

BAB II **TINJAUAN PUSTAKA**

2.1 Rekam Medis

Peraturan Menteri Kesehatan Nomor 269/MenKes/Per/III/2008, yang dimaksud rekam medis adalah berkas yang berisi catatan dan dokumen antara lain identitas pasien, hasil pemeriksaan, pengobatan yang telah diberikan, serta tindakan dan pelayanan lain yang telah diberikan kepada pasien. Catatan tersebut merupakan tulisan- tulisan yang dibuat oleh dokter atau dokter gigi mengenai tindakan-tindakan yang dilakukan kepada pasien dalam rangka pelayanan kesehatan. Rekam medis adalah rekaman atau catatan mengenai siapa, apa, mengapa, bilamana pelayanan yang diberikan kepada pasien selama masa perawatan yang memuat pengetahuan mengenai pasien dan pelayanan yang diperolehnya serta memuat informasi yang cukup untuk menemukenali (mengidentifikasi) pasien, membenarkan diagnosis dan pengobatan serta merekam hasilnya.

Alur rekam medis terdiri atas beberapa hal, yaitu dimulai dari pendaftaran, distribusi, *assembling*, *coding*, *entry*, dan *filing*. *Coding* artinya menuliskan kode dari diagnosis yang dituliskan oleh dokter. Kode ini diambil dari buku kode diagnosis internasional atau yang dikenal dengan ICD (International Classification of Diseases). *Coding* ini biasanya dikerjakan oleh petugas rekam medis. Setelah proses *coding*, biasanya dilanjutkan dengan *entry* kode diagnosis tersebut ke dalam komputer. Banyak rumah sakit yang tidak memahami pentingnya *coding* ini. Apabila *coding* tidak dilakukan tepat pada waktunya, maka berkas tidak rekam medis belum bisa disimpan dalam lemari rekam medis dan selanjutnya akan mempersulit proses pencarian berkas rekam medis tersebut ketika pasien berkunjung kembali (Huffman, 1994).

2.2 Kecerdasan Buatan

Kecerdasan buatan sebenarnya sudah dimulai sejak musim panas tahun 1956. Pada waktu itu sekelompok pakar komputer, pakar dan peneliti dari disiplin ilmu lain dari berbagai akademi, industri serta berbagai kalangan berkumpul di Dartmouth College untuk membahas potensi komputer dalam rangka menirukan atau mensimulasi kepandaian manusia. Pada mulanya kecerdasan buatan hanya ada di universitas dan laboratorium penelitian, menjelang akhir tahun 1970-an dan awal tahun 1980-an, mulai dikembangkan secara penuh dan hasilnya secara berangsur-angsur mulai dipasarkan (Victor Amrizal & Qurrotul Aini, 2013).

Definisi kecerdasan buatan atau *artificial intelligence* (AI), menurut beberapa pakar:

- a. Schalkoff (1990): AI adalah bidang studi yang berusaha menerangkan dan meniru perilaku cerdas dalam bentuk proses komputasi.
- b. Rich dan Knight (1991): AI adalah studi tentang cara membuat komputer melakukan sesuatu yang, sampai saat ini, orang dapat melakukannya lebih baik.
- c. Luger dan Stubblefield (1993): AI adalah cabang ilmu komputer yang berhubungan dengan otomasi perilaku yang cerdas
- d. Haag dan Keen (1996): AI adalah bidang studi yang berhubungan dengan penangkapan, pemodelan, dan penyimpanan kecerdasan manusia dalam sebuah sistem teknologi informasi sehingga sistem tersebut dapat memfasilitasi proses pengambilan keputusan yang biasanya dilakukan oleh manusia.

Kecerdasan buatan memiliki beberapa bidang, diantaranya adalah sebagai berikut (Victor Amrizal & Qurrotul Aini, 2013):

- a. Sistem pakar (*expert system*): Sistem pakar adalah sebuah program komputer yang dirancang untuk memodelkan kemampuan menyelesaikan masalah seperti layaknya seorang pakar (*human expert*).
- b. *Computer Vision* : *Computer Vision* adalah ilmu dan teknologi mesin yang memiliki fitur untuk melihat, di mana mesin mampu mengekstrak

informasi dari gambar yang diperlukan untuk menyelesaikan tugas tertentu.

- e. Logika Samar (*fuzzy logic*): *Fuzzy logic* menyediakan cara sederhana untuk menggambarkan kesimpulan pasti dari informasi yang ambigu, samar-samar atau tidak tepat.
- f. Jaringan saraf tiruan (*neural networks*): Jaringan saraf tiruan adalah salah satu bagian ilmu komputer yang membuat agar komputer dapat melakukan pekerjaan seperti dan sebaik yang dilakukan oleh manusia.
- g. Sistem Pendukung Keputusan: Secara umum DSS adalah sistem berbasis komputer yang interaktif, yang membantu mengambil.
- h. keputusan dengan memanfaatkan data dan model untuk menyelesaikan masalah-masalah yang terstruktur.
- i. *Data Mining*: *data mining* merupakan gabungan sejumlah disiplin ilmu computer yang didefinisikan sebagai proses penemuan pola-pola baru dari kumpulan-kumpulan data yang sangat besar, meliputi metode-metode yang merupakan irisan dari kecerdasan buatan (*artificial intelligence*).

2.3 Data Mining

Data mining merupakan analisis meninjau sekumpulan data untuk menemukan suatu hubungan yang tidak diduga dan meringkas data secara berbeda dengan sebelumnya yang bermanfaat dan dipahami oleh pemilik data. *Data mining* mewarisi banyak aspek dan teknik dari berbagai bidang ilmu, data mining memiliki akar yang panjang dari bidang ilmu seperti kecerdasan buatan (*artificial intelligent*), *machine learning*, *statistic*, *database* serta *information retrieval* (Larose, 2005).

Data mining adalah bagian dari *Knowledge Discovery in Database* (KDD). *Knowledge Discovery in Database* (KDD) adalah keseluruhan proses untuk mencari dan mengidentifikasi pola atau informasi data, dimana pola yang ditemukan bersifat sah, baru, dapat bermanfaat dan dapat dimengerti. Secara garis besar proses KDD terdiri atas beberapa tahap adalah sebagai berikut:

1. Data Selection

Pemilihan (seleksi) data dilakukan dari suatu kumpulan data operasional. Sebelum tahap penggalian informasi dalam KDD dimulai proses ini perlu dilakukan. Data hasil seleksi disimpan dalam suatu berkas, terpisah dari basis data operasional.

2. Data Cleaning

Proses *cleaning* perlu dilakukan pada data yang menjadi fokus KDD sebelum proses *data mining* dapat dilakukan. Proses *cleaning* melingkupi antara lain membuang data yang memiliki duplikasi, data yang tidak konsisten diperiksa, dan memperbaiki kesalahan pada data, seperti kesalahan cetak (*typo*), juga dilakukan proses *enrichment*, yaitu proses “memperkaya” data yang sudah ada dengan data atau informasi lain yang relevan dan diperlukan untuk KDD, seperti data atau informasi eksternal.

3. Data Transformation

Proses transformasi pada data yang telah dipilih adalah *coding*, sehingga sesuai untuk proses *data mining*. Proses *coding* dalam KDD merupakan proses kreatif dan sangat tergantung pada jenis atau pola informasi yang akan dicari dalam basis data.

4. Data Mining

Data mining merupakan proses untuk mencari suatu pola atau informasi yang menarik dalam data yang terpilih dengan teknik atau metode tertentu. *Data mining* memiliki teknik, metode, atau algoritma dalam sangat bervariasi. Pemilihan metode dan algoritma yang tepat sangat bergantung pada tujuan dan proses KDD secara keseluruhan.

5. Interpretation/Evaluation

Interpretation merupakan proses untuk menampilkan pola informasi yang dihasilkan dari proses *data mining* oleh pihak yang berkepentingan. Tahap ini

meliputi pemeriksaan terhadap pola atau informasi yang ditemukan agar tidak bertentangan dengan fakta atau hipotesis yang ada sebelumnya (Fayyad, 1996).

2.4 Clustering

Clustering adalah metode penganalisaan data untuk menemukan suatu kelompok- kelompok dari sekumpulan objek atau individu yang memiliki karakteristik yang sama. *Clustering* merupakan salah satu metode dalam *data mining*. Di dalam *clustering* terdapat dua pendekatan. Dua pendekatan utama adalah *clustering* dengan pendekatan partisi dan *clustering* dengan pendekatan hirarki. *Clustering* dengan pendekatan partisi (*partition-based clustering*) adalah mengelompokkan data dengan memilah-milah data yang dianalisa ke dalam *cluster* yang ada. *Clustering* dengan pendekatan hirarki (*hierarchical clustering*) adalah mengelompokkan data dengan membuat hirarki berupa dendogram yaitu data yang mirip ditempatkan pada hirarki yang berdekatan sedangkan yang tidak diletakkan pada hirarki yang berjauhan (Oliveira, *et al.* 2007).

2.5 K-Means Clustering

Metode *K-Means* pertama kali diperkenalkan oleh MacQueen JB pada tahun 1976. *K-Means* merupakan salah satu metode clustering non hirarki yang berusaha mempartisi data yang ada ke dalam bentuk satu atau lebih *cluster*. Metode ini mempartisi data ke dalam bentuk satu atau lebih *cluster*/kelompok, sehingga data yang memiliki karakteristik yang sama dikelompokkan dalam satu cluster yang sama dan data yang memiliki karakteristik berbeda dikelompokkan ke dalam kelompok lain (Agusta, 2007).

Algoritma *K-Means* merupakan sebuah metode sederhana untuk membagi suatu kumpulan data dalam suatu angka spesifik dari *cluster*, yaitu *k*. Algoritma atau metode *K-Means* ditemukan oleh beberapa peneliti dengan disiplin ilmu berbeda-beda yaitu oleh Lloyd (1957, 1982), Forgery (1965), Friedman dan Rubin (1967), dan terakhir adalah McQueen (1967) dalam (Wu,

et al. 2008). Disebutkan bahwa metode *K-Means* adalah metode yang cepat dan efisien yang dapat digunakan dalam *clustering* data (Larose, 2005).

Menurut Sarwono (2011), algoritma *K-Means* adalah sebagai berikut:

1. Menentukan k sebagai jumlah *cluster* yang ingin dibentuk
2. Membangkitkan nilai *random* untuk pusat *cluster* awal (*centroid*) sebanyak k
3. Menghitung jarak setiap data input terhadap masing-masing *centroid* menggunakan rumus jarak *Euclidean* (*Euclidean Distance*) hingga ditemukan jarak yang paling dekat dari setiap data dengan *centroid*.

Berikut adalah rumus persamaan *Euclidean Distance*:

$$d(x_i, \mu_j) = \sqrt{(x_i - \mu_j)^2} \dots (2.1)$$

dimana:

x_i : data kriteria

μ_j : *centroid* pada *cluster* ke- j

4. Mengklasifikasikan setiap data berdasarkan kedekatannya dengan *centroid* (jarak terkecil)
5. Memperbaharui nilai *centroid*. Nilai *centroid* baru diperoleh dari rata-rata *cluster* yang bersangkutan dengan rumus:

$$\mu^{(t+1)} = \frac{1}{N_{sj}} \sum_{j=1}^s X_j \dots (2.2)$$

dimana:

$\mu^{(t+1)}$: *centroid* baru pada iterasi ke $(t+1)$

N_{sj} : banyak data pada *cluster* s_j

6. Melakukan perulangan dari langkah 2 hingga 5 hingga anggota tiap *cluster* tidak ada yang berubah
7. Jika langkah 6 telah terpenuhi, maka nilai pusat *cluster* (μ_j) pada iterasi terakhir akan digunakan sebagai parameter untuk menentukan klasifikasi data

2.6 Odoo Framework

Odoo *Framework* adalah sebuah platform *open source* yang digunakan untuk keperluan bisnis. Aplikasi atau modul-modul yang terintegrasi dibangun di atas platform tersebut, meliputi semua bisnis dari Sales, stok, CRM dan lain sebagainya. Odoo dibangun menggunakan bahasa pemrograman *Python*, *XML*, dan *JavaScript*. Odoo termasuk sebagai *software Enterprise Resource* (Daniel Reis, 2019).

Planning (ERP). Odoo dulunya dikenal sebagai Open ERP. Odoo dibangun secara *open source*, sehingga Odoo mendukung pemanfaatan kembali *library* yang telah ada dan setiap orang dapat terlibat dalam pengembangannya. Pada platform Odoo terdiri dari tiga komponen utama, yaitu : Komponen *database PostgreSQL* sebagai *database* bawaannya, komponen *application server* Odoo serta komponen *web server*. *Database PostgreSQL* menampung semua data yang berhubungan dengan data dan konfigurasi Odoo. Selain sebagai aplikasi, Odoo juga dapat berfungsi sebagai kerangka kerja bagi para *Software Developer* atau *Framework*. Odoo menyediakan modul-modul dasar yang mendukung fungsi bisnis, sehingga setiap modul dapat dikostumisasi sesuai dengan kebutuhan. Modul-modul tersebut terdiri dari 8 kelompok aplikasi, antara lain:

1. Aplikasi *sales management* : *Customer Relationship Management* (CRM), *point of sale*, *Sales*, *subscriptions*.
2. Aplikasi *website* : *website builder*, *blog*, *e-Commerce*, *forum*, *slides*, *live chat*, *appointments*.
3. Aplikasi operasi : *inventory*, *timesheets*, *project*, *purchase*, *helpdesk*, *documents*
4. Aplikasi keuangan : *accounting*, *invoicing*, *expenses*.
5. Aplikasi marketing : *marketing automation*, *mass mailing*, *events*, *survey*.
6. Aplikasi human resources : *recruitment*, *employees*, *fleet*, *leaves*, *appraisal*.
7. Aplikasi komunikasi : *discuss*, *e-Signature*.
8. Aplikasi manufaktur : *MRP*, *PLM*, *equipment*, *quality*.

2.7 Website

Website adalah sekumpulan file hypertext yang saling terhubung yang menyajikan informasi menggunakan teks-teks dan pola-pola grafis yang terbatas dengan memanfaatkan jaringan internet untuk bisa diakses oleh pengguna (Roger S. Pressman, 2012).

2.8 Basis Data (*Database*)

Basis Data (*Database*) adalah tempat penyimpanan data dan informasi secara terstruktur dan teratur, yang digunakan sebagai pangkalan data atau tempat berkumpulnya data secara digital. Pada *database* terdapat sebuah sistem yang disebut *DBMS (Database Management System)*, *DBMS* adalah sebuah sistem untuk membantu aplikasi dan pengguna di dalam melakukan manajemen data dan *database*. Beberapa contoh *DBMS* yang umum digunakan di dunia komputer, antara lain adalah *MySQL*, *Postgre SQL*, *Oracle*, *Microsoft SQL Server*, *Maria DB*, *Mongo DB*, dan lainnya, (I Putu Agus Eka Pratama, 2018).

2.9 Definisi *PostgreSQL*

PostgreSQL merupakan sebuah sistem basis data yang disebarluaskan secara bebas menurut Perjanjian lisensi BSD. Piranti lunak ini merupakan salah satu basis data yang paling banyak digunakan saat ini, selain *MySQL* dan *Oracle*. *PostgreSQL* adalah sistem *database* yang kuat untuk urusan relasi, *open source*. Memiliki lebih dari 15 tahun pengembangan aktif dan sudah terbukti segala rancangan arsitekturnya telah mendapat reputasi tentang kuat, handal, integritas data, dan akurasi data (Regina Obe, *et al.*,2016).

2.10 Python

Python merupakan bahasa pemrograman *multi-platform* yang bersifat *free* dan *open-source*, dan dapat digunakan untuk mengembangkan aplikasi-aplikasi *desktop* maupun *web*. Python memiliki pustaka standar (*Python Standard Library*) yang sangat lengkap sehingga dapat memenuhi berbagai macam permasalahan-permasalahan didalam dunia pemrograman, sebagai alternatif dari bahasa-bahasa pemrograman lain seperti C, C++, *Java*, *PHP*, dll. Untuk

kepentingan yang spesifik, kode *python* juga dapat di integrasikan dengan pustaka lain yang ditulis di dalam bahasa C, C++, *Java* (melalui *Jython*), dan bahasa-bahasa .NET seperti Visual Basic dan C# (melalui *IronPython*) (Budi Rahardjo, 2019).

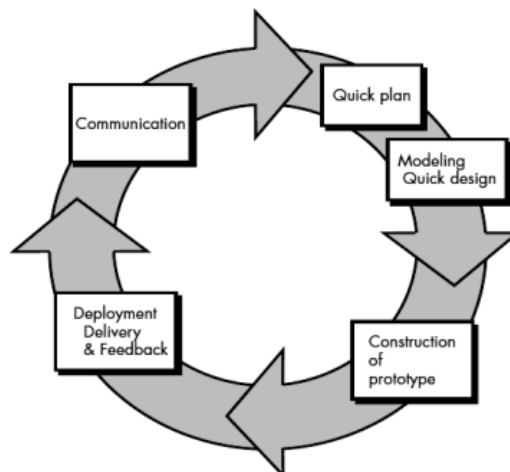
2.11 Metode Prototype

Prototype digunakan untuk menggali kebutuhan secara lebih cepat. Biasanya saat pembuatan *prototype*, keterlibatan user sangat dibutuhkan. Manfaat utama prototype adalah untuk mengurangi resiko tidak diterimanya hasil pengembangan suatu perangkat lunak serta pengulangan kerja di kemudian hari, (Munawar, 2018).

Manfaat itu diantaranya :

1. Lebih antusiasnya pengguna akhir dan pelanggan dalam proses penggalian kebutuhan dan umpan balik.
2. Mengurangi resiko proyek karena penggunaan user interface sudah dieksplorasi dari awal.
3. Kemudahan dalam penggunaan dapat lebih ditingkatkan.
4. Bisa menjembatani antara produk perangkat lunak dengan kebutuhan user.

Gambar 2.1 berikut ini adalah tahapan-tahapan dalam *Prototype* :



Gambar 2.1 Model *Prototype*

Langkah-langkah dalam metode *Prototype* dimulai dengan dilakukannya Komunikasi antara Tim Pengembang perangkat lunak dengan para pelanggan.

Tim pengembang perangkat lunak akan melakukan pertemuan-pertemuan dengan para *Stakeholder* untuk mendefinisikan sasaran keseluruhan untuk perangkat lunak yang akan dikembangkan, mengidentifikasi spesifikasi kebutuhan apa pun yang saat ini diketahui, dan menggambarkan area-area dimana definisi lebih jauh pada literasi selanjutnya merupakan keharusan. Literasi pembuatan *Prototype* direncanakan dengan cepat dan pemodelan (dalam bentuk “Rancangan Cepat”) dilakukan. Suatu rancangan cepat berfokuskan pada representasi semua aspek perangkat lunak yang akan terlihat oleh para pengguna akhir (Misalnya rancangan antarmuka pengguna User atau format tampilan). Rancangan Cepat (*Quick Design*) akan memulai konstruksi pembuatan *Prototype*. Lalu kemudian akan diserahkan kepada para *stakeholder* dan kemudian mereka akan melakukan evaluasi-evaluasi tertentu terhadap *Prototype* yang telah dibuat sebelumnya, kemudian akan memberikan umpan balik, (Roger S. Pressman, 2012 : 50).

2.12 Unified Modelling Language (UML)

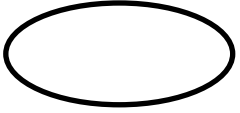
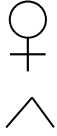

UML (*Unified Modelling Language*) adalah salah satu alat bantu yang sangat handal di dunia pengembangan system yang berorientasi objek. Hal ini disebabkan karena UML menyediakan bahasa pemodelan visual yang memungkinkan bagi pengembang sistem untuk membuat cetak biru atas visi mereka dalam bentuk yang baku, mudah dimengerti. UML merupakan kesatuan dari bahasa pemodelan yang dikembangkan oleh Booch, *Object Medeling Technique* (OMT) dan *Object Oriented Software Engineering* (OOSE). Metode ini menjadikan proses analisis dan desain ke dalam empat tahapan interatif, yaitu : identifikasi kelas-kelas dan objek-objek, identifikasi semantic dari hubungan objek dan kelas tersebut, perincian *interface* dan implementasi. UML dibangun atas model 4+1 *view*. Yaitu *LogicalView*, *Development View*, *Process View*, *Physical View* dan *Scenario*. Model ini di dasarkan pada fakta bahwa struktur sebuah system dideskripsikan dalam 5 *View*, yang salah satunya adalah *Scenario* (Munawar, 2018).


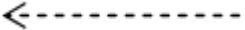
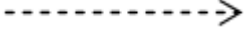

2.12.1 Use Case Diagram

Use Case adalah deskripsi fungsi sebuah system dari perspektif pengguna. *Use Case* bekerja dengan cara mendeskripsikan tipikal interaksi antara pengguna sebuah system dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah system dipakai. Urutan langkah-langkah yang menerangkan antara pengguna dan system disebut sebagai *Scenario*. Setiap *Scenario* menggambarkan urutan kejadian. Setiap urutan di inialisasi oleh orang, system yang lain, perangkat keras atau urutan waktu. Dengan demikian, secara singkat bias dikatakan *Use Case* adalah serangkaian *Scenario* yang digabungkan bersaa-sama oleh tujuan umum pengguna. *Use Case* dibuat berdasarkan kebutuhan Aktor. *Use Case* harus merupakan 'apa' yang dikerjakan software aplikasi, bukan 'bagaimana' software aplikasi mengerjakannya (Munawar, 2018).

Tabel 2.1 pada halaman berikut ini adalah Simbol-simbol yang digunakan dalam *Use Case Diagram* :

Tabel 2.1 Simbol *Use Case Diagram*

No	Simbol	Keterangan
1	Nama Use Case 	Fungsionalitas yang disediakan system sebagai unit yang saling bertukar pesan antara unit atau factor, biasanya dinyatakan dengan menggunakan kata kerja awal frase nama <i>Use Case</i>
2	Aktor 	Orang, proses atau sistem lain yang berinteraksi dengan system yang akan dibuat di luar system yang akan dibuat itu sendiri. Jadi, walaupun symbol dari aktor adalah gambar orang, tapi aktor belum tentu orang. Biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.
3	Asosiasi 	Komunikasi antara aktor dan <i>Use Case</i> , atau <i>Use Case</i> dan Aktor
4	Generalisasi	Hubungan <i>Generalisasi</i> dan <i>Spesialisasi</i>



		(Umum - Khusus) antara dua buah <i>Use Case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
5	Ekstensi / <i>Extend</i> 	Relasi <i>Use Case</i> tambahan ke sebuah <i>Use Case</i> dimana <i>Use Case</i> yang ditambahkan dapat berdiri sendiri walaupun tanpa <i>Use Case</i> tambahan.
6	<i>Include</i>  <i>Uses</i> 	Relasi <i>Use Case</i> tambahan ke sebuah <i>Use Case</i> , dimana <i>Use Case</i> yang ditambahkan memerlukan <i>Use Case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>Use Case</i> ini.

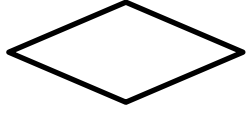
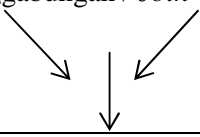

2.12.2 Activity Diagram

Diagram adalah bagian penting dari UML yang menggambarkan aspek dinamis dari Sistem. Logika Prosedural, proses bisnis dan aliran kerja suatu bisnis bisa dengan mudah di deskripsikan dalam *Activity Diagram*. *Activity Diagram* mempunyai peran seperti halnya *Flowchart*, akan tetapi perbedaannya dengan *Flowchart* adalah *Activity Diagram* bisa mendukung perilaku paralel sedangkan *Flowchart* tidak bisa, (Munawar, 2018 : 127).

Tabel 2.2 berikut ini adalah Simbol-simbol yang digunakan dalam *Activity Diagram* :

Tabel 2.2 Simbol *Activity Diagram*

No	Simbol	Keterangan
1	Status Awal 	<i>Start point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
2	Aktivitas 	Aktivitas yang dilakukan suatu sistem.

3	Percabangan / <i>Decision</i> 	Symbol ini digunakan jika ada pilihan aktivitas lebih dari satu atau menggabungkan dua kegiatan paralel menjadi satu.
4	Penggabungan / <i>Join</i> 	Penggabungan / <i>Join</i> digunakan untuk menunjukkan adanya kegiatan yang digabungkan.
5	Status Akhir 	Status Akhir, adalah akhir dari aktifitas sebuah sistem.

2.12.3 Class Diagram

Class diagram yaitu salah satu jenis diagram pada UML yang digunakan untuk menampilkan kelas-kelas maupun paket-paket yang ada pada suatu sistem yang nantinya akan digunakan. Jadi diagram ini dapat memberikan sebuah gambaran mengenai sistem maupun relasi relasi yang terdapat pada sistem tersebut, (Munawar, 2018 : 101).

2.13 Penelitian Terkait

Tabel 2.3 berikut ini adalah penelitian terkait yang berkaitan dengan penelitian yang dilakukan saat ini:

Tabel 2.3 Penelitian Terkait

No	Nama Penulis	Judul / Tahun Terbit	Uraian
1.	Muhammad. H, & Rizky T.S	Clustering Pada Data Mining Untuk Mengetahui Potensi Penyebaran Penyakit DBD Menggunakan Metode Algoritma K-Means dan Metode Perhitungan Jarak Euclidean Distance Jurnal Skanika. Vol 01 No. 01, Maret 2018, pp 1-6	Metodologi yang digunakan pada penelitian kali ini menggunakan metode penelitian Studi Literatur dan Observasi, Metode pengembangan perangkat lunak pada penelitian ini menggunakan metode Waterfall. Maka ditarik kesimpulan bahwa metode klasterisasi k-means dapat digunakan untuk mengelompokkan daerah potensi. Dengan menggunakan 3 pengelompokan atau klasterisasi.

			<p>Penelitian sebelumnya dilakukan dengan metode yang berbeda, dengan tujuan yang berbeda dan menggunakan metode perangkat lunak <i>Prototype</i> serta cluster yang ditentukan yaitu menjadi 4 cluster atau pengelompokan.</p>
2.	Putra May Chandra Abrianto	<p>Penerapan Metode K-Means Clustering untuk Pengelompokan Pasien Penyakit Liver.</p> <p>JATI. (Jurnal Mahasiswa Teknik Informatika). Vol. 02 No. 02. Hal: 1-9. 2018.</p>	<p>Beberapa tahapan pengujian yang telah dilakukan dalam penelitian ini yaitu dengan metode <i>K-Means Clustering</i>, hasil dikelompokkan dalam 3 kategori yaitu Tinggi, Normal dan Rendah. Metode K-Means Clustering yang diterapkan dalam sistem berjalan dengan baik dimana ditunjukkan pada pengujian hasil akhir metode dengan tingkat kesesuaian 86% untuk perbandingan perhitungan sistem dengan perhitungan manual. Metode K-Means Clustering yang diterapkan dalam sistem berjalan dengan tingkat kesesuaian 66% untuk perbandingan.</p>
			<p>Penelitian dilakukan dengan metode yang sama yaitu Metode <i>K-Means Clustering</i> dimana penelitian sebelumnya hanya terfokus dengan satu penyakit, namun pada penelitian menentukannya berdasarkan beberapa variabel dengan menggunakan rumus algoritma k-means clustering (rumus ecludien distance) sehingga menghasilkan nilai yang lebih akurat.</p>
3.	Fitra. K, Ichsan. F, Erwin.W, Godlief E.S., & Mige.	<p>Penerapan algoritma K-Means untuk Pengelompokan Diagnosa Penyakit Mata berdasarkan Rentang Usia</p> <p>Jurnal SPEKTRO. Vol. 02 No. 01, Hal:1-8, Mei 2019.</p>	<p>Penyakit mata dikelompokkan menjadi 3 cluster, yaitu kelompok jumlah penyakit Sedang, penyakit Tinggi dan penyakit Rendah sedangkan kategori pasien dikelompokkan menjadi 3, yaitu kelompok usia Balita dan Anak-Anak, usia Remaja dan Dewasa serta kelompok usia Tua.</p> <p>Penyakit <i>Cataract</i> menempati urutan pertama sebagai diagnosa penyakit terbanyak berada di kelompok Usia Tua sedangkan pada kelompok usia Balita dan Anak-Anak, Remaja dan Dewasa, rentan terhadap penyakit <i>Myopia</i> sebagai diagnosa penyakit terbanyak.</p>
			<p>Penggunaan metode yang sama namun</p>

	Pengembangan yang dilakukan	tujuan dalam penelitian ini berbeda, dimana tujuannya lebih terperinci yang terdiri dari 4 cluster dengan pengujian berdasarkan rumus dari Algoritma K-Means Clustering yang akan menghasilkan data yang lebih lengkap. Penelitian ini juga tidak berfokus hanya pada satu kategori variabel namun, menggunakan 5 variabel yaitu berdasarkan diagnose, jenis kelamin, umur, status kunjungan, dan alamat sehingga nilai yang didapat dari hasil pengujian lebih akurat.	
	Castaka A.S, Ayu H.R, & Aditia G	Algoritma K-Means Untuk Pengelompokan Penyakit Pasien Pada Puskesmas Cigugur Tengah. JOINT (Journal of Information Technology), Vol. 02 No. 02, pp: 39-44, Agustus 2020.	Dalam penelitian ini penulis mengelompokan penyakit pasien kedalam penyakit akut dan tidak akut, maka jumlah <i>cluster</i> yang digunakan dalam penelitian ini sebanyak 2 <i>cluster</i> , yaitu <i>cluster_0</i> dan <i>cluster_1</i> . Dimana <i>cluster_0</i> untuk mengelompokan penyakit akut. Dengan menggunakan metode pengujian algoritma k-means clustering.
	Pengembangan yang dilakukan	Penelitian yang dilakukan menggunakan metode yang sama, dengan variabel pengamatan yang berbeda, dimana penelitian ini akan menghasilkan 10 penyakit endemi disetiap alamat yang ada dimasing-masing kelurahan, yang terdapat pada satu kecamatan. Penelitian menggunakan metode pengembangan perangkat lunak dan bahasa pemograman yang berbeda, yang menjadi salah satu pengembangan dalam penelitian ini.	