

BAB II

TINJAUAN PUSTAKA

2.1 Konsep Dasar Sistem Informasi

2.1.1 Pengertian Sistem

Istilah dari sistem berasal dari Bahasa Latin yaitu *Systema* dan Bahasa Yunani yaitu *Sustema* yang memiliki arti komponen yang saling berhubungan secara teratur dan merupakan suatu keseluruhan. Dapat diartikan juga sebagai unsur – unsur yang saling berhubungan untuk memperlancar aliran informasi, materi atau energi untuk mencapai suatu tujuan. [1]

Berikut beberapa pengertian konsep dasar sistem menurut para ahli [1]:

1. Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. Sistem ini menggambarkan suatu kejadian-kejadian dan kesatuan yang nyata adalah suatu objek nyata, seperti tempat, benda, dan orang-orang yang betul – betul ada dan terjadi.
2. Sistem adalah seperangkat elemen yang membentuk kumpulan atau prosedur-prosedur atau bagan-bagan pengolahan yang mencari suatu tujuan tertentu.
3. Sistem adalah sekumpulan objek yang mencakup hubungan fungsional Antara tiap-tiap objek dan hubungan antara ciri tiap objek, dan yang secara keseluruhan merupakan suatu kesatuan secara fungsional.
4. Sistem adalah himpunan dari bagian – bagian yang saling berhubungan yang secara bersama mencapai tujuan-tujuan sama.

Sehingga dapat ditarik kesimpulan sistem adalah jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan kegiatan atau untuk melakukan sasaran yang tertentu, yang mempermudah kegiatan-kegiatan utama tercapai dari suatu organisasi.

Karakteristik Sistem [1]:

1. Komponen

Sejumlah komponen-komponen yang saling berinteraksi yang berada pada suatu sistem, saling bekerja sama membentuk satu kesatuan. Komponen sistem ini terdiri dari komponen berupa subsistem.

2. Batasan Sistem (*Boundary*)

Merupakan bagian yang membatasi antara satu sistem dengan sistem yang lain atau dari ruang lingkup diluarnya. Dengan adanya batasan ini, sistem dipandang sebagai satu kesatuan.

3. Lingkungan Luar Sistem (*Environment*)

Luar batas dari sistem yang mempengaruhi operasi sistem dinamakan lingkungan luar sistem. Lingkungan dapat bersifat menguntungkan dan merugikan dimana harus dijaga dan dikendalikan, agar tidak mengganggu kelangsungan hidup dari sistem. Apapun berbagai macam bentuk yang berada di luar ruang lingkup dapat mempengaruhi operasi sistem.

4. Penghubung Sistem (*Interface*)

Sebagai media penghubung satu subsistem dengan subsistem yang lainnya. Melalui penghubung ini *output* dari subsistem akan menjadi *input* untuk subsistem lain. Penghubung ini dapat memungkinkan sumber-sumber daya mengalir dari subsistem ke subsistem lain.

5. Masukkan Sistem (*Input*)

Energi yang dimasukkan kedalam sistem yang dapat berupa perawatan (*maintenance input*) dan masukkan sinyal (*signal input*).

6. Keluaran Sistem (*Output*)

Hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan.

7. Pengolahan Sistem

Suatu proses yang akan merubah masukkan menjadi keluaran.

8. Sasaran Sistem (*Objective*)

Suatu sistem harus mempunyai tujuan atau sasaran. Sasaran dari sistem menentukan *input* yang dibutuhkan sistem dan keluaran yang akan dihasilkan.

2.1.2 Pengertian Informasi

Informasi atau dalam Bahasa Inggrisnya adalah *information*, berasal dari kata *informaticon* Bahasa Prancis. Kata tersebut diambil dari Bahasa Latin, yaitu "*informationem*" yang artinya "konsep, ide, garis besar". Informasi merupakan hal yang sangat diperlukan oleh suatu kegiatan dalam pengambilan suatu keputusan agar tidak terjadi kesalahan. Informasi disebut juga sebagai data yang telah diolah menjadi bentuk yang lebih berguna bagi si penerima informasi. [1]

Informasi adalah sekumpulan data atau fakta yang diorganisasi atau diolah dengan cara tertentu sehingga mempunyai arti bagi penerima. (Anggraeni dan Irviani, 2017).

Informasi memiliki karakteristik yaitu [1]:

a. Relevan

Informasi harus memiliki makna yang tinggi sehingga tidak menimbulkan keraguan bagi yang menggunakannya dan dapat digunakan secara tepat untuk membuat keputusan.

b. Andal

Suatu informasi harus memiliki keterandalan yang tinggi, informasi yang dijadikan alat pengambilan keputusan merupakan kejadian nyata dalam aktivitas perusahaan.

c. Lengkap

Informasi yang dijabarkan harus memiliki penjelasan secara detail dan terperinci dari setiap aspek peristiwa yang diukurnya.

d. Tepat Waktu

Informasi harus dalam keadaan yang telah diperbaharui atau *ter-update*, sehingga penting untuk digunakan sebagai pengambilan keputusan.

e. Dapat Dipahami

Informasi yang ditunjukkan harus jelas sehingga akan memudahkan orang – orang untuk menjabarkannya.

- a) Informasi harus memiliki derajat makna yang tinggi agar tidak menimbulkan kecurigaan oleh yang menggunakannya dan dapat digunakan secara tepat untuk pengambilan keputusan.

- b) Suatu informasi harus memiliki keandalan yang tinggi, dan informasi yang digunakan sebagai alat pengambilan keputusan merupakan kejadian nyata dalam kegiatan perusahaan.
- c) Informasi yang dideskripsikan harus memiliki penjelasan yang mendetail dan mendetail tentang setiap aspek dari peristiwa yang diukurnya.
- d) Informasi harus dalam keadaan *ter-update* atau *ter-upgrade*, sehingga penting untuk digunakan dalam pengambilan keputusan.
- e) Informasi yang ditampilkan harus jelas agar orang lebih mudah mendeskripsikannya.

2.1.3 Pengertian Sistem Informasi

Sistem informasi adalah sistem *internal* organisasi atau perusahaan yang menggabungkan persyaratan pemrosesan transaksi harian dari operasi organisasi dan fungsi manajemen dengan kegiatan strategis organisasi, sehingga memberikan informasi yang dibutuhkan pihak luar tertentu untuk pengambilan keputusan. [1]

Sistem informasi dalam suatu organisasi dapat dikatakan sebagai suatu sistem yang memberikan informasi kepada semua tingkatan dalam organisasi pada saat dibutuhkan. Sistem menyimpan, mengambil, mengubah, memproses dan mengkomunikasikan informasi yang diterima melalui penggunaan sistem informasi atau peralatan sistem lainnya. [1]

2.1.4 Komponen dan Tipe Sistem Informasi

Sistem informasi terdiri dari komponen – komponen yang disebut dengan istilah blok bangunan (*building block*), yang terdiri dari blok masukan, blok model, blok keluaran, blok teknologi, blok basis data dan blok kendali. Sebagai suatu sistem, keenam blok tersebut saling berinteraksi satu dengan yang lain membentuk satu kesatuan untuk mencapai sasaran. [1]

a) Blok Masukan (*input block*)

Mewakili data yang masuk ke dalam sistem informasi. *Input* disini termasuk metode-metode dan media yang digunakan untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen dasar.

b) Blok Model (*model block*)

Terdiri dari kombinasi prosedur, logika dan metode matematik yang akan memanipulasi data input data yang tersimpan di basis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang sudah diinginkan.

c) Blok Keluaran (*output block*)

Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

d) Blok Teknologi (*technology block*)

Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirim keluaran dan membantu pengendalian diri secara keseluruhan. Teknologi terdiri dari unsur utama:

- 1) Teknisi (*human ware* atau *brain ware*)
- 2) Perangkat lunak (*software*)
- 3) Perangkat keras (*hardware*)

e) Blok Basis Data (*database block*)

Merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya.

f) Blok Kendali (*control block*)

Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah atau bila terlanjur terjadi kesalahan dapat langsung diatasi.

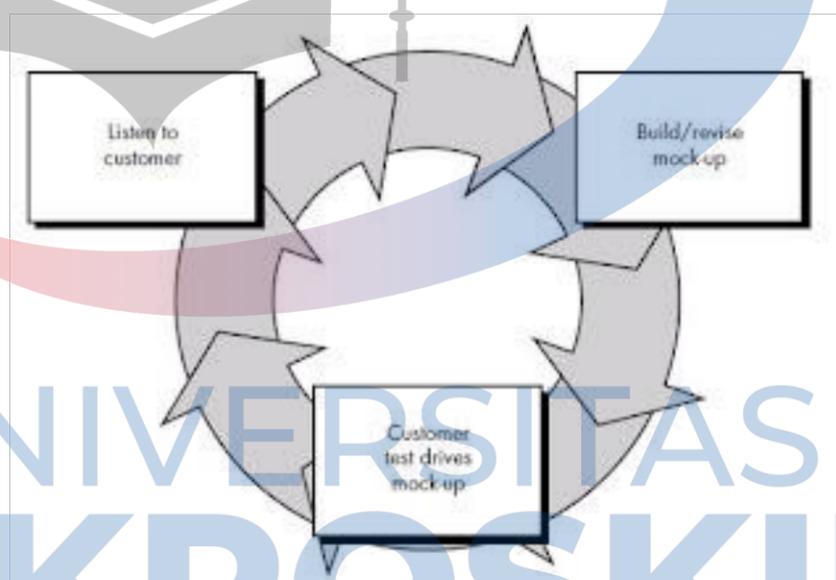
Sistem informasi juga memiliki fungsi yaitu [1]:

- 1) Dapat meningkatkan akseibilitas data dengan efisien dan efektif kepada pengguna, tanpa perantara sistem informasi.
- 2) Ketersediaan keterampilan serta kualitas dalam memanfaatkan sistem informasi secara kritis akan terjamin.
- 3) Mengembangkan proses perencanaan yang efektif.

2.2 Metode Penelitian

2.2.1 Metode *Prototype*

Metode *prototype* atau sering disebut juga dengan *prototyping* merupakan sebuah metode pengembangan sistem yang didasarkan pada konsep *working* model. Penelitian lain pun mengatakan *prototype* didefinisikan sebagai alat yang memberikan ide bagi pembuat maupun pemakai potensial tentang cara sistem berfungsi dalam bentuk lengkapnya, dan proses untuk menghasilkan sebuah *prototype* disebut *prototyping*. Metode *prototype* telah menjadi terkenal karena kemampuannya dalam menangkap kebutuhan pengguna dalam bentuk konkret. Metode *prototype* memiliki tahapan-tahapan seperti komunikasi, perencanaan cepat, pemodelan perancangan cepat, membangun *prototype* dan evaluasi dan perbaikan. [10]:



Gambar 2.1 Proses Metode *Prototype*

Tahapan-tahapan Model *Prototyping* [11]:

1. Pengumpulan Kebutuhan
Pelanggan dan pengembang bersama-sama mendefinisikan format seluruh perangkat lunak, mengidentifikasi semua kebutuhan, dan garis besar sistem yang akan dibuat.
2. Membangun *prototyping*
Dengan membuat perancangan sementara yang berfokus pada penyajian kepada pelanggan (misalnya dengan membuat *input* dan format *output*).
3. Evaluasi *Prototyping*

Evaluasi dilakukan oleh pelanggan apakah *prototyping* yang sudah dibangun sudah sesuai dengan keinginan pelanggan. Jika tidak sesuai *prototyping* akan direvisi dengan mengulang langkah 1, 2, dan 3.

Dalam setiap metode tentunya memiliki kelebihan dan kekurangannya termasuk metode *prototype* tersebut. Adapun kelebihan dan kekurangan yang dimiliki metode *prototype* adalah sebagai berikut [10]:

Tabel 2.1 Kelebihan dan Kekurangan Metode *Prototype*

Kelebihan	Kekurangan
Pengguna berperan aktif dalam perancangan ataupun pengembangan sistem.	Kualitas aplikasi belum teruji dan belum mencatumkan pemeliharaan jangka panjang.
Waktu yang digunakan lebih efisien.	Algoritma dan bahasa yang digunakan sederhana.
Adanya komunikasi antara <i>user</i> dengan pengembang.	Teknik rancangan tidak baik dilihat dari hubungan pelanggan dengan <i>computer</i> .
Pengembang dapat bekerja lebih baik.	

2.3 Alat Bantu Perancangan Sistem

2.3.1 *Unified Modelling Language* (UML)

Unified Modelling Language (UML) adalah bahasa yang telah menjadi *standard* untuk memvisualisasikan, menspesifikasikan, membangun dan dokumentasi perangkat lunak berbasis objek. UML merupakan model-model yang tercipta berhubungan langsung dengan berbagai macam bahasa pemodelan visual bukan merupakan bahasa pemrograman, sehingga dapat memungkinkan melakukan pemetaan langsung dari model-model yang dibuat dengan UML. Tipe diagram UML terbagi atas dua yaitu *structural* diagram dimana digunakan untuk mendeskripsikan relasi antar kelas dan *behavioral* diagram yang digunakan untuk mendeskripsikan interaksi antara aktor dan sebuah *use case*. Tujuan utama dari UML yaitu memberikan

model siap pakai, bahasa *visual* yang ekspresif untuk saling menukar model dengan mudah dan mengembangkan juga dimengerti secara umum, memberikan bahasa pemodelan yang bebas dari berbagai bahasa pemrograman dan proses rekayasa dan menyatukan praktek-praktek terbaik yang terdapat dalam pemodelan. Bahasa pemodelan ini dapat dikatakan juga dengan *blueprint* atau cetak biru yang digunakan di berbagai bidang keteknikan yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik, skema database dan komponen-komponen yang diperlukan dalam sistem *software*. Tahapan utama dalam metodologi ini adalah analisis, *design* sistem, *design* objek dan sebagainya. [2]

UML tersusun atas sejumlah elemen grafis membentuk 9 diagram-diagram. Dalam penelitian ini melakukan desain hanya 4 diagram yaitu *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram* dan *Class Diagram*. [2]

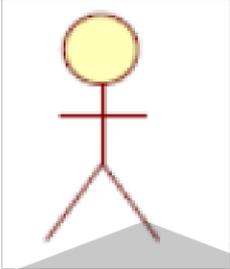
2.3.2 Use Case Diagram

Use case diagram menunjukkan fungsionalitas suatu sistem atau kelas serta siapa saja yang berhak menggunakan fungsi-fungsi tersebut. *Use case* juga mendeskripsikan interaksi satu atau lebih aktor dengan sistem informasi yang dibuat dan juga bagaimana sistem ini akan berinteraksi dengan dunia luar. Melalui diagram *use case* dapat diketahui fungsi-fungsi apa saja yang ada pada sistem. *Use case* diagram digunakan mengumpulkan kebutuhan dari sebuah sistem baik karena pengaruh *internal* maupun *eksternal*. *Use case* diagram memiliki tujuan, yaitu digunakan untuk mengumpulkan kebutuhan dari sebuah sistem, untuk mendapatkan pandangan dari luar sistem, untuk mengidentifikasi faktor yang mempengaruhi sistem baik *internal* maupun *eksternal* dan untuk menunjukkan interaksi dari para aktor dari sistem. Terdapat simbol-simbol yang digunakan dalam *Use Case Diagram* yaitu [2]:

Tabel 2.2 Simbol-Simbol *Use Case Diagram*

Gambar	Keterangan
	menggambarkan fungsionalitas yang disediakan sistem sebagai <i>unit-unit</i> yang bertukar pesan antar <i>unit</i> dengan

Tabel 2.3 Simbol-Simbol *Use Case Diagram* (sambungan)

	aktor, yang dinyatakan dengan menggunakan kata kerja
	Aktor adalah <i>Abstraction</i> dari orang atau 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. Orang atau sistem bisa muncul dalam beberapa peran. Perlu dicatat bahwa aktor berinteraksi dengan <i>use case</i> , tetapi tidak memiliki kontrol terhadap <i>use case</i>
	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 data.
	Penghubung antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengindikasikan bila aktor berinteraksi secara pasif dengan sistem
	<i>Extend</i> , merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi

Tabel 2.4 Simbol-Simbol *Use Case Diagram* (sambungan)

	<p><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. <i>Include</i> juga menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i></p>
---	---

Didalam *use case* ada penamaan yang didefinisikan sesederhana mungkin dan mudah untuk dipahami. Pada *use case* terdapat dua hal utama yaitu aktor dan *use case*.

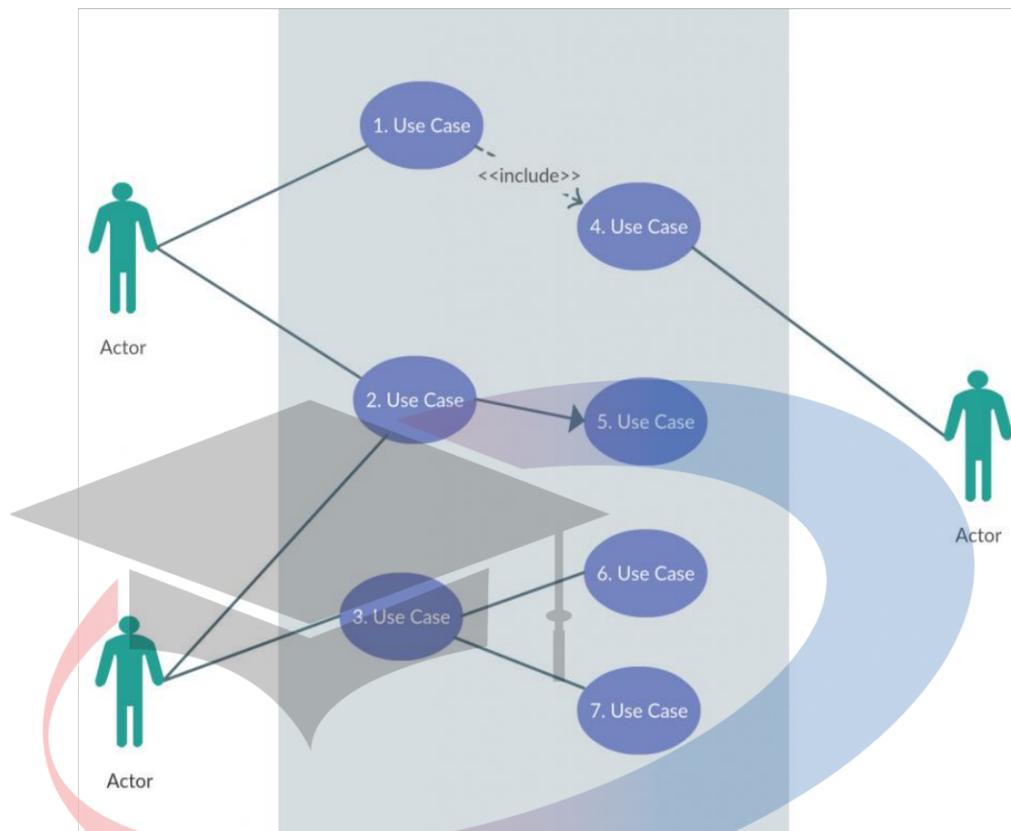
1) Aktor

Mewakili orang atau objek seperti perangkat keras atau sistem informasi lain yang berinteraksi dengan sistem. Aktor hanya berinteraksi dengan *use case* dan tidak memiliki kendali atas *use case*. Aktor dapat menerima informasi dari sistem, memasukkan informasi ke dalam sistem, atau keduanya. Simbol dari aktor berbentuk orang, tetapi belum tentu merupakan orang.

2) *Use case*

Perilaku yang dilakukan oleh pengguna sistem operasi atau aplikasi, dan juga interaksi antara aktor dengan perangkat lunak aplikasi untuk saling bertukar pesan. [2]

UNIVERSITAS
MIKROSKIL

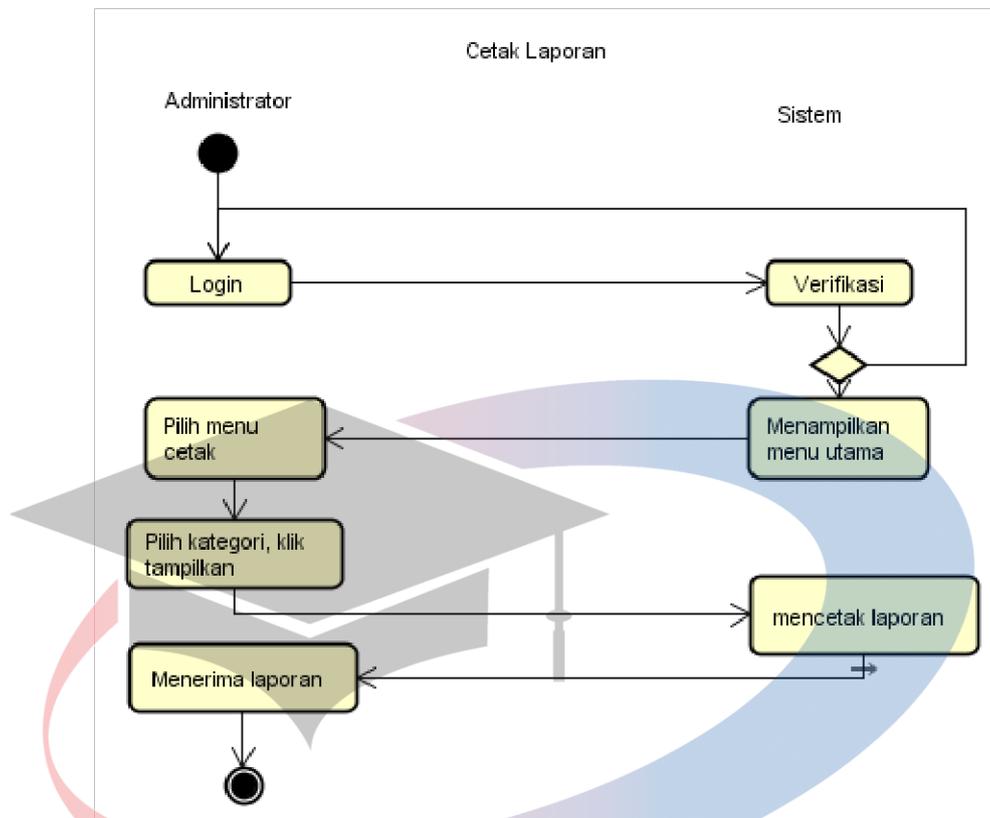


Gambar 2.2 Contoh *Use Case Diagram*

2.3.3 Activity Diagram

Activity Diagram menggambarkan berbagai aliran aktivitas atau kerja dari sebuah sistem yang sedang dirancang, namun bukan aktivitas aktor. Bagaimana aliran berawal, keputusan yang mungkin akan terjadi, dan bagaimana akhir dari suatu sistem tersebut. Diagram *Activity* adalah aktivitas-aktivitas, objek, *state*, transisi *state event*. Diagram alur kerja pada *Sequence diagram* menggambarkan perilaku sistem untuk aktivitas. *Activity diagram* juga dapat menggambarkan proses lebih dari satu aksi Dalam waktu bersamaan. *Activity diagram* juga dapat digunakan untuk menggambarkan aliran suatu sistem ke sistem yang lain, pada kasus aplikasi memiliki banyak sistem. Secara umum tujuan dari *activity diagram* bisa digambarkan sebagai berikut [2]:

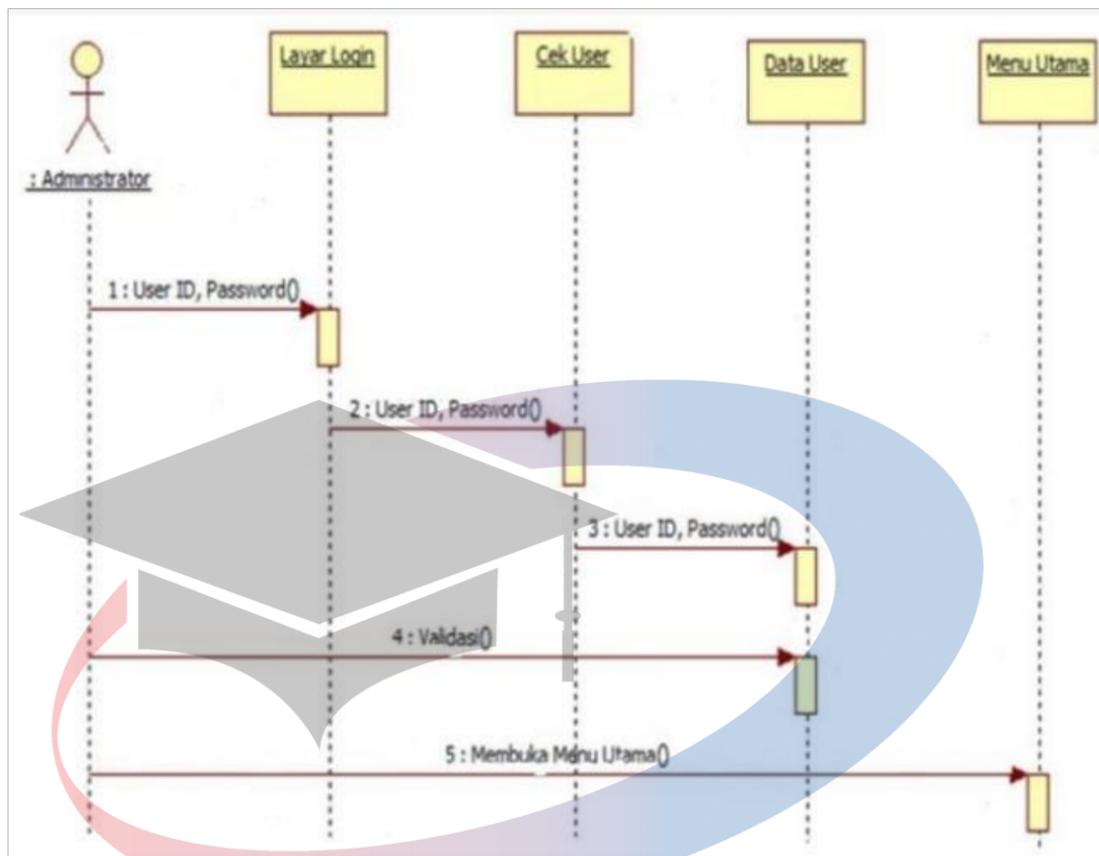
- a. Menggambarkan aliran aktivitas dari sistem
- b. Menggambarkan urutan aktivitas dari suatu aktivitas ke aktivitas lainnya
- c. Menggambarkan paralelisme, percabangan dan aliran konkuren dari sistem.



Gambar 2.3 Contoh *Activity Diagram*

2.3.4 *Sequence Diagram*

Sequence Diagram merupakan diagram yang digunakan untuk menggambarkan perilaku objek pada *use case* dengan mendeskripsikan proses objek dengan yang dikirimkan. Oleh karena itu, untuk menggambar *sequence diagram* harus diketahui objek-objek yang terlibat dalam *use case* beserta metode – metode yang dimiliki kelas yang di instansi menjadi objek itu. Kegunaan dari *sequence diagram* ini menunjukkan rangkaian pesan yang dikirim antar objek, dan menunjukkan sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem. Pada *sequence diagram* terdapat komponen utama yang terdiri dari objek yang dituliskan dengan tiga *lifeline*, pesan diwakili oleh garis dengan tanda panah, dan waktu yang ditunjukkan dengan proses vertikal. Tujuan dari *Sequence diagram* dapat digambarkan sebagai model interaksi tingkat tinggi antara objek aktif dalam suatu sistem, model interaksi antara *instance* objek dalam kolaborasi yang merealisasikan *use case* dan menunjukkan model interaksi generik atau contoh spesifik dari suatu interaksi. [2]

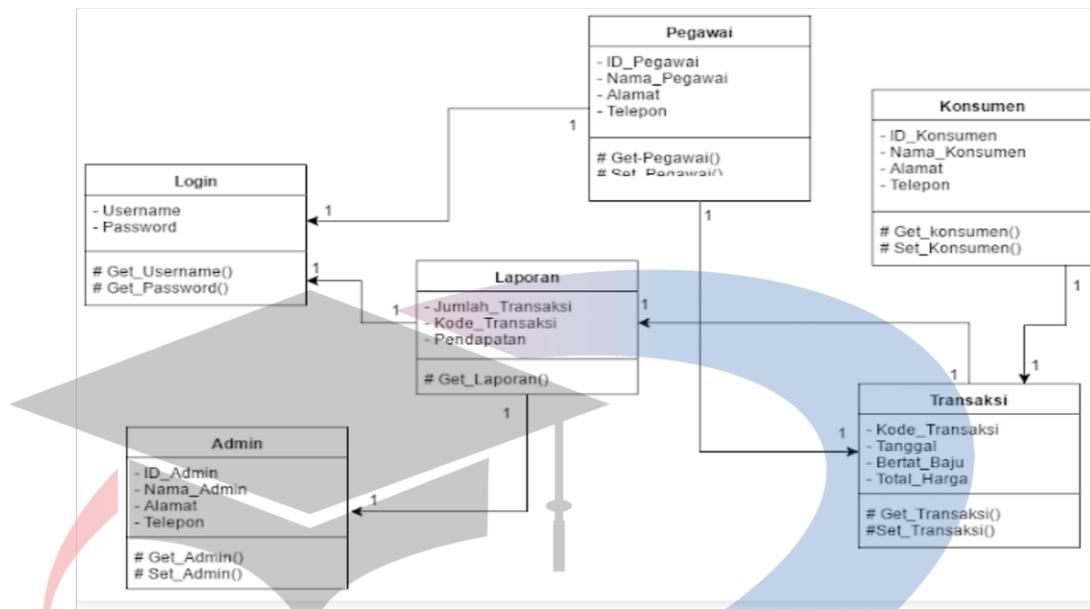
Gambar 2.4 Contoh *Sequence Diagram*

2.3.5 Class Diagram

Class Diagram digunakan untuk menggambarkan struktur dalam objek sistem. Diagram ini menunjukkan *class object*. Kelas memiliki atribut dan metode atau operasi. Atribut adalah variabel – variabel yang mendeskripsikan properti dengan bentuk sebaris teks dalam kelas tersebut, sedangkan metode adalah fungsi yang dimiliki oleh kelas yang dalam *class diagram* dilambangkan menggunakan simbol – simbol. *Class diagram* secara khas meliputi: Kelas (*Class*), Relasi *Associations*, *Generalitation* dan *Aggregation*, Atribut, Operasi (*Method*) dan *visibility*, tingkat akses objek *eksternal* kepada suatu operasi atau atribut. *Class diagram* menggambarkan jenis objek dalam sistem dan berbagai hubungan statis yang ada di antara mereka. *Class diagram* juga menunjukkan sifat dan operasi kelas dan keterbatasan yang terkandung dalam hubungan objek. Tujuan dari *class diagram* adalah untuk memodelkan pandangan statis suatu aplikasi. Secara lebih rinci, tujuan dari *class diagram* dapat diringkas sebagai berikut [2]:

- a. Analisis dan desain pandangan statis aplikasi

- b. Menjelaskan tanggung jawab suatu sistem
- c. Basis untuk diagram komponen dan penyebaran (*deployment*)



Gambar 2.5 Contoh *Class Diagram*

2.4 Analisis PIECES

Analisis PIECES adalah suatu sistem yang digunakan untuk mengidentifikasi masalah. Dalam menganalisis suatu sistem terdapat beberapa yang perlu dilakukan dan dilihat yaitu dari kinerja, informasi, ekonomi, keamanan, efisiensi, dan pelayanan. Panduan ini dikenal dengan PIECES (*Performance, Information, Economic, Control, Efficiency, Services*). [3]

Adapun kriteria wajib pada analisis PIECES yaitu [3]:

1. *Performance* (kinerja)

Masalah kinerja terjadi ketika tugas-tugas bisnis yang dijalankan tidak mencapai sasaran. Kinerja diukur dengan jumlah produksi dan waktu tanggap. Kinerja adalah suatu kemampuan sistem dalam menyelesaikan tugas dengan cepat sehingga sasaran dapat segera tercapai.

2. *Information* (informasi)

Evaluasi terhadap kemampuan sistem informasi dalam menghasilkan informasi yang bermanfaat perlu dilakukan untuk menyikapi peluang dan menangani masalah yang muncul. Dalam hal ini meningkatkan kualitas

informasi tidak dengan menambah jumlah informasi, karena terlalu banyak informasi malah akan menimbulkan masalah baru.

3. *Economic* (ekonomi)

Alasan ekonomi barangkali merupakan motivasi paling umum bagi suatu proyek. Pijakan dasar bagi kebanyakan manajer adalah biaya atau rupiah. Persoalan ekonomis dan peluang berkaitan dengan masalah biaya.

4. *Control* (pengendalian)

Tugas-tugas bisnis perlu dimonitor dan dibetulkan jika ditemukan kinerja yang dibawah standar. Kontrol dipasang untuk meningkatkan kinerja sistem, mencegah, atau mendeteksi kesalahan sistem, menjamin keamanan data, informasi atau mendeteksi kesalahan sistem.

5. *Efficiency* (efisiensi)

Efisiensi menyangkut bagaimana menghasilkan *output* sebanyak-banyaknya dengan *input* yang sekecil mungkin.

6. *Service* (layanan)

Peningkatan kualitas layanan yang diberikan oleh sistem kepada *user*, dengan kualitas layanan yang sangat *user friendly*.

2.5 Pemesanan

Pemesanan adalah kegiatan yang dilakukan oleh seorang konsumen sebelum membeli suatu produk. Untuk mengikuti perkembangan teknologi yang semakin pesat sebuah perusahaan dituntut untuk selalu berkembang dan memiliki sebuah sistem pemesanan yang baik. Pemesanan dalam artian umum adalah perjanjian yang dilakukan oleh 2 orang yaitu konsumen dan penyedia produk yang dapat berupa pemesanan suatu jasa atau sebuah barang. [4]

2.6 Laundry

Laundry Department adalah bagian di hotel atau bagian dari *Housekeeping* yang bertanggung jawab atas *laundry* semua linen, baik itu *house laundry* maupun *guest laundry*. Usaha *laundry* merupakan sebuah bisnis yang berkaitan dengan pelayanan jasa *laundry* pakaian dengan mesin cuci maupun mesin pengering otomatis dan cairan pembersih serta pewangi khusus. *Laundry* adalah kata benda yang mengacu pada

tindakan mencuci pakaian, tempat dimana mencuci dilakukan. *Laundry* juga diartikan sebagai kegiatan mencuci pakaian atau bahan tekstil lainnya dan juga sebagai sebuah tempat untuk mencuci pakaian atau bahan tekstil lainnya. [6]

2.7 *Mobile*

Mobile adalah kata sifat yang berarti dapat bergerak atau dapat digerakkan dengan bebas dan mudah. Namun *mobile* dapat pula diartikan sebuah benda yang berteknologi tinggi dan dapat bergerak tanpa menggunakan kabel. Contohnya seperti *smartphone*, PDA, dan *tablet*, *mobile* juga bisa diartikan kendaraan bermotor yang dapat bergerak. *Mobile* bersifat bebas seperti air dan dapat mengalir kemanapun. *Mobile* dapat berubah dan diubah dengan mudah. Jadi ada gambaran awal tentang *mobile*. Pengertian yang dibahas disini memang umum tidak sebatas *smartphone* atau *handphone*. *Mobile* bisa saja sebuah program yang dapat menjangkau seluruh tempat bisa diakses dengan mudah dan dapat diganti kapan saja tanpa kesulitan. [8]

2.8 *Android*

Android merupakan suatu sistem operasi yang berbasis *Linux* untuk telepon pintar (*smartphone*) ataupun pada komputer tablet. *Android* menyediakan *platform* terbuka bagi para pengembang dalam menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam peranti bergerak. Sebagai perangkat yang canggih, *Android* mempunyai beberapa keunggulan yang patut dipertimbangkan, yaitu sebagai berikut [7]:

- a. Dukungan produsen *hardware*
Smartphone dengan sistem operasi *android* kini menjadi sangat populer dan tidak lepas dari dukungan produsen *hardware*, seperti Sony dengan Xperia, HTC, LG dan berbagai merek lainnya.
- b. Harga yang bersahabat
 Harga *smartphone android* sesuai dengan kualitasnya, yang berarti tidak kemahalan. Ini juga merupakan keunggulan tersendiri karena pemakai membayar dan mendapatkan kualitas yang setara.
- c. Memiliki banyak varian

Produsen *smartphone android* sangat beragam. Mulai dari merek lokal hingga merek *branded*

d. Luwes dan menyenangkan

Pengguna dapat dengan bebas melakukan kustomisasi sesuai selera, dan dapat mengubah tampilan secara total hingga kemampuan *smartphone* itu sendiri.

e. Praktis dan banyak aplikasi tersedia

Umumnya *android* sangat nyaman digunakan untuk *browsing*, *download*, dan lainnya dengan berlangganan paket internet atau dengan *wifi*.

2.9 Database

Basis data terdiri dari 2 kata, yaitu basis dan data. Basis memiliki arti sebagai markas, gudang atau tempat berkumpul. Data adalah fakta yang mewakili suatu objek seperti manusia, hewan, barang atau keadaan, yang direkam dalam bentuk angka, huruf, simbol, teks gambar, bunyi, atau kombinasinya. Sehingga basis data dapat didefinisikan sebagai kumpulan file yang saling berhubungan yang disimpan dalam media penyimpanan elektronik. Komponen penting dalam sistem basis data yaitu [5]:

a. Data

Informasi yang disimpan dalam suatu struktur tertentu yang terintegrasi.

b. Basis Data

Basis data menyimpan data serta struktur sistem basis data baik untuk entitas maupun objek-objek secara detail. Basis data merupakan inti dari sistem basis data.

c. *Database Management System* (DBMS)

Merupakan perangkat lunak yang digunakan untuk melakukan pengolahan basis data pada suatu penyimpanan data. DBMS merupakan perantara *user* dengan basis data dengan memakai bahasa basis data yang sudah ditentukan oleh perusahaan DBMS. DBMS juga dapat membantu dalam memelihara serta pengolahan data dalam jumlah yang besar. Sebagai contoh perangkat lunak yang di gunakan yaitu *Microsoft Access*, *Pardox*, *Sql Server*, *Mysql*, *Oracle*.

d. *Hardware*

Merupakan perangkat keras dengan media penyimpanan yang digunakan untuk menyimpan data seperti komputer karena umumnya basis data memiliki ukuran yang besar.

e. Sistem Operasi

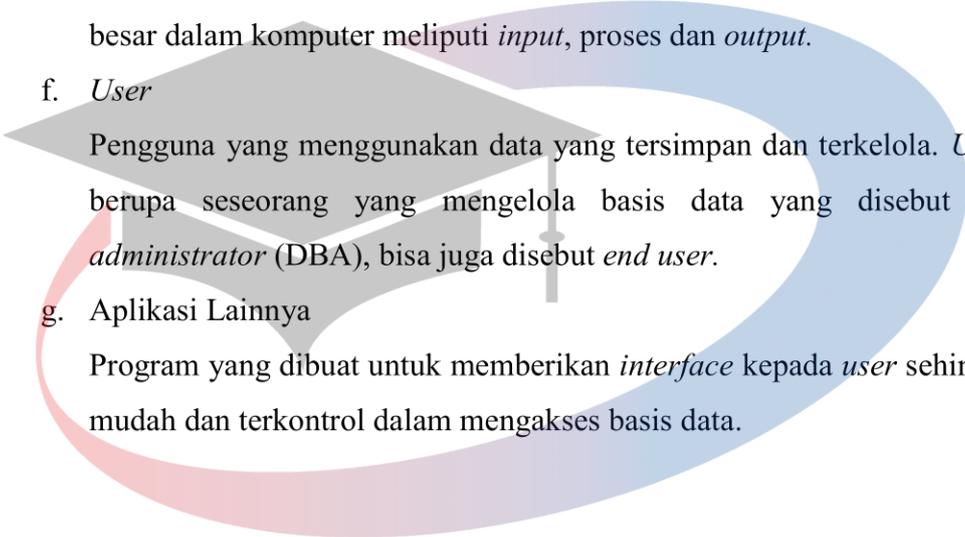
Program yang mengaktifkan dan memfungsikan sistem komputer, mengendalikan seluruh sumber daya dalam komputer, dan melakukan operasi besar dalam komputer meliputi *input*, proses dan *output*.

f. *User*

Pengguna yang menggunakan data yang tersimpan dan dikelola. *User* dapat berupa seseorang yang mengelola basis data yang disebut *database administrator* (DBA), bisa juga disebut *end user*.

g. Aplikasi Lainnya

Program yang dibuat untuk memberikan *interface* kepada *user* sehingga lebih mudah dan terkontrol dalam mengakses basis data.



UNIVERSITAS
MIKROSKIL