BAB II
LANDASAN TEORI

2.1. Konsep Dasar Sistem Informasi

2.1.1. Konsep Dasar Sistem

Menurut Tata Sutabri (2012, p.3) Sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur, komponen, atau variabel yang terorganisir, saling berinteraksi, saling tergantung satu sama lain dan terpadu. Teori sistem secara umum pertama kali diuraikan oleh *Kenneth Boulding*, terutama menekankan pentingya perhatian terhadap setiap bagian yang membentuk sebuah sistem. Model umum sebuah sistem adalah *input*, proses, dan *output*. Hal ini merupakan konsep sebuah sistem yang sangat sederhana sebab sebuah sistem dapat mempunyai beberapa masukan dan keluaran. Selain itu, sebuah sistem mempunyai karakteristik atau sifat-sifat tertentu yang mencirikan bahwa hal tersebut bisa dikatakan sebagai suatu sistem.

2.1.2. Konsep Dasar Informasi

Menurut Tata Sutabri (2012, p.22) Informasi adalah data yang telah diklasifikasi atau diinterpretasi untuk digunakan dalam proses pengambilan keputusan. Sistem pengolahan informasi mengolah data menjadi informasi atau tepatnya mengolah data dari tak berguna menjadi berguna bagi penerimanya. Nilai informasi berhubungan dengan keputusan maka informasi menjadi tidak diperlukan keputusan dapat berkisar dari keputusan berulang sederhana sampai keputusan strategis jangka panjang. Nilai informasi dilukiskan paling berarti dalam konteks sebuah keputusan.

2.1.3. Konsep Dasar Sistem Informasi

Menurut Tata Sutabri (2012, p.38) Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat

manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan laporan- laporan yang diperlukan oleh pihak luar tertentu.

2.2. Definisi Iqro'

Dalam pandangan Quraish Shihap kata iqra terambil dari akar kata yang berarti menghimpun. Dari menghimpun lahir aneka makna seperti menyampaikan, menelaah, mendalamai, meneliti, mengetahui ciri sesuatu dan membaca teks tertulis maupun tidak (Mudhoffar 2011, h.56).

2.3. Huruf Hijaiyah

Menurut Otory Surasman (2002:52) mengemukakan bahwa huruf hijaiyah merupakan kunci dasar mampu membaca AL-Qur'an. Huruf hijaiyah digunakan sebagai ejaan untuk menulis kata atau kalimat dalam AL-Qur'an.

Dari Husein bin Ali bin Abi Thalib as. : Seorang Yahudi mendatangi Nabi saw. Pada saat itu Amirul Mukminin Ali bin Abi Thalib as bersama Nabi. Yahudi itu berkata kepada Nabi saw : apa faedah dari huruf hijaiyah ? Rasulullah lalu berkata kepada Ali bin Abi Thalib as. "Jawablah" lalu Rasulullah mendoakan Ali "ya ALLAH, sukseskan Ali dan bungkam orang Yahudi itu" Lalu Ali berkata "Tidak ada satu hurufpun kecuali semua bersumber pada nama-nama ALLAH SWT". Kemudian Ali berkata adapun:

1. Alif(ا) adalah alaullah (keagungan ALLAH).
2. Ba (ب) adalah bahjatullah (kemuliaan ALLAH) (atau Al Baqi = keabadian dan badi' = pencipta langit dan bumi).
3. Ta (ت) adalah tamamul amri (kesempurnaan urusan) pada keluarga Muhammad.
4. Tsa (ث) adalah tsawab (ganjaran) bagi orang mukmin atas amalan-amalan shaleh mereka.
5. Jim (ج) artinya adalah jamalullah (keindahan ALLAH) dan jalalullah (Keagungan ALLAH)
6. Ha (ح) artinya hilm, kasih sayang ALLAH (hay = hidup, haq = yang maha benar, dan halim =lemah lembut) terhadap para pendosa.

7. **Kha** (خ) al Khabir (Yang Mahatahu) dan Mahalihat. Sesungguhnya ALLAH Mahatahu apa yang kalian kerjakan.
8. Dal (د) adalah dinullah (agama ALLAH) yang diridhai bagi hamba-hamba-Nya.
9. Dzal (ذ) adalah dzil jalali wal ikram (yang memiliki keagungan dan kemuliaan)
10. Ra (ر) adalah ar rauf (lemah-lembut) dan ar rahim (penyayang).
11. Zay (ز) adalah zalazil (kehancuran) pada hari kiamat.
12. Sin (س) adalah sanā (keluhuran) dan sarmadi (keabadian).
13. Syin (ش) adalah Sya'a kehendak ALLAH ketika Dia berkehendak. Dia berbuat sesuai kehendak-Nya. Tidak ada yang berkehendak kecuali sesuai dengan kehendak ALLAH.
14. Shad (ص) adalah shadiq (benar) dalam janji-janji-Nya kepada manusia dalam menyelamatkan orang-orang baik dari jembatan shirat dan mengadzab orang-orang zalim.
15. Dhad (ض) adalah menyesatkan orang-orang yang menentang Muhammad dan keluarga Muhammad saw.
16. Tha (ط) adalah thuba (kebahagiaan) bagi kaum mukminin dan sebaik-baik tempat kembali.
17. Dzha (ظ) adalah dzhann (anggapan) kaum mukminin yang baik kepada Allah dan anggapan orang kafir yang buruk kepada ALLAH.
18. 'Ayn (ع) adalah alim (berilmu)
19. Ghayn (غ) adalah ghina (kekayaan) yang tidak ada keperluan lagi setelahnya.
20. Fa (ف) adalah faliq (yang menumbuhkan) biji-bijian dan fauj (yang cepat penyelamatannya) dari api neraka.
21. Qaf (ق) adalah Al Qur'an yang berasal dan dibacakan di sisi ALLAH.
22. Kaf (ك) adalah al kafi (yang maha mencukupi).
23. Lam (ل) adalah laghw (permainan) orang-orang kafir ketika berbohong atas nama ALLAH.
24. Mim (م) adalah mulk (kerajaan) Allah pada hari kiamat pada hari tiada raja selain-Nya. ALLAH berfirman "Bagi siapa kerajaan pada hari ini?"

kemudian para ruh, para nabi dan rasul berkata “hanya milik ALLAH Yang esa dan Maha Perkasa” maka ALLAH berkata “Pada hari ini setiap jiwa akan diberikan balasan sesuai dengan amalan mereka, tidak akan ada yang dizalimi pada hari ini, sesungguhnya ALLAH maha cepat hisabnya”.

25. Nun (ن) adalah nawāl (pembelian ALLAH) dan nakāl (rencana) ALLAH bagi orang kafir.
26. Waw (و) adalah wayl (neraka wayl) bagi orang yang menentang-Nya dengan siksa yang sangat pedih.
27. Ha (ه) adalah penghinaan ALLAH bagi yang menentang-Nya.
28. Lam alif adalah la ilaha illallah (tidak ada Tuhan selain ALLAH) itulah kalimat tauhid termurni. Barang siapa yang mengatakannya dengan penuh keikhlasan maka baginyalah syurga.
29. Ya (ي) adalah yadullah (tangan/kekuasaan) ALLAH atas hamba-hamba-Nya, memberikan rizki-Nya. Maha suci ALLAH dari apa-apa yang mereka sekutukan.

2.4. Mobile Aplication

Mobile application adalah jenis perangkat lunak aplikasi yang dirancang untuk berjalan dan melakukan tugas-tugas tertentu pada perangkat *mobile*, seperti *handphone*, *smartphone*, dan PDA’s (Janalta Interaktif Inc, 2013).Semakin banyaknya perangkat lunak aplikasi untuk perangkat *mobile* merupakan dapat dari semakin pesatnya perkembangan teknologi *mobile computing.Mobile application* tidak hanya mampu untuk melakukan proses layanan dasar seperti layanan telepon maupun layanan pesan, tetapi sudah mampu untuk melakukan tugas-tugas yang rumit seperti seperti melakukan pencarian posisi pengguna, menampilkan dan memproses informasi dalam peta digital.

2.5. Smartphone

Menurut(Wikipedia, 2016) Ponsel cerdas atau *smartphone* adalah telepon genggam yang mempunyai kemampuan dengan penggunaan dan fungsiyan menyerupai komputer. Belum ada standar pabrik yang menentukan arti ponsel cerdas. Bagi beberapa orang, ponsel cerdas merupakan telepon yang berkerja yang menggunakan seluruh perangkat lunak sistem operasi yang menyediakan

hubungan standar dan mendasar bagi pengembangan aplikasi. Bagi yang lainnya, ponsel cerdas hanya merupakan sebuah telepon yang menyajikan fitur canggih seperti surel (surat elektronik), *internet* dan kemampuan membaca buku elektronik *(e-book)* atau terdapat papan ketik dan penyambung VGA. Dengan kata lain, ponsel cerdas merupakan komputer kecil yang mempunyai kemampuan sebuah telepon. Pertumbuhan permintaan akan alat canggih yang mudah dibawa kemana-mana membuat kemajuan besar dalam pemroses,pengingatan,layar dan sistem operasi yang diluar dari jalur telepon genggam sejak beberapa tahun ini. (https://id.wikipedia.org/wiki/Ponsel_cerdas)

2.6. Android

Menurut Abdul Kadir (2013, p.2) android bukanlah robot berbasis *artificial intelligence*, melainkan sistem operasi untuk perangkat bergerak. Android dikembangkan oleh perusahaan kecil di *Silicon Valley* yang bernama *Android Inc*. Selanjutnya, Google mengambil alih sistem operasi tersebut pada tahun 2005 dan merencanakannya sebagai sistem operasi yang bersifat "*Open Source*". Android tidak hanya ditujukan uuntuk ponsel, tetapi juga perangkat elektronik bergerak lainnya. Pada tahun 2012, android telah digunakan pada peranti-peranti berikut:

1. *Smartphone.*
2. *Tablet.*
3. Peranti pembaca buku elektronik.
4. *Netbook.*
5. MP4 *player.*
6. Tv *internet.*

Android adalah sistem operasi untuk perangkat *mobile* berbasis linux yang mencakup sistem operasi, *middleware* dan aplikasi.Android menyediakan *platform* yang terbuka bagi para pengembang untuk menciptakan aplikasi mereka.(Safaat, 2011, p.1).

Android merupakan platform yang berbasis Linux versi 2.6.untuk mengembangkan aplikasi pada platform android dibutuhkan android Software Development Kit (SDK) yang menyediakan tools dan Aplikasi Programming

Interface (API) yang diperlukan. Siste layanan seperti keamanan, manajemen memori, manajamen proses dikendalikan oleh Linux (Holla & Katti, 2012).

2.7. Versi Android

Menurut Abdul Kadir (2013, p.3) Android terus berkembang dan hal itu ditandai dengan versinya, dari aplikasi yang penulis buat penulis menggunakan versi android jelly bean.

Android jelly bean adalah versi yang memiliki fitur cepat dan halus dilengkapi grafis. Google mengumumkan android jelly bean 4.1 pada juni 2012. Sistem operasi berdasarkan linux kernel 3.0.31. android jelly bean adalah pembaharuan tambahan dengan tujuan utama meningkatan fungsi dan kinerja antarmuka pengguna, mengunakan antisipasi sentuhan, triple buffering.Versi:4.1, 4.1.1, 4.1.2, 4.2, 4.2.1, 4.2.2, 4.3, 4.3. Linux kernel : 3.0.31

2.8. Basis Data

Menurut Abdul Kadir (2013, p.264) Database adalah suatu pengorganisasian data dengan tujuan agar data dapat dimanipulasi dan diperoleh dengan cepat dan mudah.Terkait dengan database, android menyediakan SQLite.Database tersebut bersifat relasional.Artinya data dalam database disusun dalam bentuk table, yang terdiri atas baris dan kolom.Suatu database hanya dapat digunakan oleh aplikasi yang menciptakannya. Secara umum, operasi yang dilakukan di database meliputi:

- Pembuatan database.
- Pembukaan database.
- Penyisipan data di table.
- Pengubahan data di table.
- Penghapusan data di table.
- Penutupan database

2.9. Metode Pengembangan Sistem

Menurut Rosa A.S M. Shalahudin (2014, p.26) proses pengembangan mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodelogi

yang digunakan dalam mengembangkan sistem-sistem perangkat lunak sebelumnya.

2.9.1. *Metode Rational Unified Process (RUP)*

Menurut Rosa A.S M. Shalahudin (2016, p.124) Unified Process atau dikenal juga dengan proses iterative dan incremental merupakan sebuah proses pengembangan perangkat lunak yang dilakukan secara iteratif (berulang) dan incremental (bertahap dengan progress menaik). *Rational Unified Process* (RUP) adalah pendekatan pengembangan perangkat lunak yang dilakukan berulang-ulang (iterative), focus pada arsitektur (architecture-centric), lebih diarahkan berdasarkan penggunaan kasus (use case driven). RUP merupakan proses rekayasa perangkat lunak dengan pendefinisian yang baik (well defined) dan penstrukturan yang baik (well structured). RUP menyediakan pendefinisian struktur yang baik untuk alur hidup proyek perangkat lunak. RUP adalah sebuah produk proses perangkat lunak yang dikembangkan oleh *rational software* yang diakuisisi oleh IBM dibulan februari 2003.

a. Kelebihan RUP

Pendekatan iteratif/ pengulangan dari RUP dapat mengokomodir beberapa kelemahan pengembangan perangkat lunak tanpa menggunakan konsep pengulangan.

1. RUP mengakomodasikan perubahan kebutuhan perangkat lunak, Kebutuhan untuk mengubah dan menambah fitur karena perubahan teknologi atau keinginan pelanggan (*customer*) merupakan salah satu kendala yang sering dialami pengembang perangkat lunak yang berimbas pada terlambatnya waktu penyelesaian perangkat lunak. Keterlambatan ini dapat menyebabkan ketidakpuasan pelanggan (*customer*) dan pengembang menjadi frustasi.

2. Integrasi bukanlah sebuah proses besar dan cepat (*big bag*) diakhir proyek. Menempatkan integrasi dibagian akhir proyek biasanya memakan waktu yang cukup banyak, bisa mencapai 40 persen dari waktu proyek. Untuk mencegah hal ini maka pendekatan secara iteratif (pengulangan) dapat memecah proyek

menjadi bagian iterasi kecil yang diakhiri dengan integrasi kecil, yang dimana nantinya digabungkan menjadi integrasi besar.

3. Risiko biasanya ditemukan atau dialamatkan selama pada proses intergrasi awal. Pendekatan intergrasi pada RUP mengurangi risiko pada iterasi awal dimana saat semua komponen diuji.

4. Manajemen berarti membuat perubahan taktik pada produk. Taktik produk misalnya pengembangan dengan waktu singkat akan menghasilkan produk dengan fungsi yang terbatas dan dapat cepat digunakan oleh *user* sehingga memperkenalkan produk lebih cepat kemasyarakat dibandingkan produk kompetitor lain yang masih sedang dikembangkan.

5. Mendukung fasilitas penggunaan kembali lebih mudah untuk mengidentifikasi bagian umum yang sering digunakan dalam aplikasi, dimana jika diimplementasikan secara iterasi dari pada mengidentifikasi pada saat perencanaan saja.

6. Kecacatan dapat ditemukan dan dapat diperbaiki pada beberapa iterasi dan menghasilkan arsitektur yang baik dan aplikasi berkualitas tinggi, kecacatan dideteksi lebih baik mulai pada iterasi awal daripada pada tahap pengujian.

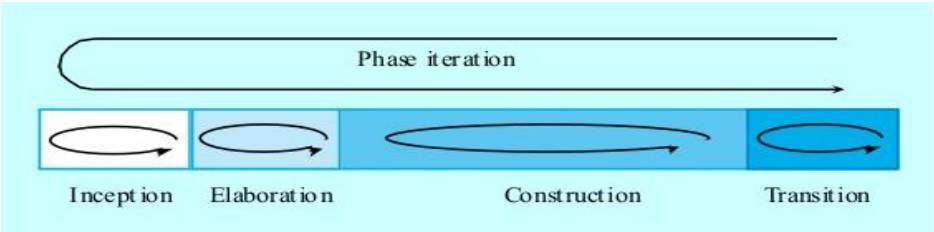
7. Lebih baik menggunakan “anggota proyek” dibandingkan susunan secara seri pada tim proyek.

8. Anggota tim belajar selama proyek berjalan, dimana anggota proyek memiliki peluang belajar dari kesalahan selama siklus pengembangan perangkat lunak dan memperbaiki kesalahan pada interasi berikutnya.

9. Pengembangan perangkat lunak dapat diperbaiki seiring proses pengembangan perangkat lunak setiap akhir proses iterasi, tidak hanya ada peninjauan mengenai target produk tetapi juga peninjauan pada kesalahan yang terjadi pada iterasi sebelumnya agar dapat diperbaiki pada iterasi berikutnya.

b. Fase RUP

RUP memiliki empat buah tahap atau fase yang dapat dilakukan pula secara iteratif. Berikut ini adalah gambar 2.1 alur hidup RUP.



Gambar 2.1 Alur Hidup RUP

Berikut ini penjelasan untuk setiap fase pada RUP, yaitu :

1. *Inception* (Permulaan)

Tahap ini lebih pada pemodelan proses bisnis yang dibutuhkan (*business modeling*) dan mendefinisikan kebutuhan akan sistem yang akan dibuat (*requiretments*).

Berikut adalah tahap yang dibutuhkan pada tahap ini:

a. Memahami ruang lingkup dari proyek (termasuk pada biaya, waktu, kebutuhannya, risiko dan lain sebagainya).
b. Membangun kasus bisnis yang dibutuhkan.

Hasil yang diharapkan dari tahap ini adalah memenuhi *lifecycle objective milestone* (batasan/tonggak objektif dari siklus) dengan kriteria berikut:

a. Umpan balik dari pendefinisian ruang lingkup, perkiraan biaya, dan perkiraan jadwal.
b. Kebutuhan dimengerti dengan pasti (dapat dibuktikan) dan sejalan dengan kasus primer yang dibutuhkan.
c. Kredibilitas dari perkiraan biaya, perkiraan jadwal, penentuan skala prioritas, risiko dan proses pengembangan.
d. Ruang lingkup purwarupa (*prototype*) yang akan dikembangkan.
e. Membangun garis dasar dengan membandingkan perencanaan aktual dengan perencanaan yang direncanakan.

Jika pada akhirnya tahap ini target yang diinginkan tidak dicapai maka dapat dibatalkan atau diulang kembali setelah dirancang ulang agar kriteria yang diinginkan dapat dicapai. Batas/tonggak objektif digunakan untuk mendeteksi apakah sebuah kebutuhan akan sistem dapat di implementasi atau tidak.

2. *Elaboration* (Perluasan/Perencanaan)

Tahap ini lebih difokuskan pada perencanaan arsitektur sistem. Tahap ini juga dapat mendeteksi apakah arsitektur sistem yang diinginkan dapat dibuat atau tidak. Mendeteksi risiko yang mungkin terjadi dari arsitektur yang dibuat. Tahap ini lebih pada analisis dan desain sistem serta implementasi sistem yang fokus pada purwarupa sistem *(prototype).*

Hasil yang diharapkan dari tahap ini adalah memenuhi *lifecycle architecture milestone* (batas/tonggak arsitektur dari siklus) dengan kriteria berikut:

a. Model kasus yang digunakan *(use case)* dimana kasus dan aktor yang terlihat telah didefinisikan dari sebagian besar kasus harus dikembangkan. Model *use case* harus 80 persen lengkap dibuat.
b. Deskripsi dari arsitektur perangkat lunak dari proses pengembangan sistem perangkat lunak telah dibuat.
c. Rancangan arsitektur yang dapat diimplementasikan dan mengimplementasikan *use case.*
d. Kasus bisnis dan proses bisnis dan daftar risiko yang sudah mengalami perbaikan (revisi) telah dibuat.
e. Rencana pengembangan untuk seluruh proyek telah dibuat.
f. Purwarupa *(prototype)* yang dapat di demonstrasikan untuk mengurangi setiap risiko teknis yang didefinisikan.

Jika pada akhir tahap ini dapat yang diinginkan tidak dicapai maka dapat dibatalkan atau diulang kembali. Batas/tonggak arsitektur digunakan untuk mendeteksi apakah sebuah kebutuhan akan sistem dapat diimplementasikan atau tidak melalui pembuatan arsitektur.

3. *Construction* (Konstruksi)

Tahap ini fokus pada pengembangan komponen dan fitur-fitur sistem. Tahap ini lebih pada implementasi dan pengujian sistem yang fokus pada implementasi perangkat lunak pada kode program. Tahap ini menghasilkan produk perangkat lunak dimana menjadi syarat dari *Intial Operational Capability Milestone* atau batas/tonggak kemampuan operasional awal.

4. *Transition* (Transisi)

Tahap ini lebih pada *deployment* atau instalasi sistem agar dapat dimengerti oleh *user*. Tahap ini menghasilkan produk perangkat lunak dimana menjadi syarat dari *Initial Operational Capability Milestone* atau batas/tonggak kemampuan operasional awal. Aktivitas pada tahap ini termasuk pada pelatihan *user*, pemeliharaan dan pengujian sistem apakah sudah memenuhi harapan u*ser*. Produk perangkat lunak juga disesuaikan dengan kebutuhan yang didefinisikan pada tahap *inception.* Jika semua kriteria objektif terpenuhi maka dianggap sudah memenuhi *Products Release Milestone* (batas/tonggak peluncuran produk) dan pengembangan perangkat lunak selesai dilakukan.

Akhir dari keempat fase ini adalah produk perangkat lunak yang sudah lengkap. Ke empat fase pada rup dijalankan secara berurutan dan iteratif dimana setiap iterasi dapat digunakan untuk memperbaiki iterasi berikutnya.

2.9.2 Definisi *Unified Modeling Language* (UML)

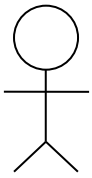
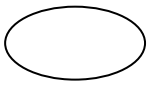
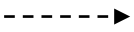
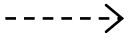
Menurut Rosa A.S M. Shalahudin (2016, p.137) UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan dengan menggunakan diagram dan teks-teks pendukung. UML hanya berfungsi untuk melakukan pemodelan, jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek. Perkembangan penggunaan UML bergantung pada level abstraksi penggunaannya, dalam penggunaan UML adalah suatu yang salah, tapi

perlu ditelaah dimanakah UML digunakan dan hal apa yang ingin divisualkan.Menurut Rosa A.S M. Shalahudin (2016, p.140) Pada UML terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori.

a. *Use Case Diagram*

Use case atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use Case* mendeskripsikan sebuah interaksi antara satru atau lebih actor dengan sistem informasi yang akan dibuat . Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Syarat penamaan pada *use case* adalah nama didefinisikan sesimpel mungkin dan dapat dipahami

Tabel 2.1 Simbol *Use Case*

Simbol	**Keterangan**
Aktor 	seseorang atau sesuatu yang berinteraksi dengan sistem yang sedang dikembangkan.
Use Case 	perungkat tertinggi dari fungsionalitas yang dimiliki sistem.
Association	relasi antara actor dan *use case.*
Generalisasi 	untuk memperlihatkan struktur pewaris yang terjadi.
Include 	menspesifikasikan bahwa *use case* sumber secara *eksplisit*
Collaboration	interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan prilaku yang lebih besar dari jumlah dan elemen-elemen nya (sinergi)

Extend <- - - - - - -	Menspesifikasikan bahwa *use case* target memperluar perilaku dari *use case* sumber pada suatu titik yang diberikan

b. *Activity Diagram*

Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram ktivitas menggambarkan aktivitas sistem bukan apa yang dilakuakan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

Tabel 2.2 Simbol *Activity Diagram*

No	**Gambar**	**Nama**	**Keterangan**
1		*Actifity*	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		*Action*	*State* dari sistem yang mencerminkan eksekusi dari suatu aksi
3		*Initial Node*	Bagaimana objek dibentuk atau diawali.
4		*Actifity Final Node*	Bagaimana objek dibentuk dan dihancurkan.
5.		*Fork Node*	Satu alairan yang pada tahap tertentu berubah menjadi beberapa aliran.

c. *Class Diagram*

Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan mtode atau operasi.

- Atribut merupakan variable-variabel yang dimiliki oleh suatu kelas.

- Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Tabel 2.3 Simbol *Class Diagram*

No	Gambar	Nama	Keterangan
1		*Generalization*	Hubungan dimana objek anak *(descendent)* berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (*ancestor*).
2		*Nary Association*	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		*Class*	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		*Collaboration*	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
5		*Realization*	Operasi yang benar-benar dilakukan oleh suatu objek.
6		*Dependency*	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri *(independent)* akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri
7		*Association*	Apa yang menghubungkan antara objek satu dengan objek lainnya

2.10. Perangkat Lunak Pendukung

Menurut Abdul Kadir (2013, p.4) pertumbuhan pasar perangkat berbasis android tentu saja mendorong pertumbuhan pengembangan aplikasi berbasis android. Peranti yang dibuutuhkan untuk pengembangan aplikasi android mencakup *Java Development Kit (JDK), Eclipse, Android Development Tools (ADT).* Penjelasan peranti masing-masing adalah sebagai berikut :

1. JDK (*Java Development Kit)*

Menurut Abdul Kadir (2013, p.4*) Java Development Kit (JDK)* adalah perangkat pengembangan aplikasi java yang bisa diunduh secara gratis di www.oracle.com/technetwork/java/javanes/download/. Perangkat ini mutlak diperlukan untuk membuat aplikasi android, mengingat aplikasi android itu berbasis java. Sebagaimana diketahui, java adalah salah satu bahasa pemrograman yang biasa digunakan untuk membuat aplikasi. Namun perlu diketahui, tidak semua pustaka dalam java digunakan di android.

2. Android SDK

Menurut Abdul Kadir (2013, p.5) Android SDK adalah kumpulan software yang berisi mengenai pustaka, debugger (alat pencari kesalahan program), emulator (peniru perangkat bergerak), dokumentasi, kode contoh, dan panduan.