodean, pengujian, dan pemeliharaan. Selain itu, model ini juga memastikan adanya kelemahan dari desain sebelumnya untuk melakukan pengembangan suatu produk. Model ini bekerja dengan baik untuk proyek-proyek kualita kontrak menjadi perhatian utama karena dokumentasi dan perencanaan

yang cukup intensif (Alshamrani dkk., 2015). Berikut ini metode diagram *waterfall* dapat ditunjukkan pada gambar 2.6.



Gambar 2.5 Diagram model *waterfall* (Westfall, 2010)

Tahap *concept* merupakan gambaran dari analisis kebutuhan perangkat lunak berisikan deskripsi lengkap dan komprehensif tentang perilaku perangkat lunak yang akan dikembangkan.

Tahap *requirement* merupakan analisis kebutuhan yang melibatkan persyaratan fungsional dan non-fungsional yang ditentukan dengan *Data Flow Diagram (DFD)*. *DFD* menggambarkan Aliran informasi yang terjadi pada pengguna dengan perangkat lunak. Persyaratan analisis seperti tujuan, ruang lingkup, perspektif, fungsi, atribut perangkat lunak, karakteristik pengguna, spesifikasi fungsi, persyaratan antarmuka dan persyaratan database. Sebaliknya persyaratan non fungsional merujuk pada kriteria, kendala, batasan dan persyaratan yang dikenakan pada desain pengoperasian perangkat lunak dari pada perilaku dan sifat sistem seperti keandalan, skalabilitas, pengujian, ketersediaan, pemeliharaan, kinerja, dan standar kualitas.

Tahap *design* adalah proses perencanaan dan pemecahan masalah untuk mencari solusi perangkat lunak. Pengembang dan perancang perangkat digunakan untuk menentukan rencana solusi yang meliputi desain algoritma, desain arsitektur perangkat lunak, skema konseptual database dan desain diagram logis, desain konsep, desain antarmuka pengguna grafis, dan definisi struktur data.

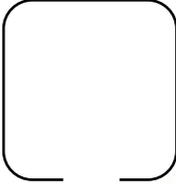
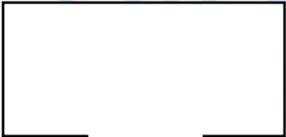
Tahap *implement* adalah realisasi persyaratan bisnis dan spesifikasi desain menjadi program yang dapat dieksekusi, *database*, situs web, atau komponen perangkat lunak melalui pemrograman dan penyebaran.

Tahap *test* adalah proses pengujian yang berfokus pada logika internal perangkat lunak, untuk memastikan semua kebutuhan dari sistem dapat berjalan dengan baik dan dapat mengetahui jika ada kesalahan pada input, proses dan output di sistem. Tahap *test* juga terdiri dari verifikasi dan validasi, verifikasi adalah proses mengevaluasi perangkat lunak menentukan apakah produk dari fase pengembangan tertentu telah memenuhi kondisi yang dijalankan pada awal fase itu, sedangkan validasi adalah proses mengevaluasi perangkat lunak selama atau pada akhir proses pengembangan untuk menentukan apakah itu memenuhi persyaratan yang ditentukan (Geraci dkk., 1991).

Tahap *operation* adalah proses memodifikasi solusi perangkat lunak setelah pengiriman dan penyebaran untuk memperbaiki output, memperbaiki kesalahan, dan meningkatkan kinerja dan kualitas. Kegiatan pemeliharaan tambahan dapat dilakukan dalam fase ini termasuk mengadaptasi perangkat lunak dengan lingkungan, mengakomodasi kebutuhan pengguna baru, dan meningkatkan keandalan perangkat lunak (Stellman dkk., 2005).

Selanjutnya, DFD digunakan untuk mencapai ketepatan, keseimbangan diagram dan konsistensi antara diagram. Simbol DFD terdiri dari empat simbol yaitu proses, aliran data, penyimpanan data dan entitas eksternal (Ibrahim dkk., 2011). Berikut simbol elemen DFD dapat dilihat pada tabel 2.1

Tabel 2.2. Simbol elemen DFD (Dennis dkk., 2006)

Simbol	Nama Elemen
 <p style="text-align: center;">0</p> <hr style="width: 20%; margin: auto;"/> <p style="text-align: center;">Name</p> <hr style="width: 20%; margin: auto;"/>	<p style="text-align: center;">Process</p>
<p style="text-align: center;">Name</p> <hr style="width: 20%; margin: auto;"/> 	<p style="text-align: center;">Data Flow</p>
	<p style="text-align: center;">Data Store</p>
 <p style="text-align: center;">Name</p>	<p style="text-align: center;">External Entity</p>

Dalam diagram aliran data, tampilan tingkat tertinggi dari sistem dikenal sebagai diagram konteks. Level berikutnya dari diagram aliran data disebut diagram aliran data level 0 yang mewakili proses utama sistem, aliran data, dan penyimpanan data pada tingkat detail yang tinggi. Setiap proses dalam diagram aliran data level $n-1$ akan didekomposisi menjadi diagram aliran data level bawah yang merupakan diagram aliran data level n . Diagram alir data diilustrasikan pergerakan data antara entitas eksternal dan proses dan penyimpanan data dalam suatu sistem (Dennis dkk., 2006).

