

BAB II

LANDASAN TEORI

2.1 Sejarah Instansi

PT. Sejahtera Kelola Abadi didirikan pada tanggal 05 Oktober 2009 bergerak dalam bidang usaha pemasaran peralatan dan pengelolaan mall. Perusahaan ini dipimpin oleh Bapak Christoper Raharja, kantor beralamat di Jl. Prof. DR. Satrio No.Kav 18, Kuningan, Karet Kuningan, Kecamatan Setiabudi, Kota Jakarta Selatan, Kode Pos 12940.

Tujuan berdirinya PT. Sejahtera Kelola Abadi antara lain adalah sebagai berikut:

- a) Menjadi perusahaan terbaik di Indonesia dalam bidang usaha pemasaran peralatan dan pengelolaan mall.
- b) Menyediakan lapangan usaha maupun lapangan kerja bagi masyarakat Indonesia.
- c) Selalu menjaga kepercayaan yang diinginkan konsumen dan menjaga nama baik PT. Sejahtera Kelola Abadi.

2.2 Kajian Terhadap Penelitian Yang Terkait Sebelumnya

1. Pertama, Penelitian terkait yang menjadi referensi pada penelitian ini adalah skripsi yang berjudul: “Penentuan Pola Penjualan Media Edukasi dengan Menggunakan Metode Algoritme Apriori dan FP-Growth”. Analisa dan pengujian

terhadap kajian penentuan pola penjualan media edukasi dengan menggunakan algoritme asosiasi Apriori dan FP-Growth (studi kasus : Cerdas-Sehat Online Shop) maka dengan menggunakan langkah-langkah algoritme Apriori dari tahap pemilihan sampai evaluasi berdasarkan data transaksi penjualan media edukasi untuk menentukan pola penjualan menghasilkan 5 pola aturan asosiasi minimum support sebesar 10% dan minimum confidence sebesar 50% menghasilkan nilai confidence tertinggi yaitu 100%.

Sedangkan algoritme FP- Growth menghasilkan 5 pola aturan asosiasi dengan minimum support count 2 menghasilkan nilai support count tertinggi yaitu 8. Dan juga menggunakan 2 atribut yaitu no.order dan kode produk dapat menemukan kecenderungan pola kombinasi antar itemset sehingga dapat dijadikan informasi yang sangat penting dalam pengambilan keputusan yang berguna untuk mempersiapkan jenis stok barang apa yang diperlukan kedepanya bagi penjualan di Cerdas– Sehat Online Shop.

2. Kedua, Penelitian terkait yang menjadi referensi pada penelitian ini adalah skripsi yang berjudul: “PENERAPAN DATA MINING MENGGUNAKAN ALGORITMA FP-TREE DAN FP-GROWTH PADA DATA TRANSAKSI PENJUALAN OBAT”. Data Mining dengan Algoritma FP-Growth dapat diimplementasikan dengan menggunakan database penjualan obat karena dapat menemukan pola kombinasi itemsets. Sehingga informasi tersebut dapat membantu mengembangkan strategi penjualan terhadap konsumen.

Metode Association Rule dengan menggunakan Algoritma FP-Growth dengan parameter support dan confidence dapat memperoleh korelasi barang pembelian untuk lebih meningkatkan penjualan. Setelah dilakukan pengimplementasian Algoritma FP-Growth pada Rapidminer, maka hasil pengolahan data penjualan obat paling banyak terjual pada apotek adalah Fungoral, Alopurinol, Diatizem HCL, Batugin dan Ketoconazole.

2.3 Sistem Informasi

Sistem informasi mencakup sejumlah komponen (manusia, komputer, teknologi informasi dan prosedur kerja), ada sesuatu yang diproses (data menjadi informasi), dan dimaksudkan untuk mencapai suatu sasaran atau tujuan. Kumpulan perangkat keras dan perangkat lunak yang dirancang untuk mentransformasikan data ke dalam bentuk informasi yang berguna. (Abdul, 2018)

Menurut Alter sistem informasi adalah kombinasi antar prosedur kerja, informasi, orang dan teknologi informasi yang diorganisasikan untuk mencapai tujuan dalam sebuah organisasi.

Menurut Wilkinson sistem informasi adalah kerangka kerja yang mengordinasikan sumber daya (manusia, komputer) untuk mengubah masukan (input) menjadi keluaran (informasi), guna mencapai sasaran-sasaran perusahaan. (Abdul, 2018)

2.4 ALGORITMA APRIORI

Algoritma Apriori adalah salah satu algoritma pada data mining untuk mencari frequent item/itemset pada transaksional database. Algoritma apriori pertama kali diperkenalkan oleh R.Agarwal dan R Srikant untuk mencari frequent tertinggi dari suatu database, Kaur et al. Penggunaan bottom-up pendekatan berulang. Untuk menentukan asosiasi rule mining sebuah transaksi database, diperlukan waktu dalam melakukan proses frequent itemset, menghasilkan kombinasi data yang cukup banyak, Abdullah. Proses ini dilakukan untuk mencari minimum nilai support dan minimum nilai confidence.

Algoritma apriori sangat mudah dipahami, tetapi ada beberapa kekurangan pada algoritma tersebut:

1. Database Scanning: Database transaksi perlu dipindai berulang kali untuk menemukan frequent itemset. Jika ada n item dalam database, membutuhkan minimal n kali memindai database.

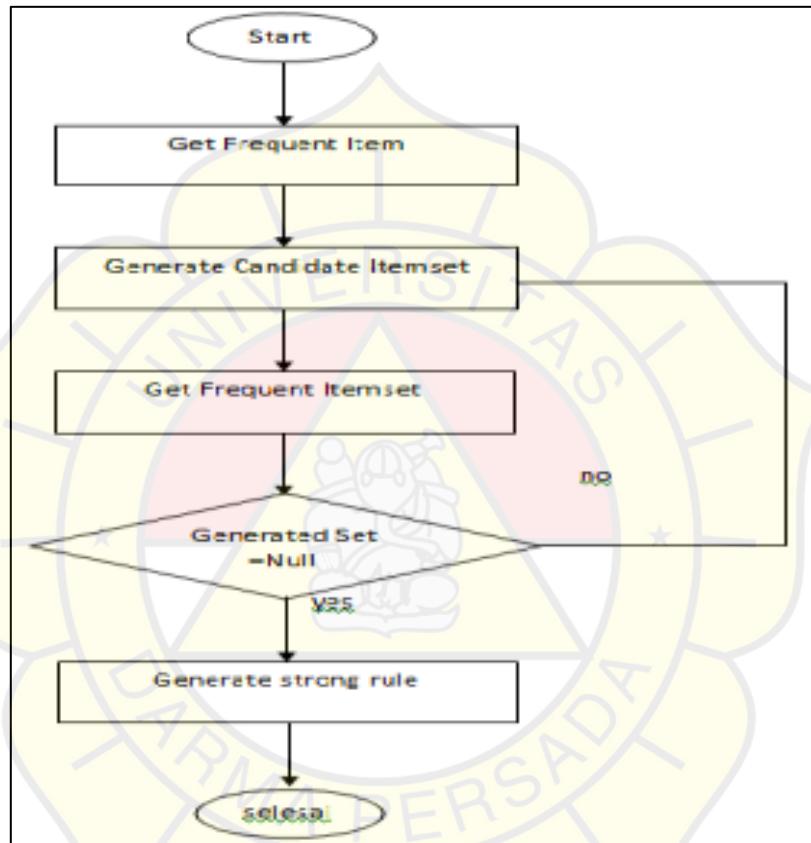
2. Pengaturan minimal frequent item/itemset untuk menentukan nilai support minimum.

3. Aturan Asosiasi rule mining dalam mendapatkan nilai minimum confidence

Langkah-langkah algoritma apriori sebagai berikut:

1. Join(penggabungan). Pada proses ini setiap item dikombinasikan dengan item yang lainnya sampai tidak terbentuk kombinasi lagi.

2. Prune(pemangkasan). Pada proses ini, hasil dari item yang telah dikombinasikan tadi lalu dipangkas dengan menggunakan minimum support yang telah ditentukan. Dua proses utama tersebut merupakan langkah yang akan dilakukan untuk mendapat frequent itemset pada algoritma Apriori.



Gambar 2.1 Deskripsi Algoritma Apriori, Sumber :Hasil Penelitian

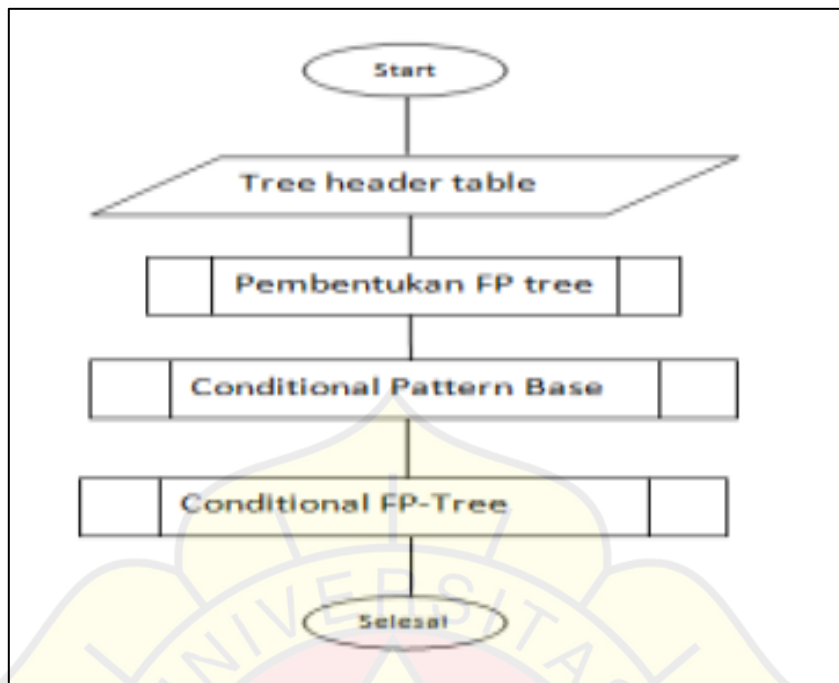
2.5 ALGORITMA FP-GROWTH

Algoritma FP-Growth merupakan salah satu alternatif algoritma yang cukup efektif untuk mencari himpunan data yang paling sering muncul (frequent itemset) dalam sebuah kumpulan data yang besar. Algoritma FP-Growth merupakan algoritma

Association Rules yang cukup sering dipakai. Algoritma FP-Growth ini dikembangkan dari algoritma apriori. Algoritma apriori menghasilkan kombinasi yang sangat banyak sehingga data yang dihasilkan tidak efisien (Kezia Sumangkut, et al, 2018).

Algoritma FPtree dibagi menjadi tiga langkah utama, yaitu:

1. Tahap Pembangkitan Conditional Pattern Base merupakan subdatabase yang berisi prefix path (lintasan e:1 prefix) dan pattern (pola akhiran). Pembangkitan conditional pattern base didapatkan melalui FP-tree yang telah dibangun sebelumnya.
2. Tahap Pembangkitan Conditional FPtree pada tahap ini, support count dari setiap item pada setiap conditional pattern base dijumlahkan, lalu setiap item yang memiliki jumlah support count lebih besar sama dengan minimum support count akan dibangkitkan dengan conditional FPtree.
3. Tahap Pencarian frequent itemset apabila conditional FP-tree merupakan lintasan tunggal(single path), maka didapatkan frequent itemset dengan melakukan kombinasi item untuk setiap conditional FP-tree. Jika bukan lintasan tunggal, maka dilakukan pembangkitan FP-growth secara rekursif. Ketiga tahap tersebut merupakan langkah yang akan dilakukan untuk mendapatkan frequent itemset. Dengan menggunakan FP-Growth, kita dapat melakukan Pettern Frequent itemset dengan tidak membutuhkan waktu yang cukup lama



Gambar 2.2 Deskripsi FP_Growth, Sumber : Hasil Penelitian

2.6 WEBSITE

Menurut Triyono (2018:23), Website atau situs dapat diartikan sebagai kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau bergerak, animasi, suara, dan atau gabungan dari semuanya baik yang bersifat dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman.

Berdasarkan pengertian para ahli diatas dapat disimpulkan bahwa website adalah kumpulan dari keseluruhan halaman-halaman web yang berisi sebuah data atau informasi baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan

jaringan-jaringan halaman.

Website pada dasarnya merupakan media yang berisi berbagai informasi yang diakses melalui jaringan internet. Website pada umumnya diakses untuk mencari informasi terkait suatu hal yang dibutuhkan oleh yang mengakses website tersebut, pada suatu website memungkinkan untuk terhubung ke website lain.

2.7 APLIKASI WEB

Pada umumnya website ditempatkan pada web server agar dapat diakses oleh penggunaanya dan dibuat menggunakan teknologi-teknologi tertentu untuk disebut sebagai aplikasi web, berikut ini merupakan penjelasan mengenai hal tersebut.

2.7.1. Web Server

Web server adalah suatu sistem yang bertugas menerima request HTTP atau HTTPS dari komputer client, yaitu web browser dan memberikan response HTTP atau HTTPS berupa data seperti halaman web yang terdiri dari dokumen HTML dan objek terkait seperti gambar dan lainnya (Supono & Putratama, 2018).

Web server dapat diakses melalui internet ataupun jaringan lokal, ada beberapa jenis software yang dapat digunakan dalam membangun web server antara lain Apache, Nginx, Lighttpd dan berbagai web server lainnya.

2.7.2. Teknologi Aplikasi Web

1. Mypertext Markup Language (HTML)

Hypertext Markup Language (HTML) merupakan bahasa pemrograman web yang memberitahukan web browser bagaimana menyusun dan menyajikan konten di halaman web (Solichin, 2018).

HTML merupakan dokumen ASCII atau teks biasa. HTML bukan merupakan bahasa pemrograman, HTML hanya berisi perintah-perintah yang telah terstruktur berupa tag-tag penyusun yang nantinya akan diterjemahkan menjadi halaman web oleh web browser. Dokumen HTML dapat dibuat dengan text editor seperti Visual Studio Code, Sublime Text 3, Notepad++ dan lain-lain.

2. Hypertext Preprocessor (PHP)

PHP adalah pemrograman interpreter yaitu proses penerjemahan berisi kode sumber menjadi kode mesin yang dimengerti komputer dilakukan secara langsung pada saat baris kode tersebut dijalankan. PHP disebut sebagai Server Side Programming, hal ini dikarenakan seluruh prosesnya dijalankan pada server (Sibero, 2019).

PHP merupakan bahasa pemrograman dengan hak cipta terbuka atau yang juga dikenal dengan istilah open source. Pemrograman PHP dapat ditulis dalam dua bentuk yaitu penulisan baris kode PHP pada file tunggal dan penulisan kode PHP pada halaman HTML.

3. JavaScript

JavaScript merupakan bahasa pemrograman yang dikembangkan untuk dapat berjalan pada web browser (Sibero, 2019). JavaScript adalah bahasa script yang dikembangkan untuk membuat dokumen yang dinamis. JavaScript merupakan bahasa pemrograman berorientasi objek.

JavaScript memiliki keistimewaan untuk ditambahkan pada kode HTML dan membuat dokumen menjadi lebih interaktif. JavaScript berfokus pada proses pengolahan data di sisi client dan menyajikan komponen web yang lebih interaktif serta berfungsi untuk menambah fungsionalitas dan kenyamanan halaman web (Solichin, 2018).

2.8 DATABASE

Database merupakan sistem terkomputerisasi yang berfungsi untuk memelihara suatu informasi dan membuat informasi tersebut tersedia pada saat dibutuhkan (Sukanto & Shalahuddin, 2018). Jadi, database merupakan suatu sekumpulan data atau informasi yang telah diolah yang disimpan untuk dapat digunakan kembali pada saat data atau informasi tersebut dibutuhkan

2.8.1 MySQL

Menurut (Sibero, 2019), “MySQL adalah suatu *Relational Database Management Sistem* (RDBMS) yaitu aplikasi sistem yang menjalankan fungsi pengolahan data.” MySQL merupakan salah satu aplikasi DBMS yang sudah banyak oleh para pemrogram aplikasi web

MySQL adalah *Relational Database Management System* (RDBMS) yang berfungsi untuk melakukan pengelolaan data dalam membangun suatu aplikasi web. MySQL menggunakan konsep utama dalam database, yaitu SQL (*Structured Query Language*). SQL merupakan struktur bahasa untuk berkomunikasi dengan database.

2.9 METODE PENGEMBANGAN SISTEM

2.9.1. Waterfall

Waterfall merupakan suatu proses pengembangan perangkat lunak yang berurutan yang terus mengalir ke bawah seperti air terjun, yang terdiri dari fase perencanaan, pemodelan, implementasi, pengujian dan pemeliharaan (Pressman, 2012).

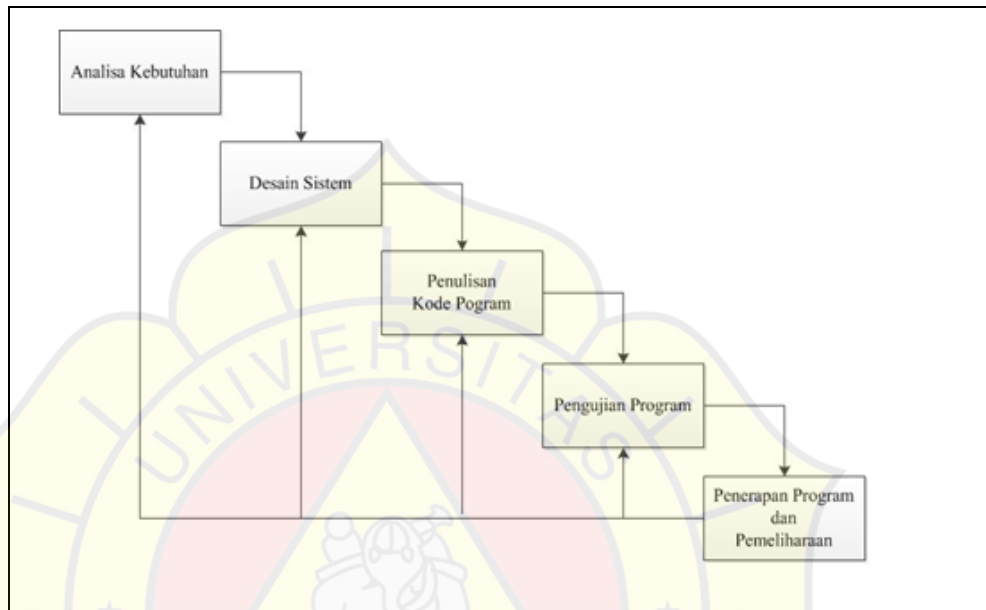
1. Perancangan Perancangan adalah proses pengumpulan kebutuhan secara lengkap yang kemudian dianalisis serta mendefinisikan kebutuhan yang harus dipenuhi oleh sistem yang akan dibangun.

2. Pemodelan Pemodelan atau desain sistem adalah tahap untuk mendefinisikan tampilan sistem secara keseluruhan dan menentukan alur perangkat lunak dalam membangun sistem.

3. Implementasi Implementasi adalah tahapan yang mana seluruh desain kemudian diubah menjadi kode program . Kode program yang dihasilkan yang masih berupa modulmodul akan diintegrasikan menjadi sistem yang lengkap.

4. Pengujian Pengujian adalah tahap penggabungan modul-modul yang kemudian dilakukan pengujian apakah sistem tersebut telah sesuai dengan desain dan tidak ada kesalahan pada setiap fungsi sistem.

5. Pemeliharaan Pemeliharaan adalah tahap setelah sistem telah diimplementasikan dan dilakukan pengembangan akan dilakukan pemeliharaan serta perbaikan pada sistem apabila terdapat kesalahan pada sistem tersebut.



Gambar 2. 3 Metode Waterfall (Sumber : Pressman)

System Development Life Cycle Waterfall Siklus (SDLC) Waterfall dijalankan secara berurutan, mulai dari langkah pertama hingga langkah keenam. Setiap langkah yang telah selesai harus dikaji ulang, kadang-kadang bersama *expert user*, terutama dalam langkah spesifikasi kebutuhan dan perancangan sistem untuk memastikan bahwa langkah telah dikerjakan dengan benar dan sesuai harapan.

2.10 PEMODELAN UML

UML (Unified Modelling Language) merupakan suatu standar bahasa yang digunakan untuk mendefinisikan requirement, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek. (Sukamto & Shalahuddin, 2018).

UML melakukan pemodelan secara visual untuk merancang atau membangun sistem yang berorientasi objek. UML terdiri dari diagram-diagram sistem menurut aspek atau sudut pandang tertentu. Diagram pada UML berfungsi untuk menggambarkan permasalahan atau solusi pada suatu model. Diagramdiagram pada UML terdiri dari use case diagram, activity diagram, sequence diagram dan lain-lain. Diagram UML dikelompokkan menjadi 3 kategori, yaitu:

1. Structure Diagram

Structure Diagram adalah diagram-diagram yang digunakan untuk menggambarkan suatu struktur statis dari suatu sistem yang akan dimodelkan.

2. Behavior Diagram

Behavior Diagram adalah diagram-diagram yang digunakan untuk menggambarkan tingkah laku suatu sistem atau serangkaian perubahan yang terjadi pada suatu sistem.

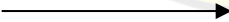
3. Interaction Diagram

Interaction Diagram adalah diagram-diagram yang digunakan untuk menggambarkan interaksi pada suatu sistem dengan sistem lainnya maupun interaksi antar subsistem pada suatu sistem.

2.10.1. Use Case Diagram

Use case diagram digunakan untuk mengetahui fungsi di dalam suatu sistem dan pengguna yang akan menggunakan fungsi-fungsi tersebut (Sukanto & Shalahuddin, 2018).

Tabel 2.1 Komponen *Use Case* Diagram

No	Simbol	Keterangan
1	Use Case 	Merupakan sebuah komponen yang menggambarkan seseorang atau sesuatu (seperti perangkat atau sistem lainnya) yang berinteraksi dengan sistem.
2	Actor 	Use case adalah gambaran fungsionalitas dari suatu sistem, sehingga pengguna sistem paham dan mengerti mengenai kegunaan sistem yang akan dibangun.
3	Asosiasi / Association 	Komunikasi antara aktor dan use case yang berpartisipasi pada use case atau use case memiliki interaksi dengan aktor
4	Ekstensi / <i>Extend</i> <<extend>>	Kelakuan yang hanya berjalan di bawah kondisi tertentu seperti menggerakkan alarm.

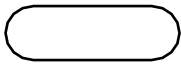
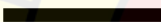
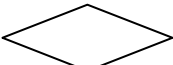
5	Generalisasi / <i>Generalization</i> 	<i>Generalization</i> disebut juga <i>inheritance</i> (pewarisan), sebuah elemen dapat merupakan spesialisasi dari element lainnya.
6	Menggunakan / <i>Include</i> / <i>uses</i> <<include>>	Kelakuan yang harus terpenuhi agar sebuah <i>event</i> dapat terjadi, di mana pada kondisi ini sebuah use case adalah bagian dari use case lainnya.

2.10.2 Activity Diagram

Activity diagram menggambarkan *workflow* atau aktivitas dari suatu sistem. *Activity* diagram menggambarkan aktivitas yang dapat dilakukan oleh suatu sistem. (Sukamto & Shalahuddin, 2018).

Tabel 2.2 Komponen Dalam *Activity* Diagram

<i>Activity Diagram</i>	Keterangan
<i>Start State</i> 	<i>Start State</i> , sebagai tanda awal proses dari <i>activity</i> diagram.
<i>State</i> 	<i>State</i> , berfungsi menampung <i>event</i> dalam <i>activity</i> diagram.

<i>Activity</i> 	<i>Activity</i>, memiliki fungsi yang sama dengan state. Menampung <i>event</i> atau aktivitas pada proses sistem.
<i>State Transition</i> 	<i>State Transition</i>, berfungsi untuk menunjukkan aliran atau urutan dari <i>event</i> atau aktivitas pada diagram.
<i>Transition to self</i> 	<i>Transition to self</i>, berfungsi untuk menunjukkan transisi sebuah <i>event</i> yang mengarah ke <i>event</i> itu sendiri.
<i>Horizontal Synchronization</i> 	<i>Horizontal Synchronization</i>, berfungsi untuk menyinkronisasikan 2 cabang <i>event</i> yang posisinya <i>horizontal</i>.
<i>Vertical Synchronization</i> 	<i>Vertical Synchronization</i>, berfungsi untuk menyinkronisasikan 2 cabang <i>event</i> yang posisinya vertikal.
<i>Decision</i> 	<i>Decision</i>, digunakan ketika terjadi pemilihan 2 kondisi <i>event</i> pada diagram.

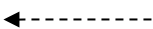
<i>End State</i> 	<i>End State</i> , sebagai tanda akhir dari <i>activity</i> diagram.
---	--

2.10.3 Sequence Diagram.

Sequence diagram menggambarkan tingkah laku objek pada use case dengan mendeskripsikan waktu hidup suatu objek dan pesan yang dikirim atau diterima antar objek tersebut. Dalam menggambar *sequence* diagram perlu diketahui objek yang terlibat pada suatu use case beserta metode yang dimiliki kelas yang didefinisikan menjadi objek tersebut (Sukamto & Shalahuddin, 2018).

Tabel 2.3 Komponen Dalam *Sequence* Diagram

<i>Sequence</i> Diagram	Keterangan
<i>Actor</i> 	<i>Actor</i> , menggambarkan seseorang atau sesuatu (seperti perangkat atau sistem lain) yang berinteraksi dengan sistem.
<i>Boxes</i> 	<i>Boxes</i> , sebuah kontak yang tampil pada posisi paling atas diagram, yang mewakili <i>object</i> , <i>use case</i> , <i>class</i> , dan <i>actor</i> .

<i>Return Message</i> 	<i>Return Message</i>, menggambarkan pesan atau hubungan antara obyek yang menunjukkan urutan kejadian yang terjadi.
<i>Lifeline</i> 	<i>Lifeline</i>, eksekusi obyek selama sequence (<i>message</i> dikirim atau diterima dan aktifasinya).
<i>Message to Self</i> 	<i>Message to Self</i>, menggambarkan pesan atau hubungan obyek itu sendiri yang menunjukkan urutan kejadian yang terjadi
<i>Object Message</i> 	<i>Object Message</i>, menggambarkan pesan atau hubungan antar obyek yang menunjukkan urutan kejadian yang terjadi.