

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Penerapan Teknologi Informasi (TI) menjadi sebuah cara atau metode melakukan transparansi bidang tugas pertanggung jawaban. Teknologi memudahkan siapapun berinteraksi untuk mendapatkan informasi terkini. Pemanfaatan TI yang terbuka untuk umum dan dapat diakses oleh siapapun memungkinkan pengawasan terhadap kinerja pemerintah. Teknologi informasi tidak hanya terbatas pada teknologi komputer (perangkat keras dan perangkat lunak) yang akan digunakan untuk memproses dan menyimpan informasi, melainkan juga mencakup teknologi komunikasi untuk mengirim informasi. Pada masa pandemi pemerintah gencar memberikan Bantuan Dana Langsung (BLT). Bantuan Langsung Tunai yang disingkat BLT merupakan program bantuan pemerintah berjenis pemberian uang tunai atau beragam bantuan lainnya, baik bersyarat maupun tidak bersyarat. Dalam buku Panduan Pendataan Penerima BLT-Dana desa, dijelaskan kriteria calon penerima BLT dana desa bermanfaat membantu. (Sinaga, 2021)

(Natasuwarna, 2019) Data mining merupakan bagian dari tahapan proses Knowledge Discovery in Database ( KDD ) dengan menggunakan metode Clustering. Metode Algoritma DBSCAN merupakan metode klasifikasi yang cukup bagus karena tingkat akurasi yang lebih baik di bandingkan dengan metode lain. Algoritma Clustering yang masuk kedalam kelompok Unsupervised learning yang

dipakai untuk mengelompokkan data kedalam beberapa bagian dengan sistem partisi . Kesederhanaan DBSCAN menyatakan bahwa algoritma sudah diadopsi dibanyak bidang. Karena mampu dengan cepat dan efisien dalam mengelompokkan sejumlah data yang besar, termasuk outlier .

Peraturan Menteri Desa dan Pembangunan Daerah Tertinggal dan Transmigrasi (Permendes PDTT) Nomor 6 Tahun 2020 Tentang Perubahan pada Peraturan Menteri Desa, Pembangunan Daerah Terpelosok, dan Transmigrasi Nomor 11 Tahun 2019 Tentang Prioritas Penggunaan Dana Desa Tahun 2020, merupakan dasar yuridis dan implementatif BLT kepada penduduk yang tidak mampu di desa.

Bahtera Makmur yang berada di Kecamatan Bagan Sinembah Kabupaten Rokan Hilir di tahun 2023 merupakan salah satu desa yang masyarakatnya merupakan penerima BLT di Kabupaten Rokan Hilir. BLT di Bahtera Makmur Kecamatan Bagan Sinembah diputuskan dan ditetapkan bersama melalui Musyawarah untuk menentukan warga yang terdampak kurang mampu secara ekonomi dan keuangan sebagai penerima BLT. Banyaknya data calon penerima BLT dan sulitnya pengolahan data sesuai dengan karakteria penerima BLT menyebabkan sering terjadi kesalahan dalam penetapan penerima BLT. Pengelolaan data BLT di Desa Bahtera Makmur masih menggunakan sistem manual, yang masih belum terbuka sehingga sering terjadi permasalahan seperti kesalahan kriteria penerima bantuan dan tidak tepat sasaran pada penerimanya. Pengelolaan data dan keputusan yang kurang transparan menimbulkan ketidak seimbangan dan kecemburuan sosial di tengah masyarakat.(Azhari & Suhartini, 2021)

Tujuan utama yang ingin dicapai dalam penelitian ini, adalah membuat Aplikasi penerima BLT Berbasis Web dengan menerapkan kriteria penerima BLT sebagai Clustering sehingga diperoleh atau dihasilkan informasi yang cepat dan akurat penerima BLT secara online di Desa Bahtera Makmur. Berdasarkan keadaan Desa Bahtera Makmur yang telah diuraikan di atas maka peneliti memutuskan untuk menyelesaikan permasalahan yang ada dengan cara membangun sebuah aplikasi berbasis web yang berjudul.

**“IMPLEMENTASI ALGORITMA DBSCAN ( Density Based Clustering Of Applications With Noise ) Pada Aplikasi Penerima BLT ( Bantuan Langsung Tunai ) Online DI DESA BAHTERA MAKMUR Kec. Bagan Sinembah Kab. Rokan Hilir - RIAU”**. Dengan adanya sistem ini diharapkan dapat membantu petugas dalam mencari data penerima BLT secara cepat, tepat dan akurat . Selain untuk menentukan penduduk yang berhak menerima BLT sistem ini juga dapat dipergunakan untuk mengetahui berapa banyak total masyarakat yang tidak mampu di Desa Bahtera Makmur.

## **1.2 Rumusan Masalah**

Adapun rumusan masalah dalam penelitian ini ialah :

1. Bagaimana merancang aplikasi online tentang pengelompokkan karakteristik calon penerima Dana BLT.
2. Bagaimana menerapkan Algoritma DBSCAN dalam membangun aplikasi berbasis website untuk mengetahui calon penerima Dana BLT Desa Bahtera Makmur.

### **1.3 Batasan Masalah**

Agar penelitian tetap dalam jalur yang seharusnya, maka penelitian mengenai hal ini harus diberi batasan. Adapun ruang lingkup batasan masalah tersebut adalah :

1. Sistem pengelompokan penerima BLT di Desa Bahtera Makmur dengan menggunakan Algoritma DBSCAN.
2. Aplikasi dalam memprediksi penerima Dana BLT ini menggunakan aplikasi berbasis website yang dirancang dengan bahasa pemrograman PHP dan database MySQL.
3. Aplikasi memprediksi penerima Dana BLT di rancang dengan menggunakan UML (*Unified Modelling Language*).

### **1.4 Tujuan Penelitian**

Adapun Tujuan dari penelitian ini ialah :

1. Dengan adanya aplikasi ini dapat memudahkan pegawai untuk mendata penduduk di Desa Bahtera Makmur yang berjumlah 252 Kartu Keluarga untuk mencari data – data yang cocok untuk menerima Dana BLT di Desa Bahtera Makmur.
2. Website ini dapat memberikan informasi yang detail bagi penerima Dana.
3. Website berguna untuk mengetahui karakteristik bagi calon penerima Dana BLT.

### **1.5 Manfaat Penelitian**

1. Bagi Desa

Manfaat dari penelitian itu sendiri dapat mengetahui karakteristik calon penerima Dana BLT.

## 2. Bagi Mahasiswa

Hasil penelitian ini dapat memberikan dorongan bagi mahasiswa untuk mengetahui tentang Dana BLT.

## 3. Bagi Perguruan Tinggi

Hasil penelitian ini dapat digunakan sebagai materi pengajaran.

### **1.6 Sistematika Penulisan**

Penulisan penelitian ini dibagi dengan beberapa bab dengan tujuan untuk menunjukkan penyelesaian masalah yang sistematis. Pembagian bab adalah sebagai berikut.

#### **Bab I : Pendahuluan**

Pada bab ini menjelaskan tentang mengenai latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metode penelitian dan sistematika penulisan.

#### **Bab II : Tinjauan Pustaka**

Pada bab ini menjelaskan tentang landasan teori yang berkaitan dengan penelitian ini, seperti pengertian sistem informasi, website, komponen perancangan sistem, alat bantu perancangan sistem, flowchart, profil desa, dan peneliti yang terkait.

#### **Bab III : Metode Penelitian**

Pada bab ini menjelaskan tentang metode dan tahapan penelitian yang digunakan dan dilakukan selama proses penelitian berlangsung, seperti metode penelitian yang digunakan, instrument *functionality* sistem, penerapan metode penelitian, Teknik pengumpulan data, serta rancangan aplikasi yang akan dibuat.

#### **Bab IV : Hasil dan Pembahasan**

Pada bab ini menjelaskan tentang hasil dari penelitian yang telah dilakukan berdasarkan metode dan tahapan yang sudah dilakukan sebelumnya, seperti, analisis perancangan model sistem, hasil rancangan tampilan design sistem, penerapan pemrograman sistem, hasil pengembangan sistem dan hasil dari aplikasi yang telah dirancang.

#### **Bab V : Penutup**

Pada bab ini menjelaskan tentang kesimpulan dari keseluruhan penelitian yang telah dilakukan serta saran dari peneliti.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Implementasi**

Menurut (Schlindwein & Ison, 2018) implementasi adalah tindakan atau pelaksanaan dari sebuah rencana yang telah disusun dengan matang, cermat dan terperinci. Jadi, implementasi dilakukan jika sudah ada perencanaan yang baik dan matang, atau sebuah rencana yang telah disusun jauh jauh hari sebelumnya, sehingga sudah ada kepastian dan kejelasan akan rencana tersebut.

Berikut ini kumpulan pengertian implementasi menurut para ahli sebagai berikut :

1. Sedangkan menurut Leo Agustino (2014), “implementasi merupakan suatu proses yang dinamis, dimana pelaksana kebijakan melakukan suatu aktivitas atau kegiatan, sehingga pada akhirnya akan mendapatkan suatu hasil yang sesuai dengan tujuan atau sasaran kebijakan itu sendiri”.
2. Sedangkan pada jurnal ilmiah yang ditulis oleh Prabowo & Tiwuk pada tahun 2021, implementasi adalah sebuah tindakan guna menyelesaikan suatu hal yang dapat memberi dampak ataupun akibat. Dapat juga diartikan sebagai pelaksanaan suatu kegiatan terencana secara sungguh-sungguh dan menerapkan norma yang berlaku dalam mewujudkan tujuannya.

## 2.2 Aplikasi

(Syarifuddin, 2017) Aplikasi merupakan sebuah subkelas perangkat lunak komputer yang secara langsung menggunakan kapabilitas komputer dengan maksud melaksanakan suatu pekerjaan sesuai kehendak pengguna. Hal ini berbeda dengan perangkat lunak sistem yang memadukan bermacam-macam kapabilitas komputer, namun tidak secara langsung mengimplementasikannya demi menyelesaikan pekerjaan tertentu.

Berikut ini merupakan pengertian aplikasi menurut beberapa Ahli :

1. Menurut Rachamd Hakim S (2018), Aplikasi adalah perangkat lunak yang digunakan untuk tujuan tertentu, seperti mengubah dokumen, mengatur windows & permainan *game* dan sebagainya.
2. Menurut Harip Santoso (2017), Aplikasi adalah suatu kelompok file (*form, class, report*) yang bertujuan untuk melakukan aktivitas tertentu yang saling terkait, misalnya aplikasi *payroll*, aplikasi *fixed asset* dan lain-lain. Aplikasi sendiri berasal dari kata *application* yang artinya penerapan lamaran penggunaan.

Selain itu aplikasi berfungsi untuk sebagai pelayanan kebutuhan aktifitas yang dilakukan oleh manusia seperti sistem untuk *software* jual beli, permainan atau *game online*, pelayanan masyarakat dan hampir semua proses yang dilakukan oleh manusia dapat dibantu dengan menggunakan suatu aplikasi. Beberapa aplikasi jika digabungkan akan menjadi suatu paket atau sering juga disebut dengan *application suite*, dimana aplikasi tersebut memiliki posisi antar muka yang mempunyai kesamaan sehingga dapat dengan mudah digunakan atau pelajari

pengguna tiap aplikasi tersebut (Halawa, 2016)

### **2.3 Website**

Menurut (Elgamar, 2020:3), *website* adalah suatu media yang terdiri dari beberapa halaman yg saling berkaitan satu sama lain, dan berfungsi menjadi media untuk menampilkan suatu berita, baik berbentuk gambar, video, teks, suara, ataupun gabungan berasal semuanya. *Website* bersifat multiplatform yang artinya bisa dibuka dari segala perangkat atau *device* yang terhubung menggunakan jaringan internet. Walaupun teknologi ini telah relatif lama digunakan, namun waktu ini masih aneka macam perusahaan-perusahaan yg masih menggunakan website dalam menampilkan profil perusahaan (*company profile*), menjual produk, ataupun menjadi sistem yg bisa digunakan sang pelanggan.

Aplikasi berbasis *web* pada umumnya dibangun menggunakan bantuan dari struktur HTML (Hypertext Markup Language), serta dengan kombinasi dari beberapa bahasa pemrograman lain, seperti PHP ataupun Javascript. *Website* juga bisa dipercantik tampilannya menggunakan bantuan CSS (*Cascading Style Sheets*). mengenai *database* atau media penyimpanan, relatif banyak yang bisa dipergunakan, salah satunya adalah MYSQL

### **2.4 Komponen Perancangan Sistem**

Perancangan sistem ialah suatu langkah awal dari keseluruhan pembangunan sistem informasi pengolahan data penduduk berbasis komputerisasi. Perancangan sistem pula biasanya membutuhkan jangka waktu yang sedikit lama daripada pemecahan suatu masalah yang pada umumnya menampilkan aliran data utama pada sistem. Beberapa komponen perancangan sistem informasi pengolahan data penduduk yang dipakai adalah sebagai berikut:

### 2.4.1 XAMPP

(Siregar & Sari, 2018) 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), merupakan web server yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis. XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi beberapa program. Berfungsi sebagai server yang berdiri sendiri (localhost). Dari definisi tersebut, penulis menyimpulkan bahwa XAMPP adalah sebuah software yang berfungsi untuk menjalankan website berbasis PHP dan menggunakan pengolah data MySQL di komputer lokal. XAMPP berperan sebagai server web pada komputer. XAMPP juga dapat disebut sebuah cpanel server virtual, yang dapat membantu melakukan preview sehingga dapat memodifikasi website tanpa harus online atau terakses dengan internet.

### 2.4.2 HTML (*Hyper Text Markup Language*)

*Hyper Text Markup Language* atau HTML merupakan suatu bahasa yang standar serta sering dipakai untuk menyajikan halaman web. Dengan menggunakan HTML kita dapat melakukan beberapa hal yaitu membuat table dalam halaman web, mempromosikan halaman web secara *online*, mengelola tampilan serta kandungan dari halaman web, membuat form yang bisa dipakai dalam mengatur

pengregistrasian serta transaksi via web, dapat menambah objek-objek misalnya citra, java applet, audio, animasi pada halaman web, video, serta dapat memperlihatkan area gambar (canvas) pada *browser*.

Di dalam HTML seluruh tag mempunyai sifat yang dinamis, maksudnya kode HTML tidak bisa dibuat menjadi sebuah *file executable program*. Ini dikarenakan HTML hanya suatu bahasa *scripting* yang bisa dieksekusi jika diproses didalam *browser* (pengakses web), HTML didukung oleh beberapa *browser* antara lain yaitu: *Netscape Navigator, Opera, Internet Explorer, Mozilla* dan lain sebagainya. Jadi kita Ketika ingin mengakses halaman dari HTML anda bisa mengklik *view-source* untuk melihat bagaimana bentuk pengkodeannya mulai dari seluruh tag serta isi dari halaman web tersebut. HTML adalah suatu kode *scripting* serta bukanlah sebuah program *compiler* sehingga memerlukan editor untuk menulis kode program, berikut beberapa editor yang bisa dipakai adalah *Front page, Home Site* atau *Note Pad* sebagai editor standar *windows*, serta *Macromedia Dreamweaver*(Latuconsina & Priyambadha, 2019).

### **2.4.3 PHP (*Hypertext Preprocessor*)**

Menurut Sri Haryanti (2011), PHP (*hypertext preprocessor*) adalah sebuah bahasa pemrograman yang berbentuk *scripting*, sistem kerja dari program ini adalah sebagai *interpreter* bukan *compiler*. Artinya semua sintaks yang diberikan akan sepenuhnya dijalankan pada server sedangkan yang dikirimkan ke browser hanya hasilnya saja. Ketika seorang pengguna internet membuka suatu situs yang menggunakan fasilitas *server side scripting PHP*, maka terlebih dahulu server yang bersangkutan akan memproses semua perintah PHP di server lalu mengirimkan hasilnya dalam format HTML ke web server pengguna internet.

Sehingga kode asli yang ditulis dengan PHP tidak terlihat di browser pengguna. PHP tergolong sebagai perangkat lunak open source yang diatur dalam aturan general purpose licences (GPL). Pemrograman PHP sangat cocok dikembangkan dalam lingkungan web, karena PHP bisa dilekatkan pada script HTML atau sebaliknya. PHP dikhususkan untuk pengembangan web dinamis. Maksudnya, PHP mampu menghasilkan website yang secara terus menerus hasilnya bisa berubah ubah sesuai dengan pola yang diberikan.

#### **2.4.4 Mysql**

Menurut Sri Haryanti (2011), MySQL merupakan turunan dari salah satu konsep utama dalam basis data sejak lama, yaitu SQL (structured query language). SQL adalah sebuah konsep pengoperasian basis data terutama dapat dikerjakan dengan mudah dan otomatis. Kepopuleran MySQL dimungkinkan karena kemudahannya untuk digunakan, cepat secara kinerja query, dan mencukupi untuk kebutuhan database perusahaan-perusahaan skala menengah kecil. MySQL merupakan database yang digunakan oleh situs-situs terkemuka di internet untuk menyimpan datanya.

MySQL dimiliki dan disponsori oleh sebuah perusahaan komersial Swedia, MYSQL AB dimana hak cipta dan semua kode sumbernya dipegang oleh perusahaan itu sendiri. Adapun yang mendirikan MYSQL AB adalah David Axmark, Allan Larsson dan Michael “Monty” Widenius.

MySQL adalah sebuah implementasi dari sistem manajemen basis data relasional (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (*General Public License*). Setiap pengguna dapat secara bebas menggunakan MySQL, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan

produk turunan yang bersifat komersial. MySQL sebenarnya merupakan turunan dalam satu konsel utama dalam basis data yang telah ada sebelumnya; SQL (*Structured Query Language*). SQL adalah sebuah konsel pengoperasian basisdata, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah dan otomatis.

## **2.5 UML (*Unified Modelling Language*)**

Menurut Brigida dalam Winda Fandella (2015:14) *Unified Modelling Language*”(UML) merupakan salah satu alat bantu yang dapat digunakan dalam bahasa pemrograman yang berorientasi objek.

UML merupakan bahasa visual yang ditemukan oleh Grady Booch, Ivar Jacobson dan James Rumbaugh untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung.

Diagram pada UML dibagi menjadi dua kelompok, yaitu *structure diagram* dan *behavior diagram* yang mana *structure diagram* menggambarkan data dan hubungan statis dalam suatu sistem informasi. *Structure diagram* terdiri dari *package*, *object*, *component*, *class*, *deployment*, *composite structure*, dan *profile diagram*. Sementara itu, *behavior diagram* menggambarkan hubungan dinamis diantara objek yang mewakili sistem informasi bisnis. *Behavior diagram* terdiri dari *sequence*, *timing*, *interaction*, *overview*, *activity*, *use case*, *protocol*, *state machine*, *communication* dan *Behavior state machine diagram*.

### **2.5.1 Use Case Diagram**

*Use Case Diagram* adalah sebuah diagram yang wajib dikerjakan pertama kali pada saat pemodelan perangkat lunak berorientasi objek dibuat. *Use Case*

menggambarkan suatu hubungan timbal balik antara satu actor atau lebih dari sistem yang akan dibangun. *Use Case Diagram* juga berfungsi untuk mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat. *Use Case Diagram* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. (Muhammad Romzi & Kurniawan, 2020)

Berikut adalah simbol-simbol yang ada pada dalam *use case diagram*:

Table 2.1 Daftar Simbol *Use Case Diagram*

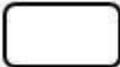
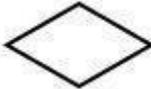
Simbol	Nama	Keterangan
	Aktor	Adalah Penggunaan dari sistem. Nama aktor diisi dengan katabenda
	<i>Use Case</i>	Adalah pekerjaan yang dikerjakan aktor. Nama use case menggunakan kata kerja.
	Asosiasi	Keterkaitan antar aktor dan usecase
<<use>>	<i>Include</i>	Keterkaitan antara use case dan, include menjelaskan jika sebelum suatu pekerjaan dikerjakan maka wajib melakukan pekerjaan lain terlebih dahulu
<<extends>>	<i>Extends</i>	Keterkaitan antara use case dan use case, extends menjelaskan jika pekerjaan yang dikerjakan tidak sesuai

		atau memiliki suatu kondisi khusus, maka kerjakan pekerjaan itu.
--	--	--

### 2.5.2 Activity Diagram

*Activity Diagram* mendeskripsikan aliran kerja (*workflow*) dengan cara menggambarkan kegiatan dari sebuah sistem. *Activity diagram* mendeskripsikan aliran kerja dengan penggambaran kegiatan dari suatu menu yang terdapat pada software. *Activity diagram* mengilustrasikan kegiatan kerja sistem bukan kegiatan yang dapat dilakukan oleh aktor, jadi kesimpulannya *activity diagram* adalah kegiatan yang bisa dikerjakan oleh sistem. Simbol-simbol dari *activity diagram* adalah sebagai berikut:

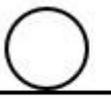
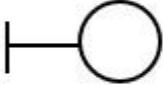
Tabel 2.2 Daftar Simbol *Activity Diagram*

<b>Nama</b>	<b>Simbol</b>	<b>Keterangan</b>
<i>Innitial State</i>		Menggambarkan awal suatu aktivitas
<i>Final State</i>		Menggambarkan berakhirnya aktivitas
<i>Activity</i>		Menggambarkan kegiatan yang dikerjakan oleh sistem
<i>Decision</i>		Menggambarkan pilihan serta mengambil keputusan dari pilihan yang telah disediakan
<i>Control Flow</i>		Menggambarkan arah dari kegiatan sebuah sistem.

### 2.5.3 Sequence diagram

*Sequence diagram* merupakan diagram yang menjelaskan interaksi objek berdasarkan urutan waktu. *Sequence diagram* dapat menggambarkan urutan atau tahapan yang harus dilakukan untuk dapat menghasilkan sesuatu, seperti yang tertera pada *Use Case Diagram*.

Table 2.4 Daftar Simbol *Sequence Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menggambar orang yang sedang berinteraksi dengan sistem.
2		<i>Entity Class</i>	Menggambarkan hubungan yang akan dilakukan
3		<i>Boundary Class</i>	Menggambarkan sebuah gambaran dari form
4		<i>Control Class</i>	Menggambarkan penghubung antara boundary dengan tabel
5		<i>A focus of Control &amp; A Life Line</i>	Menggambarkan tempat mulai dan berakhirnya message
6		<i>A message</i>	Menggambarkan Pengiriman Pesan

## **2.6 Data Mining**

### **2.6.1 Pengertian Data Mining**

(Sudipa, 2024) Data mining adalah ekstraksi informasi atau pola yang penting atau menarik dari data yang ada di database yang besar. Dalam jurnal ilmiah, data mining juga di kenal dengan nama *Knowledge Discovery in database* (KDD).

Data mining dapat didefinisikan sebagai salah satu set teknik Yang digunakan Secara menyeluruh dan membawa ke permukaan relasi – relasi yang kompleks pada set data yang berbentuk tabulasi seperti banyak implementasikan dalam teknologi manajemen basis data relasional. Akan tetapi, teknik – teknik data mining dapat juga di aplikasikan pada repretasi data yang lain, seperti domain data spatial, berbasis teks, dan multimedia.

### **2.6.2 Sejarah Data Mining**

Data mining muncul sekitar tahun 90-an. Data Mining memang salah satu cabang ilmu komputer yang relatif baru. Dan sampai sekarang orang masih memperdebatkan untuk menepatkan data mining menyamgkut database, kecerdasan buatan ( *artificial intelligence* ), statistic, dan sebagainya. Ada pihak yang berpendapat bahwa data mining tidak lebih dari machine lerning atau analisis statistic yang berjalan diatas database. Namun pihak lain berpendapatan bahwa peran database berperanan penting di data mining karena data mining mengakses data yang ukurannya besar (bisa sampai terabyte) dan disini terlihat penting database terutama dalam optimisasi query-nya.

### **2.6.3 Cara Kerja Data Mining**

Cara kerja data mining yaitu “menggali” hal – hal penting yang belum diketahui sebelumnya atau memprediksi apa yang akan terjadi. Teknik yang digunakan untuk melaksanakan tugas ini disebut pemodelan. Pemodelan di sini dimaksudkan sebagai kegiatan untuk membangun sebuah model pada situasi yang telah diketahui “ jawabannya” dan kemudian menerapkannya pada situasi lain yang akan di cari jawabannya.

## **2.7 Algoritma**

### **2.7.1 Pengertian Algoritma**

Algoritma adalah suatu langkah atau metode yang telah direncanakan secara matang sehingga terurut dan terorganisir dengan baik dan biasanya digunakan untuk memecahkan suatu masalah dengan memberikan suatu instruksi sehingga menjadi suatu tindakan.Sedangkan dalam Kamus Besar Bahasa Indonesia (KBBI), algoritma adalah suatu prosedur sistematis untuk menyelesaikan masalah matematika dalam langkah-langkah terbatas atau urutan pengambilan keputusan yang logis untuk memecahkan masalah tersebut.

Algoritma merupakan jantung ilmu komputer atau pemrograman. Banyak cabang ilmu computer yang diacu dalam terminologi algoritma. Namun, jangan beranggapan algoritma selaluidentik dengan ilmu komputer saja. Dalam kehidupan sehari-haripun banyak terdapatproses yang dinyatakan dalam suatu algoritma. Cara-cara membuat kue atau masakan yangdinyatakan dalam suatu resep juga dapat disebut sebagai algoritma. Pada setiap resep selaluada urutan

langkah-langkah membuat masakan. Bila langkah-langkahnya tidak logis, tidak dapat dihasilkan masakan yang diinginkan. Ibu-ibu yang mencoba suatu resep masakan akan membaca satu per satu langkah-langkah pembuatannya lalu ia mengerjakan proses sesuai yang ia baca. Secara umum, pihak (benda) yang mengerjakan proses disebut pemroses (*processor*). Pemroses tersebut dapat berupa manusia, komputer, robot atau alat-alat elektronik lainnya. (Rizq Daffa Jodi, 2020)

### 2.7.2 Struktur Dasar Algoritma

Adapun struktur dasar pada Algoritma adalah sebagai berikut

a. Sekuensial ( runtunan )

Pada struktur ini sekuensial ini langkah – langkah yang dilakukan dalam algoritma di proses secara berurutan. Di mulai dari langkah pertama, kedua, dan seterusnya. Pada dasarnya suatu program memang menjalankan suatu proses dari yang dasar seperti struktur ini.

b. Struktur seleksi

Struktur seleksi menyatakan pemilihan langkah yang didasarkan oleh suatu kondisi atau pengambilan suatu keputusan. Struktur ini ditandai selalu dengan bentuk *flowchart decision* ( *flowchart* yang berbentuk belah ketupat ). Banyak contoh yang dapat kita terapkan pada struktur jenis ini jika itu menyangkut keputusan, diantaranya: diskon yang berbeda berdasarkan jumlah barang yang ingin dibeli

c. Stuktur perulangan

Stuktur ini memberikan perintah atau tindakan yang di lakukan beberapa kali. Misalnya jika teman mau menuliskan kata “belajar c” sebanyak sepuluh kali. Akan lebih efisien jika teman menggunakan stuktur ini dari pada sekedar menuliskannya berturut- turut sebanyak sepuluh kali.

## 2.8 Algoritma DBSCAN

Algoritma Density-based Spatial Clustering of Application with Noise (DBSCAN) merupakan metode clustering yang berbasis kepadatan (*density-based*) dari posisi amatan data dengan prinsip mengelompokkan data yang relatif berdekatan. DBSCAN sering diterapkan pada data yang banyak mengandung noise, hal ini dikarenakan DBSCAN tidak akan memasukkan data yang dianggap noise kedalam cluster manapun.

(Gunawan, 2021) Algoritma DBSCAN ini sendiri merupakan singkatan dari *density-based spatial clustering of applicatian with noise*. Seperti namanya, algoritma ini menggunakan sebuah metode *clustering* berbasis *density* atau *density based*.

*Clustering* sendiri merupakan proses yang bertujuan untuk membagi data kedalam kelompok tertentu berdasarkan kategori atau tingkat kemiripan data. *Clustering* sendiri biasa digunakan untuk mengolah data dalam berbagai bidang seperti segmentasi pasar, data *spatial*, serta *clastring profiling*.

Seperti disebutkan sebelumnya, DBSCAN merupakan algoritma yang menggunakan algoritma clustering berbasis *density* atau kepadatan. Algoritma

DBSCAN banyak digunakan pada data yang mengandung banyak noise atau gangguan. DBSCAN cenderung memisahkan data yang mengandung noise agar tidak bercampur dengan *cluster* apapun yang ada dalam data.

Algoritma DBSCAN memiliki Komponen-komponen dalam proses analisis clustering yang ada pada algoritma DBSCAN antara lain:

a. Epsilon

Epsilon kekerabatan dari sebuah profil atau Eps-neighborhood dari sebuah profil, *Eps-neighborhood* dari sebuah profil,  $N_{eps}(P)$ , didefinisikan sebagai

$$N_{eps}(p) = \{q \in D \mid dist(p, q) \leq Eps\} \quad (2.1)$$

$D$  = basis data yang dianalisis

$q$  = profil lain.

$Eps$  = nilai ambang jarak antar profil untuk dapat dimasukkan ke dalam *cluster* yang sama profil  $p$  dapat berkerabat dengan profil  $q$  (berada dalam satu cluster yang sama) jika jarak dari  $p$  ke  $q$  tidak lebih dari nilai  $Eps$ .

b. Minimum Points

Meskipun  $p$  berada dalam *Eps-neighborhood* dari  $q$ , akan tetapi jika hanya dua profil itu saja yang berkerabat, maka akan ada kasus dimana terdapat banyak *cluster* dengan jumlah anggota yang sedikit. Untuk mengantisipasi hal tersebut, diperkenalkan istilah minimum *points* atau *MinPts*. *MinPts* merupakan nilai ambang yang merepresentasikan jumlah minimal profil yang berada dalam *Eps-neighborhood* profil  $p$  agar dapat terbentuk *cluster*. Dengan nilai ambang ini, maka ada tiga klasifikasi jenis profil di DBSCAN, yaitu profil yang berada di berada di luar daerah padat disebut *outlier*, profil yang berada di pangkal daerah padat

disebut *border point*, dan profil yang berada di dalam daerah padat disebut *core point*.

c. *Directly density-reachable*

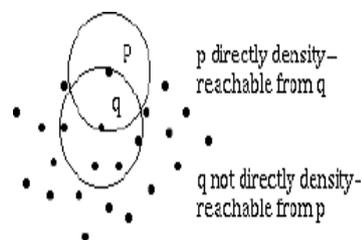
Sebuah profil  $p$  dikatakan *directly density-reachable* terhadap profil  $q$  jika

$$P \in N_{\text{eps}}(q), \text{ dan}$$

$$|N_{\text{eps}}(q)| \geq \text{MinPts} \text{ (} q \text{ merupakan core point)}$$

Dari definisi tersebut dapat diketahui bahwa agar profil  $p$  *directly density-reachable* terhadap profil  $q$ , maka harus memenuhi dua syarat yaitu profil  $p$  berada pada *Eps-neighborhood* profil  $q$  dan profil  $q$  merupakan *core point*.

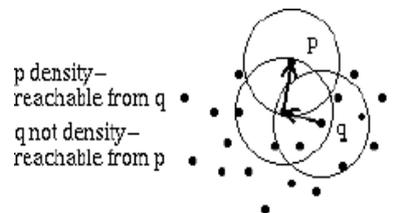
*Directly density-reachable* bersifat simetris jika  $p$  dan  $q$  keduanya adalah *core point*. Artinya, jika  $p$  *directly density-reachable* terhadap  $q$ , maka  $q$  *directly density-reachable* terhadap  $p$ .



Gambar 2.1 Contoh sepasang profil yang *directly density-reachable*

d. *Density-reachable*

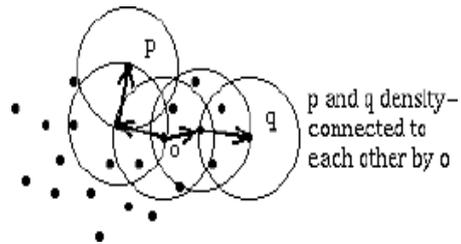
Sebuah profil  $p$  dikatakan *density-reachable* terhadap profil  $q$  jika terdapat rantai  $p_1 \dots p_n$ , dengan  $p_1 = p$  dan  $p_n = q$ , sedemikian sehingga  $p_{i+1}$  bersifat *directly density-reachable* terhadap  $p_i$ . Dari definisi tersebut dapat diketahui bahwa dua buah profil dikatakan *density-reachable* jika ada satu rantai profil sedemikian sehingga dari profil satu ke profil lain di dalam rantai tersebut bersifat *directly density-reachable*. Sifat *density-reachable* tidak menjamin dua *border point* bersifat *density-reachable*.



Gambar 2.2 Contoh sepasang profil yang *density-reachable*

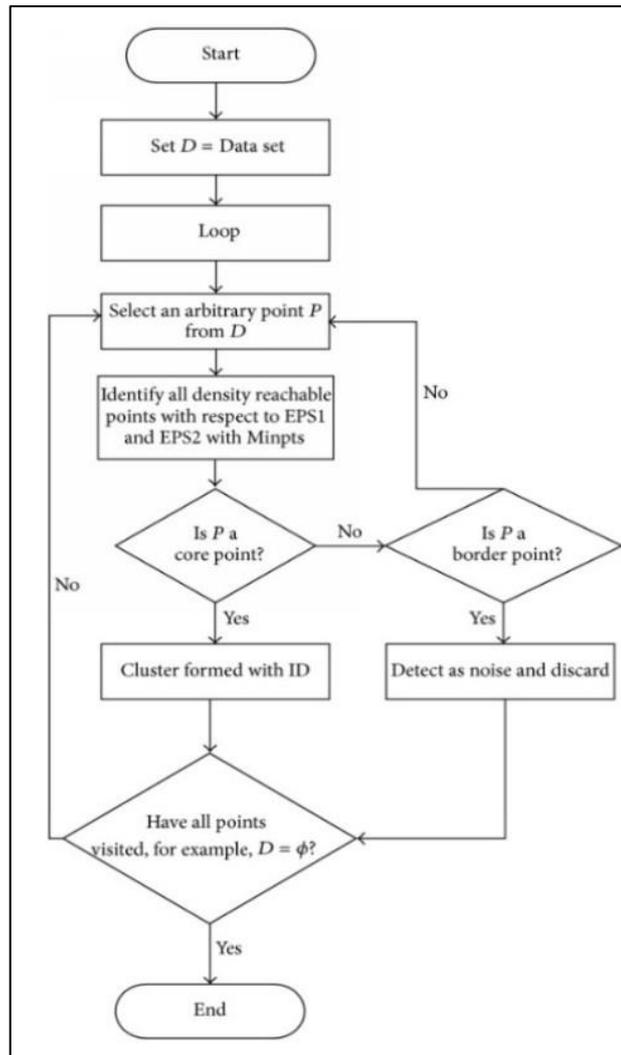
e. *Density-connected*

Sebuah profil  $p$  dikatakan *density-connected* terhadap profil  $q$  jika terdapat profil  $o$  sedemikian sehingga profil  $p$  dan profil  $q$  bersifat *density-reachable* terhadap poin  $o$ . Dengan demikian, setidaknya dua profil di dalam satu *cluster* bersifat *density-connected*. *Density-connected* bersifat simetris dan refleksif. Artinya, jika profil  $p$  bersifat *density-connected* terhadap poin  $q$ , maka profil  $q$  bersifat *density-connected* terhadap profil  $p$ .



Gambar 2.3 Contoh sepasang profil yang *density-connected*

Nilai *eps* dan *MinPts* harus diketahui untuk dapat menjalankan Algoritma DBSCAN. Algoritma DBSCAN dimulai dengan memilih satu profil *p* secara acak, kemudian mencari profil-profil lain yang *density-reachable* terhadap profil *p*. Jika *p* merupakan *core point*, maka terbentuk suatu *cluster*. Akan tetapi jika *p* adalah *border point*, maka DBSCAN akan mengambil profil lain dari basis data. Dalam proses tersebut, terdapat kemungkinan dua *cluster* bergabung jika kedua *cluster*.



Gambar 2.4 Flowchart Algoritma DBSCAN

Berdasarkan gambar 5, dalam algoritma DBSCAN dilakukan input data yang akan digunakan, kemudian dilakukan proses *loop*. Pada proses *loop* ini dilakukan pengecekan tiap nilai yang *density reachable*. Jika data yang dicek ini merupakan titik inti maka akan dibentuk *cluster*. Jika tidak, maka akan dicek apakah titik itu merupakan titik tepi, jika iya maka titik itu akan melakukan proses pengecekan kembali. Jika tidak, maka akan dimasukkan kedalam kategori *noise*. Proses ini berlanjut hingga semua titik diperiksa.