BAB III

OBJEK DAN METODE PENELITIAN

3.1. Objek Penelitian

Objek penelitian yang dilakukan oleh penulis disini, yaitu pada Sekolah Menengah Atas Angkasa Lanud Husein Sastranegara Bandung, berikut uraian mengenai SMA Angkasa.

3.1.1. Sejarah Singkat SMU Angkasa Lanud Husein Sastranegara Bandung

SMU Angkasa pada awal berdirinya bernama SMA Angkasa. Sekolah ini didirikan dengan latar belakang adanya animo orang tua murid siswa SMP Angkasa yang cukup besar untuk mengusulkan dibuat sekolah lanjutan atau SLTA selain STM di lingkungan sekitar pangkalan udara (Husein Sastranegara). Hal itu disambut baik oleh pengurus yayasan yang telah memiliki tiga sekolah TK 4 SD 1 SMP dan 1 STM. Maka mulailah direncanakan pendirian SMU Angkasa ini. Sejak awal bangunan ini digunakan oleh dua sekolah yaitu bersama SMK Angkasa yang dulun ya STM Angkasa atau STM Garuda. Mengacu kepada program yang dikembangkan oleh Badan Kerjasama Pendidikan (BKSP) saat itu yaitu untuk memajukan pembinaan umum pendidikan keluarga di lingkungan

Lanuma Husein (sekarang Lanud) maka resmilah SMA Angkasa didirikan pada tahun 1980 tepatnya pada tanggal 30 Agustus.

Pada awalnya pembukaan SMA Angkasa memiliki tujuan khusus, yaitu Menampung lulusan SMP Angkasa terutama putra-putri anggota TNI AU yang tidak dapat diterima di sekolah negeri atau siswa dari daerah lain yang pindah karena orang tua pindah tugas dinas.

Tujuan lain yang bersifat umum adalah membantu usaha pemerintah dalam menanggulangi penyaluran lulusan SMP dan merealisasikan program kerja PIA Ardya Garini Daerah V Cabang Lnud Husein Sastranegara dalam rangka pengembangan dan pembinaan sekolah umum di lingkunan TNI AU.

3.1.2. Visi dan Misi Sekolah

Visi dan Misi SMA Angkasa Lanud Husein Sastranegara Bandung yaitu :

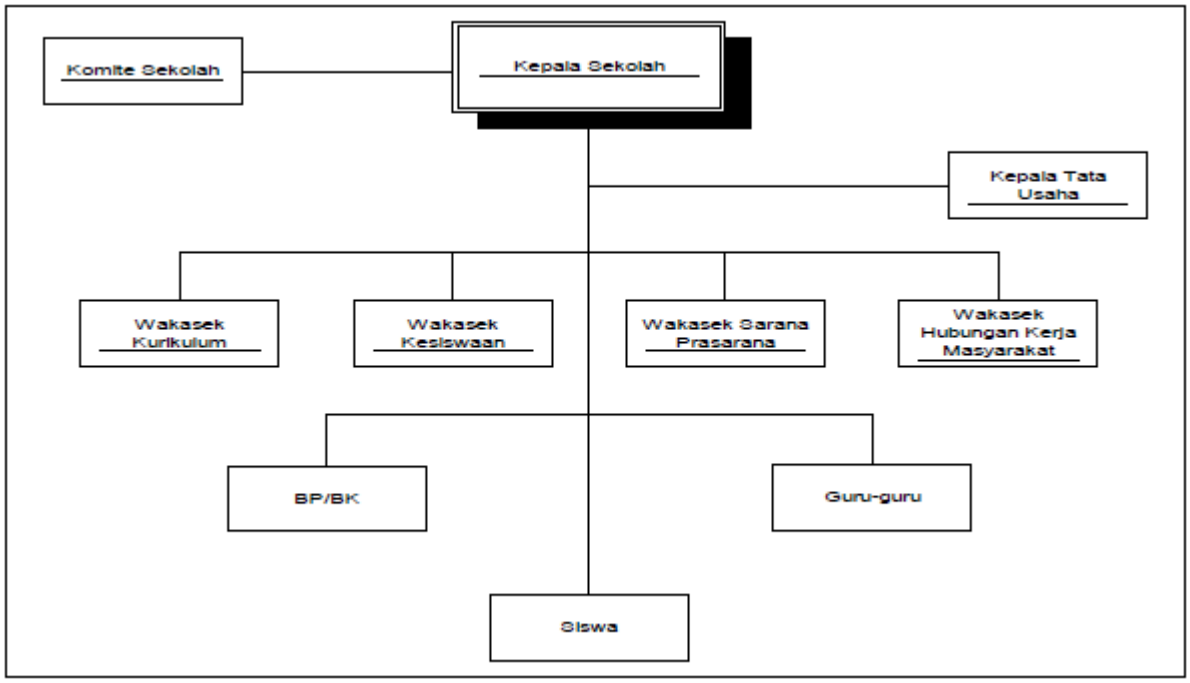
Visi :

Meningkatkan prestasi melalui usaha peningkatan kualitas diri.

Misi :

Mewujudkan SMA Angkasa sebagai wahana pengembangan pendidikan, kebudayaan, dan ilmu pengetahuan dan teknolohi yang berbasiskan kepada keimanan dan ketakwaan kepada Tuhan Yang Maha Esa.

3.1.3. Struktur Organisasi Sekolah



Gambar 3.1. Struktur Organisasi Sekolah SMA Angkasa

[Sumber : SMA Angkasa Lanud Husein Sastranegara Bandung]

3.1.4. Deskripsi Tugas

1. Kepala Sekolah

Kepala Sekolah mempunyai wewenang dan kewajiban dalam mengatur segala sesuatu di lingkungan persekolahan. Tugasnya meliputi :

A. Mengatur dan mengadakan urusan pendidikan.

B. Mengatur dan mengadakan urusan Tata Usaha sekolah.

C. Mengatur dan mengadakan urusan keuangan sekolah.

D. Mengatur dan mengadakan urusan kepegawaian.

E. Mengatur dan mengadakan urusan perpustakaan laboratorium.

F. Mengatur dan mengadakan urusan sarana dan prasarana.

G. Menyelenggarakan hubungan dengan orang tua dan masyarakat.

H. Mengusulkan mutasi tenaga pengajar dan administrasi ke Kepala Kanwil.

I. Mengatur pembinaan kesiswaan.

J. Membina hubungan antara pemimpin, guru, dan murid.

K. Melihat hasil laporan data siswa, data guru, data kelas, data mata

L. pelajaran, data nilai akhir siswa (raport), dan laporan iuran SPP per bulan.

2. Wakasek Kurikulum

Bagian kurikulum membantu kegiatan wakil Kepala Sekolah untuk urusan kurikulum, tugasnya meliputi :

A. Menyusun program pengajaran.

B. Menyusun pembagian tugas guru dan jadwal pelajaran.

C. Menyusun supervisi bagi guru.

D. Melaksanakan pemilihan bagi guru teladan dan membina kegiatan lomba bidang akademik tingkat kabupaten dan propinsi.

E. Menyusun laporan pelaksanaan pelajaran.

F. Menerapkan standar lulus siswa.

G. Menyusun jadwal pelaksanaan ulangan serta UN.

H. Menyusun program pemantapan kelas XII.

I. Mengatur jadwal penerimaan buku laporan pendidikan dan ijazah.

J. Mengkoordinasikan dan mengarahkan penyusunan perangkat pembelajaran.
K. Melihat hasil laporan data siswa, data guru, data kelas, data mata
L. pelajaran, data nilai akhir siswa (raport) yang telah dibuat oleh admin.

3. Guru

Guru bertanggung jawab kepada Kepala Sekolah dan mempunyai tugas melaksanakan proses belajar mengajar secara efektif dan efisien, meliputi :

A. Membuat program pengajaran Analisa materi dan pembelajaran (AMP).
    1) Program tahunan/semester.
    2) Silabus.
    3) Program rencana pengajaran.
    4) Lembar kegiatan siswa (LKS).
B. Melaksanakan kegiatan pembelajaran, penilaian dan analisis hasil ulangan harian, semester/tahunan.
C. Mengisi daftar nilai siswa.
D. Menghitung dan mengumpulkan angka kredit untuk kenaikan pangkatnya.
E. Menginput data nilai siswa setelah mengadakan ulangan harian, nilai
F. tugas, UTS, dan UAS.

4. Wali kelas

Wali kelas membantu kepala sekolah dalam kegiatan-kegiatan sebagai berikut

a) Pengelolaan kelas
b) Pen yusunan dan pembuatan statistik bulanan siswa
c) Pengisian daftar kumpulan nilai siswa
d) Pembuatan catatan khusus dan mutasi siswa
e) Pembagian buku laporan siswa

5. Bagian Tata Usaha (TU)

Bagian Tata Usaha bertanggung jawab kepada Kepala Sekolah, tugasnya meliputi :

a) Menyusun program kerja ketatausahaan sekolah.
b) Membina dan mengembangkan karir pegawai Tata Usaha Sekolah.
c) Mengkoordinasikan dan melaksanakan 7K (Keamanan, Kebersihan,
d) Ketertiban, Kerindangan, Keindahan, Kebersamaan, dan Kekeluargaan).
e) Menyusun laporan pelaksanaan kegiatan secara berkala setelah mengadakan evaluasi kegiatan pengadministrasian.
f) Memimpin, membina, mengatur, dan mengkoordinir pelaksanaan seluruh urusan ketatausahaan yang dibawahinya.
g) Memelihara hubungan yang harmonis antara sesama karyawan Tata Usaha dengan guru, maupun karyawan Tata Usaha dengan personil sekolah lainnya.

h) Membuka surat-surat Dinas yang dialamatkan kepada Kepala Sekolah, kecuali surat yang bertanda rahasia, memeriksa proses surat dan membubuhi paraf.

i) Memeriksa dan menandatangani agenda kegiatan harian pegawai setiap hari pada saat menjelang pulang.

j) Menginput data pembayaran iuran SPP siswa, mencetak bukti pembayaran, dan mencetak laporan SPP bulanan.

6. Siswa

Siswa berkewajiban untuk :

a. Mematuhi segala peraturan dan tata tertib yang dibuat sekolah.
b. Masuk sekolah dan datang tepat waktu.
c. Membayar iuran wajib sekolah (SPP) sesuai waktu yang telah ditentukan.

3.2. Metode Penelitian

Dalam menunjang pelaksanaan penelitian ini perlu adanya metode untuk mengatasi masalah-masalah yang ada, maka diperlukan metode dan pengumpulan data.

3.2.1. Desain Penelitian

Dalam melakukan penelitian, desain penelitian merupakan kerangka atau perincian prosedur kerja yang akan dilakukan pada waktu meneliti, sehingga diharapkan dapat memberikan gambaran dan arah mana yang akan dilakukan dalam melaksanakan penelitian tersebut, desain penelitian yang baik dapat

memudahkan kita dalam melakukan penelitian dan sesuai dengan tujuan yang telah ditetapkan sebelumnya.

Sesuai dengan tujuan yang ingin dicapai dalam penelitian ini, penulis mengumpulkan data sebagai bahan laporan dengan menggunakan metode deskriptif, yaitu metode yang menggambarkan atau menguraikan keadaan situasi pada tempat observasi, melakukan penelitian dan kemudian melakukan analisis sehingga dapat diperoleh suatu kesimpulan.

3.2.2. Jenis dan Metode Pengumpulan Data

Dalam suatu penelitian alat pengumpulan data sangatlah mempengaruhi kualitas data yang akan didapatkan sekaligus menentukan kualitas dari penelitian. Jenis data yang digunakan dalam penelitian ini yaitu data primer dan data sekunder.

3.2.2.1. Sumber Data Primer

Jenis pengumpulan data primer merupakan penelitian yang mengumpulkan data langsung dari lapangan penelitian atau tempat penelitian untuk mengetahui keadaan penelitian yang akan dijalankan. Metode yang dipakai dibagi ada beberapa cara sebagai berikut :

a. Observasi

   Observasi merupakan proses untuk mendapatkan data dengan mengadakan pengamatan ditempat lokasi penelitian secara langsung yaitu di SMA

Angkasa Lanud Husein Sastranegara Bandung, kemudian hasil dari pengamatan tersebut dicatat dan dianalisis lebih lebih lanjut.

b. Wawancara

Wawancara merupakan proses memperoleh keterangan untuk tujuan penelitian dengan cara melakukan tanya jawab secara langsung dengan narasumber yang terkait dengan objek penelitian yaitu tentang mata pelajaran fisika yang berada di SMA Angkasa Lanud Husein Sastranegara Bandung. Metode ini dilakukan agar mendapatkan data serta informasi secara langsung dari narasumbernya.

3.2.2.1 Sumber data Sekunder (Dokumentasi)

Data sekunder merupakan pengumpulan data dengan cara mempelajari data yang telah tersedia atau dikumpulkan terlebih dahulu oleh pihak lain seperti buku-buku, literatur internet atau artikel-artikel ilmiah yang dapat dikaji sebagai bahan rujukan dan landasan teoritis dalam pemecahan masalah.

Cara yang digunakan untuk mengumpulkan data sekunder adalah metode dokumentasi, yaitu dengan mengumpulkan data dan informasi yang diperlukan dari sumber-sumber yang ada. Berikut dokumen yang diamati dalam tahap analisis sistem yaitu :

a. Dokumen sejarah berdirinya SMA Angkasa Lanud Husein Sastranegara Bandung, visi dan misi, struktur organisasi, dan prosedur kerja (*job descriptio*n).

3.2.3. Metode Pendekatan / Pengembangan Sistem

Metode pendekatan dan pengembangan sistem adalah beberapa metode yang digunakan pada penelitian ini, diantaranya metode pengembangan sistem dan metode pendekatan sistem dan alat bantu analisis dan perancangan, Dibawah ini akan diuraikan tentang pemahaman dari metode tersebut.

3.2.3.1. Metode Pendekatan Sistem

Metode pendekatan sistem merupakan serangkaian langkah-langkah pemecahan masalah yang memastikan bahwa masalah dipahami, solusi alternative dipertimbangkan dan solusi yang dipilih bekerja. Proses pemecahan masalah secara sistematis bermula dari John Dewey, seorang professor filosofi di Columbia University pada awal abad ini. Dalam bukunya (1910) diidentifikasi ada 3 seri penilaian dalam memecahkan suatu kontroversi yang memadai :

1. Mengenali kontroversi.
2. Menimbang klaim alternatif.
3. Membentuk penilaian.

Metode pendekatan sistem yang digunakan adalah metode pendekatan terstruktur yaitu suatu proses untuk mengimplementasikan urutan langkah untuk menyelesaikan suatu masalah dalam bentuk program. Pendekatan terstruktur dilengkapi dengan alat-alat (*tools)* dan teknik-teknik yang dibutuhkan dalam pengembangan sistem, sehingga hasil akhir dari sistem yang dikembangkan akan diperoleh sistem yang strukturnya didefinisikan dengan baik dan jelas.

3.2.3.2. Metode Pengembangan Sistem

Dalam perancangan sebuah sistem hal yang sangat penting dan harus diperhatikan agar menghasilkan suatu sistem yang baik dan dapat melengkapi informasi yang dibutuhkan merupakan kajian teknis formal. Untuk itu digunakan metode – metode perancangan sistem. Pengembangan sistem yang penulis gunakan dalam pelaporan tugas akhir ini adalah dengan menggunakan pendekatan model SDLC (*System Development Life Cycle*).

Dikutip dari http://id.wikipedia.org/wiki/SDLC. *Siklus hidup pengembangan sistem (Systems Development Life Cycle) adalah proses pembuatan dan pengubahan sistem serta model dan metodologi yang digunakan untuk mengembangkan sistem-sistem tersebut. Konsep ini umumnya merujuk pada sistem komputer atau informasi.*

SDLC berfungsi untuk menggambarkan tahapan-tahapan utama dan langkah-langkah dari setiap tahapan yang secara garis besar terbagi dalam tiga kegiatan utama, yaitu :

a. *Analysis*

Digunakann untuk membuat keputusan, apabila sistem saat ini mempunyai masalah atau sudah tidak berfungsi secara baik, dan hasil analisisnya digunakan sebagai dasar untuk memperbaiki system.

b. *Design* (perancangan)

Tahapan perancangan / desain memiliki tujuan untuk mendesain sistem baru yang dapat menyelesaikan masalah-masalah yang dihadapi perusahaan yang

diperoleh dari pemlihan alternatif sistem yang terbaik. Kegiatan yang dilakukan dalam tahapan perancangan ini meliputi perancangan output, input, dan file.

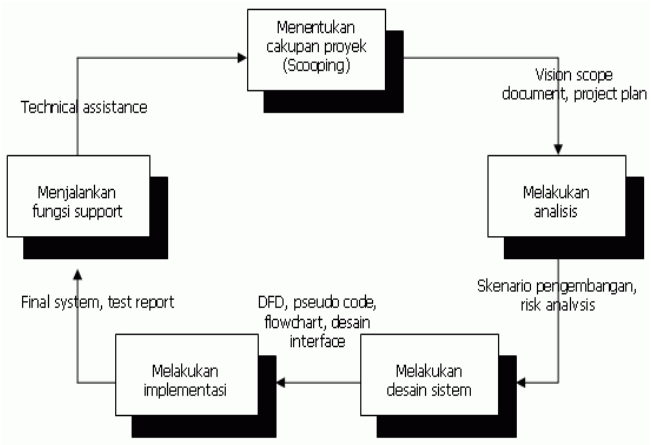
c. Implementation

Memiliki beberapa tujuan yaitu untuk melakukan kegiatan spesifikasi rancangan logikal ke dalam kegiatan yang sebenarnya dari sistem informasi yang akan dibangunnya atau dikembangkan ny, lalu mengimplementasikan sistem yang baru tersebut ke dalam salah satu bahasa pemrograman yang paling sesuai. Pada tahap ini juga harus menjamin bahwa sistem yang baru dapat berjalan secara optimal. Kegiatan yang dilakukan dalam tahap implementasi ini adalah pembuatan program dan test data, pelatihan dan pergantian sistem.

Dalam penerapan tahapan pengembangan sistem informasi penulis menggunakan sistem pengembangan *prototype*. *Prototyping* merupakan salah satu metode pengembangan perangat lunak yang banyak digunakan. Dengan metode prototyping ini pengembang dan pelanggan dapat saling berinteraksi selama proses pembuatan sistem. Sering terjadi seorang pelanggan hanya mendefinisikan secara umum apa yang dikehendakinya tanpa menyebutkan secara detal output apa saja yang dibutuhkan, pemrosesan dan data-data apa saja yang dibutuhkan. Sebaliknya disisi pengembang kurang memperhatikan efesiensi algoritma, kemampuan sistem operasi dan *interface* yang menghubungkan manusia dan komputer.

Untuk mengatasi ketidakserasian antara pelanggan dan pengembang , maka harus dibutuhakan kerjasama yanga baik diantara keduanya sehingga pengembang akan mengetahui dengan benar apa yang diinginkan dengan tidak

mengesampingkan segi-segi teknis dan pelanggan akan mengetahui proses-proses dalm menyelasaikan system yang diinginkan. Dengan demikian akan menghasilkan sistem sesuai dengan jadwal waktu penyelesaian yang telah ditentukan. Kunci agar model *prototype* ini berhasil dengan baik adalah dengan mendefinisikan aturan-aturan main pada saat awal, yaitu pelanggan dan pengembang harus setuju bahwa *prototype* dibangun untuk mendefinisikan kebutuhan. *Prototype* akan dihilangkan sebagian atau seluruhnya dan perangkat lunak actual direkayasa dengan kualitas dan implementasi yang sudah ditentukan.



Gambar 3.2 Proses *Prototype*

(Sumber: http://www.ristinet.com/2010/04/16)

a) Pengumpulan kebutuhan

Pelanggan dan pengembang bersama-sama mendefinisikan format seluruh perangkatlunak, mengidentifikasikan semua kebutuhan, dan garis besar sistem yang akandibuat.

b) Membangun *prototyping*

Membangun prototyping dengan membuat perancangan sementara yang berfokuspada penyajian kepada pelanggan (misalnya dengan membuat input dan formatoutput)

c) Evaluasi *protoptyping*

Evaluasi ini dilakukan oleh pelanggan apakah *prototyping* yang sudah dibangun sudahsesuai dengan keinginann pelanggan. Jika sudah sesuai maka langkah 4 akan diambil.Jika tidak *prototyping* direvisi dengan mengulangu langkah 1, 2 , dan 3.

d) Mengkodekan Sistem

Dalam tahap ini *prototyping* yang sudah disepakati diterjemahkan ke dalam bahasa pemrograman yang sesuai

e) Menguji Sistem

Setelah sistem sudah menjadi suatu perangkat lunak yang siap pakai, harus ditesdahulu sebelum digunakan. Pengujian ini dilakukan dengan *White Box*, *Black Box*, *Basis Path*, pengujian arsitektur dan lain-lain

f) Evaluasi Sistem

Pelanggan mengevaluasi apakah sistem yang sudah jadi sudah sesuai dengan yangdiharapkan . Jika ya, langkah 7 dilakukan; jika tidak, ulangi langkah 4 dan 5.

g) Menggunakan system

Perangkat lunak yang telah diuji dan diterima pelanggan siap untuk digunakan.

Adapun alasan mengapa penulis menggunakan metode ini karena metode ini mempunyai bebrapa keunggulan diantaranya :

Keunggulan *prototyping* adalah:

1. Adanya komunikasi yang baik antara pengembang dan pelanggan
2. Pengembang dapat bekerja lebih baik dalam menentukan kebutuhan pelanggan
3. Pelanggan berperan aktif dalam pengembangan system
4. Lebih menghemat waktu dalam pengembangan system
5. Penerapan menjadi lebih mudah karena pemakai mengetahui apa yang diharapkannya.

Kelemahan *prototyping* adalah :

1. Pelanggan kadang tidak melihat atau menyadari bahwa perangkat lunak yang ada belum mencantumkan kualitas perangkat lunak secara keseluruhan dan juga belum memikirkan kemampuan pemeliharaan untuk jangka waktu lama.
2. Pengembang biasanya ingin cepat menyelesaikan proyek. Sehingga menggunakan algoritma dan bahasa pemrograman yang sederhana untuk membuat prototyping lebih cepat selesai tanpa memikirkan lebih lanjut bahwa program tersebut hanya merupakan cetak biru sistem.
3. Hubungan pelanggan dengan komputer yang disediakan mungkin tidak mencerminkan teknik perancangan yang baik

Prototyping bekerja dengan baik pada penerapan-penerapan yang berciri sebagai berikut:

1. Resiko tinggi Yaitu untuk maslaha-masalah yang tidak terstruktur dengan baik, ada perubahan yang besar dari waktu ke waktu, dan adanya persyaratan data yang tidak menentu.
2. Interaksi pemakai penting. Sistem harus menyediakan dialog on-line antara pelanggan dan komputer.
3. Perlunya penyelesaian yang cepat
4. Perilaku pemakai yang sulit ditebak
5. Sitem yang inovatif. Sistem tersebut membutuhkan cara penyelesaian masalah dan penggunaan perangkat keras yang mutakhir
6. Perkiraan tahap penggunaan sistem yang pendek.

3.2.3.3. Alat Bantu Analisis & Perancangan

A. *Unified Modeling Languange* (UML)

Unified Modelling Language (UML) adalah keluarga notasi grafis yang didukung oleh meta-model tunggal, yang membantu pendeskripsian dan desain sistem perangkat lunak, khususnya sistem yang dibangun menggunakan pemograman berorientasi objek (Fowler, 2005:1). Selain itu UML juga dapat diartikan sebagai sebuah bahasa yang telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem perangkat lunak (Dharwiyanti dan Wahono, 2003:2). *Unified Modelling Language* (UML)

menawarkan sebuah standar untuk merancang model sebuah sistem. Tujuan UML adalah :

1. Memberikan model yang siap pakai, bahasa pemodelan visual yang ekspresif untuk mengembangkan dan saling menukar model dengan mudah dan dimengerti secara umum.
2. Memberikan bahasa pemodelan yang bebas dari berbagai bahasa pemrograman dan proses rekayasa.
3. Menyatukan praktik-praktik terbaik yang terdapat dalam pemodelan.

UML menyediakan beberapa *notasi dan artifact standar* yang bisa digunakan sebagai alat komunikasi bagi para pelaku dalam proses analisis dan desain. Artifact didalam UML didefinisikan sebagai informasi dalam bentuk yang digunakan atau dihasilkan dalam proses pengembangan perangkat. Contohnya adalah *source code* yang dihasilkan oleh proses pemrograman.

a) Notasi-notasi dalam Unified Modelling Languaage (UML)

1. **Notasi Dasar**

Merupakan bentuk pemodelan yang digunakan untuk mempresentasikan elemen-elemen dalam UML, terdiri dari :

1.Notasi Struktual, Terdiri dari

Notasi Aktor

Merupakan segala sesuatu yang berinteraksi dengan system aplikasi komputer. Aktor dapat berupa orang, perangkat, atau objek lain dalam sistem yang sama.

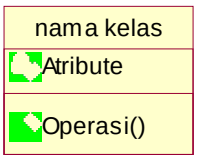


aktor

Gambar 3.3: Simbol Aktor (*Modul Unaki,2003)*

2. Notasi Kelas

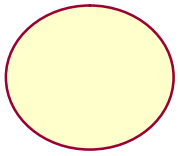
Merupakan pembentuk utama dari sistem berorientasi objek karena kelas menunjukan kumpulan objek yang mempunyai attribute dari operasi yang sama. Kelas digunakan untuk mengabstraksikan elemen-elemen dari sistem yang sedang dibuat.



Gambar 3.4 : Simbol Notasi Kelas(*Modul Unaki,2003)*

3. Notasi Antar Muka *(interface)*

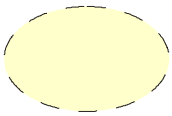
Merupakan kumpulan operasi tanpa implementasi dari suatu kelas. Implementasi operasi dalam antar muka dijabarkan oleh operasi dalam kelas, sehingga keberadaan anatar muka selalu disertai oleh kelas yang mengimplementasikan operasinya.



Gambar 3.5 : Simbol Notasi Antar Muka (*Modul Unaki,2003*)

4. *Notasi Kolaborasi*

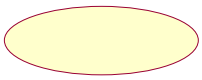
   Mempresentasikan sekelompok peranan dan elemen lainya yang bekerja bersama untuk menyediakan perilaku bersama dari semua elemen yang ada didalamnya



Gambar 3.6 : Simbol Notasi Kolaborasi (*Modul Unaki, 2003)*

5. *Notasi Use Case*

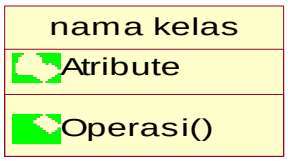
   Gambaran fungsionalitas dari suatu sistem, sehingga pengguna sistem paham dan mengerti mengenai kegunaan sistem yang akan dibangun



Gambar 3.7: Simbol Notasi Use Case (*Modul Unaki, 2003)*

6. *Notasi kelas aktif*

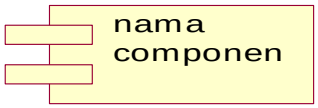
   Mirip dengan sebuah kelas, hanya saja objek - objeknya merepresentasikan elemen - elemen yang memiliki konkuren dengan elemen-elemen yang lain.



Gambar 3.8 : Simbol Notasi Kelas Aktif (*Modul Unaki, 2003)*

7. *Notasi Komponen*

   Merupakan bentuk fisik yang dapat dipindahkan dari sebuah sistem yang membentuk dan menyediakan realisasi dari sekumpulan antar muka.



Gambar 3.9 : Simbol Notasi Komponen (*Modul Unaki,2003)*

8. *Notasi Node*

Merupakan elemen fisik yang harus ada pada saat aplikasi dijalankan dan mempresentasikan sebuah sumber daya komputasi, yang memiliki memori, dan kemampuan pengolahan data.



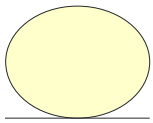
Gambar 3.10 : Simbol Notasi Node (*Modul Unaki, 2003)*

9. *Objek Desain*

Sruktur sebuah sistem berbasis objek dapat didentifikasikan dan diklasifikasikan dalam tiga tipe objek yaitu :

1) Objek entity

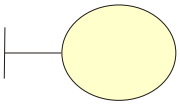
Merupakan sebuah objek yang berisi informasi yang berhubungan dengan kehidupan nyata yang bersifat menetap dan disimpan dalam suatu database yang memungkinkan pemanggilan dan manipulasi di wktu yang akan datang.



Gambar 3.11 : Simbol Objek Entitiy (*Modul Unaki, 2003)*

2) Objek Antar Muka

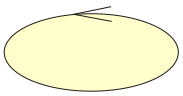
Merupakan sebuah objek yang menyediakan peralatan dimana pengguna dapat berkomunikasi dengan system



Gambar 3.12 : Simbol Objek Antar Muka (*Modul Unaki, 2003)*

3) Objek Control

Merupakan sebuah objek yang berisi logika aplikasi yang bukan merupakan tanggung jawab objek entity. Contoh logika tersebut adalah aturan dan perhitungan bisnis yang melibatkan banyak objek.



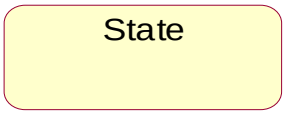
Gambar 3.13 : Simbol Objek Kontrol (*Modul Unaki, 2003)*

2. Notasi Perilaku

Notasi Perilaku terdiri dari :

1. Notasi State

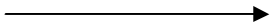
Adalah suatu kondisi selama kehidupan sebuah objek ketika objek memenuhi beberapa kondisi, melakukan beberapa aksi, atau menunggu beberapa aksi, atau menuggu sebuah event.



Gambar 3.14 :Simbol Notasi State (*Modul Unaki, 2003)*

2. Notasi Interaksi.

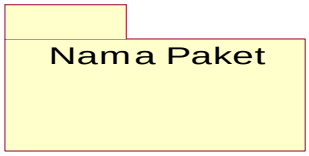
   Mempresentasikan pertukaran pesan antar objek dalam sebuah objek dalam sebuah konteks tertentu guna melakukan sebuah tujuan tertentu..



Gambar 3.15 : Simbol Notasi Interaksi (*Modul Unaki, 2003)*

3. Notasi Pengelompokan

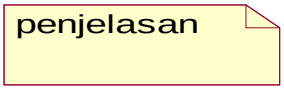
Jika sistem hanya mempunyai sedikit kelas, dapat diatur dengan mudah. Sebagian besar sistem dibuat dari banyak kelas sehingga diperlukan mekanisme untuk mengelompokan, untuk memudahkan dalam penggunaan kembali.



Gambar 3.16 : Simbol Notasi Penggelompokan (*Modul Unaki, 2003,)*

4. Notasi Penjelasan

Penjelasan digunakan untuk memberikan penjelasan dan catatan dalam pemodelan yang dilakukan

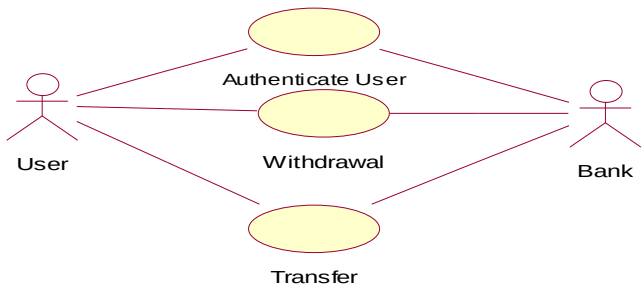


Gambar 3.17 :Simbol Notasi Penjelasan (*Modul Unaki, 2003)*

B. *Diagram Unified Modeling Languange* (DUML)

1) *Use Case* Diagram

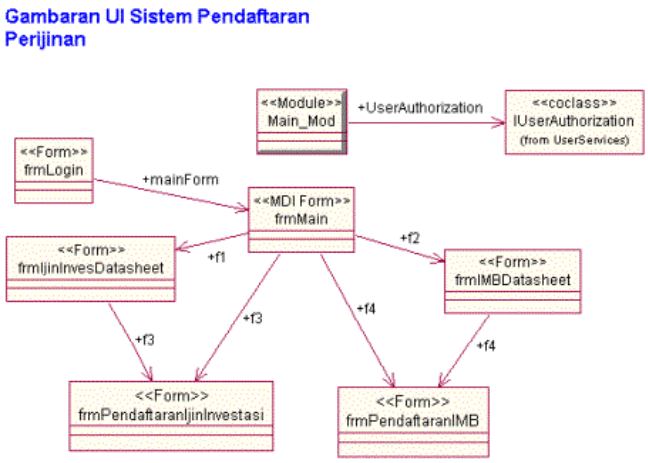
Use Case Diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah "apa" yang diperbuat sistem, dan bukan "bagaimana". *Use Case* dilakukan oleh satu *actor* yang digambarkan dengan simbol orang yang dihubungkan dengan garis yang menunjukan hubungan komunikasi. Setiap *Use Case* harus diberi nama yang menyatakan apa hal yang dicapai dari hasil interaksi dengan *Actor*. Nama *Use Case* boleh terdiri dari beberapa kata dan tidak boleh ada *Use Case* yang memiliki nama yang sama.



Gambar 3.18 : Use Case Diagram (*Modul Unaki, 2003)*

2) *Class* Diagram

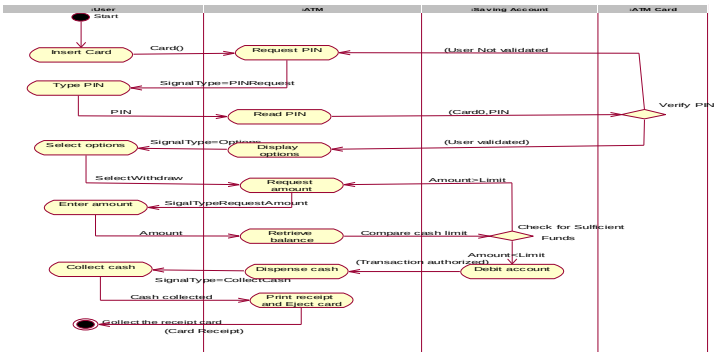
Class Diagram menggambarkan tipe-tipe objek dalam *system* dan berbagai jenis hubungan atau relasi statis yang ada diantara mereka. Diagram ini memberikan gambaran umum dari sistem. Seperti tipe-tipe dari objek dengan menunjukan kelasnya dan *relationship* yang diantara mereka, serat penjelasan detail tiap-tiap kelas ke dalam model suatu *system. Class diagram* bersifat *static* (tidak berubah) yang akan menunjukan apa itu interaksi tapi tidak menjelaskan apa yang terjadi ketika mereka melakukan interaksi.



Gambar 3.19 : Class Diagram (*Modul Unaki, 2003*)

3) Activity Diagram

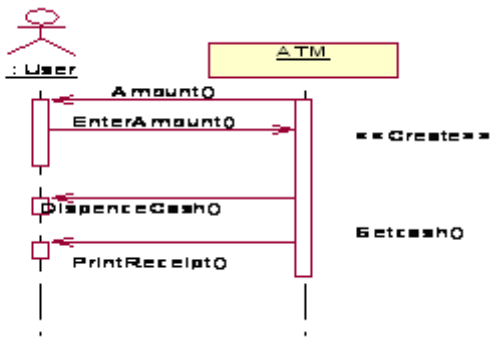
Activity diagram memodelkan alur kerja *(work flow).* Sebuah proses bisnis dan urutan aktifitas dalam suatu proses. Diagram ini sangat mirip dengan sebuah *flowchart* karena kita dapat memodelkan sebuah alur kerja dari aktifitas keaktifitas lainnya atau dari suatu aktifitas kedalam keadaan sesaat (*state).*



Gambar 3.20 : Activity Diagram (*Modul Unaki, 2003)*

4) Sequence Diagram

Sequence Diagram merupakan gambaran interaksi antar objek di dalam dan di sekitar sistem berupa *massage* yang digambarkan terhadap waktu. Diagram ini secara khusus berasosiasi dengan *Use Case. Sequence Diagram* juga digunakan untuk menggambarkan skenario atau rangkain langkah-langkah apa yang seharusnya terjadi sebagai respons dari sebuah *event* untuk menghasilkan sesuatu didalam *Use Case* sebagai *output.* Untuk *message* digambarkan sebagai garis berpanah dari satu objek ke objek lainnya. Pada fase desain berikutnya, *message* dipetakan menjadi operasi/metode dari *class. Activation bar* menunjukan lamanya eksekusi sebuat proses, biasanya diawali dengan sebuah *message*.



Gambar 3.21 : Sequance Diagram (*Modul Unaki, 2003)*

3.3 Pengujian Software

Roger Pressman (2002 : 59). Pengujian Software (perangkat Lunak) adalah elemen kritis dari jaminan kualitas perangkat lunak dan mempresentasikan kajian pokok dari spesifikasi, desain dan pengkodean.

Metode pengujian yang digunakan oleh penulis adalah *black-box tasting*. *Black Box Tasting* digunakan untuk menguji fungsi-fungsi dari perangkat lunak yang dirancang. Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak. Dengan demikian, pengujuan *black box* memungkinkan perekayasa perangkat lunak mendapat serangkain kondisi input yang sepenuhnya menggunakan semua persyaratan fungsional untuk semua program. Pengujian *black box* merupakan pendekatan komlementer yang kemungkinan besar mampu mengungkap kelas kesalahan. Pengujian *black box* berusaha menemukan kesalahan dalm kategori sebagai berikut :

1. Fungsi yang tidak benar atau hilang
2. Kesalahan antar muka
3. Kesalahan dalam struktur data atau akses database eksternal
4. Kesalahan kinerja
5. Inisialisasi kesalahan terminasi