

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terkait

2.1.1. Sistem

Menurut Rosa A.S dan M. Shalahuddin (2016) Sistem adalah sebagai suatu kumpulan atau himpunan dari unsur, komponen, atau variabel yang terorganisir, saling berinteraksi, saling tergantung satu sama lain, dan terpadu. Model umum sebuah sistem adalah *input*, proses, dan *output*. Hal ini merupakan konsep sebuah sistem yang sangat sederhana sebab sebuah sistem dapat mempunyai beberapa masukan dan keluaran. Selain itu, sebuah sistem mempunyai karakteristik atau sifat-sifat tertentu yang mencirikan bahwa hal tersebut bisa dikatakan sebagai suatu sistem.

2.1.2. Sistem Pendukung Keputusan

Konsep Sistem Pendukung Keputusan (SPK) atau Decision Support System (DSS), dikenalkan pertama kali pada tahun 1970-an oleh Little. Menurut Little (1970), *Decision Support System* adalah kumpulan prosedur model, yang digunakan sebagai data dan pertimbangan untuk membantu manajer dalam mengambil keputusan (Turban, 2011:88).

Sistem Penunjang Keputusan (SPK) juga merupakan suatu sistem interaktif yang membantu pengambilan keputusan melalui penggunaan data dan model-model keputusan untuk memecahkan masalah-masalah yang sifatnya semi terstruktur dan tidak terstruktur (Dona Yuliawati, 2014).

Istilah SPK mengacu pada suatu sistem yang memanfaatkan dukungan komputer dalam proses pengambilan keputusan. Untuk memberikan pengertian yang lebih maka ada beberapa definisi mengenai SPK oleh beberapa ahli. Sistem Penunjang Keputusan merupakan implementasi teori-teori pengambilan keputusan yang telah

diperkenalkan oleh ilmu-ilmu seperti *operation research* dan *management science*, hanya bedanya adalah bahwa jika dahulu untuk mencari penyelesaian masalah yang dihadapi harus dilakukan perhitungan iterasi secara manual (biasanya untuk mencari nilai minimum, maksimum, atau optimum), saat ini computer PC telah menawarkan kemampuannya untuk menyelesaikan persoalan yang sama dalam waktu relatif singkat (Kusrini, 2011).

2.1.3. Pengertian Kinerja Karyawan

Istilah kinerja berasal dari kata *Job Performance* (prestasi kerja) atau prestasi sesungguhnya yang dicapai oleh seseorang. Pengertian Kinerja adalah hasil kerja secara kualitas dan kuantitas yang dicapai oleh seorang pegawai dalam melaksanakan tugasnya sesuai dengan tanggung jawab yang diberikan padanya .

2.1.4. Penilaian Kinerja Karyawan

Dessler (2009) berpendapat bahwa kinerja (prestasi kerja) karyawan adalah prestasi aktual karyawan dibandingkan dengan prestasi yang diharapkan dari karyawan.

Menurut Ilyas (2001) kinerja adalah penampilan hasil karya personel baik kuantitas maupun kualitas dalam suatu organisasi. Kinerja dapat merupakan penampilan individu maupun kerja kelompok personel. Penilaian prestasi kinerja adalah proses yang meliputi : (1) penetapan standar prestasi kerja, (2) penilaian prestasi kerja aktual karyawan dalam hubungan dengan standar – standar ini, dan (3) memberi umpan balik kepada karyawan dengan tujuan memotivasi orang tersebut untuk menghilangkan kemerosotan prestasi kerja.

2.1.5. Indikator Kinerja Utama / KPI (*Key Performance Indicator*)

Indikator pada hakikatnya adalah data atau informasi yang digunakan menyatakan keadaan sesuatu. Menurut Moh. Faisal Amir (2015, 125) Indikator kinerja adalah penanda yang menyatakan bahwa apakah sebuah tugas kerja yang dibebankan kepada kelompok karyawan sudah dapat diselesaikan dengan baik atau terdapat masalah tertentu yang perlu diurai. Indikator kinerja utama (*Key Performance Indicator*) menjadi unsur yang sangat penting dalam pengelolaan perusahaan, departemen atau organisasi khususnya untuk mengendalikan usaha agar tetap sehat.

2.1.6. Pengambilan Keputusan

1) Pengertian Keputusan

Keputusan adalah suatu reaksi terhadap beberapa solusi alternatif yang dilakukan secara sadar dengan cara menganalisa kemungkinan-kemungkinan dari alternatif tersebut bersama konsekuensinya. Setiap keputusan akan membuat pilihan terakhir, dapat berupa tindakan atau opini. Itu semua bermula ketika perlu untuk melakukan sesuatu tetapi tidak tahu apa yang harus dilakukan. Untuk itu keputusan dapat dirasakan rasional atau irrasional dan dapat berdasarkan asumsi kuat atau asumsi lemah. (Turban, 2005:3).

2) Pengertian Pengambilan Keputusan

Pengambilan keputusan merupakan proses pemilihan alternatif 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. (Turban, 2005:3)

Menurut Dewanto, J.I, dan Adhikara, A.MF. (2015) Konsep Sistem Pendukung Keputusan (SPK) pertama kali diungkapkan pada awal tahun 1970-an oleh Michael S. Scott Morton dengan istilah *Management Decision System*. Sistem tersebut adalah suatu sistem yang berbasis komputer yang ditunjukan untuk membantu pengambil keputusan dengan memanfaatkan data dan model tertentu untuk memecahkan berbagai persoalan yang tidak terstruktur.

2.1.7. Definisi *Analytic Hierarchy Process* (AHP)

Analytical Hierarchy Process (AHP) merupakan suatu model pendukung keputusan yang dikembangkan oleh Thomas L. Saaty. AHP menguraikan masalah multi faktor atau multi criteria yang kompleks menjadi suatu hirarki. Menurut Saaty (2003), hirarki didefinisikan sebagai suatu representasi dari sebuah permasalahan yang kompleks dalam suatu struktur multi level dimana level pertama adalah tujuan, yang diikuti level faktor, kriteria, sub kriteria, dan seterusnya ke bawah hingga level terakhir dari alternatif. Dengan hirarki, suatu masalah yang kompleks dapat diuraikan ke dalam kelompok-kelompoknya yang kemudian diatur menjadi suatu bentuk hirarki sehingga permasalahan akan tampak lebih terstruktur dan sistematis.

Metode *Analytic Hierarchy Process* (AHP) dikembangkan oleh Thomas L. Saaty, seorang matematikawan di Universitas Pittsburgh Amerika Serikat sekitar tahun 1980. Tujuan utama AHP adalah untuk membuat rangking alternatif dari keputusan dan memilih salah satu yang terbaik bagi kasus multi criteria yang menggabungkan faktor kualitatif dan kuantitatif di dalam keseluruhan evaluasi alternative-alternatif yang ada.

AHP digunakan untuk mengkaji permasalahan yang dimulai dengan mendefinisikan permasalahan tersebut secara seksama kemudian menyusunnya ke dalam suatu hirarki. AHP memasukkan pertimbangan dan nilai-nilai pribadi secara logis. Proses ini bergantung pada imajinasi, pengalaman, dan pengetahuan untuk

menyusun hirarki suatu permasalahan dan bergantung pada logika dan pengalaman untuk member pertimbangan.

Beberapa prinsip yang harus dipahami dalam menyelesaikan permasalahan menggunakan AHP, yaitu :

1. Penyusunan Hirarki

Merupakan langkah penyederhanaan masalah ke dalam bagian yang menjadi elemen pokoknya, kemudian ke dalam bagian-bagiannya lagi, dan seterusnya secara hirarki agar lebih jelas, sehingga mempermudah pengambilan keputusan untuk menganalisis dan menarik kesimpulan terhadap permasalahan tersebut.

2. Menentukan Prioritas

AHP melakukan perbandingan berpasangan antar dua elemen pada tingkat yang sama. Kedua elemen tersebut dibandingkan dengan menimbang tingkat preferensi elemen yang satu terhadap elemen yang lain berdasarkan kriteri tertentu

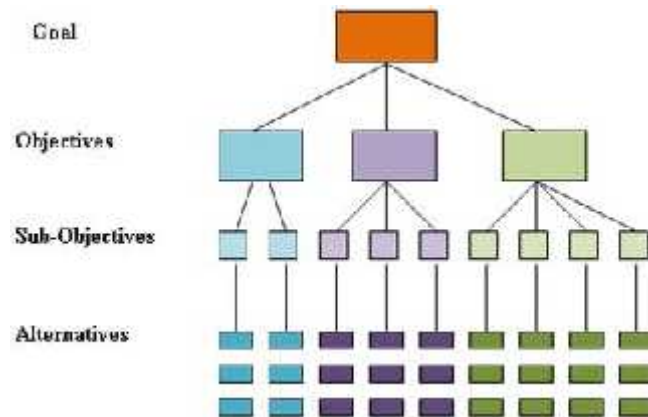
3. Konsistensi logis

Konsistensi logis merupakan prinsip rasional dalam AHP. Konsistensi berarti dua hal, yaitu:

- a. Pemikiran atau objek yang serupa dikelompokkan menurut homogenitas dan relevansinya
- b. Relasi antar objek yang didasarkan pada kriteria tertentu, saling membenarkan secara logis.

2.1.8. Hierarki

Hirarki adalah gambaran dari permasalahan yang kompleks dalam struktur banyak tingkat dimana paling atas adalah tujuan dan diikuti tingkat kriteria, subkriteria dan seterusnya ke bawah sampai pada tingkat yang paling bawah adalah tingkatalternatif. Hirarki menggambarkan secara grafis saling ketergantungan elemen-elemen yang relevan, memperlihatkan hubungan antar elemen yang homogeny dan berhubungan dengan sistem sehingga menjadi satu kesatuan yang utuh. Struktur AHP ditunjukkan seperti pada gambar II.1 berikut:



Gambar 2.1. Struktur Hierarki AHP

i. Penilaian Kriteria Dan Alternatif

Kriteria dan alternatif dilakukan dengan perbandingan berpasangan. Menurut Saaty (1988), untuk berbagai persoalan, skala 1 sampai 9 adalah skala terbaik untuk mengekspresikan pendapat. Nilai dan definisi pendapat kualitatif dari skala perbandingan Saaty dapat diukur menggunakan table analisis seperti tabel dibawah ini:

Tabel 2.1. Skala Penilaian Perbandingan Pasangan

Intensitas Kepentingan	Keterangan
1	Kedua elemen sama pentingnya
3	Elemen yang satu sedikit lebih penting daripada Elemen yang lainnya
5	Elemen yang satu lebih penting daripada yang Lainnya
7	Satu elemen jelas lebih mutlak penting dari pada elemen lainnya
9	Satu elemen mutlak penting dari pada elemen Lainnya

2,4,6,8	Nilai-nilai antara dua nilai pertimbangan- Pertimbangan yang berdekatan
Kebalikan	Jika aktivitas i mendapat satu angka dibandingkan dengan aktivitas j , maka j memiliki nilai kebalikannya dibandingkan dengan i

Sumber: Kusrini (2007)

ii. Prosedur *Analytical Hierarchy Process (AHP)*

Pada dasarnya, prosedur atau langkah-langkah dalam metode *Analytical Hierarchy Process (AHP)* menurut Kusrini (2007:135), adalah :

1. Mendefinisikan masalah dan menentukan solusi yang diinginkan, lalu menyusun hierarki dari permasalahan yang dihadapi.
2. Menentukan prioritas elemen

Langkah pertama dalam menentukan prioritas elemen adalah:

- a. Membuat perbandingan pasangan, yaitu membandingkan elemen secara berpasangan sesuai criteria yang diberikan.
- b. Matriks perbandingan berpasangan diisi menggunakan bilangan untuk merepresentasikan kepentingan relatif dari suatu elemen terhadap elemen yang lainnya.
3. Sintesis

Pertimbangan-pertimbangan terhadap perbandingan berpasangan disintesis untuk memperoleh keseluruhan prioritas. Hal-hal yang dilakukan dalam langkah ini adalah:

- a. Menjumlahkan nilai-nilai dari setiap kolom pada matriks
- b. Membagi setiap nilai dari kolom dengan total kolom yang bersangkutan untuk memperoleh normalisasi matriks.
- c. Menjumlahkan nilai-nilai dari setiap baris dan membaginya dengan jumlah elemen untuk mendapatkan nilai rata-rata.
4. Mengukur Konsistensi

Dalam pembuatan keputusan, penting untuk mengetahui seberapa baik konsistensi

yang ada karena kita tidak menginginkan keputusan berdasarkan pertimbangan dengan konsistensi yang rendah. Hal-hal yang dilakukan dalam langkah ini adalah sebagai berikut:

- a. Kalikan setiap nilai pada kolom pertama dengan prioritas relative elemen pertama, nilai pada kolom kedua dengan prioritas relative elemen kedua dan seterusnya.
- b. Jumlahkan setiap baris. Hasil dari penjumlahan baris dibagi dengan elemen prioritas relative yang bersangkutan.
- c. Jumlahkan hasil bagi di atas dengan banyaknya elemen yang ada,

Hasilnya disebut λ maks.

5. Hitung Consistency Index (CI) dengan rumus:

$$CI = (\lambda_{\max} - n) / n$$

Keterangan :

n = banyaknya elemen.

6. Hitung Rasio Konsistensi / Consistency Ratio (CR) dengan rumus: $CR = CI / RC$

Keterangan :

$CR = \text{Consistency Ratio}$

$CI = \text{Consistency Index}$

$IR = \text{Index Random Consistency}$

7. Memeriksa konsistensi hierarki. Jika nilainya lebih dari 10%, maka penilaian data *judgement* harus diperbaiki. Namun jika Rasio Konsistensi (**CI/CR**) kurang atau sama dengan 0,1, maka hasil perhitungan bisa dinyatakan benar. Daftar Indeks Random Konsistensi (IR) yang nilainya dapat dilihat pada table di bawah ini:

Tabel 2.2. Daftar Indeks Random Konsistensi

Ukuran Matriks	NilaiIR
1,2	0.00
3	0.58
4	0.90
5	1.12
6	1.24
7	1.32
8	1.41
9	1.45
10	1.49
11	1.51
12	1.48
13	1.56
14	1.57
15	1.59

b. Alat-Alat Dalam Pengembangan Sistem

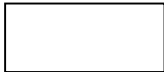
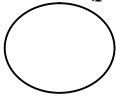
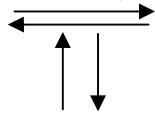
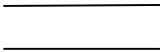
Alat yang digunakan dalam proses pengembangan system antara lain *Data Flow Diagram* (DFD), *Entity Relationship Diagram* (ERD) dan bagan alir dokumen (*flowchart*).

i. *Data Flow Diagram (DFD)*

Data Flow Diagram (DFD) atau dalam bahasa Indonesia menjadi Diagram Alir Data (DAD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*Output*). DFD dapat digunakan untuk

merepresentasikan sebuah sistem atau perangkat lunak pada beberapa level yang lebih detail untuk merepresentasikan aliran informasi atau fungsi yang lebih detail. DFD menyediakan mekanisme untuk pemodelan fungsional ataupun pemodelan aliran informasi. Oleh karena itu, DFD lebih sesuai digunakan untuk memodelkan fungsi-fungsi perangkat lunak yang akan diimplementasikan menggunakan program-program terstruktur membagi-bagi bagiannya dengan fungsi-fungsi dan prosedur-prosedur. (Rosa A.S M. Shalahuddin, 2014). Simbol-simbol DFD disajikan pada Tabel 2.3.

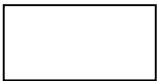
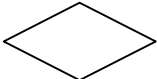
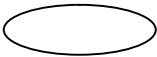
Tabel 2.3.Simbol-simbol *Data Flow Diagram (DFD)*

Simbol	Keterangan
External Entity 	Merupakan sumber atau tujuan dari aliran data dari atau ke sistem
Proses (process) 	Merupakan kegiatan atau kerja yang dilakukan oleh orang, mesin atau komputer dari hasil suatu arus data yang masuk ke dalam proses untuk dihasilkan arus data yang akan keluar dari proses.
Arus data (data flow) 	Menggambarkan arus data
Simpanan data (data store) 	Merupakan komponen yang berfungsi untuk menyimpan data atau file

ii. Entity Relationship Diagram (ERD)

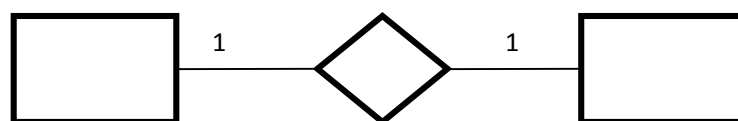
Entity Relationship Diagram (ERD) merupakan cara penggambaran entitas beserta hubungannya dengan entitas lain dan dengan disertakan atribut setiap entitas di dalamnya (Shalahuddin, 2011). Pada Tabel 2.4 disajikan simbol-simbol ERD.

Tabel 2.4. Simbol-simbol *Entity Relationship Diagram (ERD)*

Simbol	Keterangan
	Entitas merupakan individu yang mewakili sesuatu yang nyata eksistensinya dan dapat dibedakan dari sesuatu yang lain.
	Relasi menunjukkan adanya hubungan antara entitas yang berbeda.
	Atribut menggambarkan properti yang menjelaskan criteria dari entitas.
	<i>Link</i> merupakan garis untuk menandakan hubungan.

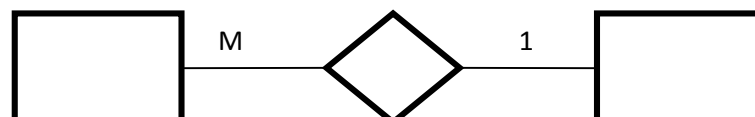
Relasi atau *relationships* memiliki tiga *type* yaitu :

- 1) *One to One*, satu *entity* hanya berelasi hanya dengan satu *entity* lain, *relationships One to One* disajikan pada Gambar II.2.



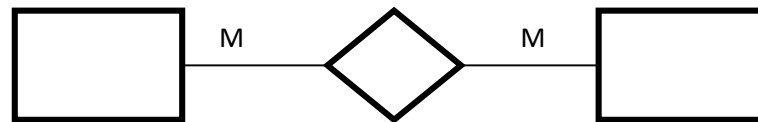
Gambar 2.2. *Relationships One to One*

- 2) *One to Many*, satu *entity* berelasi dengan lebih dari satu *entity*, *relationships One to Many* disajikan pada Gambar 2.3.



Gambar 2.3. *Relationships One to Many*

- 3) *Many to Many*, banyak *entity* bias berelasi dengan banyak *entity* lainnya, *relationships Many to Many* disajikan pada Gambar 2.4.



Gambar 2.4. *Relationships Many to Many*

iii. Bagan Alir (*Flowchart*)

Bagan alir (*flowchart*) adalah suatu bagan yang menggambarkan arus logika dari data yang akan diproses dalam suatu program atau sistem dari awal sampai akhir. Bagan alir merupakan alat yang berguna bagi *programmer* untuk mempersiapkan program yang rumit (Shalahuddin, 2011). Pada Tabel 3 disajikan simbol-simbol *Flowchart*.

Tabel 2.5. Simbol-simbol *Flowchart*

Simbol	Keterangan
	Simbol terminal (<i>Terminal Symbol</i>) digunakan untuk menunjukkan awal dan akhir program.
	Simbol pengolahan (<i>Processing Symbol</i>) digunakan untuk pengolahan aritmatika dan pemindahan data.
	Simbol keputusan (<i>Decision Symbol</i>) digunakan untuk mewakili operasi perbandingan logika.
	Simbol proses terdefinisi (<i>Predefined Process Symbol</i>) digunakan untuk proses yang detailnya dijelaskan terpisah.
	Simbol <i>input/output</i> , menyatakan proses input atau output tanpa tergantung jenis peralatannya
	Simbol Arus/ <i>Flow</i> , menyatakan prosedur yang dilakukan dari atas kebawah, bawah keatas, dari kiri kekanan, dari kanan kekiri

c. *Microsoft SQL Server*

Menurut Nugroho (2013:26) MySQL adalah software atau program Database Server. Sedangkan SQL adalah bahasa pemrogramannya, bahasa permintaan (query) dalam database server termasuk dalam MySQL itu sendiri. SQL juga dipakai dalam *software database server* lain, seperti *SQL Server*, *Oracle*, *PostgreSQL* dan lainnya.

Menurut Nugroho (2013:29) adapun perintah yang terdapat pada MySQL adalah sebagai berikut :

Tabel 2.4 Perintah Dasar MySQL

Perintah	Keterangan
Show databases	Perintah ini digunakan untuk menampilkan atau melihat daftar database yang sudah ada (sudah dibuat)
Use	Perintah ini digunakan untuk masuk atau mengakses database yang sudah ada
Show tables	Perintah ini digunakan untuk melihat atau menampilkan semua tabel yang ada di dalam database aktif (yang sudah dibuka, sudah di use)
Desc/describe	Perintah ini digunakan untuk melihat struktur tabel
Quit	Perintah ini digunakan untuk keluar MySQL Server

(Sumber: Nugroho (2013:29))

d. *Visual Basic.net*

Visual basic.net merupakan sebuah bahasa pemrograman yang menawarkan *Intergrated Development Environment* (IDE) visual untuk membuat program perangkat lunak berbasis operasi *Microsoft Windows* menggunakan model pemrograman (COM). *Visual Basic* merupakan turunan bahasa pemrograman

BASIC dan menawarkan pengembangan perangkat lunak komputer berbasis grafik dengan cepat. Beberapa bahasa skrip seperti *Visual Basic for Applications* (VBA) dan *Visual Basic Scripting Edition* (VB-Script), mirip seperti halnya Visual Basic, tetapi cara kerjanya yang berbeda. (Stefano, 2014).

e. Microsoft Visio 2007

Microsoft Visio (atau sering disebut Visio) adalah sebuah program aplikasi komputer yang sering digunakan untuk membuat diagram, diagram alir (*flowchart*), *brainstorm*, dan skema jaringan yang dirilis oleh *Microsoft Corporation*. Aplikasi ini menggunakan grafik vektor untuk membuat diagram-diagramnya. Visio aslinya bukanlah buatan *Microsoft Corporation*, melainkan buatan *Visio Corporation*, yang diakuisisi oleh *Microsoft* pada tahun 2000. Versi yang telah menggunakan nama *Microsoft Visio* adalah *Visio 2002*, *Visio 2003*, *Visio 2007*, *Visio 2013*, dan *Visio 2016* yang merupakan versi terbaru (Rosa A.S M. Shalahuddin, 2014).

f. Black Box Testing

Pengujian menggunakan sekumpulan aktifitas validasi, dengan pendekatan *black box testing*. Menurut Shalahuddin dan Rosa (2011), *blackbox testing* adalah menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan. Pengujian kotak hitam dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai dengan spesifikasi yang dibutuhkan. Kasus uji yang dibuat untuk melakukan pengujian black box testing harus dibuat dengan kasus benar dan kasus salah. Menurut Pressman (2010), *black box testing* juga disebut pengujian tingkah laku, memusat pada kebutuhan fungsional perangkat lunak. Teknik pengujian black box memungkinkan memperoleh serangkaian kondisi masukan yang sepenuhnya

menggunakan semua persyaratan fungsional untuk suatu program. Beberapa jenis kesalahan yang dapat diidentifikasi adalah fungsi tidak benar atau hilang, kesalahan antarmuka, kesalahan pada struktur data (pengaksesan basis data), kesalahan performansi, kesalahan inisialisasi dan akhir program.

1. *Equivalence Partitioning*

Equivalence Partitioning merupakan metode *black box testing* yang membagi domain masukan dari program kedalam kelas-kelas sehingga *test case* dapat diperoleh. *Equivalence Partitioning* berusaha untuk mendefinisikan kasus uji yang menemukan sejumlah jenis kesalahan, dan mengurangi jumlah kasus uji yang harus dibuat. Kasus uji yang didesain untuk *Equivalence Partitioning* berdasarkan pada evaluasi dari kelas ekuivalensi untuk kondisi masukan yang menggambarkan kumpulan keadaan yang valid atau tidak. Kondisi masukan dapat berupa spesifikasi nilai numerik, kisaran nilai, kumpulan nilai yang berhubungan atau kondisi *Boolean*.

Kesetaraan kelas dapat didefinisikan menurut panduan berikut:

1. Jika masukan kondisi menentukan kisaran, satu sah dan dua diartikan tidak valid kesetaraan kelas.
2. Jika masukan membutuhkan nilai, kondisi tertentu satu sah dan dua tidak valid kesetaraan kelas diartikan.
3. Jika masukan kondisi menentukan anggota dari set, satu sah dan satu tidak valid kesetaraan kelas diartikan.
4. Jika kondisi yang input, *boolean* satu sah dan satu tidak valid kelas diartikan.

Sebagai contoh, pemeliharaan data untuk aplikasi bank yang sudah diotomatisasikan. Pemakai dapat memutar nomor telepon bank dengan menggunakan mikro computer yang terhubung dengan *password* yang telah ditentukan dan diikuti dengan perintah-perintah. Data yang diterima adalah :

1. Kode Area : kosong atau 3 digit
2. Prefix : 3 digit atau tidak diawali 0 atau 1
3. Suffix : 4 digit

4. Password : 6 digital fanumerik

5. Perintah : *check, deposit,dll*

Selanjutnya kondisi masukan digabungkan dengan masing-masing data elemen, dapat ditentukan sebagai berikut :

1. Kode Area: kondisi masukan, *Boolean*-kode area mungkin ada atau tidak. Kondisi masukan, kisaran nilai ditentukan antara 200-999.
2. Prefix : kondisi masukan kisaran lebih besar 200 atau tidak diawali 0 atau 1
3. Suffix : kondisi masukan nilai 4 digit.
4. Password : kondisi masukan *Boolean*-pw mungkin diperlukan atau tidak. Kondisi masukan nilai dengan 6 karakter string.
5. Perintah : kondisi masukan diatur dengan berisi perintah-perintah yang telah didefinisikan.

Menerapkan pedoman untuk derivasi kelas kesetaraan, uji kasus untuk setiap masukan domain item data dapat dikembangkan dan dilaksanakan. Uji kasus dipilih sehingga jumlah terbesar dari atribut dari kelas kesetaraan tersebut dilakukan sekaligus.

Beberapa kata kunci dalam pengujian perangkat lunak yang dapat diperhatikan, yaitu:

1. Dinamis

Pengujian perangkat lunak dilakukan pada masukan yang bervariasi. Masukan ini ditentukan sebelum pengujian dilakukan dengan batasan yang disesuaikan dengan kemampuan perangkat lunak. Masukan tidak harus sesuatu yang dimungkinkan terjadi pada penggunaan program lebih lanjut, melainkan meliputi keseluruhan batasan yang dapat dijangkau perangkat lunak dan dilakukan pemercontohan (*sampling*) secara acak untuk proses pengujian.

2. Terbatas

Meskipun pengujian dilakukan pada perangkat lunak sederhana sehingga rumit sekalipun, pengujian dilakukan dengan memenuhi batasan-batasan tertentu sesuai dengan kemampuan program. Batasan ini juga

diberlakukan pada masukan-masukan yang dipilih untuk pengujian. Tidak semua kemungkinan masukan diujikan pada perangkat lunak karena akan memakan waktu yang cukup panjang mengingat begitu banyaknya kemungkinan yang bias terjadi. Untuk mengatasi hal ini, pemilihan masukan-masukan pada proses pengujian secara acak yang diperkirakan mampu memenuhi kebutuhan pengujian perangkat lunak akan dilakukan.

3. Tertentu

Pengujian dilakukan dengan batasan tertentu disesuaikan dengan harapan pada fungsi, respon, dan karakteristik perangkat lunak tersebut. Batasan tersebut akan disesuaikan dengan teknik-teknik pengujian yang ada. Pemilihan criteria pengujian yang paling tepat merupakan hal yang kompleks. Dalam praktiknya, analisis risiko pengujian dan pengalaman terhadap pengujian-pengujian sejenis akan diperlukan.

4. Harapan

Kata kunci ini memiliki keadaan-keadaan yang diharapkan, baik berupa respon sistem terhadap masukan maupun karakteristik responnya. Dalam hal ini, batasan-batasan hasil pengujian yang diharapkan harus ditentukan. Dengan demikian, dapat diketahui apakah perangkat lunak tersebut telah memenuhi hasil pengujian yang diharapkan atau memerlukan pembenahan kembali, baik berupa perbaikan maupun pengembangan perangkat lunak.