



BAB II

TINJAUAN PUSTAKA

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

Adapun penelitian terkait yang akan digunakan sebagai sumber acuan yang relevan dan terkini yaitu:

Berdasarkan penelitian dari Brilian Sidhatama, dkk (2020). Tujuan Penelitian ini adalah untuk pengembangan berupa suatu aplikasi yang mendukung penerapan *knowledge management system* pada rumah sakit muhammadiyah. Aplikasi ini diharapkan dapat membantu pihak administrasi rumah sakit muhammadiyah dalam pengetahuan tersebut kedalam suatu wadah serta meningkatkan budaya *knowledge sharing*. Kelebihan Penelitian ini Aplikasi KMS yang dibangun dapat membantu rumah sakit muhammadiyah dalam mengumpulkan pengetahuan tersebut kedalam suatu wadah, sehingga tidak akan kehilangan pengetahuan yang dimiliki karyawan dibagian administrasi

Berdasarkan penelitian dari Yudi Akbar (2018), Tujuan penelitian ini adalah, *knowledge management system (KMS)*, untuk meningkatkan kualitas SDM dalam suatu organisasi dengan memperbaiki komunikasi antar karyawan dan meningkatkan penguasaan dengan melakukan *transfer* pengetahuan. *Knowledge Management System (KMS)* merupakan salah satu cara untuk menumbuhkan budaya saling berbagi *knowledge (sharing knowledge)* antar individu maupun organisasi. Pada penelitian ini, penulis mencoba mengembangkan model

knowledge management system yang sesuai dengan kondisi yang ada pada Biro Administrasi Akademik Akademi Pariwisata Nusantara.

Berdasarkan penelitian Purwantoro, dkk (2018). Tujuan penelitian *Knowledge management system* merupakan cara untuk proses sosialisasi (*Socialization*), yaitu Suatu kegiatan tatap muka antar individu (warga masyarakat dan aparat desa dukuh karya) dalam berinteraksi untuk dapat saling berbagi pengetahuannya, *Externalization*. Proses menyajikan *tacit knowledge* kedalam bentuk yang lebih mudah untuk dikomunikasikan dan dimanfaatkan dalam bentuk komunikasi. Menganalisa infrastruktur yang tersedia tersedia pada desa Dukuh karaya pada bidang teknologi lebih banyak menggunakan konsep manual dengan media kertas, meski menggunakan komputer sebagai alat bantu, namun pertukaran informasi tetap menggunakan konsep tradisional dan membutuhkan waktu yang lama dalam proses pelayanan kepada masyarakat.

Berdasarkan penelitian Yance Sonatha, dkk, (2018). Tujuan penelitian yang dapat ditarik dari proses pengembangan KMS. Pertama aplikasi *Knowledge Management System* telah berhasil dikembangkan dan terbukti dapat membantu dan mempermudah pengolahan data dokumentasi IT di PT. Gamatechno Indonesia seperti dokumentasi produk, dokumentasi rapat dan dokumentasi pelatihan. Selain itu aplikasi *Knowledge Management System* juga ini dapat membantu karyawan IT dalam membuat menyimpan data *log problem* dan karyawan IT dapat dengan mudah mencari dan mengunduh dokumentasi yang terdapat di dalam Aplikasi *Knowledge Management System*.

Berdasarkan penelitian yang dilakukan oleh Cahyo Adi Suprpto, Setiawan Assegaff (2017). Tujuan penelitian ini adalah untuk merancang sebuah sistem yang mendukung adanya pendokumentasian dan penyebaran *Knowledge* yang lebih baik keseluruh pendidik dan tenaga kependidikan di SMAN 6 Kota Jambi. Bagaimana merancang suatu *Knowledge Management System* yang dapat mentransfer *Knowledge* dari dokumen, panduan dan solusi dari kasus yang pernah terjadi agar dapat didistribusikan ke seluruh Guru dan Tenaga Kependidikan secara merata. Faktor yang akan diteliti yaitu komponen utama dari *Knowledge Management* (KM) yaitu *people, process, dan technology* dan untuk perancangan *Knowledge Management System* menggunakan teori *Knowledge roadmap* dari Tiwana.

Berdasarkan penelitian dari Rikardo Ramadi (2016) Tujuan Penelitian yang diharapkan dari penelitian ini adalah membangun sistem informasi *Knowledge Management System* yang di gunakan untuk menyimpan dan memanfaatkan *knowledge* yang dimiliki karyawan sehingga mempermudah karyawan untuk menemukan solusi dari masalah yang di hadapi. Untuk metode pengumpulan data peneliti melakukan dengan cara observasi langsung, studi pustaka, dan wawancara. *Knowledge management* dilakukan dengan penciptaan pengetahuan, berbagi pengetahuan dan penerapan pengetahuan. Penciptaan pengetahuan dilakukan dengan model SECI (*sosialization, externalization, combination dan internalization*). Metode pengembangan system dengan *Rapid Application Development (RAD)* dan dimodelkan dengan menggunakan sistem analisis menggunakan *Unified Modelling Language (UML)*. Pengujian validasi menggunakan *Focus Group Discussion (FGD)*.

II.2. Tinjauan Pustaka

II.2.1 Penerapan *Knowledge Management*

Penerapan *Knowledge Management* adalah kerangka kerja untuk merancang tujuan, struktur, dan proses organisasi sehingga organisasi dapat menggunakan pengetahuan mereka untuk belajar dan menciptakan nilai bagi klien dan komunitas mereka. Oleh karena itu, manajemen pengetahuan adalah upaya untuk memperluas pengetahuan dengan kerangka kerja dengan merancang tujuan, struktur, dan proses organisasi untuk menciptakan dan mempertahankan peningkatan nilai organisasi dengan menggunakan apa yang diketahui untuk dipelajari melalui penggunaan teknologi informasi yang ada. (Winardi, Catra Butu, dkk, 2020)

II.2.2 *Knowledge Management System*

Menurut Fahmi Aldi Choirunsyah dan Hery Heryanto (2019), mengatakan bahwa *Knowledge Management System* adalah suatu alat yang bertujuan mendukung manajemen pengetahuan, dan merupakan pengembangan dari alat manajemen informasi yang mengintegrasikan berbagai aspek ilmu computer dalam mendukung kolaborasi antara lingkungan kerja dengan informasi dan sistem pengelolaan dokumen.

Menurut Cahyo Adi Suprpto dan Setiawan Assegaff (2017), mengatakan bahwa *Knowledge Management System* adalah suatu teknologi informasi yang memungkinkan organisasi untuk mengelola pengetahuan secara efektif dan efisien.

Menurut Rikaro Ramadi (2016), mengatakan bahwa *Knowledge Management System* merupakan strategi untuk meningkatkan efektifitas dan peluang/ kesempatan pengembangan kompetensi. Beberapa hal yang perlu dilakukan untuk menumbuhkan budaya berbagi pengetahuan diantaranya :

1. Menciptakan *know-how* dimana setiap pegawai berkesempatan dan bebas menentukan cara baru untuk menyelesaikan tugas dan berinovasi serta peluang untuk mensinergikan pengetahuan eksternal kedalam institusi.
2. Menangkap dan mengidentifikasi pengetahuan yang dianggap bernilai dan direpresentasikan dengan cara yang logis.
3. Penempatan pengetahuan yang baru dengan format yang mudah diakses oleh seluruh pegawai dan pejabat.
4. Pengelolaan pengetahuan untuk menjamin kekinian informasi agar dapat diriview untuk relevansi dan akurasinya.
5. Format pengetahuan yang disediakan di portal adalah format yang user friendly agar semua pegawai dapat mengakses dan mengembangkan setiap saat.

Menurut Entis Sutrisna (2018), *knowledge* dibagi menjadi dua jenis yaitu *Explicit Knowledge* dan *Tacit Knowledge*, yang dapat dijabarkan sebagai berikut:

1. *Explicit Knowledge*

Adalah *knowledge* tertulis, terarsip, tersebar (cetak maupun elektronik) diekspresikan dengan kata-kata dan angka, serta dapat disampaikan dalam bentuk ilmiah, spesifikasi atau manual dan bisa sebagai bahan pembelajaran untuk orang lain.

2. *Tacit Knowledge*

Adalah *knowledge* dari para pakar baik individu maupun masyarakat yang berbentuk *know-how*, pengalaman, skill, pemahaman, maupun petunjuk praktis berada didalam benak orang yang mengetahui.

Menurut Wiwik Suharso (2016), dalam model SECI terjadi empat proses transfer pengetahuan, yaitu :

1. *Socialization (tacit ke tacit)*, adalah proses transfer informasi diantara orang-orang dengan cara *conversation* atau percakapan.
2. *Externalization*, yaitu transfer dari *tacit knowledge* ke *explicit knowledge*.
3. *Combination*, adalah transfer dari *explicit knowledge* ke *explicit knowledge*.
4. *Internalization*, adalah transfer dari *explicit knowledge* ke *tacit knowledge*.

Tabel II.1. Model SECI



Gambar II.1. Konsep Pembagian Informasi /Model Seci (Sumber : Entis Sutrisna : 2018).

Menurut Fauzan Adzima dan Herman Sjahrudin (2019), *Knowledge management* memiliki elemen yang saling terkait satu sama lain, yaitu :

1. *People*

Adalah peran yang sangat penting untuk memberikan kontribusi sebagai penghasil *knowledge* itu sendiri dan penyebar *knowledge*.

2. *Process*

Merupakan tanggapan responden terhadap salah satu bagian dari *knowledge management*. *Process* juga merupakan hal yang berhubungan dengan proses pengambilan nilai-nilai *knowledge* ke dalam suatu media dan kemudian di distribusikan ke setiap individu lainnya untuk digunakan kembali.

3. *Technology*

Merupakan tanggapan responden dan aspek ketiga yang menjadi sebuah alat dalam mendukung unsur *people* maupun *process* dalam meningkatkan pengetahuan terhadap nilai-nilai *knowledge*.

II.2.3. Sentra Pelayanan Kepolisian Terpadu

SPKT merupakan pusat pelayanan kepolisian dalam melayani masyarakat, untuk peningkatan dan memudahkan pelayanan maka diperlukan perangkat lunak yang lebih akurat. Apabila sistem sudah terkomputerisasi, maka petugas bisa melakukan tugasnya lebih cepat agar tidak memakan waktu lama serta tidak menimbulkan antrian masyarakat yang terlalu banyak dan pengarsipan pun akan lebih efektif. Dan pengawasan oleh Kepala Administrasi SPKT dapat secara langsung dilakukan dengan melihat laporan yang telah tersedia pada sistem. (Wahyuni, N. S. 2021)

Salah satu unit atau fungsi yang terdapat di POLRI adalah Sentra Pelayanan Kepolisian Terpadu (SPKT) merupakan unit pelayanan Kepolisian yang disiagakan dan ditempatkan oleh POLRI, dari tingkat wilayah hukum terkecil POLRI yaitu Polsek sampai tingkat pusat yaitu Markas Besar dan ada di seluruh wilayah Republik Indonesia. Dengan fungsi utama memberi pelayanan Kepolisian kepada masyarakat yang membutuhkan, dalam bentuk penerimaan dan penanganan pertama laporan atau pengaduan dari masyarakat seperti Laporan adanya tindak pidana, laporan kehilangan, laporan adanya suatu kejadian maupun bencana, permintaan bantuan Kepolisian dan penyelesaian perselisihan antar warga. (Erni Widarti, dkk, 2020)

II.2.4. Website

Website atau *Situs Web* adalah sejumlah halaman *web* yang memiliki topik saling terkait, terkadang disertai pula dengan berkas-berkas gambar, video, atau berkas lainnya. *Situs web* merupakan kumpulan dari halaman *web* yang sudah dipublikasikan di jaringan *internet* dan memiliki *domain/URL (Unifed Resource Locator)* yang dapat diakses semua pengguna *internet* dengan cara mengetikkan alamatnya. Ditinjau dari aspek *content* atau isi, *web* dapat dibagi menjadi 2 jenis, yaitu *web statis* dan *web dinamis*. Selain dari sisi *content/isi*, *web statis* dan *web dinamis* dapat dilihat dari aspek teknologi yang digunakan untuk membuat jenis *web* tersebut. (Muhamad Tabrani : 2017)

Website merupakan kumpulan halaman halaman yang berisi informasi yang disimpan diinternet yang bisa diakses atau dilihat melalui jaringan internet pada

perangkat perangkat yang bisa mengakses internet itu sendiri seperti komputer. Definisi kata web adalah Web sebenarnya penyederhanaan dari sebuah istilah dalam dunia komputer yaitu *world wide web* yang merupakan bagian dari teknologi Internet.

World wide Web atau disingkat dengan nama *www*, merupakan sebuah sistem jaringan berbasis Client-Server yang mempergunakan protokol HTTP (*Hyperteks Transfer Protocol*) dan TCP/IP Control Protocol / Internet Protocol) sebagai medianya. Karena kedua sistem ini mempunyai hubungan yang sangat erat, maka untuk saat ini sulit untuk membedakan antara *HTTP* dengan *www*.(Rulia Puji Hastanti (Rulia Puji Hastanti, dkk, 2015)

II.2.5 PHP

PHP adalah bahasa pemrograman script *server-side* yang didesain untuk pengembangan web. *PHP* disebut bahasa pemrograman *server-side* karena skrip *PHP* berjalan pada sisi server. Hal ini berbeda dengan bahasa pemrograman *client-side* seperti *JavaScript* yang berjalan pada *web browser*. Dalam beberapa tahun perkembangannya, *PHP* menjadi bahasa pemrograman web yang *powerful* dan tidak hanya digunakan untuk membuat halaman web sederhana, tetapi juga website populer yang digunakan oleh jutaan orang (Achmad Behori dan Badrul Alamin, 2018).

Script *PHP* adalah bahasa program yang berjalan pada sebuah *webserver*, atau sering disebut *server-side*. Oleh karena itu, *PHP* dapat melakukan apa saja yang bisa dilakukan program CGI lain, yaitu mengolah data dengan tipe apapun,

menciptakan halaman web yang dinamis, serta menerima dan menciptakan *cookies*, dan bahkan PHP bisa melakukan lebih dari itu. PHP dapat berjalan pada semua jenis sistem operasi, antara lain pada Linux dan varian Unix (*HP-UX*, *Solaris* dan *OpenBSD*), pada *Ms Windows*, *Mac*. (Chrisantus Trisianto, 2018).

PHP dirancang untuk dapat bekerja sama dengan *database server* dan dibuat sedemikian rupa sehingga pembuatan dokumen HTML yang dapat mengakses *database* menjadi begitu mudah. Tujuan dari bahasa *scripting* ini adalah untuk membuat aplikasi dimana aplikasi tersebut dibangun oleh PHP pada umumnya akan memberikan hasil kepada *web browser*, tetapi proses keseluruhannya dijalankan di *server* (Eka Wida Fridayanthie dan Tias Mahdiati, 2016).

II.2.6. DataBase

Database atau basis data adalah kumpulan data yang disimpan secara sistematis didalam komputer dan dapat diolah atau dimanipulasi menggunakan perangkat lunak (program aplikasi) untuk menghasilkan informasi. Pendefinisian basis data meliputi spesifikasi berupa tipe data, struktur, dan juga batasan-batasan data yang akan disimpan. Basis data merupakan aspek yang sangat penting dalam sistem informasi dimana basis data merupakan gudang penyimpanan data yang akan diolah lebih lanjut. Basis data menjadi penting karena dapat menghindari duplikasi data, hubungan antar data yang tidak jelas, organisasi data, dan juga update yang rumit. (Chrisantus Trisianto, 2018).

II.2.7 MySQL

MySQL adalah *database* yang menghubungkan *script* php menggunakan perintah *query* dan *escape character* yang sama dengan php”. Selain itu ia bersifat *open source* (anda tidak perlu membayar untuk menggunakannya) pada berbagai *platform* (kecuali untuk jenis *enterprise*, yang bersifat komersial). Perangkat lunak *mysql* sendiri bisa didownload dari <http://www.mysql.com>. *MySQL* termasuk jenis *RDMS (Relation Database Management System)*. Itulah sebabnya istilah seperti *table*, baris, kolom digunakan dalam *mysql*. Pada *mysql* sebuah *database* mengusung satu atau sejumlah *table*. Tabel terdiri atas sejumlah baris dan setiap baris mengandung satu atau beberapa kolom. *MySQL* merupakan *software* yang dikembangkan oleh komunitas umum, dan hak cipta untuk kode sumber dimiliki oleh penulisnya masing- masing. *MySQL* dimiliki dan disponsori oleh sebuah perusahaan komersial swedia yaitu *MySQL AB*. Beberapa kelebihan *mysql* antara lain:

1. *Free* (bebas di *download*).
2. Stabil dan tangguh.
3. Fleksibel dengan berbagai pemograman.
4. *Security* yang baik.
5. Kemudahan manajemen *database*.
6. Mendukung transaksi.
7. Perkembangan *software* yang cukup cepat. (Muhammad Tabrani : 2017).

II.2.8. Normalisasi

Menurut (Dwi Puspitasari, dkk, 2016) bahwa normalisasi adalah proses pembentukan struktur basis data sehingga sebagian besar ambiguity bisa dihilangkan. Normalisasi merupakan sebuah teknik dalam logical desain sebuah basis data relasional yang mengelompokkan atribut dari suatu table sehingga membentuk struktur tabel yang normal. Adapun kriteria table dikatakan normal adalah ketika tidak ada kerangkapan data (redundansi data). Tujuan dari normalisasi adalah untuk menghilangkan kerangkapan data sehingga meminimumkan pemakaian *storage* yang dipakai oleh *base relations* (file), untuk mengurangi kompleksitas, untuk mempermudah pemodifikasian data.

1. Proses Normalisasi

Adapun proses normalisasi adalah sebagai berikut :

- a. Data diuraikan dalam bentuk table, selanjutnya dianalisis berdasarkan persyaratan tertentu kebeberapa tingkat.
- b. Apabila tabel yang diuji belum memenuhi persyaratan tertentu, maka tabel tersebut perlu dipecah menjadi beberapa tabel yang lebih sederhana sampai memenuhi bentuk yang optimal.

2. Tahapan Normalisasi

- a. Bentuk Normal Pertama / 1 NF

Adapun aturan-aturan yang ada dalam bentuk normal pertama yaitu :

- 1) Tidak adanya atribut multi-value, atribut komposit atau kombinasinya.
- 2) Mendefinisikan atribut kunci.

- 3) Setiap atribut dalam tabel tersebut harus bernilai *atomic* (tidak dapat dibagi-bagi lagi).

b. Bentuk Normal Kedua / 2NF

Adapun aturan-aturan yang ada dalam bentuk normal kedua yaitu :

- 1) Sudah memenuhi dalam bentuk normal kesatu (1NF).
- 2) Semua atribut bukan kunci hanya boleh tergantung (functional dependency) pada atribut kunci.
- 3) Jika ada ketergantungan parsial maka atribut tersebut harus dipisah pada tabel yang lain.
- 4) Perlu ada tabel penghubung ataupun kehadiran foreign key bagi atribut-atribut yang telah dipisah tadi.

c. Bentuk Normal Ketiga / 3NF

Adapun aturan-aturan yang ada dalam bentuk normal ketiga yaitu :

- 1) Sudah memenuhi dalam bentuk normal kedua (2NF).
- 2) Tidak ada ketergantungan transitif (dimana atribut bukan kunci tergantung pada atribut bukan kunci lainnya).

II.2.9. UML (*Unified Modeling Language*)


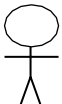

Unified Modelling Language (UML) merupakan satu kumpulan konvensi pemodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem *software* yang terkait dengan objek. UML merupakan salah satu alat bantu yang sangat handal dalam bidang pengembangan sistem berorientasi objek karena UML menyediakan bahasa pemodelan visual yang memungkinkan pengembang sistem membuat *blue print* atas visinya dalam bentuk yang baku.


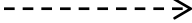

UML berfungsi sebagai jembatan dalam mengkomunikasikan beberapa aspek dalam sistem melalui jumlah elemen grafis yang bisa dikombinasikan menjadi diagram. (Omni Alfina, dkk, 2019).

a. *Use Case Diagram*

Use case diagram merupakan pemodelan untuk kelakuan sistem informasi yang akan dibangun. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibangun. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada pada sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Berikut ini adalah simbol-simbol *use case diagram*, seperti yang terlihat pada Tabel III.2

Tabel II.2. Simbol *Use Case Diagram*

Gambar	Keterangan	Deskripsi
	<i>Use case</i>	Menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal nama <i>use case</i> .
	Aktor	Sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem.
	Asosiasi	Penghubung antara aktor dan <i>use case</i> , digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengindikasikan aliran data.

	Asosiasi	Penghubung antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengindikasikan bila aktor berinteraksi secara pasif dengan sistem.
	<i>Include</i>	Merupakan di dalam <i>use case</i> lain (<i>required</i>) atau pemanggilan <i>use case</i> oleh <i>use case</i> lain, contohnya adalah pemanggilan sebuah fungsi program.
	<i>Extend</i>	Merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi.

(Sumber : Julianto Simatupang, 2019).

b. Class Diagram

Class Diagram juga menunjukkan atribut-atribut dan operasi-operasi dari sebuah kelas dan *constraint* yang berhubungan dengan objek yang dikoneksikan. *Class Diagram* secara khas meliputi : Kelas (*Class*), Relasi *Associations*, *Generalisation* dan *Aggregation*, atribut (*Attributes*), operasi (*operation/method*) dan *visibility*, tingkat akses objek eksternal kepada suatu operasi atau atribut. Hubungan antar kelas mempunyai keterangan yang disebut dengan *Multiplicity* atau *Cardinality* dapat dilihat pada Tabel II.2.

Tabel II.3. Simbol *Class Diagram*

<i>Multiplicity</i>	Penjelasan
1	Satu dan hanya satu
0..*	Boleh tidak ada atau 1 atau lebih
1..*	1 atau lebih
0..1	Boleh tidak ada, maksimal 1
n..n	Batasan antara. Contoh 2..4 mempunyai arti minimal 2 maksimal 4




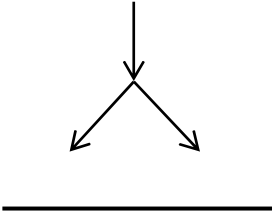
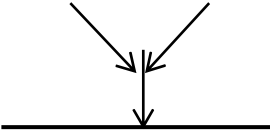
(Sumber : Julianto Simatupang, 2019).

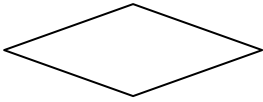
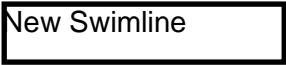
c. *Activity Diagram*

Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana setiap alir berawal, keputusan yang mungkin terjadi, dan bagaimana suatu aktivitas berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa kegiatan. Sebuah aktivitas dapat direalisasikan oleh satu *use case* atau lebih (Isworo Nugroho, dkk, 2017).

Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Simbol-simbol yang digunakan dalam *activity Diagram* dapat dilihat pada Tabel II.3.

Tabel II.4. Simbol *Activity Diagram*

Gambar	Keterangan	Deskripsi
	<i>Start point</i>	Diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
	<i>End point</i>	Akhir aktifitas.
	<i>Activites</i>	Menggambarkan suatu proses/kegiatan bisnis.
	<i>Fork</i> (Percabangan).	Digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan pararel menjadi satu.
	<i>Join</i> (penggabungan)	Digunakan untuk menunjukkan adanya dekomposisi.



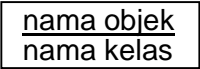
	<i>Decision Points</i>	Menggambarkan pilihan untuk pengambilan keputusan, <i>true</i> , <i>false</i> .
	<i>Swimlane</i>	Pembagian <i>activity</i> diagram untuk menunjukkan siapa melakukan apa.


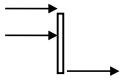


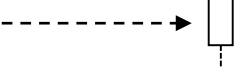
(Sumber : Julianto Simatupang, 2019).

d. *Sequence Diagram*

Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar *sequence diagram* maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu". Membuat *sequence diagram* dibutuhkan untuk melihat skenario yang ada pada *use case*. *Sequence diagram* dapat dilihat pada Tabel II.4.

Tabel II.5. Simbol *Sequence Diagram*

Gambar	Deskripsi
Aktor  Atau <u>nama aktor</u>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri.
Garis hidup / lifeline 	Menyatakan kehidupan suatu objek.
Objek 	Menyatakan objek yang berinteraksi pesan

<p><i>Activation</i> </p>	<p>Garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i>..</p>
<p>Pesan tipe create << create >> </p>	<p>Mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.</p>
<p>Pesan tipe send 1 : masukan </p>	<p>Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.</p>
<p>Pesan tipe return 1 : keluaran </p>	<p>Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian.</p>
<p>Pesan tipe destroy </p>	<p>Menyatakan suatu objek mengakhiri hidup objek lain, arah panah yang mengarah pada objek yang diakhiri, sebaiknya jika ada create maka ada destroy.</p>

(Sumber : Julianto Simatupang, 2019).

