

BAB II

LANDASAN TEORI

2.1. Kajian Pustaka

Kajian pustaka dari penelitian pembuatan media pembelajaran siswa SMK untuk mata pelajaran fisika (studi kasus Hukum Ohm dan Hukum Kirchoff) didasarkan pada aplikasi media pembelajaran lain yang sudah pernah dibuat adalah aplikasi pembuatan media pembelajaran Fisika yang dibangun oleh Sarofi (2014). Sarofi pada tugas akhir dengan judul “Pengembangan Media Pembelajaran Fisika Interaktif Berbasis *Macromedia Flash* Pada Materi Listrik Dinamis Untuk Siswa SMP/MTs Kelas IX” menyebutkan bahwa banyak siswa kelas IX di MTs Wahid Hasyim Yogyakarta yang belum mencapai nilai kriteria ketuntasan minimal karena materi yang dianggap sulit untuk dipahami. Aplikasi ini berfungsi untuk mempermudah pengguna aplikasi, terutama siswa SMP/MTs kelas IX semester 2, dalam mempelajari Mata Pelajaran Fisika pada materi listrik dinamis melalui *Personal Computer/ Laptop*. Rancangan aplikasi ini di dalamnya mengandung unsur teks, gambar, animasi dan video. Namun terdapat kelemahan pada media yang dibuat yaitu sistem pembelajaran harus dilakukan di dalam kelas, dan tidak dapat dilakukan jika diluar kelas.

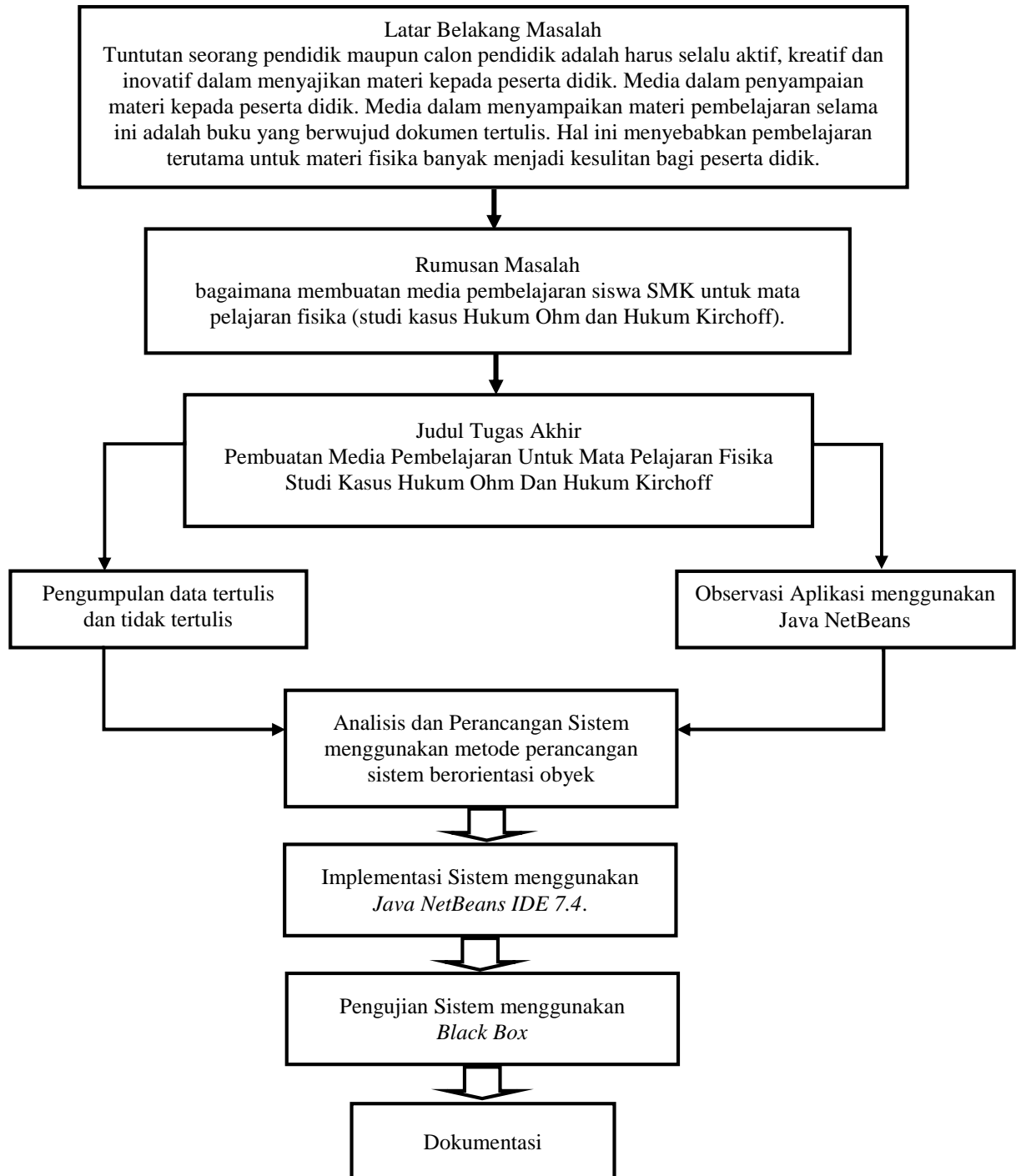
Aththibby (2011) pada karya ilmiah dengan judul “Perancangan Media Pembelajaran Fisika Berbasis Animasi Komputer Untuk Sekolah Menengah Atas Pokok Bahasan Hukum Newton Tentang Gerak” menyebutkan bahwa media diperlukan untuk mempermudah siswa dalam memahami materi. Penggunaan aplikasi produk dari penelitian ini relatif mudah dan sederhana, hanya memasukan *input* (program) lalu tinggal meng-klik menggunakan mouse sehingga *Output* akan ditampilkan pada layar.

Pradipta (2013) pada tugas akhir dengan judul “Pengembangan Media Pembelajaran Fisika berbasis Multimedia Dengan *Swishmax 4* Pada Materi Kinematika Gerak Lurus Untuk Siswa SMA” menyebutkan bahwa Kinematika gerak lurus merupakan bab yang harus dikuasai siswa karena akan dibutuhkan pada materi-materi selanjutnya misalnya pada dinamika, perpaduan gerak, dan

gerak parabol. Sehingga untuk membuat siswa mau belajar, dibuat media pembelajaran yang mampu menarik siswa untuk belajar. Media pembelajaran berbasis animasi komputer ini masih belum menampilkan skor dan pembahasan soal setelah kuis.

Penelitian ini merupakan penelitian tentang bahasan yang hampir sama dengan penelitian sebelumnya namun memiliki beberapa perbedaan diantaranya pada media pembelajaran ini dilengkapi dengan halaman materi, latihan dan pembahasan pada tiap materi kompetensi dasar. Selain itu dilengkapi dengan gambar agar dapat lebih memvisualisasikan tentang materi yang disampaikan. Aplikasi Pembelajaran ini dibuat dengan menggunakan metode pengembangan sistem metode SDLC menggunakan pendekatan sistem yang disebut dengan pendekatan air terjun (*waterfall approach*) dengan melalui beberapa tahapan mulai dari perancangan sistem sampai dengan pemeliharaan sistem.

2.2. Kerangka Pemikiran

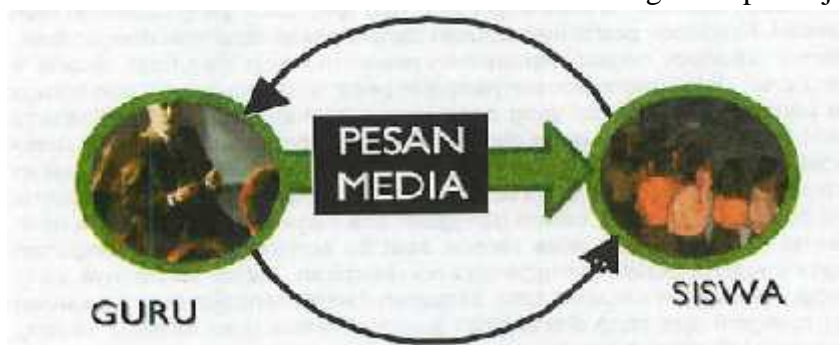


Gambar 2.1. Diagram Kerangka Pemikiran

2.3.Landasan Teori

2.3.1. Media Pembelajaran

Menurut Susilana dan Riyana (2009 : 1-2), pembelajaran merupakan suatu kegiatan yang melibatkan seseorang dalam upaya memperoleh pengetahuan, ketrampilan dan nilai-nilai positif dengan memanfaatkan berbagai sumber untuk belajar. Pembelajaran dapat melibatkan dua pihak yaitu siswa sebagai sumber untuk belajar. Pembelajaran dapat melibatkan dua pihak yaitu siswa sebagai pembelajar dan guru sebagai fasilitator. Yang terpenting dalam kegiatan pembelajaran adalah terjadinya proses belajar (*learning process*). Sebab sesuatu dikatakan hasil belajar kalau memenuhi beberapa ciri berikut : a. belajar sifatnya disadari, timbul dalam dirinya motivasi-motivasi untuk memiliki pengetahuan yang diharapkan sehingga tahapan-tahapan dalam belajar sampai pengetahuan itu dimiliki secara permanen (*retensi*) betul-betul disadari sepenuhnya. b. hasil belajar diperoleh dengan adanya proses, dalam hal ini pengetahuan diperoleh tidak secara spontanitas, instant, namun bertahap (*sequensial*). Seorang anak bisa membaca tentu tidak diperoleh hanya dalam waktu sesaat namun berproses cukup lama, kemampuan membaca diawali dengan kemampuan mengeja, mengenal huruf, kata dan kalimat. Seseorang yang tiba-tiba memiliki kecakapan seperti lari dengan kecepatan tinggi karena akibat *doping*, bukanlah hasil dari kegiatan belajar, namun efek dari obat atau zat kimia yang dikonsumsi. c. Belajar membutuhkan interaksi, khususnya interaksi yang sifatnya manusiawi. Seorang siswa akan lebih cepat memiliki pengetahuan karena bantuan dari guru, pelatih ataupun instruktur. Komunikasi dua arah antara siswa dan guru dapat terjadi.



Gambar 2.2. Proses Pembelajaran

Sumber : Susilana dan Riyana, 2009 : 4

Gambar 2.2., menunjukkan bahwa dalam proses pembelajaran itu terdapat pesan-pesan yang harus dikomunikasikan. Pesan tersebut biasanya merupakan isi dari suatu topik pembelajaran. Pesan-pesan tersebut disampaikan oleh guru kepada siswa melalui suatu media dengan menggunakan prosedur pembelajaran tertentu yang disebut metode (Susilana dan Riyana, 2009:4).

Susilana dan Riyana (2009:6), media secara harfiah kata tersebut berasal dari kata latin, merupakan bentuk jamak dari kata “medium”. Secara harfiah kata tersebut mempunyai arti perantara atau pengantar. Akan tetapi sekarang kata tersebut digunakan, baik untuk bentuk jamak maupun *mufrad*. Kemudian telah banyak pakar dan juga organisasi yang memberikan batasan mengenai pengertian media. Beberapa di antaranya mengemukakan bahwa media adalah sebagai berikut:

- a. Teknologi pembawa pesan yang dapat dimanfaatkan untuk keperluan pembelajaran. Jadi media adalah perluasan dari guru (Schram, 1982).
- b. *National Education Association* (NEA) memberikan batasan bahwa media merupakan sarana komunikasi dalam bentuk cetak maupun *audio visual*, termasuk teknologi perangkat kerasnya.
- c. Briggs berpendapat bahwa media merupakan alat untuk memberikan perangsang bagi siswa supaya terjadi proses belajar.
- d. *Association of Education Communication Technology* (AECT) memberikan batasan bahwa media merupakan segala bentuk dan saluran yang dipergunakan untuk proses penyaluran pesan.
- e. Sedangkan Gagne berpendapat bahwa berbagai jenis komponen dalam lingkungan siswa yang dapat merangsang siswa untuk belajar.
- f. Segala sesuatu yang dapat digunakan untuk menyalurkan pesan yang dapat merangsang pikiran, perasaan, perhatian dan kemauan siswa untuk belajar. (Miarso, 1989)

Susilana dan Riyana (2009:6-7), menyatakan bahwa selain pengertian media yang telah diuraikan di atas, masih terdapat pengertian lain yang dikemukakan oleh beberapa ahli. Beberapa pengertian media pembelajaran yang dimaksud adalah berikut ini,

- a. Teknologi pembawa pesan yang dapat dimanfaatkan untuk keperluan pembelajaran (Schramm, 1982)
- b. Sarana fisik untuk menyampaikan isi/materi pembelajaran seperti buku, film, video, slide dan sebagainya.
- c. Sarana komunikasi dalam bentuk cetak maupun pandang dengar, termasuk teknologi perangkat kerasnya.

Susilana dan Riyana (2009:7), mengatakan media pembelajaran selalu terdiri atas dua unsur penting, yaitu unsur peralatan atau perangkat keras (*hardware*) dan unsur pesan yang dibawanya (*message/software*). Dengan demikian media pembelajaran memerlukan peralatan untuk menyajikan pesan, namun yang terpenting bukanlah peralatan itu, tetapi pesan atau informasi belajar yang dibawakan oleh media tersebut.

Menurut Ragiell (2016), dalam kegiatan pembelajaran terdapat proses belajar mengajar yang pada dasarnya merupakan proses komunikasi. Dalam proses komunikasi tersebut, guru bertindak sebagai komunikator (*communicator*) yang bertugas menyampaikan pesan pendidikan (*message*) kepada penerima pesan (*communican*) yaitu anak. Agar pesan-pesan pendidikan yang disampaikan guru dapat diterima dengan baik oleh anak, maka dalam proses komunikasi pendidikan tersebut diperlukan wahana penyalur pesan yang disebut media pendidikan/pembelajaran.

Agar proses komunikasi pembelajaran berjalan secara efektif dan efisien, guru perlu menggunakan media untuk merangsang siswa dalam belajar. Jadi pada prinsipnya media bermanfaat untuk menunjang proses pembelajaran, hal ini bukan saja membuat penyajian menjadi lebih konkrit, tetapi juga ada beberapa kegunaan yang lain.

Menurut Tim Pengembang Ilmu Pendidikan FIP-UPI (2007 : 207), perkembangan media pembelajaran memang mengikuti perkembangan teknologi pendidikan. Apabila ditelaah lebih lanjut, perkembangannya paradigma dalam teknologi pendidikan mempengaruhi perkembangan media pembelajaran, adalah sebagai berikut :

- a. Dalam paradigma pertama, media pembelajaran sama dengan alat peraga audio visual yang dipakai oleh instruktur untuk melaksanakan tugasnya.
- b. Dalam paradigma kedua, media dipandang sebagai sesuatu yang dikembangkan secara sistemik serta berpegang kepada kaidah komunikasi.
- c. Dalam paradigma ketiga, media dipandang sebagai bagian integral dalam sistem pembelajaran dan karena itu menghendaki adanya perubahan pada komponen-komponen lain dalam proses pembelajaran.
- d. Media pembelajaran, dalam paradigma keempat, lebih dipandang sebagai salah satu sumber yang dengan sengaja dan bertujuan dikembangkan dan atau dimanfaatkan untuk keperluan belajar.

Media pembelajaran jika kita telaah lebih lanjut sebenarnya memiliki beberapa fungsi diantaranya :

- a. Media pembelajaran dapat mengatasi keterbatasan pengalaman yang dimiliki oleh peserta didik.
- b. Media pembelajaran dapat melampaui batas kelas.
- c. Media pembelajaran memungkinkan adanya interaksi langsung antara peserta didik dan lingkungannya.
- d. Media menghasilkan keseragaman pengamatan.
- e. Media dapat menanamkan konsep dasar yang benar, konkrit dan realistis.
- f. Media membangkitkan keinginan dan minat baru.
- g. Media membangkitkan motivasi dan merangsang anak untuk belajar.
- h. Media memberikan pengalaman yang integral/menyeluruh dari yang konkrit sampai yang abstrak.

2.3.2. Fisika

Menurut KBBI (2016), Fisika adalah ilmu tentang zat dan energi (seperti panas, cahaya, dan bunyi). Kaitannya adalah segala sesuatu yang ada di alam sekitar kita ini. Fisika menjadi dasar bagi ilmu pengetahuan untuk ilmu-ilmu lain yang berkaitan.

Menurut Young dan Freedman (2002:1), dua alasan kita belajar fisika. Pertama, fisika adalah salah satu ilmu yang paling dasar dari ilmu pengetahuan.

Ilmuwan dari segala disiplin ilmu memanfaatkan ide-ide dari fisika, mulai dari ahli kimia yang mempelajari struktur molekul sampai ahli palaentologi yang berusaha merekonstruksi bagaimana dinosaurus berjalan. Fisika juga merupakan dasar dari semua ilmu rekayasa dan teknologi. Tidak ada insinyur yang dapat merancang alat-alat praktis tanpa terlebih dahulu mengerti prinsip-prinsip dasar yang digunakan. Untuk merancang sebuah pesawat antariksa atau sebuah perangkat tikus yang lebih baik. Anda harus mengerti hukum-hukum dasar fisika.

Menurut Young dan Freedman (2002:1-2), fisika merupakan ilmu eksperimental. Fisikawan mengamati fenomena alam dan berusaha menemukan pola dan prinsip yang menghubungkan fenomena-fenomena ini. Pola ini disebut teori fisika atau ketika mereka sudah benar-benar terbukti dan digunakan luas, disebut hukum atau prinsip fisika. Perkembangan teori fisika memerlukan kreativitas dalam setiap tahapnya. Fisikawan harus belajar untuk mengajukan pertanyaan yang tepat, merancang percobaan untuk mencoba menjawab pertanyaan-pertanyaan itu dan menarik kesimpulan yang tepat dari hasilnya.

Materi fisika terkhusus adalah materi Hukum Ohm (hambatan) dan Hukum Kirchoff (arus dalam rangkaian). Hukum Ohm dan Hukum Kirchoff sendiri merupakan hukum yang membahas tentang arus listrik. Hukum Ohm sendiri merupakan suatu hukum yang menyatakan bahwa besar arus listrik yang mengalir melalui sebuah pengantar selalu berbanding lurus dengan beda potensial yang diterapkan kepadanya. Sedangkan Hukum Kirchoff adalah suatu hukum yang dipelajari pada bidang elektronika yang berfungsi untuk menganalisis arus dan tegangan yang masuk pada rangkaian.

Menurut Pauliza (2008:53), pada 1827, seorang ahli fisika bernama George Simon Ohm menyimpulkan bahwa kuat arus yang mengalir melalui penghantar sebanding dengan tegangan atau beda potensial suatu penghantar listrik tersebut. Hasil eksperimen selalu konstan. Oleh karena itu dihasilkan hubungan tersebut bersifat ohmik. Pernyataan yang dikemukakan Ohm kemudian dikenal dengan hukum Ohm.

Menurut Sudirman (2014:78), hambatan pada arus listrik didefinisikan sebagai perbandingan antara tegangan listrik dengan arus listrik dengan arus listrik yang melewatinya. Hambatan diberi simbol R dan satuannya ohm (Ω).

Masih menurut Sudirman (2014:82), terdapat 2 bagian Hukum Kirchoff yang sangat bermanfaat untuk menganalisis arus dalam rangkaian. Hukum yang pertama berkaitan dengan arah arus dalam menghadapi titik percabangan, hukum yang kedua bermanfaat untuk menganalisis arus dalam suatu rangkaian dalam suatu rangkaian yang kompleks untuk menentukan tegangan jatuh dalam suatu resistor.

2.3.3. Java NetBeans

Java adalah bahasa pemrograman yang berorientasi objek (OOP) dan dapat dijalankan pada berbagai *platform* sistem operasi. Perkembangan Java tidak hanya terfokus pada satu sistem operasi, tetapi dikembangkan untuk berbagai sistem operasi dan bersifat *open source*. Sebagai sebuah bahasa pemrograman, Java dapat membuat seluruh bentuk aplikasi, desktop, *web* dan lainnya, sebagaimana dibuat dengan menggunakan bahasa pemrograman konvensional yang lain menurut Joyce Avestro (2007:3).

Java pertama kali diluncurkan pada tahun 1995 sebagai bahasa pemrograman umum (*general purpose programming language*) dengan kelebihan dia bisa dijalankan di *web browser* sebagai *applet* menurut Utama (2002:4).



Gambar 2.3. Tampilan awal *Java NetBeans*

Keterangan tampilan awal jendela Java NetBeans yang ditampilkan pada Gambar 2.3 adalah sebagai berikut:

1. *Menu Bar*
2. *Project Bar*
3. *Navigation Bar*
4. *Output Bar*

Menurut Hariyanto (2014:3), Java telah berkembang dari semula ditujukan untuk pemrograman *applet* di *web browser* menjadi bahasa pemrograman pengembangan aneka ragam aplikasi, mulai dari yang berjalan di *handheld device* seperti *handphone*, PDA (*Personal Digital Assistant*) sampai aplikasi tersebar skala *enterprise* di beragam komputer *server*. Java merupakan bahasa orientasi objek untuk pengembangan aplikasi mandiri, aplikasi berbasis *internet*, aplikasi untuk perangkat cerdas yang dapat berkomunikasi lewat *Internet*/jaringan komunikasi. Melalui teknologi Java, dimungkinkan perangkat *audio stereo* di rumah terhubung jaringan komputer. Java tidak lagi hanya bahasa untuk membuat *applet* yang memperindah halaman *web* tapi Java telah menjadi bahasa untuk pengembangan aplikasi skala *enterprise* berbasis jaringan besar.

Menurut Wahana Komputer (2010:6), Java memang sejak awal dirancang untuk memenuhi kemampuan pemrograman berorientasi objek. Tidak seperti bahasa pemrograman lainnya yang mulanya hanya diperuntukkan bagi pemrograman prosedural. Pemrograman berorientasi objek adalah pendekatan atau metodologi perancangan program berdasarkan objek. Metode ini menggantikan metode prosedural yang telah lama digunakan. Dalam pemrograman berorientasi objek semua hal dapat dianggap objek. Contohnya adalah manusia, bunga, tanaman, komputer, bahkan *database* dapat dianggap sebuah objek. Selain itu, pemrograman berorientasi objek memiliki kelebihan-kelebihan, seperti sangat fleksibel, modular, penggunaan kembali kode program melalui enkapsulasi data, *inheritance* dan *polimorphism*.

Menurut Yuniar Supardi (2010:2), versi pertama Java, dirilis pada awal tahun 1996, dengan sebutan JDK 1.1 (*Java Development Kit* versi 1.1), kemudian

muncul Java 2 yang dilengkapi dengan Swing, yaitu teknologi GUI (*Graphical Use Interface*) yang dapat menghasilkan aplikasi dekstop.

Menurut Hendry (2008:29), mengapa harus GUI? Jawabnya singkat saja tuntutan zaman. Di masa sekarang ini kita bisa melihat pertumbuhan kota-kota di Indonesia yang dipenuhi dengan *mall-mall*. Setali tiga uang, tuntutan akan aplikasi yang mudah dipakai dan menarik perhatian mata menjadi hal yang lumrah. Hal inilah yang menyebabkan teknik pemrograman berbasis GUI menjadi kebutuhan pokok dalam dunia teknologi informasi sekarang.




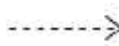






2.3.4. Star UML

Menurut Triandini dan Suardika (2012:1), StarUML adalah *platform* pemodelan perangkat lunak yang mendukung UML (*Unified Modeling Language*). Star UML berbasiskan pada UML versi 1.4, menyediakan sebelas jenis Diagram yang berbeda dan mendukung notasi UML 2.0. StarUML juga secara aktif mendukung pendekatan MDA (*Model Driven Architecture*) dengan mendukung konsep UML *profile*. StarUML unggul dalam hal kustomisasi lingkungan kerja pengguna dan memiliki ekstensibilitas tinggi pada fungsionalitasnya. StarUML mengklaim diri sebagai salah satu alat pemodelan perangkat lunak terkemuka yang menjamin dapat memaksimalkan produktivitas dan kualitas proyek perangkat lunak anda

StarUML menyediakan secara sederhana dan kuat arsitektur *plug-in* sehingga siapapun dapat mengembangkan *plug-in* modul dalam bahasa COM-kompatibel (C++, Delphi, VB, dll). Hal yang paling penting dalam pengembangan perangkat lunak adalah *usability*. StarUML diimplementasikan untuk memberikan berbagai fitur yang *user-friendly* seperti dialog cepat, manipulasi keyboard, ikhtisar diagram, dll




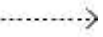


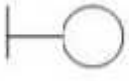

Berikut ini simbol-simbol yang digunakan UML diagram disajikan pada Tabel 2.1 sampai dengan Tabel 2.6.

Tabel 2.1. Simbol-simbol dalam *Use Case Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).
3		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
5		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		<i>Note</i>	Elemen fisik yang ada saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.






Tabel 2.1. menunjukkan simbol-simbol yang digunakan pada saat kita membuat *Use Case Diagram*. *Use Case Diagram* menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah apa yang dibuat sistem dan bukan bagaimana. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem.

Tabel 2.2. Simbol-simbol dalam *Use Case Realization Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
3		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
4		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).
5		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		Objek Antarmuka	merupakan kumpulan operasi tanpa implementasi dari suatu class.
8		Objek Kontrol	Memodelkan perilaku mengatur satu atau beberapa use case.

Tabel 2.1. menunjukkan simbol-simbol yang digunakan pada saat kita membuat *Use Case Diagram Realization*. Menggambarkan bagaimana sebuah use-case direalisasikan dalam bentuk kolaborasi dari berbagai objek. Di dalam realisasi use-case terdapat satu atau beberapa diagram kelas. Untuk menentukan kelas-kelas yang terdapat dalam sebuah realisasi use-case, dipergunakan diagram interaksi.

Tabel 2.3. Simbol-simbol dalam *Activity Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran



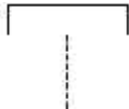


Tabel 2.3. menunjukkan simbol-simbol yang digunakan pada saat kita membuat *Activity Diagram*. Salah satu cara untuk memodelkan aliran kerja (workflow) dari *business use case* dalam bentuk grafik. Diagram ini juga menunjukkan langkah-langkah dalam aliran kerja, titik-titik keputusan dalam aliran kerja, siapa yang bertanggung jawab menyelesaikan masing-masing aktivitas dan objek-objek yang digunakan dalam aliran kerja.

Elemen-elemen yang dapat kita temukan dalam *Activity Diagram* antara lain:

- 1) *Swimlanes*, menunjukkan siapa yang bertanggung jawab melakukan aktivitas dalam suatu diagram.
- 2) *Activities State*, adalah kegiatan dalam aliran kerja.
- 3) *Action State*, adalah langkah-langkah dalam sebuah *activity*. *Action* bisa terjadi saat memasuki *activity*, meninggalkan *activity*, saat di dalam *activity*, atau pada *event* yang spesifik.
- 4) *Business object*, adalah entitas-entitas yang digunakan dalam aliran kerja.
- 5) *Transition*, menunjukkan bagaimana aliran kerja itu berjalan dari satu aktivitas ke aktivitas lainnya.

- 6) *Decision point*, menunjukkan dimana sebuah keputusan perlu dibuat dalam aliran kerja.
- 7) *Synchronization*, menunjukkan dua atau lebih langkah dalam aliran kerja berjalan secara serentak à *Fork* dan *Join*.
- 8) *Start state*, menunjukkan dimana aliran kerja itu dimulai.
- 9) *End state*, menunjukkan dimana aliran kerja itu berakhir.







Tabel 2.4. Simbol-simbol dalam *Sequence Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan
2		<i>Object</i>	Merupakan <i>instance</i> dari sebuah <i>class</i> dan dituliskan tersusun secara horizontal. Digambarkan sebagai sebuah <i>class</i> (kotak) dengan nama obyek didalamnya yang diawali dengan sebuah titik koma.
3		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
4		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
5		<i>Activation</i>	sebuah obyek yang akan melakukan sebuah aksi.

Tabel 2.4. menunjukkan simbol-simbol yang digunakan pada saat kita membuat *Sequence Diagram*. Diagram ini menggambarkan kelakuan objek pada

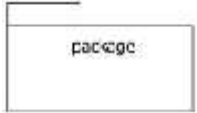
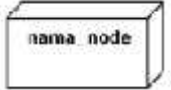


use case dengan mendiskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek.

Tabel 2.5. Simbol-simbol dalam *Class Diagram*

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

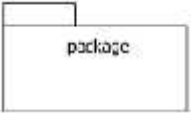
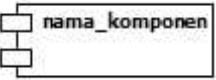

Tabel 2.5. menunjukkan simbol-simbol yang digunakan pada saat kita membuat *Class Diagram*. Diagram ini menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan di buat untuk membangun sistem. kelas memiliki apa yang disebut atribut dan metode atau oprasi. Diagram kelas dibuat agar pembuat program atau programmer membuat kelas-kelas sesuai rancangan di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron.

Tabel 2.6. Simbol-simbol dalam *Deployment Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Package</i>	Bungkusan satu atau lebih <i>node</i>
2		<i>Node</i>	Biasanya mengacu pada perangkat keras (<i>hardware</i>), perangkat lunak yang tidak dibuat sendiri (<i>software</i>), jika di dalam node disertakan komponen untuk mengkonsistensikan rancangan maka komponen yang diikut sertakan harus sesuai dengan komponen yang telah didefinisikan sebelumnya pada diagram komponen.
3		<i>Dependency</i>	Kebergantungan antar <i>node</i> , arah panah mengarah pada node yang dipakai
4		<i>Link</i>	Relasi antar <i>node</i>

Tabel 2.6. menunjukkan simbol-simbol yang digunakan pada saat kita membuat *Deployment Diagram*. Diagram ini menggambarkan gambaran proses-proses berbeda pada suatu sistem yang berjalan dan bagaimana relasi di dalamnya.

Tabel 2.7. Simbol-simbol dalam *Component Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Package</i>	Bungkusan satu atau lebih <i>node</i>
2		<i>Component</i>	Komponen sistem
3		<i>Dependency</i>	Kebergantungan antar <i>node</i> , arah panah mengarah pada node yang dipakai

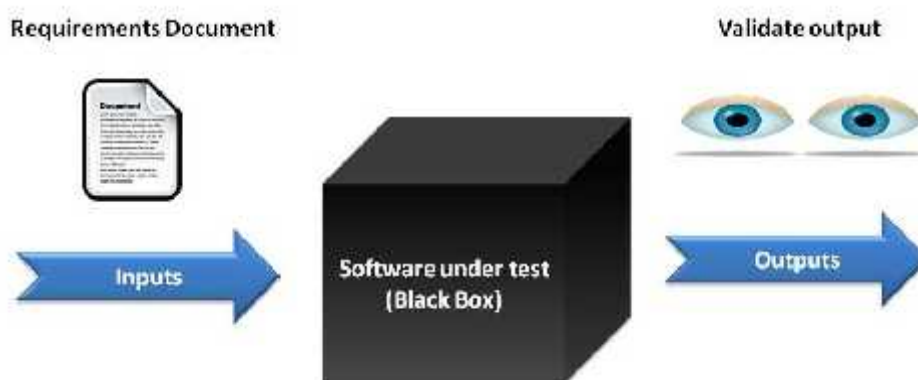
Tabel 2.7. menunjukkan simbol-simbol yang digunakan pada saat kita membuat *Component Diagram*. Diagram ini menunjukkan struktur dan hubungan antar komponen *software* termasuk ketergantungan (*dependency*) diantara komponen-komponen tersebut.

2.3.5. Metode Pengujian *Black Box*

Pengujian perangkat lunak adalah sangat diperlukan dalam suatu sistem informasi, dimana dengan melakukan suatu pengujian akan ditemukan kesalahan atau *error* yang muncul dari sistem perangkat lunak tersebut. Dengan demikian tentunya seorang programmer akan bisa mengetahui dan apa yang harus dikerjakan selanjutnya.

Metode *Black Box* memungkinkan perancang perangkat lunak mendapatkan serangkaian kondisi *input* yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program menurut Rouf (2011:4).

Menurut Al Fatta (2007:172), terfokus pada apakah unit program memenuhi kebutuhan (*requirement*) yang disebutkan dalam spesifikasi. Pada *Black Box testing*, cara pengujian hanya dilakukan dengan menjalankan atau mengeksekusi unit atau modul, kemudian diamati apakah hasil dari unit itu sesuai dengan proses bisnis yang diinginkan. Pada Gambar 2.4. Jika ada unit yang tidak sesuai *output*-nya maka untuk menyelesaikannya, diteruskan pada pengujian yang kedua, yaitu *White Box Testing*.



Gambar. 2.4 Skema Metode *Black Box*

Menurut Limaye (2009:108-109), keunggulan dari Metode *Black Box* adalah biasa digunakan untuk menguji perilaku dari suatu aplikasi untuk mewujudkan kebutuhan konsumen.

- a. Metode *Black Box* adalah satu-satunya metode untuk membuktikan sesuatu harus dilakukan atau tidak oleh software dimana itu dapat menyebabkan masalah bagi pengguna/konsumen.
- b. Metode *Black Box* adalah satu-satunya metode untuk menunjukkan bahwa *software* itu aktif dan dapat bekerja.
- c. Beberapa tipe pengujian hanya dapat dilakukan oleh Metode *Black Box*, sebagai contoh unjuk kerja dan keamanan.

Kelemahan dari Metode *Black Box* adalah

- a. Beberapa kesalahan logika dalam koding dapat terlewatkan dalam Metode *Black Box* karena Metode *Black Box* dijalankan berdasarkan kebutuhan bukan desain.
- b. Beberapa tes yang tidak perlu mungkin dibutuhkan untuk menjalankan beberapa bagian yang lain.