BAB II

LANDASAN TEORI

2.1 Konsep Dasar Sistem

Terdapat 2 kelompok pendekatan didalam mendefinisikan sistem,yaitu yang menekankan pada prosedurnya dan yang menekankan pada komponen atau elemennya. Pengertian sistem menurut **JOG[4]** adalah :

> ***" Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berkumpul bersama-sama untuk melakukan suatu kegiatan untuk menyelesaikan suatu sasaran tertentu".***

Berdasrkan definisi sistem tersebut diatas dapat ditarik kesimpulan bahwa sistem merupakan suatu jaringan kerja yang saling berhubungan dan berkumpul bersama-sama untuk mencapai suatu jaringan tertentu.

2.1.1 Karakteristik Sistem

Dengan adanya pendekatan sistem tersebut maka sifat-sifat atau karakteristik dapat ditentukan sabagai berikut :

1. Adanya komponen Sistem

   Dimana komponen ini terdiri dari subsistem-subsistem yang paling berhubungan atau berinteraksi membentuk suatu kesatuan.

2. Adanya batasan Sistem

   Batasan sistem *(Boundary)* merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lain atau dengan lingkungan luarnya.

3. Lingkungan luar Sistem

   Lingkungan luar sistem *(Environment)* adalah apapun diluar batas dari sistem yang mempengaruhi operasi sitem, baik itu yang menguntungkan ataupun merugikan.

4. Penghubung Sistem

   Merupakan suatu media penghubung antar satu subsistem dengan subsistem yang lain sehingga dapat berintegrasi membentuk satu kesatuan.

5. Masukan Sistem

   Masukan adalah merupakan masukan sistem yang akan diolah menjadi informasi, misalnya program dan data.

6. Keluaran Sistem

   Keluaran adalah hasil dari proses yang terjadi dan diklarifikasikan menjadi keluaran yang berguna, yaitu informasi dan sisa pembuangan, misalnya panas yang dihasilkan Komputer.

7. Pengolahan Sistem

   Suatu sistem dapat mempunyai suatu bagian pengolah yang akan merubah masukan menjadi keluaran.

8. Sasaran Sistem

   Sistem pasti mempunyai tujuan atau sasaran yang akan menentukan masukan data keluaran.

2.1.2 Klasifikasi Sistem

Sistem dapat diklasifikasikan dari beberapa sudut pandang, diantaranya adalah sebagai berikut :

1. Sistem diklasifikasikan sebagai sistem abstrak dan sistem fisik

   a) Sistem Abstrak

      Sistem Abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak ma,pak secara fisik

b) Sistem Fisik

Sistem Fisik adalah suatu sistem dari unsure-unsur yang bekerja bersama untuk mencapai suatu tujuan tertentu, dimana sistem-sistem tersebut lebih sekedar kerangka-kerangka konsep belaka.

2. Sistem diklasifikasikan sebagai sistem tertutup dan sistem terbuka

a) Sistem Tertutup

Sistem Tertutup adalah sistem yang tidak berhubungan dan tidak terpengaruh dengan lingkungan luar.

b) Sistem Terbuka

Sistem Terbuka adalah sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya.

3. Sistem diklasifikasikan sebagai sistem Alamiah dan sistem Buatan manusia

a) Sistem Alamiah

Sistem Alamiah adalah sistem yang terjadi melalui proses alam dan tidak dibuat oleh manusia.

b) Sistem Buatan Manusia

Sistem Buatan Manusia adalah sistem yang dirancang manusia dan melibatkan interaksi antara manusia dengan mesin.

4. Sistem diklasifikasikan sebagai sistem tertentu dan sistem tak tentu

a) Sistem Tertentu

Sistem Tertentu adalah beroperasi dengan tingkah laku yang sudah dapat diprediksi interaksi diantara bagian-bagiannya dapat dideteksi dengan pasti sehingga keluarannya dapat diramalkan.

b) Sistem Tak Tentu

Sistem Tak Tentu adalah sistem yang kondisi masa depannya tidak dapat diprediksikan karena mengandung unsur probabilitas.

2.2 Konsep Dasar Informasi

Definisi informasi sendiri adalah data yang diolah menjdi bentuk yang lebih berguna bagi yang menerimanya. Jadi informasi adalah faktor terpenting dalam sistem untuk pengambilan keputusan

Menurut **GOR[1]** definisi umum untuk informasi dalam pemakai sistem informasi adalah suatu data yang telah diolah sedemikian rupa, menjadi sebuah bentuk yang lebih berguna dan lebih berarti bagi si pemakai dalam mengambil suatu keputusan sekarang atau yang akan datang.

Atau bisa dikatakan juga masih menurut **GOR[1],** informasi adalah suatu bahasa informasi yang dirumuskan sehingga kumpulan simbol-simbol teratur yang menyatakan jumlah dan tindakan, jadi sistem pengolahan informasi yaitu pengolahan data dari bentuk tak berguna menjadi lebih berguna bagi si pemakai.

Informasi yang baik harus memiliki kualitas sebagai berikut :

1. Dapat Dipercaya, yaitu bebas dari kesalahan atau bisa secara akurat menggambarkan kejadian atau aktivitas organisasi.
2. Relevan, yaitu menambah pengetahuan atau nilai bagi para pembuat keputusan, dengan cara mengurangi ketidakpastian, menaikkan kemampuan untuk memprediksi, atau menegaskan ekspektasi semula.
3. Lengkap, yaitu tidak menghilangkan data penting yang dibutuhkan oleh para pemakai.

4. Tepat waktu, yaitu disajikan pada saat yang tepat untuk mempengaruhi proses pembuatan keputusan.
5. Mudah dipahami, yaitu disajikan dalam format yang mudah dimengerti.

   Dapat diuji kebenarannya, yaitu memungkinkan dua orang yang kompeten untuk menghasilkan informasi yang sama secara independent

2.3 Konsep Dasar Sistem Informasi

Sistem informasi adalah suatu keseluruhan yang terdiri atas sejumlah variabel (subsistem) yang saling berinteraksi suatu sistem pada dasarnya adalah suatu susunan yang teratur dari kegiatan dalam melaksanakan dan memudahkan kegiatan utama dari suatu organisasi.

. Sistem informasi merupakan suatu proses sistem manusia dan mesin yang terintegrasi secara harmonis untuk menyediakan informasi yang dapat digunakan oleh user untuk mendukung fungsi operasi, manajemen dan pengambilan keputusan di dalam organisasi baik secara manual maupun dengan kemampuan teknologi informatika (komputer).

> **"*Sistem Informasi adalah suatu sistem dari suatu sistem organisasi yang mempertemukan pengolahan transaksi harian, mendukung operasi, bersifat manajerial, dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dan laporan yang diperlukan*" GOR[1]**

Kegiatan-kegiatan di dalam suatu informasi meliputi :

1. Input, kegiatan menyediakan data untuk diproses.
2. Proses, menggambarkan bagaimana suatu data diproses untuk menghasilkan informasi yang bernilai lebih.
3. Output, kegiatan yang menghasilkan keluaran berupa informasi hasil dari proses.

4. Penyimpanan, kegiatan untuk memelihara dan menyimpan data.
5. Kontrol (pengendali), suatu aktifitas untuk menjamin bahwa sistem informasi berjalan sesuai dengan jalan yang diharapkan.

Sistem informasi mempunyai beberapa komponen antara lain:

1. Perangkat keras *(hardware)*, merupakan komponen fisik berupa peralatan input dan peralatan proses.
2. Perangkat lunak *(software)*, merupakan intruksi-intruksi yang membuat komputer melakukan pekerjaan tertentu.
3. Sumber daya manusia *(brainware)*, sebagai pengoperasi sistem.
4. Data, yaitu fakta-fakta perkiraan dan pendapat-pendapat yang belum memiliki arti guna.
5. Prosedur, yaitu instruksi-instruksi yang digunakan dalam mengoperasikan sistem.

2.4 Metode Pendekatan Sistem

Metode pendekatan sistem yang digunakan adalah *Object Oriented* yang menekankan pada solusi logic berbasis objek. Teknologi *object oriented* merupakan paradigma baru dalam rekayasa perangkat lunak yang didasarkan pada objek dan kelas.

2.4.1 Metode Berorientasi Objek

Object-oriented mencakup bidang aplikasi yang sangat luas.Para pengguna sistem komputer dan sistem lain yang didasarkan atas teknologi komputer merasakan efek *object-oriented* dalam bentuk meningkatnya aplikasi software yang mudah digunakan dan servis yang lebih fleksibel, yang muncul dalam

berbagai bidang industri, seperti dalam perbankan, telekomunikasi, dan sebagainya.Sedangkan bagi *software engineer, object-oriented* berpegaruh dalam bahasa pemrograman, metodologi rekayasa, manajemen proyek, hardware, dan sebagainya.

2.4.2 Object Oriented Analysis and Design (OOAD)

Object Oriented mencakup bidang aplikasi yang sangat luas. Para pengguna sistem komputer dan sistem yang lain didasarkan atas teknologi komputer merasakan efek *object-oriented* dalam bentuk meningkatnya aplikasi software yang mudah digunakan dan servis yang lebih fleksibel, yang muncul dalam berbagai bidang industri, sedangkan bagi *software engineer, object-oriented* berpengaruh dalam bahasa pemograman, metodologi rekayasa, manajemen proyek, *hardware* dan sebagainya.

Dalam konteks OOAD, kita dapat mendefinisikannya sebagai berikut :

a) *Object-Oriented Analysis*

object oriented analysis(OOA) adalah metode analisis yang memeriksa *requirements* (syarat/keperluan yang harus dipenuhi suatu sistem) dari sudut pandang kelas-kelas dan objek-objek yang ditemui dalam ruang lingkup permasalahan.

Dalam *object oriented analysis(OOA),* terdapat lima lapisan (tidak bersifat sequential), yaitu :

1) Menentukan lapisan Subjek
2) Menentukan lapisan kelas dan Objek
3) Menentukan lapisan Struktur
4) Menentukan lapisan Attribute

5) Menentukan lapisan Service

b) *Object Oriented Design (OOD)*

object oriented design(OOD) adalah metode untuk mengarahkan arsitektur software yang didasarkan pada manipulasi objek-objek sistem atau subsistem.

Dalam *object oriented design(OOD),* terdapat lima lapisan (tidak bersifat sequential), yaitu :

1) Menentukan lapisan Subjek
2) Menentukan lapisan kelas dan Objek
3) Menentukan lapisan Struktur
4) Menentukan lapisan Attribute
5) Menentukan lapisan Service

2.4.3 Konsep Dasar dalam OOAD

Konsep dasar dalam OOAD terdiri dari Objek, Kelas, Encapsulation, Association dan Aggregation.

1. Objek (*object*)

Adalah "benda", secara fisik atau konseptual, yang dapat kita temui disekeliling kita. Hardware, Software, Dokumen, Manusia, dan bahkan konsep semuanya adalah contoh objek.

a) State

Adalah kondisi objek tersebut atau himpunan dari keadaan yang menggambarkan objek tersebut. *State* dinyatakan dengan nilai dari atribut (*attribute*) objeknya.

b) Atribut

Adalah nilai internal suatu objek yang mencerminkan antara lain karakteristik objek, kondisi sesaat koneksi dengan objek lain, dan identitas. Perubahan *state* dicerminkan oleh perilaku (*behavior*) objek tersebut.

c) Behavior

Suatu objek mendefinisikan bagaimana sebuah objek bertindak (beraksi) dan memberi reaksi. *Behavior* ditentukan oleh himpunan semua tau beberapa operasi yang dapat dilakukan dalam objek itu sendiri. *Behavior* dari sebuah objek dicerminkan oleh *interface, service,* dan *method* dari objek tersebut.

d) Interface

Adalah pintu untuk mengakses service objek.

e) Service

Adalah fungsi yang bisa diemban objek

f) Method

Adalah mekanisme internal objek yang mencerminkan perilaku objek tersebut. Sebagai contoh, jika printer sebuah merupakan, perilaku atau servicenya mencetak apapun yang dia terima.

2. Kelas *(class)*

Kelas (*class*) adalah definisi umum (pola,*template* atau cetak biru) untuk himpunan objek sejenis. Kelas menentapkan spesifikasi perilaku (*behaviors*) dan

atribut objek-objek tersebut. *Class* adalah keniskalan (abstraksi) dari entitas dalam dunia nyata. Objek adalah "contoh" (*instance*) dari sebuah kelas.

3. *Encapsulation*

Adalah proses menyembunyikan detil implementasi sebuah objek. Satu-satunya jalan untuk mengakses data objek tersebut adalah *interface*. *Interface* melindungi *internal state* sebuah objek dari sebuah objek dari pihak luar.

Dalam *object-oriented programming,* kode dan data disatukan dalam sebuah "benda" yang tersembunyi isinya, yaitu objek. Pengguna objek tidak perlu mengetahui isi dalam kotak tersebut. Untuk dapat berkomunikasi dengan objek, diperlukan pesan (*message*).

Message dapat didefinisikan sebagai permintaan untuk objek penerima (*receiver object*) untuk membawa metode yang ditunjukan dan mengembalikan *result* dari aksi tersebut kepada objek pengirim (*sender objek*).

4. *Association dan Aggregation*

Association (asosiasi) adalah hubungan antar objek yang saling membutuhkan. *Aggregation* (agregasi) adalah bentuk khusus dari asosiasi yang menggambarkan seluruh bagian suatu objek merupakan bagian dari *objek* lainnya.

2.5 UML (*Unified Modeling Langguage)*

UML (Unified Modelling Langguage) merupakan bahasa pemodelan yang paling sukses dari tiga metode OO (*Object-oriented)* yang telah ada sebelumnya, yaitu Booch, OMT, and OOSE. UML merupakan kesatuan dari ketiga metode pemodelan tersebut dan ditambah kemampuan lebih karena mengandung metode

tambahan untuk mengatasi masalah pemodelan yang tidak dapat ditangani ketiga metode tersebut.

2.5.1 Pengenalan UML

UML (Unified Modeling Language) merupakan pengganti dari metode analisis berorientasi object dan design berorientasi object (OOA&D) yang dimunculkan sekitar akhir tahun 80-an dan awal tahun 90-an.

UML merupakan gabungan dari metode Booch, Rumbaugh (OMT) dan Jacobson.Tetapi UML ini akan mencakup lebih luas daripada OOA&D. Pada pertengahan pengembangan UML dilakukan standarisasi proses dengan OMG (Object Management Group) dengan harapan UML akan menjadi bahasa standar pemodelan pada masa yang akan datang.

Bahasa pemodelan merupakan bagian terpenting dari metode. Ini merupakan bagian kunci tertentu untuk komunikasi. Jika anda ingin berdiskusi tentang desain dengan seseorang, maka Anda hanya membutuhkan bahasa pemodelan bukan proses yang digunakan untuk mendapatkan desain.

UML merupakan bahasa standar untuk penulisan *blueprint* software yang digunakan untuk visualisasi, spesifikasi, pembentukan dan pendokumentasian alat-alat dari sistem perangkat lunak.

2.5.2 Pengertian UML

UML adalah bahasa untuk menspesifikasi, memvisualisasi, membangun dan mendokumentasikan *artifact* (bagian dari informasi yang digunakan atau

dihasilkan oleh proses pembuatan perangkat lunak, *artifact* tersebut dapat berupa model deskripsi atau perangkat lunak) dari sistem perangkat lunak.

Selain itu UML adalah bahasa pemodelan yang menggunakan konsep orientasi *object.* UML menyediakan notasi-notasi yang membantu memodelkan sistem dari berbagai perspektif. UML tidak hanya digunakan dalam pemodelan perangkat lunak, namun hampir dalam semua bidang yang membutuhkan pemodelan.

2.5.3 Bagian-bagian UML

Bagian-bagian utama dari UML adalah *view,diagram, model element, dan general mechanism.*

1. View

View digunakan untuk melihat sistem yang dimodelkan dari beberapa aspek yang berbeda. View bukan melihat grafik, tapi merupakan suatu abstraksi yang berisi sejumlah diagram.

Beberapa jenis view dalam UML antara lain : *use case view,logical view, component view, concurrency view* dan *deployment view.*

a. Use Case View

Mendeskripsikan fungsionalitas sistem yang seharusnya dilakukan sesuai yang diinginkan *external actors. Actor* yang berinteraksi dengan sistemdapat berupa user atau sistem lainnya.

View ini digambarkan dalam use case diagram dan kadang-kadang dengan *activity diagrams.*

b.Logical View

Mendeskripsikan bagaimana fungsionalitas dari sistem, struktur statis *(class, object dan relationship)* dan kolaborasi dinamis yang terjadi

c.Component View

Mendeskripsikan implementasi dan ketergantungan modul. Komponen yang merupakan tipe lainnya dari *code module* diperlihatkan dengan struktur dan ketergantungannya juga alokasi sumber daya komponen dan informasi administrative lainnya.

d.Councurrency view

Membagi sistem ke dalam proses dan prosesor, view ini digambarkan dalam diagram dinamis dan diagram implementasi serta digunakan untuk pengembang (developer), pengintegrasi (integrator), dan penguji (tester).

e. Deployment view

Mendeskripsikan fisik dari sistem seperti komputer dan perangkat (nodes) dan bagaimana hubungan dengan lainnya

2. Diagram

Diagram merupakan bagian dari suatu *view* tertentu dan ketika digambarkan biasanya dialokasikan untuk *view* tertentu. Ada beberapa jenis diagram antara lain :

a) ***Use Case Diagram***

Use Case adalah deskripsi fungsi yang disediakan oleh sistem dalam bentuk teks sebagai dokumentasi dari ***use case symbol*** namun dapat juga dilakukan dalam ***activity diagram***.

b) ***Class Diagram***

Class Diagram menggambarkan struktur statis ***class*** di dalam sistem. ***Class*** merepresentasikan sesuatu yang ditangani oleh sistem. ***Class*** dapat berhubungan dengan yang lain melalui berbagai cara : *associated* (terhubung satu sama lain), *dependent* (satu class tergantung/ menggunakan class yang lain), *specialed* (satu *class* merupakan spesialisasi dari *class* lainnya), atau *Package* (grup bersama sebagai satu unit).

c) ***State Diagram***

State Diagram menggambarkan *state* (kondisi) yang dimiliki oleh suatu *object* dari suatu *class* dan keadaan yang menyebabkan *state* berubah. Kejadian dapat berupa *object* lain yang mengirim pesan.

d) ***Sequence Diagram***

Sequence Diagram menggambarkan kolaborasi dinamis antara sejumlah *object.* Kegunaannya untuk menunjukan rangkaian pesan yang dikirim antara *object* juga interaksi antara *object,* sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem.

e) ***Colaboration Diagram***

Colaboration Diagram menggambarkan kolaborasi dinamis seperti *sequence digram.* Dalam menunjukan pertukaran pesan, Colaboration Diagram

f) ***Activity diagram***

Menggambarkan rangkaian aliran dari aktivitas, digunakan untuk mendeskripsikan aktivitas yang dibentuk dalam suatu operasi sehingga

dapat juga digunakan untuk aktivitas lainnya seperti *use case* atau interaksi.

g) ***Component diagram***

Menggambarkan alokasi semua kelas dan objek kedalam komponen-komponen dalam desain fisik sistem software. Diagram ini memperlihatkan pengaturan dan kebergantungan antara komponen-komponen software seperti *source code*, *binary code*, dan komponen tereksekusi (*executable component).*

h) ***Deployment diagram***

Menggambarkan arsitektur fisik dari perangkat keras dan perangkat lunak sistem, menunjukkan hubungan komputer dengan perangkat *(nodes)* satu sama lain dan jenis hubungannya. Diagram ini memperlihatkan pemetaan software kepada hardware.

2.5.4 Indikator Multiplicity

Walaupun *multiplicity* ditentukan untuk *class*, multiplicity menentukan banyaknya object yang terlibat dalam relasi. Multiplicity menentukan banyaknya object yang terhubung satu dengan yang lainnya.Indikator Multiplicity terdapat pada masing-masing akhir garis relasi, baik pada asosiasi maupun agregasi.Beberapa contoh multiplicity adalah :

e. Tepat Satu

0..* Nol atau lebih

1..* Satu atau lebih

0..1 Nol atau Satu

5...8 range 5 s.d.8

4..6,9 range 4 s.d.6 dan 9

2.5.5 Business Modelling

Dalam dunia bisnis dan industri, terdapat banyak sistem manual dan otomatis yang muncul secara regular.Setiap sistem tersebut memiliki satu atau banyak *workflow.Business Modelling* menggambarkan semua *workflow* yang terjadi dalam suatu organisasi.

Secara formal, kita definisikan business modeling sebagai segala teknik pemodelan yang digunakan untuk menggambarkan model sebuah bisnis.*Business modeling* dapat digunakan untuk meninjau, meningkatkan, dan membuat sebuah bisnis.

Tujuan Business Modelling adalah sebagai berikut :

1) Memahami struktur dan dinamika organisasi
2) Memahami masalah-masalah dalam mencapai target organisasi dan menemukan potensi untuk kemajuan organisasi.
3) Yakin bahwa para *customer, end user, dan developer* mempunyai sebuah pemahaman yang benar mengenai organisasi.
4) Dapat menurunkan/mendapatkan requirement software aplikasi yang akan kita buat yang diperlukan untuk mendukung pencapaian target organisasi.

Untuk mencapai tujuan tersebut, harus membuat dua jenis model bisnis, yaitu model use-case bisnis (*business use-case model*) dan model objek bisnis

(*business object model*) yang berisi proses-proses, peranan-peranan, dan tanggung jawab organisasi untuk mencapai targetnya.

2.5.6 Proses Bisnis

Dalam suatu organisasi mungkin terdapat banyak proses bisnis.Proses Bisnis (*business process*) adalah sekumpulan aktivitas yang dirancang untuk menghasilkan keluaran (*output*) tertentu bagi customer tertentu.Sebuah proses bisnis menekankan pada bagaimana sebuah pekerjaan dikerjakan dalam sebuah organisasi, dengan berfokus pada produk yang dihasilkan proses tersebut.proses disini adalah urutan aktivitas tertentu terhadap waktu dan tempat, dengan sebuah titik awal, sebuah titik akhir, dan sebagai *input* dan *output* yang didefinisikan dengan baik.

2.5.7 Stereotype dalam Model Bisnis

Rational Rose memiliki beberapa jenis elemen yang digunakan dalam *business modeling*, yaitu :

1) ***Business Actor***

Business Actor (aktor bisnis) menggambarkan peran yang dimainkan oleh seseorang atau sesuatu yang dengannya bisnis berinteraksi.Sebuah business actor menggambarkan seorang customer atau partner bisnis, tetapi sebuah sistem informasi yang berhubungan dengan bisnis kita dapat berperan sebagai sebuah business actor juga.

2) ***Business Use-Case***

Business Use-Case merupakan urutan tindakan yang dimainkan suatu bisnis yang menghasilkan sebuah nilai yang dapat dilihat dan ditujukan untuk suatu *business actor* tertentu.Setiap *business use-case* mewakili satu proses bisnis.

3) ***Business Use-Case Realization***

Business Use-Case Realization memberikan sudut pandang dari dalam (*internal view*) terhadap *business use-case*, yakni menentukan bagaimana pekerjaan (dalam *business use-case*) tersebut diatur dan dimainkan untuk mencapai hasil yang diinginkan. Sebuah *Business Use-Case Realization* menentukan *business worker* dan *business entity* yang terlibat ketika menjalankan sebuah *business use-case*.

4) ***Business Worker***

Business Worker (pekerja bisnis) adalah sebuah kelas yang mewakili seorang manusia yang bertindak dalam bisnis.Sebuah *Business Worker* berinteraksi dengan para worker lainnya dan memainkan entitas bisnis saat berpartisipasi dalam realisasi suatu *business use-case*.

5) ***Business Entity***

Business Entity (entitas bisnis) menggambarkan "benda" yang ditangani atau digunakan oleh business worker selama melaksanakan suatu *business use-case*

2.5.8 Requirement

Requirement adalah kondisi atau kemampuan yang harus dipenuhi oleh software aplikasi yang akan dibuat.*Requirement* untuk software aplikasi dibagi ke dalam dua kelompok, yaitu *Function requirement dan Non-function requirement*. *Function requirement* menentukan tindakan yang harus dapat dimainkan sebuah software aplikasi.Requirement ini sering dimodelkan dalam bentuk use-*case-use-case* dalam *use-case model. Function requirement* juga menentukan masukan dan keluaran dari sebuah software aplikasi. *Function requirement* mencakup fitur, kemampuan (capabilities), dan factor *sequrity* software aplikasi.*Non-Function requirement* menggambarkan atribut dari software aplikasi dan lingkungannya. *Requirement* ini biasanya masuk di dalam use-case dan tercakup dalam sifat (*property*) *use-case* tersebut.

2.5.9 Model Analisis

Model analisis (Analysis Model) menggambarkan realisasi dari *use-case-use-case* dalam *use-case model*, tujuan akhir sebuah model analisis adalah untuk membuat pemetaan awal mengenai perilaku yang disyaratkan dalam sistem aplikasi ke dalam elemen-elemen pemodelan.

Elemen model yang terdapat dalam model analisis disebut kelas analisis (*analysis class*).Kelas analisis adalah kelas ber-*stereotype* “boundary”, “control”, “entity” yang menggambarkan sebuah konsep awal mengenai “benda” dalam sistem aplikasi yang memiliki tanggung jawab dan perilaku.*Boundary* adalah kelas yang memodelkan interaksi antara satu atau lebih *actor* dengan sistem.*Control* digunakan untuk memodelkan “perilaku mengatur”, khusus untuk

satu atau beberapa *use-case* saja.*Entit*y(entitas) memodelkan informasi yang harus disimpan oleh sistem.

2.6 Basis Data

Basis data merupakan kumpulan dari data-data yang saling terkait dan saling berhubungan satu dengan lainnya. Basis data adalah kumpulan kumpulan *file* yang saling berkaitan.

Salah satu pemakai sistem informasi manajemen *database* yaitu administratur basis data (*Database Administrator*/*DBA*) fungsi ini bertanggung jawab untuk melakukan koordinasi, pengawasan, dan pengelolaan *database.*

Basis data memiliki tujuan yang diantaranya adalah sebagai berikut :

1. Kemampuan dalam menangani data dalam jumlah besar.
2. Efisiensi pengolahan data yang meliputi *speed, space* dan *accuracy.*
3. Kemampuan pemakaian *resource* basis data secara bersama-sama *(Sharebility)*.
4. Meniadakan duplikasi data yang tidak perlu dan menghindari inkonsistensi data.

2.7 Arsitektur Aplikasi

Arsitektur aplikasi dapat pula dikatakan sebagai struktur desain dari suatu sistem komputer dengan semua kelengkapannya yamg telah siap untuk digunakan oleh user.

2.7.1 Pengertian Jaringan Komputer

Jaringan komputer adalah sekumpulan komputer pribadi saling berhubungan antara satu dengan yang lainnya menggunakan *protokol* komunikasi sehingga dapat saling berbagi informasi, program penggunaan bersama perangkat keras seperti printer, harddisk. Selain itu jaringan komputer dapat diartikan sebagai kumpulan sejumlah terminal komunikasi yang berada di berbagai lokasi yang terdiri dari lebih dari satu komputer yang saling berhubungan. Dalam sebuah jaringan komputer dapat diartikan sebagai kumpulan sejumlah terminal komunikasi yang berada di berbagai lokasi yang terdiri dari satu komputer yang saling berhubungan. Dalam sebuah jaringan komputer biasanya terhubung sejumlah komputer ke sebuah/ beberapa *serve*r.

Server adalah komputer yang difungsikan sebagai "pelayanan" pengiriman data dan penerimaan data diantara komputer-komputer yang tersambung.

Tujuan dibangunnya suatu jaringan komputer adalah membawa informasi secara tepat dan tanpa adanya kesalahan dari sisi pengirim (*transmisi*) menuju ke sisi penerima (*receiver*) melalui media komunikasi.

TCP/IP (Trasmission Control Protocol/Internet Protocol) merupakan protokol standard internet yang digunakan untuk melakukan koneksi ke internet protokol. TCP/IP memiliki beberapa subyek protocol yang berbeda yang beroperasi pada lapisan yang berbeda dan mempunyai tugas masing-masing. Berkat adanya protokol ini setiap komputer dapat berhubungan secara fleksibel dengan host-host yang terkoneksi.

2.7.2Jenis-Jenis Jaringan Komputer

Jenis-jenis jarinangan computer terdapat 2 jenis, diantaranya adalah :

1. **Model peer to peer**

   Menurut model ini, setiap host dapat menawarkan layanan ke peer lain dan juga mengambil layanan dari peer lain. Model ini cocok untuk jaringan kecil.

2. **Model Client/Server**

   Model ini memisahkan secara jelas, mana yang dapat memberikan layanan jaringan (server) dan mana yang hanya memberikan layanan (client). Beberapa komputer diset-up sebagai server yang memberikan sumber daya (resource) dari jaringan : printer, modem, dan saluran lainnya kepada komputer lain yang dikoneksi kejaringan yang berfungsi sebagai client. Dalam perancangan sistem informasi Pengolahan Data Siswa ini jenis jaringan yang digunakan adalah Model Client Server

2.7.3 Topologi Jaringan Komputer

Yang dimaksud dengan topologi jaringan adalah susunan fisik bagaimana node-node saling dihubungkan. Ada tiga yang digunakan, yaitu :

1. **Topologi Bus**

Ethernet menggunakan satu kawat (kabel) yang berfungsi sebagai medium untuk mentransmisikan data. Node yang merupakan bagian dari jaringan dihubungkan seluruhnya ke kabel tersebut.

Node-node yang terhubung mengirim dan menerima data jaringan melalui kabel sebagai pembawa sinyal dan melihat apakah data tersebut ditujukan buat dirinya.

2. Topologi Ring

IBM Token Ring. Salah satu node dihubungkan dengan node yang ada didepan dan dibelakangnya sehingga membentuk ring. Setiap node mendapat giliran menggunakan jaringan dengan mengirimkan "token". Node yang mendapatkan giliran da[pat mengirimkan data dan node lain menerima data, serta melihat apakah data yang ditujukan kepadanya. Bila ditujukan buat dirinya,datapun disimpan, bila tidak data diteruskan ke node didepannya.

3. Topologi Star atau Hub

Susunan atau skema dari topologi ini mirip sebuah bintang. Topologi ini memiliki satu hub pusat dari mana data ditransmisikan ke seluruh node dalam jaringan. Skema ini mempunyai kelebihan disbanding dua skema sebelumnya, yaitu bila terjadi kerusakan pada kabel tidak membawa dampak bagi seluruh node, tapi hanya node yang bersangkutan saja sehingga aktivitas jaringan tidak terganggu secara total. Ini berbeda dengan skema bus atau ring, dimana bila terjadi kerusakan pada kabel berakibat pada seluruh jaringan.

2.7.4 Manfaat Jaringan Komputer

Manfaat yang didapat dari membangun jaringan komputer adalah sebagai berikut :

1. Sharing Resources

   Sharing resources bertujuan agar seluruh program, peralatan/ periperal lainnya dapat dimanfaatkan oleh setiap orang yang ada pada jaringan komputer tanpa terpengaruh pada jaringan komputer tanpa terpengaruh oleh lokasi maupun pengaruh dari pemakai. Dengan kata lain, seorang pemakai

yang letaknya jauh sekalipun dapat memanfaatkan data maupun informasi yang lainnya tanpa mengalami kesulitan.

2. Media komunikasi

   Jaringan komputer memungkinkan terjadinya komunikasi antara pengguna, baik untuk *teleconference* maupun untuk mengirim pesan/informasi yang penting lainnya.

3. Integrasi data

   Pembangunan jaringan komputer dapat mencegah ketergantungan pada komputer pusat. Setiap proses data tidak harus dilakukan 1 komputer saja, melainkan dapat didistribusikan ke tempat lainnya oleh sebab inilah maka dapat terbentuk data yang terintegrasi sehingga dengan demikian memudahkan pemakai untuk memperoleh dan mengolah informasi setiap saat.

4. Pengembangan dan Pemeliharaan

   Dengan adanya jaringan komputer ini, maka perkembangan peralatan dapat dilakukan dengan mudah dan menghemat biaya. Jaringan komputer bisa memudahkan pemakai dalam merawat *harddisk* dan peralatan lainnya. Misalnya untuk memberikan perlindungan serangan *virus* maka pemakai cukup memusatkan perhatian pada *harddisk* yang ada di komputer pusat.

5. Keamanan Data

   Sistem jaringan komputer memberikan perlindungan terhadap data jaminan keamanan data tersebut di berikan melalui pengaturan hak akses para pemakai dan password, serta teknik perlindungan terhadap harddisk sehingga data mendapatkan perlindungan yang efektif.

6. Sumber Daya Lebih Efisien dan Informasi Terkini

   Dengan adanya pemakaian sumber daya secara bersama-sama maka pemakai bisa mendapatkan hasil dengan maksimal dan kualitas yang tinggi. Selain itu data atau informasi yang di akses selalu terbaru, karena setiap ada perubahan yang terjadi dapat secara langsung di ketahui oleh setiap pemakai.

2.8 Pengertian Client / server

Sistem *client-server* mempunyai dua komponen utama yaitu komputer *client* dan komputer *server. Server* merupakan komputer induk yang melakukan pemrosesan terbanyak untuk memenuhi permintaan-permintaan dari komputer *client* dan bertindak sebagai *server database* yang menyimpan data. *Client* yaitu komputer atau *workstation* yang melakukan pengiriman permintaan-permintaan data pada *server* kemudian menampilkan data tersebut pada *interface* aplikasi yang dimilikinya. Selain itu *client* juga mempunyai kemampuan untuk mengubah atau menghapus data itu.

Sistem *client-server* merupakan suatu sistem *client* komputer yang melibatkan proses-proses *client* yang meminta suatu pelayanan data kepada computer *server* yang menyediakan layanan data tersebut. Sehingga *client* maupun *server* sama-sama melakukan pekerjaan. Dengan adanya kombinasi *client* (*Front-end)* dan *server (back-end*) ini maka kumpulan dari modul-modul program tidak dieksekusi dalam memori yang sama namun terbagi dalam komputer *client-server.*

Hal ini menjadikan konfigurasi bagi komputer *client* dan komputer *server* bisa berbeda seperti kapasitas memori, kecepatan *prosesor* atau alat masukan dan

keluaran yang disesuaikan dengan fungsi kerja dari elemen-elemen tersebut. Bagi *server* yang menjalankan tugas pengelolaan suatu *database* digunakan suatu konfigurasi yang khusus menangani tugasnya tersebut dengan sistem operasi yang dikhususkan bagi *server* seperti windows NT server, windows 2000 server, sedangkan komputer *client* menggunakan konfigurasi yang umum bagi sebuah komputer *desktop* yang terhubung ke jaringan dengan sistem operasi seperti windows 98, windows ME, windows Xp dan lain-lain.

2.9 Perangkat Lunak Pendukung

Perangkat Lunak ini terdiri dari Sistem Operasi (SO) dan paket aplikasi database.Adapun spesifikasi Perangkat Lunak tersebut adalah :

2.9.1 Sekilas Tentang Delphi

Borland Delphi merupakan suatu bahasa pemrograman yang memberikan berbagai fasilitas aplikasi visual.Keunggulan bahasa pemrograman ini terletak pada produktivitas, kualitas, pengembangan perangkat lunak, kecepatan kompilasi, pola desain yang menarik serta diperkuat dengan pemrogramannya yang terstruktur.Keunggulan lain dari Delphi adalah dapat digunakan untuk merancang program aplikasi yang memiliki tampilan seperti program aplikasi lain yang berbasis windows.

Borland Delphi 7.0 merupakan pengembangan dari Borland Delphi 6.0 lebih banyak keunggulan dan fasilitas baru yang diberikan oleh Borland Delphi 7.0 dibandingkan dengan versi-versi sebelumnya.

2.9.2 Microsoft Access

Microsoft Access atau lebih dikenal dengan Access merupakan salah satu software database yang berjalan di bawah sistem Windows, karena disamping Microsoft Access, masih banyak software-software database lainnya yang dapat kita temukan. Dengan Microsoft Access, kita dapat merancang, membuat dan mengelola database dengan cara mudah dan cepat.

Beberapa kelebihan dari Microsoft Access diantaranya adalah :

1. Fasilitas subdtasheets baru memungkinkan kita melihat record yang berkaitan dengan sebuah dataset.
2. Dapat berfungsi sebagai front end untuk data Microsoft Sql server
3. Dilengkapi dengan Tool Web yang lebih canggih.
4. Memungkinkan kita untuk mempublikasikan data ke web, namun kita tidak dapat melihat hasil halaman Web di Access.
5. Dapat mengoperasikan secara langsung dari database Access untuk membuka dabn browse halaman web yang menunjukan data accesss.
6. Adanya fasilitas bantuan page wizard yang baru menciptakan halaman web sederhana di mana kita dapat menyunting data table.

Di dalam mengelola database, Microsoft Access memiliki sarana atau obyek yang dapat mempermudah dalam kerja anda. Obyek-obyek ini satu sama lain saling berhubungan atau saling terkait. Adapun obyek-obyek tersebut adalah :

Tabel 2.1 Obyek-obyek pada MS Access

Object	Kegunaan
Tables	Merupakan tempat atau sarana untuk penyimpanan data
Queries	Digunakan untuk menyaring data dengan berbagai criteria dan urutan yang anda inginkan.
Forms	Digunakan untuk memasukan dan mengedit data atau informasi yang ada dalam suatu database dengan menggunakan bentuk

	tampilan formulir
Reports	Digunakan untuk menampilkan atau mencetak data atau informasi dalam bentuk laporan
Pages	Digunakan untuk membuat halaman web berupa data access page
Macros	Digunakan untuk mengotomatisasikan database anda dengan perintah atau cara yang lebih sederhana tanpa membutuhkan pemograman.
Modules	Merupakan program kecil atau prosedur yang kegunaanya adalah untuk perancangan modul aplikasi pengolahan database tingkat lanjut.

Ada beberapa istilah yang harus diketahui pada saat bekerja di dalam Access diantarnya yaitu :

a. Tabel

Adalah sekumpulan data yang memiliki topic tertentu. Table mengatur data di dalam kolom (dikaitkan dengan field) dan baris (dikaitkan dengan record).

b. Field

Tempat atau dimana data atau informasi dalam kelompok sama atau sejenis dimasukan. Field pada umumnya tersimpan dalam bentuk kolom secara vertical pada table.

c.Record

Merupakan data lengkap dalam jumlah tunggal yang biasanya tersimpan dalam bentuk baris secara horizontal pada table.

2.9.3 Rational Rose

Rational Rose adalah software yang memiliki perangkat-perangkat pemodelan secara visual untuk membangun suatu solusi dalam rekayasa software dan pemodelan bisnis. Rational Rose dikeluarkan oleh perusahaan software

bernama Rational Software, perusahaan yang mencetuskan ide pembentukan konsorsium bagi perusahaan-perusahaan yang memakai standar UML sebagai bahasa pemodelan di perusahaannya, Rational Rose memakai UML sebagai bahasa pemodelannya, ditambah beberapa fitur lain yang membuat Rational Rose menjadi software pemodelan visual yang terkemuka. Bebarapa fitur yang terkemuka diantaranya Rational Rose memiliki Rational Unified Process (RUP) selain itu, Rational Rose memiliki kemampuan membuat solusi client/server, yang kemudian diterapkan dan didistribusikan dalam lingkungan perusahaan.

Rational Rose memiliki keunggulan, diantaranya :

1. Bahasa yang digunakan adalah bahasa pemodelan standar yaitu UML, akan meningkatkan komunikasi intra tim.
2. Rational Rose mendukung round-trip engineering, sehingga kita dapat men-generate model kedalam kode (java, C++, Visual Basic, dan sebagainya) dan melakukan reverse engineering untuk menampilkan arsitektur software dari kode yang ada. Hal ini dapat dilakukan secara bolak-balik sebagai proses interative selama proses rekayasa software.
3. Model dan kode senantiasa singkron selama dalam development cycle.
4. Membangun software menggunakan Rational Rose memudahkan dalam memperbaiki software tersebut karena apabila suatu saat ditemukan requirement baru, dapat digambarkan kembali software tersebut dalam UML.
5. Para user Rational Rose dapat berkomunikasi walaupun bekerja dalam sistem operasi yang berbeda (Windows atau UNIX).
6. Dengan menggunakan Rose Web Publisher suatu tim dapat mengkomunikasikan model dan spesifikasinya dalam web browser.

7. Mendukung rekayasa software untuk sistem client/server sehingga Rational Rose merupakan software pemodelan visual yang tangguh dalam lingkungan client/server, e-business, dan lingkungan perusahaan terdistribusi (kantor-kantornya terletak dalam tempat yang terpisah-pisah)

2.10 **Kasus yang dianalisis / dirancang**

SDN Karyamulya adalah suatu instansi pendidikan yang dimana terjadi proses pengolahan data siswa, perhitungan nilai, dan proses registrasi atau daftar ulang. Di bawah ini penjelasan tentang Pengolahan Data Siswa, Sekolah Dasar, dan Siswa Sekolah Dasar. Berikut penjelasannya di bawah ini :

2.10.1 **Pengertian Pengolahan Data**

Pengolahan data *(data prosecing)* adalah manipulasi dari data ke dalam bentuk yang lebih berguna dan lebih berarti, berupa suatu informasi yang lebih berarti dari suatu kejadian. Jadi Pengolahan Data Electronic (PDE) atau Electronoc Data Processing (EDP) adalah manipulasi dari data ke dalam bentuk yang lebih berarti berupa suatu informasi dengan menggunakan suatu alat elektronik, yaitu Komputer.

2.10.2 Pengertian Sekolah Dasar

Sekolah Dasar (disingkat **SD**) adalah jenjang paling dasar pada pendidikan formal di Indonesia. Sekolah Dasar ditempuh dalam waktu 6 tahun, mulai dari Kelas 1 sampai Kelas 6. Saat ini murid Kelas 6 tidak diwajibkan mengikuti Ujian Nasional_ (dahulu Ebtanas) yang mempengaruhi kelulusan siswa. Lulusan Sekolah Dasar dapat melanjutkan pendidikan ke Sekolah Menengah

Pertama (atau sederajat).

Sekolah Dasar diselenggarakan oleh pemerintah maupun swasta. Sejak diberlakukannya otonomi daerah pada tahun 2001, pengelolaan Sekolah Dasar Negeri di Indonesia yang sebelumnya berada di bawah Departemen Pendidikan Nasional, kini menjadi tanggung jawab kabupaten. Sedangkan Departemen Pendidikan Nasional hanya berperan sebagai regulator dalam bidang standar nasional pendidikan. Secara struktural, Sekolah Dasar Negeri berada di bawah Unit Pelaksana Teknis Dinas Pendidikan di kecamatan.

2.10.3 Siswa Sekolah Dasar

Siswa Sekolah Dasar atau biasa juga disebut Pelajar Sekolah Dasar, umumnya berusia 7-12 tahun. Di Indonesia, setiap warga negara berusia 7-15 tahun tahun wajib mengikuti pendidikan dasar, yakni Sekolah Dasar (atau sederajat) 6 tahun dan Sekolah Menengah Pertama (atau sederajat) 3 tahun.