BAB 2
TINJAUAN PUSTAKA

2.1. Profil Laboratorium INDI TF-ITB

Profil laboratorium INDI merupakan pembahasan mengenai sejarah, visi dan misi, struktur organisasi serta deskripsi jabatan pada perusahaan yang akan menjadi tempat dibangunnya sistem ini.

2.1.1 Sejarah Laboratorium INDI TF-ITB

Sejarah Laboratorium INDI dimulai dari dibentuknya Kelompok Keahlian Instrumentasi dan Kontrol (KK-INK)) pada tahun 1999 dimana kelompok ini adalah salah satu Kelompok Keahlian di lingkungan Fakultas Teknologi Industri - Institut Teknologi Bandung (ITB) dan laboratorium INDI dipimpin oleh Bpk. Dr. –eng. Parsaulian Siregar.

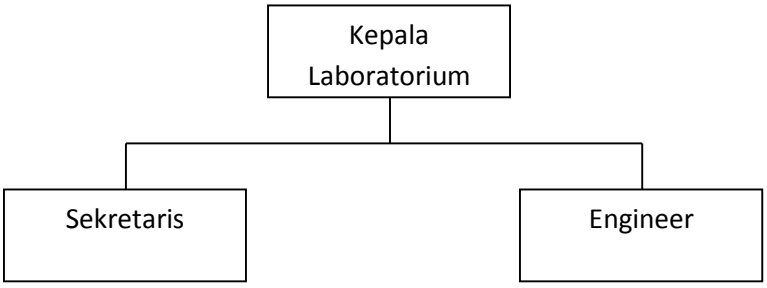
Konsep pendidikan Teknik Fisika yang yang menjadi fokus utama di laboratorium INDI adalah bagaimana menyelesaikan masalah-masalah di bidang instrumentasi dan kontrol. Laboratorium INDI berorientasi pada ilmu rekayasa yang ditunjang dengan pengetahuan dasar yang kokoh. Dalam dunia industri seorang lulusan instrumen memiliki kemampuan khusus yang seolah-olah dapat dijadikan sebagai “jembatan” penghubung antara berbagai bidang teknik lainnya. Dengan berbekal penguasaan dasar-dasar keilmuan yang lebih luas seorang sarjana Teknik Fisika mampu menganalisa berbagai problem yang melibatkan beberapa bidang teknik yang berbeda.

2.1.2 Visi dan Misi

Visi dan Misi Labratorium Instrumentasi adalah menjadi pelopor dalam pengembangan keilmuan instrumentasi dan kontrol untuk menunjang proses pendidikan serta penelitian yang dapat memberikan kontribusi nyata dalam pengembangan keilmuan di tingkat nasional dan internasional.

2.1.3 Struktur Organisasi Laboratorium INDI

Struktur organisasi instansi adalah gambar bagan yang menjelaskan posisi dan hierarki struktur kerja karyawan di dalam laboratorium. Struktur organisasi laboratorium INDI dapat dilihat pada Gambar 2-1 Struktur Organisasi Laboratorium INDI.



Gambar 2-1 Struktur Organisasi Laboratorium INDI

2.1.4 Deskripsi Kerja Laboratorim INDI

Deskripsi kerja Struktur Organisasi adalah penjelasan tentang bagian kerja. Deskripsi kerja dapat dilihat pada Tabel 2-1 Deskripsi Kerja di Laboratorium INDI TF-ITB.

Tabel 2-1 Deskripsi Kerja di Laboratorium INDI

No	Jabatan	Deskripsi Kerja
1	Kepala Laboratorium INDI	Pengajar, Komunikator, pengambil keputusan, pemimpin, pengelola dan eksekutor
2	Sekertaris	Meliputi pengetikan, pembuatan surat menyurat, membuat laporan, pengaturan jadwal dan absensi.
3	Engineer	Sebagai instruktur dalam mengoperasikan peralatan laboratorium, membantu dalam pelaksanaan *internal project.*

2.2. Landasan Teori

Bab ini akan membahas tentang landasan teori yang digunakan untuk melakukan penelitian tugas akhir. Adapun landasan teori yang akan dibahas adalah tentang sistem pendukung keputusan (SPK), *flowmeter,* metode *Multyfactor Evaluation Proccess, tools* pemodelan perangkat lunak dan *tools* implementasi yang meliputi bahasa pemrograman yang digunakan.

2.2.1 SPK (Sistem Pendukung Keputusan)

Pengambilan keputusan merupakan proses pemilihan alternative tindakan untuk mencapai tujuan atau sasaran tertentu. Pengambilan keputusan dilakukan dengan pendekatan sistematis terhadap permasalahan melalui proses pengumpulan data menjadi informasi serta ditambah dengan faktor – faktor yang perlu dipertimbangkan dalam pengambilan keputusan.

SPK biasanya dibangun untuk mendukung solusi atas suatu masalah atau untuk mengevaluasi suatu peluang. SPK yang seperti itu disebut aplikasi SPK, digunakan dalam pengambilan keputusan. Aplikasi SPK menggunakan CBIS (*Computer Based Information System*) yang fleksibel, interaktif, dan dapat diadaptasi, yang dikembangkan untuk mendukung solusi atas masalah manajemen spesifik yang tidak terstruktur. Aplikasi SPK menggunakan data, memberikan antarmuka pengguna yang mudah, dan dapat menggabungkan pemikiran pengambil keputusan.

Menurut Keen dan Scoot Morton [3] Sistem Pendukung Keputusan merupakan penggabungan sumber – sumber kecerdasan individu dengan kemampuan komponen untuk memperbaiki kualitas keputusan. Sistem Pendukung Keputusan juga merupakan sistem informasi berbasis komputer untuk manajemen pengambilan keputusan yang menangani masalah – masalah semi struktur .

Dengan pengertian diatas dapat dijelaskan bahwa sistem pendukung keputusan bukan merupakan alat pengambilan keputusan, melainkan merupakan sistem yang membantu pengambil keputusan dengan melengkapi mereka dengan informasi dari data yang telah diolah dengan relevan dan diperlukan untuk

membuat keputusan tentang suatu masalah dengan lebih cepat dan akurat. Sehingga sistem ini tidak dimaksudkan untuk menggantikan pengambilan keputusan dalam proses pembuatan keputusan.

Alter (p2002) mendefinisikan Sistem pendukung keputusan atau *Decision Support Sistems* (DSS) adalah sistem informasi interkatif yang menyediakan informasi, pemodelan, dan pemanipulasian data yang digunakan untuk membantu pengambilan keputusan pada situasi yang semi terstruktur dan situasi yang tidak terstruktur dimana tak seorang pun tahu secara pasti bagaimana keputusan seharusnya dibuat. Konsep DSS dikemukaakan pertama kali oleh Scoot-Morton pada tahun 1971. Beliau mendefinisikan cikal bakal DSS tersebut sebagai sistem berbasis komputer yang interaktif, yang membantu pengambil keputusan dengan menggunakan data dan model untuk memecahkan persoalan-persoalan tak terstruktur (Turban, McLean, dan Wetherbe, 1999). [7]

2.2.1.1 Tujuan dari SPK

1. Membantu manajer dalam pengambilan keputusan atas masalah semiterstruktur.
2. Memberikan dukungan atas pertimbangan manajer dan bukannya dimaksudkan untuk menggantikan fungsi manajer.
3. Meningkatkan efektivitas keputusan yang diambil manajer lebih daripada perbaikan efisiensinya.
4. Kecepatan komputasi.
5. Peningkatan produktivitas.
6. Dukungan kualitas.
7. Berdaya saing.
8. Mengatasi keterbatasan kognitif dalam pemrosesan dan penyimpanan.

2.2.1.2 Karakteristik Sistem Pendukung Keputusan (SPK)

Berdasarkan pengertian Sistem Pendukung Keputusan maka dapat ditentukan karakteristik antara lain [3]:

1. SPK menyediakan dukungan bagi pengambil keputusan utamanya pada situasi semi terstruktur dan tak terstruktur dengan memadukan pertimbangan manusia dan informasi terkomputerisasi. Berbagai masalah tak dapat diselesaikan (atau tak dapat diselesaikan secara memuaskan) oleh sistem terkomputerisasi lain, seperti EDP atau MIS, tidak juga dengan metode atau *tool* kuantitatif standar.
2. Dukungan disediakan untuk berbagai level manajerial yang berbeda, mulai dari pimpinan puncak sampai manajer lapangan.
3. Dukungan disediakan bagi individu dan juga bagi group. Berbagai masalah organisasional melibatkan pengambilan keputusan dari orang dalam group. Masalah yang strukturnya lebih sedikit seringkali hanya membutuhkan keterlibatan beberapa individu dari departemen dan level organisasi yang berbeda.
4. SPK menyediakan dukungan ke berbagai keputusan yang berurutan atau saling berkaitan.
5. SPK mendukung berbagai fase proses pengambilan keputusan: intelligence, design, choice dan implementation.
6. SPK mendukung berbagai proses pengambilan keputusan dan *style* yang berbeda-beda; ada kesesuaian diantara SPK dan atribut pengambil keputusan individu (contohnya *vocabulary* dan *style* keputusan).
7. SPK selalu bisa beradaptasi sepanjang masa. Pengambil keputusan harus reaktif, mampu mengatasi perubahan kondisi secepatnya dan beradaptasi untuk membuat SPK selalu bisa menangani perubahan ini. SPK adalah fleksibel, sehingga user dapat menambahkan, menghapus, mengkombinasikan, mengubah, atau mengatur kembali elemen-elemen dasar (menyediakan respon cepat pada situasi yang tak diharapkan). Kemampuan ini memberikan analisis yang tepat waktu dan cepat setiap saat.
8. SPK mudah untuk digunakan. User harus merasa nyaman dengan sistem ini. User friendliness, fleksibelitas, dukungan grafis terbaik, dan antarmuka

bahasa yang sesuai dengan bahasa manusia dapat meningkatkan efektivitas SPK. Kemudahan penggunaan ini diiimplikasikan pada mode yang interaktif.

9. SPK mencoba untuk meningkatkan efektivitas dari pengambilan keputusan (akurasi, jangka waktu, kualitas), lebih daripada efisiensi yang bisa diperoleh (biaya membuat keputusan, termasuk biaya penggunaan komputer).
10. Pengambil keputusan memiliki kontrol menyeluruh terhadap semua langkah proses pengambilan keputusan dalam menyelesaikan masalah. SPK secara khusus ditujukan untuk mendukung dan tak menggantikan pengambil keputusan. Pengambil keputusan dapat menindaklanjuti rekomendasi komputer sembarang waktu dalam proses dengan tambahan pendapat pribadi atau pun tidak.
11. SPK mengarah pada pembelajaran, yaitu mengarah pada kebutuhan baru dan penyempurnaan sistem, yang mengarah pada pembelajaran tambahan, dan begitu selanjutnya dalam proses pengembangan dan peningkatan SPK secara berkelanjutan.
12. User/pengguna harus mampu menyusun sendiri sistem yang sederhana. Sistem yang lebih besar dapat dibangun dalam organisasi user tadi dengan melibatkan sedikit saja bantuan dari *spesial*is di bidang Sistem Informasi (SI).
13. SPK biasanya mendayagunakan berbagai model (standar atau sesuai keinginan user) dalam menganalisis berbagai keputusan. Kemampuan pemodelan ini menjadikan percobaan yang dilakukan dapat dilakukan pada berbagai konfigurasi yang berbeda. Berbagai percobaan tersebut lebih lanjut akan memberikan pandangan dan pembelajaran baru.
14. SPK dalam tingkat lanjut dilengkapi dengan komponen knowledge yang bisa memberikan solusi yang efisien dan efektif dari berbagai masalah yang pelik.

2.2.1.3 Keuntungan SPK:

1. Mampu mendukung pencarian solusi dari masalah yang kompleks.

2. Respon cepat pada situasi yang tak diharapkan dalam kondisi yang berubah-ubah.
3. Mampu untuk menerapkan berbagai strategi yang berbeda pada konfigurasi berbeda secara cepat dan tepat.
4. Pandangan dan pembelajaran baru.
5. Memfasilitasi komunikasi.
6. Meningkatkan kontrol manajemen dan kinerja.
7. Menghemat biaya.
8. Keputusannya lebih tepat.
9. Meningkatkan efektivitas manajerial, menjadikan manajer dapat bekerja lebih singkat dan dengan sedikit usaha.
10. Meningkatkan produktivitas analisis.

2.2.1.4 Komponen Sistem Pendukung Keputusan

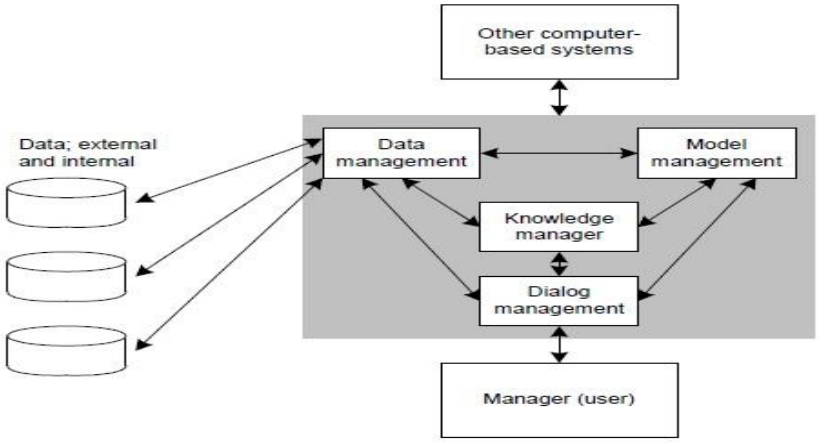
Sistem pendukung keputusan memiliki beberapa komponen yang terdiri dari Data Management termasuk database yang mengandung data yang relevan untuk berbagai situasi dan diatur oleh software yang disebut Database Management Sistem (DBMS) serperti terlihat pada Gambar 2-3 Konseptual Sistem Pendukung Keputusan Model

Data Management. Termasuk database, yang mengandung data yang relevan untuk berbagai situasi dan diatur oleh software yang disebut Database Management Systems (DBMS).

1 *Model Management*. Melibatkan model finansial, statistikal, management science, atau berbagai model kuantitatif lainnya, sehingga dapat memberikan ke sistem suatu kemampuan analitis, dan manajemen software yang diperlukan.

2 *Communication* (dialog subsystem). User dapat berkomunikasi dan memberikan perintah pada SPK melalui subsistem ini. Ini berarti menyediakan antarmuka.

3 *Knowledge Management*. Subsistem optional ini dapat mendukung subsistem lain atau bertindak sebagai komponen yang berdiri sendiri



Gambar 2-2 Konseptual Sistem Pendukung Keputusan Model

2.2.2 Flowmeter

Flowmeter adalah alat yang digunakan untuk mengukur massa atau laju aliran volumetrik cairan atau gas. Sebelum menetapkan *flowmeter*, juga dianjurkan untuk menentukan apakah aliran informasi akan lebih berguna jika disajikan dalam unit massa atau volumetrik. Ketika mengukur aliran bahan yang mempunyai tekanan, aliran volumetrik tidak terlalu berarti, kecuali kepadatan adalah konstan. Ketika kecepatan (volumetrik aliran) dari cairan mampat diukur, faktor gelembung udara akan menyebabkan kesalahan, karena itu, udara dan gas harus dipindahkan sebelum mencapai fluida meter. (Noor Yudha Priyantini, 2010).

Tidak semua *fluida* yang berpindah dinamakan *fluida* bergerak. Yang dimaksud fluida bergerak adalah jika fluida tersebut bergerak lurus terhadap sekitar. Aliran fluida dikatakan aliran garis lurus apabila aliran fluida yang mengalir mengikuti suatu garis (lurus melengkung) yang jelas ujung pangkalnya. Aliran garis lurus juga disebut aliran berlapis atau aliran laminar (*laminar flow*).

Kecepatan- kecepatan partikel di tiap titik pada garis arus, searah dengan garis singgung di titik itu. Dengan demikian garis arus tidak pernah

berpotongan. Pada fluida yang tak termampatkan, hasil kali antara kelajuan aliran fluida dan luas penampangnya selalu tetap. Jadi A.v = konstan, atau disebut debit (Q). Debit adalah volume fluida (m^3) yang mengalir melewati suatu penampang dalam selang waktu tertentu. Dirumuskan dengan persamaan 2.1 berikut ini:

$$Q=V/t \quad \text{.....(2.1)}$$

Keterangan :

Q = debit (m^3 / s)

V = volume fluida (m^3)

t = waktu fluida mengalir (s)

Sistem kontrol fluida adalah sebuah alat yang dapat mengatur jumlah debit air yang akan dikeluarkan. Dengan sistem digital, sistem kontrol ini dirancang untuk mempermudah dalam pengemasan atau penakaran cairan dengan batas keluaran yang ditentukan. Rancangan alat ini berupa perangkat keras dimana perangkat yang satu dengan yang lainnya berhubungan dan saling mendukung, adapun perangkat keras tersebut terdiri dari Mikrokontroler, piringan derajat, *optocoupler*, *water meter* termodifikasi, *solenoid*, pompa air dan LCD karakter. Sedangkan perangkat lunaknya berupa program pada mikrokontroler dengan menggunakan bahasa pemrograman *assembly* sehingga dapat mengontrol perangkat tersebut baik berupa *input* maupun *output*.

2.2.2.1 Pengelompokan *Flowmeter (Categorization)*

Beberapa jenis *flowmeter* yang sering digunakan di dalam industri proses dapat dikelompokkan seperti terlihat pada Tabel 2-2 Pengelompokan jenis *flowmeter*:

Tabel 2-2 Pengelompokkan Jenis Flowmeter [4]

No	Flowmeter	
1.	Differential Pressure	Orifice
		Venturi Tube
		Flow Nozzle
		Pitot Tube
		Pitot Tube (Averaging)
		Anubar
		Elbow Taps

No	Flowmeter	
		Wedge
		V-Cone
		Dall Tube
2.	Variable Area	Rotameter
		Movable Vane
		Weir, Flume
3.	Positive Displacement	Nutating Disc
		Oscillating Piston
		Oval Gear
		Roots
4.	Turbine	
5.	Thermal	
6.	Target	
7.	Ultrasonic	Doppler
		Transit Time
8.	Magnetic	
9.	Coriolis	
10.	Vortex	

2.2.2.2 *Custody Transfer*

Custody Transfer merupakan pemindahan produk untuk perdagangan / transaksional di bawah pengawasan terkait. *Custody transfer* lebih mensyaratkan akurasi yang relatif lebih tinggi.

Dalam regulasi PERTAMINA (Panduan PERTAMINA PPDN 1990): [4]

1. kesalahan pengukuran ketinggian produk BBM di dalam tangki timbun (*Tank Gauging*) untuk *Custody Transfer* adalah sebesar maksimal 0.01%
2. Sedangkan untuk pengukuran temperatur, ketelitian ditetapkan sampai pada 0.25^{o}C

2.2.3 Metode Multi-Factor Evaluation Process (MFEP)

Multi Factor Evaluation Process (MFEP) adalah metode kuantitatif yang menggunakan '*weighting system*'. Dalam pengambilan keputusan multi faktor, pengambil keputusan secara subyektif dan intuitif menimbang berbagai faktor yang mempunyai pengaruh penting terhadap alternatif pilihan mereka. Untuk keputusan yang berpengaruh secara strategis, lebih dianjurkan menggunakan sebuah pendekatan kuantitatif seperti MFEP. Dalam MFEP pertama-tama seluruh kriteria yang menjadi faktor penting dalam melakukan pertimbangan diberikan pembobotan (*weighting*) yang sesuai. Langkah yang

sama juga dilakukan terhadap alternatif-alternatif yang akan dipilih, yang kemudian dapat dievaluasi berkaitan dengan faktor–faktor pertimbangan tersebut. Metode MFEP menentukan bahwa alternatif dengan nilai tertinggi adalah solusi terbaik berdasarkan kriteria yang telah dipilih.

2.2.3.1 Penentuan Faktor Penting (*Weight Factor*)

Dalam MFEP pertama-tama ditentukan kriteria yang menjadi faktor penting dalam melakukan pertimbangan. Faktor-faktor penting tersebut ditentukan oleh perusahaan berdasarkan kepentingan dari alternatif-alternatif yang akan dibandingkan satu sama lain. Langkah yang sama dilakukan terhadap semua jenis alternatif apabila terdapat lebih dari satu jenis alternatif. Faktor penting (*Weight Factor)* tersebut dapat dilihat pada Tabel 2-3 *Weight Factor*.

Tabel 2-3 Weight Factor (WF)

Alternatif	WF1	WF2	WF3
Alternatif 1			
Alternatif 2			
Alternatif 3			

2.2.3.2 Pemberian Bobot

Langkah selanjutnya adalah memberikan pembobotan kepada faktor-faktor yang digunakan dimana total pembobotan adalah 1 (Σ pembobotan= 1). Pemberian bobot pada faktor penting dapat dilihat pada rumus 2.2.

$$WF1 + WF2 + WF3 = 1 \quad \text{……...……..……...…} \quad (2.2)$$

Di mana:
WF = *Weight Factor*

Berdasarkan analisis penghitungan rumus 2-2, diberikan ilustrasi *factor weight* dari masing-masing *weight factor*. Tabel *factor weight* dari masing-masing *weight factor* dapat dilihat pada Tabel 2-4 *Factor weight*.

Tabel 2-4 Factor Weight

Weight Factor	*Factor Weight*
WF1	0.60
WF2	0.15
WF3	0.25
Total	**1**

2.2.3.3 Evaluasi *Factor Weight*

Setelah dilakukan pembobotan, alternatif-alternatif yang telah ditentukan sebelumnya lalu dievaluasi dengan menghitung Factor Weight dari setiap alternatif. Sebagai ilustrasi evaluasi faktor dapat dilihat pada Tabel 2-5 Tabel Factor Evaluation

Tabel 2-5 Tabel Factor Evaluation

Weight Factor	**Alternatif 1**	**Alternatif 2**	**Alternatif 3**
WF1	a11	a12	a13
WF2	a21	a22	a23
WF3	a31	a32	a33

Data evaluasi faktor penting dari tiap alternatif dapat dianalisis dengan menggunakan rumus 2.3.

$$X = (WF1 * a11) + (WF2 * a21) + (WF3 * a31) + (WF.... * a....) \text{ ……..…….. } (2.3)$$

Di mana:
X = *Weighted Evaluation*
WF = *Weight Factor*
a = *Factor Evaluation*

Dengan adanya informasi yang terdapat pada Tabel 2.3, didapatkan jumlah total nilai evaluasi untuk setiap alternatif. Setiap alternatif mempunyai sebuah nilai evaluasi bagi ketiga faktor yang menjadi pertimbangannya, dan kemudian nilai faktor tersebut dikalikan dengan faktor evaluasi dan dijumlahkan untuk mendapatkan total nilai evaluasi untuk setiap alternatif. Data total nilai evaluasi dapat dianalisis dengan menggunakan rumus 2.4 .

$$X = \frac{x1 + x2 + x3 + x...)}{n} \quad \text{……...…….……... } (2.4)$$

Di mana:

X= *Total Weighted Evaluation*
x = *Weighted Evaluation*
n = Jumlah *Weighted Evaluation*

Dengan adanya informasi yang terdapat pada Tabel 2-3, didapatkan jumlah total nilai evaluasi untuk setiap alternatif. Seperti yang dapat dilihat pada Tabel 2-6, di mana alternatif 1 memiliki nilai evaluasi total 1,17, dari Tabel 2-7 didapat bahwa alternatif 2 memiliki nilai evaluasi total 1,49, dan dari Tabel 2-8 diperoleh nilai alternatif 3 adalah 2,17.

Tabel 2-6 Tabel Alternatif 1

Factor Name	***Factor Weight***	***Factor Evaluation***	***WEIGHTED EVALUATION***
WF1	0.60	2	1,2
WF2	0.15	6	0,9
WF3	0.25	5	1,25
TOTAL	1		**1,17**

Tabel 2-7 Tabel Alternatif 2

Factor Name	***Factor Weight***	***Factor Evaluation***	***WEIGHTED EVALUATION***
WF1	0.60	5	3
WF2	0.15	3	0,45
WF3	0.25	4	1
TOTAL	1		**1,49**

Tabel 2-8 Tabel Alternatif 3

Factor Name	***Factor Weight***	***Factor Evaluation***	***WEIGHTED EVALUATION***
WF1	0.60	8	4,8
WF2	0.15	2	0,3
WF3	0.25	5	1,25
TOTAL	1		**2,17**

Berdasarkan penghitungan *weighted evaluation* dari 3 (tiga) alternatif, dapat dilihat bahwa alternatif 3 memiliki nilai total *weighted evaluation* yang paling besar. Hal ini dapat menjadi rekomendasi dalam menentukan pilihan dari 3 (tiga) alternatif yang tersedia.

2.2.4 Tools Pemodelan

Tools pemodelan adalah alat atau perangkat dan metode yang digunakan untuk memodelkan perancangan perangkat lunak simulasi. Adapun *tools* pemodelan yang digunakan dalam penelitian ini adalah :

1. *ERD (Entity relational Diagram)*
2. DFD (*Data Flow Diagram*)

2.2.4.1 ERD (Entity Relational Diagram)

Entity Relationship Diagram atau biasa dikenal dengan diagram E-R secara grafis menggambarkan isi sebuah *database*. ERD terbagi atas tiga komponen, yaitu entitas (*entity*), atribut (*atribute*), dan relasi atau hubungan (*relation*). Secara garis besar entitas merupakan dasar yang terlibat dalam sistem. Atribut atau *field* berperan sebagai penjelas dari entitas, dan relasi atau hubungan menunjukkan hubungan yang terjadi antara dua entitas[5].

1. Entitas (*Entity*)

   Entitas (*entity*) menunjukkan objek-objek dasar yang terkait di dalam sistem. Objek dasar dapat berupa orang, benda atau hal lain yang keterangannya perlu disimpan dalam basis data.
2. Atribut (*Atribute*)

   Atribut sering juga disebut sebagai properti (*property*) merupakan keterangan-keterangan yang terkait pada sebuah entitas yang perlu disimpan sebagai basis data.Atribut berfungsi sebagai penjelas sebuah entitas.
3. Relasi (*Relation*)

   Relasi atau hubungan adalah kejadian atau transaksi yang terjadi di antara dua entitas yang keterangannya perlu disimpan dalam basis data.

2.2.4.2 Data Flow Diagram (DFD)

Data Flow Diagram (DFD) adalah representasi grafik dari sebuah sistem. DFD menggambarkan komponen-komponen sebuah sistem, aliran-aliran data di mana komponen-komponen tersebut, dan asal, tujuan, dan

penyimpanan dari data tersebut. *Data Flow Diagram* digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir atau lingkungan fisik dimana data tersebut akan disimpan. *Data Flow Diagram* juga digunakan pada metodologi pengembangan sistem yang terstruktur. [2]

2.2.5 Client Server

Client – Server merupakan sebuah paradigma dalam teknologi informasi yang merujuk kepada cara untuk mendistribusikan aplikasi ke dalam dua pihak, yaitu pihak klien dan pihak server. Prinsip kerjanya yaitu membagi sebuah aplikasi menjadi dua bagian terpisah tetapi masih dalam sebuah kesatuan. Dua bagian tersebut yaitu komponen klien dan komponen server.

Komponen klien akan menyiapkan data yang dimasukkan pengguna dan kemudian mengirimkannya pada komponen server. Komponen server kemudian memproses data tersebut dan mengembalikan hasil pemrosesan tersebut kepada komponen klien. Klien menerima data tersebut lalu menampilkannya pada pengguna melalui aplikasi yang berinteraksi dengan pengguna.

2.2.6 Bahasa Pemrograman

Pascal merupakan salah satu bahasa pemrograman tingkat tinggi. Pemrograman tingkat tinggi menandakan bahwa Pascal banyak menggunakan bahasa manusia dalam penulisan sintaksnya. Beberapa bahasa pemrograman tingkat tinggi lainnya yang ada yaitu BASIC dan DELPHI. Sedangkan bahasa pemrograman tingkat rendah yaitu bahasa pemrograman yang masih banyak menggunakan tanda-tanda sehingga bahasa ini cenderung lebih “kotor”. Beberapa bahasa pemrograman tingkat rendah yang sering digunakan adalah C, C++ dan Java.

2.2.5.1 Borland Delphi

Delphi adalah pemrograman yang berorientasi objek yang diturunkan dari bahasa pemrograman Pascal. Pascal adalah bahasa pemrogramnan yang berbasis DOS dibuat pada tahun 1971 oleh Niklaus Wirth dari Swiss, nama Pascal sendiri diambil dari nama matematikawan dan ahli filsafat di Prancis yaitu Blaise Pascal. Pascal dirilis tahun 1983 oleh Borland International salah satu perusahaan software yang terkemuka di California.

Pada tahun 1993 Borland International mengembangkan bahasa pemrograman pascal bersifat Visual yang disebut Delphi. Delphi terus dikembangkan guna mengikuti perkembangan jaman dan sampai sekarang sudah keluar versi terbaru yaitu Delphi XE5 atau yang dikenal dengan Embarcadero XE5.

Delphi adalah alat perancang aplikasi secara cepat dan praktis "Rapid Application Development(RAD)" dan dikembangkan dengan cara visualisasi untuk lebih memudahkan perancang dalam menghasilkan aplikasi-aplikasi yang "rileble" handal. Pada prinsipnya, pemrograman ini sama dengan pemrograman visual lainnya, misalnya Visual Foxpro, Visual Basic, Power Builder apalagi saudara dekatnya Borland C++.

Borland Delphi XE adalah paket bahasa pemrograman yang bekerja dalam sistem operasi windows. Delphi merupakan bahasa pemrograman yang mempunyai cakupan kemampuan yang luas dan sangat canggih. Berbagai jenis aplikasi dapat dibuat dengan Delphi, termasuk aplikasi untuk mengolah teks, grafik, angka, *database* dan aplikasi web.

Secara umum, kemampuan Delphi adalah menyediakan komponen-komponen dan bahasa pemrograman yang handal, sehingga memungkinkan untuk dapat membuat program aplikasi sesuai dengan keinginan, dengan tampilan dan kemampuan yang canggih.

Untuk mempermudah *programmer* dalam membuat program aplikasi, Delphi menyediakan fasilitas pemrograman yang sangat lengkap. Fasilitas

pemrograman tersebut dibagi dalam 2 kelompok, yaitu *object* dan bahasa pemrograman. Secara ringkas, *object* adalah suatu komponen yang mempunyai bentuk fisik dan biasanya dapat dilihat (*visual*). *Object* biasanya dipakai untuk melakukan tugas tertentu dan mempunyai batasan-batasan tertentu. Sedangkan bahasa pemrograman secara singkat dapat disebut sebagai sekumpulan teks yang mempunyai arti tertentu dan disusun dengan aturan tertentu serta untuk menjalankan tugas tertentu. Delphi menggunakan struktur bahasa pemrograman *Object* Pascal yang sudah sangat dikenal di kalangan pemrogram profesional. Gabungan dari *object* dan bahasa pemrograman ini sering disebut sebagai bahasa pemrograman berorientasi *object* atau *Object Oriented Programing* (OOP).

Delphi telah memberikan komponen-komponen yang mudah dalam membangun aplikasi. Dalam grafis pun delphi dapat mengolah data yang membuat tampilan gambar seperti grafik equalizer dengan menggunakan komponen khusus. Secara matematis delphi 2010 telah menyediakan operator untuk mengolah data sehingga dalam membuat aplikasinya pun lebih mudah, adapun tingkatan operator dalam delphi 2010 sehingga jika muncul fungsi matematis maka akan diolah secara otomatis. [5]

2.6.5.3 MySql

MySQL merupakan *software* sistem manajemen *database* (*Database Management System*-DBMS) yang sangat popular dikalangan pemrogram *web*, terutama di lingkungan Linux dengan menggunakan *script* PHP dan Perl. *Software database* ini kini telah tersedia juga pada *platform* sistem operasi windows. MySQL merupakan *database* yang paling popular digunakan untuk membangun aplikasi *web* yang menggunakan *database* sebagai sumber dan pengolahan datanya. Kepopuleran MySQL dimungkinkan karena kemudahannya untuk digunakan, cepat secara kinerja *query*, dan mencukupi untuk kebutuhan *database* perusahaan-perusahaan skala menengah kecil. MySQL merupakan *database* yang digunakan oleh situs-situs terkemuka di

internet untuk menyimpan datanya. MySQL dan PHP dianggap sebagai pasangan *software* pembangun aplikasi *web* yang ideal. [6]

Perintah-Perintah SQL

1. Menambah Data (*Insert*)

   Perintah SQL yang digunakan untuk menambah data ke *database* adalah perintah *INSERT*. Perintah ini mempunyai bentuk umum sebagai berikut:

   INSERT [INTO] nama_tabel [(daftar *field*/kolom)] *VALUES* (daftar_data)

2. Memilih/Mengakses Data (*Select*)

   Perintah SQL yang digunakan untuk memilih data adalah perintah *SELECT*. Bentuk umum perintah ini adalah sebagai berikut:

   SELECT daftar_select *FROM* daftar_tabel

   [*WHERE* kondisi_pencarian]

   [*GROUP BY* daftar_group_by]

   [*HAVING* kondisi_pencarian]

   [*ORDER BY* daftar_order[*ASC* | *DESC*]]

3. Mengubah/Mengubah Data (*Update*)

   Perintah SQL yang digunakan untuk mengubah data adalah perintah *UPDATE SET*. Perintah ini mempunyai bentuk umum sebagai berikut:

   UPDATE nama_table *SET* *field*1=databaru1 [data2=databaru2]

   [*WHERE* kondisi_update]

4. Menghapus Data (*Delete*)