

## BAB II

### TINJAUAN PUSTAKA

#### 2.1 Interaksi Manusia dan Komputer

Interaksi Manusia dan Komputer adalah disiplin ilmu yang berhubungan dengan perancangan, evaluasi dan implementasi dalam sistem komputer interaktif untuk digunakan oleh manusia, serta studi fenomena – fenomena besar yang berhubungan dengannya. Terdapat delapan aturan emas dalam membuat suatu tampilan *user interface*. [3] Delapan *Golden Rules* yang dapat menjadi pedoman bagi perancangan antarmuka aplikasi yang baik yaitu [4]:

1. *Strive for consistency*

Konsistensi dibutuhkan antar halaman dalam satu aplikasi ataupun antara aplikasi yang masih berhubungan. Gunanya adalah supaya *user*, terutama *novice user*, tetap dapat mengenali halaman yang dilihat masih dalam lingkup atau masih memiliki hubungan dengan aplikasi yang digunakan. Dengan demikian akan membuat *user* nyaman dalam mengeksplorasi aplikasi tanpa takut berpindah aplikasi.

2. *Cater to universal usability*

Dalam merancang antarmuka aplikasi, seorang *interface designer* harus memperhitungkan jenis variasi *user* nya. Baik itu dari segi latar belakang budaya dan bahasa, juga variasi tingkat pemahaman *user* terhadap aplikasi. Pada poin ini yang lebih sering dipikirkan adalah perbedaan kebudayaan *user*, sehingga aplikasi harus dirancang dalam berbagai macam bahasa. Tidak harus demikian, tetapi lebih efektif jika *universal usability* diterapkan pada variasi tingkat pemahaman *user* terhadap aplikasi. *User* yang baru menggunakan aplikasi, atau *user* yang sudah terbiasa menggunakan aplikasi, tentu memiliki preferensi antarmuka yang berbeda, misalnya ada *shortcut* untuk suatu fungsi tertentu bagi *user* yang sudah sering menggunakan aplikasi, sehingga dapat lebih memudahkan *user* untuk menggunakan fungsi tersebut.

### 3. *Offer informative feedback*

*Informative feedback* tidak harus selalu dengan jawaban dari aplikasi ke *user*, tetapi dapat berupa perubahan antarmuka setiap *user* melakukan aksi, dengan demikian *user* paham bahwa aksinya sudah direspon oleh aplikasi.

### 4. *Design dialogs to yield closure*

Untuk poin ini sebenarnya termasuk dari bagian *informative feedback*, dengan menyampaikan bahwa proses yang dijalankan oleh *user* sudah selesai, *user* paham bahwa dia tidak perlu menunggu apakah masih akan ada tahapan lain setelah menyelesaikan suatu proses.

### 5. *Prevent errors*

Poin ini untuk menjaga agar *user* tidak melakukan kesalahan dalam menjalankan proses. Sangat diperlukan supaya *user* tidak merasa jenuh dalam mencoba menggunakan aplikasi, karena tidak bisa menemukan format atau aksi yang tepat pada saat mencoba suatu fungsi. Biasanya berupa petunjuk pengisian formulir sesuai format yang diterima oleh aplikasi, sehingga *user* dapat mengisi formulir dengan tepat pada percobaan pertama.

### 6. *Permit easy reversal of actions*

Poin ini merupakan salah satu poin yang cukup penting untuk menunjang UX dari suatu aplikasi. Biasanya yang dianggap sebagai pemenuhan poin ini adalah tombol *back*. Namun sebenarnya, tombol *back* hanyalah untuk kembali kehalaman sebelumnya, namun belum tentu membatalkan aksi. Contoh paling nyata dari poin ini adalah pada aplikasi *online shop*, *user* dapat mencoba melakukan pembelian, tetapi pada saat selesai memilih barang, *user* dapat melakukan pembatalan barang yang ingin dibeli. Dengan demikian *user* merasa nyaman saat mencoba untuk melakukan eksplorasi pada aplikasi, karena barang yang dicoba untuk dibeli tidak langsung terproses beli, tetapi *user* dapat melakukan *cancel delete* barang yang sudah dipilih.

### 7. *Support internal locus of control*

Poin ini terutama sangat disukai oleh *user* yang sudah terbiasa menggunakan aplikasi, karena biasanya *user* ingin memiliki tampilan yang bisa diatur oleh *user* sendiri sesuai preferensi dari *user*. Hal ini dapat meningkatkan kepuasan *user* terhadap aplikasi yang sangat mempengaruhi UX terhadap aplikasi tersebut.

#### 8. *Reduce short-term memory load*

Pada poin ini biasanya orang lebih memusatkan pada desain tata letak menu dan tombol. Tetapi sebenarnya akan lebih efektif jika diterapkan pada proses saat *user* harus memberikan *input* ke sistem. Dengan menerapkan poin ini, maka *user* tidak perlu mengingat data yang harus di-input ke sistem. Karena data yang harus di-input, sudah disediakan oleh sistem. Misalnya pada *online shop*, *user* memilih barang yang diinginkan, dan dari situ sistem langsung menerima input kode barang yang diinginkan oleh *user* untuk diproses pada proses pembayaran nantinya.

Demikian ulasan delapan *Golden Rules* dari Ben Shneiderman, dapat dilihat bahwa 8 *Golden Rules* ini masih sangat dapat diaplikasikan baik pada aplikasi berbasis *desktop* maupun berbasis *mobile*, karena dengan 8 *Golden Rules* pun dapat meningkatkan tingkat kepuasan *user* terhadap aplikasi yang sangat mempengaruhi UX yang merupakan suatu tolak ukur keberhasilan suatu aplikasi *mobile*. [4]

*Human Computer Interaction* yaitu studi yang mempelajari hubungan interaksi antara manusia, komputer dan penugasan. Prinsipnya bagaimana manusia dan komputer dapat secara interaktif menyelesaikan penugasan dan bagaimana sistem yang interaktif tersebut dapat dibuat. Adapun pemahaman lain terhadap HCI dimana ketika membangun sebuah sistem informasi, seorang desainer atau pengembang sistem harus “memperhatikan faktor interaksi manusia dan komputer karena sistem informasi yang dibuat oleh manusia dan tujuannya untuk manusia”. Kunci utama HCI adalah daya guna (*usability*). *Usability* adalah tingkat produk yang dapat digunakan dan telah ditetapkan oleh pengguna, untuk mencapai tujuan secara efektif, efisien dan memuaskan dalam menggunakannya. HCI memfokuskan desain sistem pada pengguna atau biasa disebut dengan *user center design* (UCD). UCD yaitu filosofi perancangan yang menempatkan pengguna sebagai pusat dari proses pengembangan sistem. Dan prinsip-prinsip dalam merancang *user interface* adalah sebagai berikut [5]:

- a. *User Familiarity* / Mudah dikenali
- b. *Consistency* / Konsistensi
- c. *Minimal Surprise* / Tidak membuat pengguna terkejut
- d. *Recoverability* / Pemulihan

e. *User guidance* / Bantuan

Dengan memperhatikan aspek kebutuhan pengguna maka dapat diciptakan suatu sistem yang sesuai dan tepat bagi pengguna. Sistem yang tepat bagi pengguna akan memberikan kenyamanan kepada pengguna didalam menggunakan sistem, dengan demikian tujuan penerapan sistem akan dapat dicapai dan tidak akan mengalami kegagalan. Agar peran *interface* sebagai jembatan antara sistem dengan penugasan oleh pengguna berjalan dengan baik.[5]

## 2.2 User Interface (UI)

*User interface* merupakan bagian dari komputer dan perangkat lunak yang dapat dilihat, didengar, disentuh, dan diajak bicara, baik secara langsung maupun dengan proses pemahaman tertentu. Antar muka yang baik merupakan jendela untuk melihat kemampuan sistem serta jembatan bagi kemampuan perangkat lunak dalam membantu memenuhi kebutuhan *user* tidak terkecuali *user* yang mengalami cacat (*disability*).[6]

Sistem informasi memiliki desain *interface* yang berbeda-beda sesuai dengan fungsi dan kebutuhan penggunanya. *User interface* yang bagus atau menarik adalah kemudahan pengguna dalam menggunakannya. Prinsip dalam membuat desain antar muka (*user interface*) sangat penting karena jika terlalu memusingkan, pengguna harus belajar lagi cara menggunakannya dan akan membuat mereka merasa kesulitan menggunakan sistem tersebut. Sebagai bagian dari pengembangan suatu sistem yang memenuhi standart HCI, evaluasi pada kemudahan penggunaan (*usability*) harus dilakukan. *Usability* sebagai suatu pengalaman pengguna dalam berinteraksi dengan aplikasi atau suatu situs website sampai pengguna dapat mengoperasikannya dengan mudah dan cepat.[7]

Ada beberapa metode yang pernah digunakan untuk mengevaluasi *user interface* seperti *Technology Acceptance Model* (TAM), *system usability scale* (SUS), GOMS, terdapat pula metode evaluasi *user interface* lain yang dapat mengetahui tingkat *user interface* pada sebuah sistem yaitu dengan metode *User Centered Systems Design*. Dalam metode *User Centered Systems Design* ini, evaluasi sistem terpusat pada *interface* menurut cara pandang *user*. *User* merupakan object yang penting didalam pengembangan dan pembangunan sistem yang baik.

Kemudahan penggunaan (*usability*) merupakan isu yang krusial dalam HCI, karena hal itu menjadi aspek penting untuk menilai kualitas dari antarmuka (*interface*) pengguna.[7]

*User interface*, berfungsi untuk menginput pengetahuan baru ke dalam basis pengetahuan sistem pakar (ES), menampilkan penjelasan sistem dan berikan panduan pemakaian sistem secara menyeluruh *step by step* sehingga *user* mengerti apa yang dilakukan terhadap suatu sistem. Yang terpenting dalam membangun *user interface* adalah kemudahan dalam memakai atau menjalankan sistem, interaktif, komunikatif, sedangkan kesulitan dalam mengembangkan atau membangun suatu program jangan terlalu diperlihatkan.[8]

*User interface* berfungsi untuk menghubungkan atau penterjemah informasi antar pengguna dengan sistem operasi, sehingga komputer dapat digunakan. Dengan demikian, *user interface* juga bisa diartikan sebagai mekanisme inter-relasi atau integrasi total dari perangkat keras dan lunak yang membentuk pengalaman berkomputer. *User interface* dari sisi *software* bisa berbentuk *Graphical User Interface (GUI)* atau *Command Line Interface (CLI)*, sedangkan dari sisi *hardware* bisa berbentuk *Apple Desktop Bus (ADB)*, *USB* dan *fireware*.[9]

Adapun tujuan *interface* adalah sebagai berikut:

1. Memudahkan input data masukan oleh user.
2. Membuat program menjadi menarik.
3. Menampilkan hasil pemrosesan data dalam program.
4. Memudahkan pembacaan data yang dihasilkan oleh program.

### 2.3 User Experience (UX)

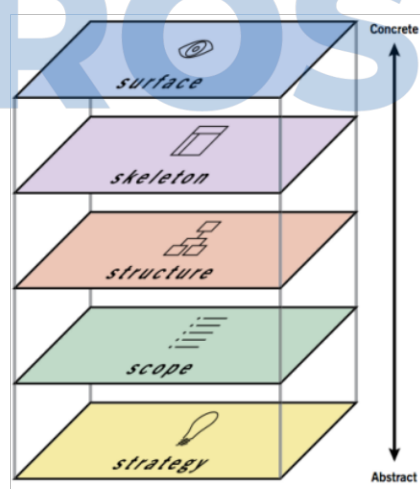
*User Experience* merupakan kualitas pengalaman seseorang ketika berinteraksi dengan desain tertentu. *User experience* menilai kepuasan dan kenyamanan seseorang terhadap sebuah produk, sistem, dan jasa. Seberapa pun bagus fitur sebuah produk, sistem, atau jasa, tanpa khalayak yang dituju dapat merasakan kepuasan, kaidah dan kenyamanan dalam berinteraksi maka tingkat *user experience* menjadi rendah.[6]

*User Experience (UX)* merupakan bagaimana layanan atau produk berkerja ketika seseorang datang atau menggunakannya, bukan mengenai kinerja internalnya.

Untuk itu UX merupakan bagian yang penting dalam berbisnis. UX memiliki elemen dasar antara lain[8]:

1. *Startegy Plane*, merupakan lapisan paling bawah yang berupa startegi bisnis mendasar yang menentukan tujuan dari produk dan kebutuhan penggunaanya.
2. *Scope Plane*, merupakan translasi dari strategi untuk diteruskan dalam pembuatan spesifikasi fungsional.
3. *Structure Plane*, dibentuk arsitektur informasi yang disusun untuk memfasilitasi pemahaman pengguna.
4. *Skeleton Plane*, terdiri atas bagian, meliputi:
  - a. *Information design*, merupakan presentasi dari informasi untuk memudahkan pemahaman.
  - b. *Interface design*, merupakan pengaturan elemen antarmuka untuk memungkinkan pengguna dalam berinteraksi.
  - c. *Navigation design*, merupakan kumpulan elemen yang memungkinkan pengguna untuk menjelajahi arsitektur informasi.
5. *Surface Plane*, merupakan pengalaman sensorik kepengguna meliputi (warna, gambar, ikon) dibuat untuk menyelesaikan produk.

Kelima elemen tersebut merupakan urutan dari lapisan paling bawah (mendasar dan abstrak) hingga lapisan paling atas (visual). Lapisan tersebut digambarkan pada gambar 2.1 sebagai berikut[8]:



Gambar 1.1 Lima Elemen Dasar UX

UX yang baik diperlukan proses penelitian terhadap pengguna atau *User Research*. *User Research* adalah proses penggambaran bagaimana manusia bisa mengartikan dan menggunakan produk atau layanan.[8]

## 2.4 Usability

Secara umum *usability* mengacu pada sejauh mana *user* dapat belajar dan menggunakan suatu produk untuk mencapai tujuannya dan sejauh mana kepuasan *user* dalam menggunakan produk tersebut.[11]

Kuesioner yang dapat digunakan untuk mengukur *usability* adalah USE, terdapat beberapa aspek pengukuran *usability* yaitu efisiensi, efektivitas dan kepuasan. Beberapa penelitian yang sudah dilakukan menunjukkan bahwa kebanyakan evaluasi produk mengacu pada tiga dimensi. Hasil beberapa pengamatan juga menunjukkan adanya korelasi dan saling mempengaruhi antara parameter *ease of use* dan *usefulness*. Faktor *usefulness* biasanya kurang penting jika sistem tersebut bersifat sistem internal dimana penggunaanya bersifat wajib. Kuesioner dibuat dalam bentuk skor lima point dengan model skala *likert*, untuk pengukuran tingkat persetujuan user terhadap *statement* hasil pengukuran kemudian diolah dengan metoda statistik deskriptif dan dilakukan analisis baik terhadap masing-masing parameter atau terhadap keseluruhan parameter.[13]

USE merupakan salah satu paket kuesioner non komersial yang dapat digunakan untuk penelitian *usability* sistem. Menurut Jacob Nielsen, *usability* adalah atribut kualitas yang menjelaskan atau mengukur seberapa mudah penggunaan suatu antar muka (*interface*). Kata "*usability*" juga merujuk pada suatu metode untuk meningkatkan kemudahan pemakaian selama proses desain.[13]

*Usability* didefinisikan melalui lima komponen kualitas yaitu [11]:

- a. *Learnability* adalah mengukur semudah apa *user* dapat mempelajari cara penggunaan produk tersebut untuk pertama kali.
- b. *Efisiensy* adalah mengukur secepat apa *user* dapat melakukan tugasnya.
- c. *Memorability* adalah sejauh mana *user* mengingat langkah-langkah atau proses yang dilakukan dalam mencapai tujuannya.
- d. *Error* adalah sebanyak apa *user* melakukan *error*, dan sejauh mana akibat *error* tersebut, serta apakah mudah bagi *user* untuk mengatasi *error* tersebut.

- e. *Satisfaction* adalah bagaimana perasaan *user* ketika menggunakan produk atau tanggapan terhadap desain produk secara keseluruhan. Inti utama *usability* adalah menjawab pertanyaan, apakah produk tersebut sesuai dengan kebutuhan *user*.

#### 2.4.1 Pengukuran Usability

Mengukur *usability* berarti mengukur efektifitas, efisiensi dan kepuasan *user*.

Untuk itu dapat dilakukan dua cara yaitu [11] :

- a. Mengandalkan asumsi pembuat program atau diri sendiri
- b. Menggunakan *usability matric*.

Hasil pengukuran *usability* dapat dimanfaatkan untuk beberapa hal berikut [11]:

- a. Mendapatkan masukan dari data, lebih obyektif dari pada pendapat sendiri
- b. Dapat digunakan untuk membandingkan *usability* dua produk
- c. Dapat mengklasifikasikan masalah (jika ada)
- d. Membuat prediksi penggunaan produk yang sebenarnya
- e. Memberikan ilustrasi pada manajemen berdasarkan fakta

Saat ini, terdapat beberapa jenis metrik atau teknik pengukuran *usability*, yang secara umum dapat dibagi menjadi dua kategori yaitu [11]:

- a. *Desired Quality* merupakan pengamatan berupa ukuran selesai atau tidaknya suatu tugas (*yes/no*), atau tercapai tidaknya suatu hasil, atau diterima/tidaknya suatu pernyataan (*agree/disagree*)
- b. Pengukuran Kuantitatif adalah mengukur dalam skala angka tertentu, misalnya x% user dapat menyelesaikan tugasnya kurang dari 1 menit.

Pengukuran *usability* dapat dilakukan dengan tahapan-tahapan sebagai penelitian lainnya yaitu [11] :

- a. Pemilihan kuesioner : memilih paket kuesioner yang akan digunakan. Setiap paket kuesioner memiliki asumsi dasar tertentu, kerangka pemikiran dan pendekatan yang berbeda-beda
- b. Memilih partisipan : menentukan partisipan yang representatif, membagi berdasarkan kelompok seperti umur, jenis kelamin dan lain – lain



- c. Menentukan ukuran sampel : menentukan ukuran partisipan yang representatif untuk dijadikan obyek pengumpulan data
- d. Mengolah dan interpretasi data sesuai dengan karakteristik data penelitian.

#### 2.4.2 Jenis Kuesioner untuk mengukur Usability

Pada umumnya, pengukuran *usability* dilakukan menggunakan serangkaian kuesioner. Pada saat ini terdapat beberapa jenis kuesioner yang dapat digunakan untuk mengukur *usability* seperti [11]:

- a. *System usability scale* (SUS) yang ditawarkan secara komersial dalam bentuk paket
- b. *Post-Study System Usability Questionnaire* (PSSUQ) merupakan paket kuesioner yang dirilis oleh IBM yang terdiri atas 19 item instrumen pengukuran
- c. WAMMI dan SUPR-Q untuk mengukur *website*
- d. *Single Ease Question* (SEQ) yang terdiri dari satu pertanyaan singkat
- e. USE (*Usefulness Satisfaction, and Ease of use*) serta beberapa paket kuesioner lainnya.

#### 2.5 System Usability Scale (SUS)

*System usability scale* (SUS) merupakan kuesioner yang dapat digunakan untuk mengukur *usability* sistem komputer menurut sudut pandang subjektif pengguna. SUS dikembangkan oleh Jhon Brooke sejak 1986. Hingga saat ini, SUS banyak digunakan untuk mengukur *usability* dan menunjukkan beberapa keunggulan, antara lain[12]:

1. SUS dapat digunakan dengan mudah, karna hasilnya berupa skor 0 – 100;
2. SUS sangat mudah digunakan, tidak membutuhkan perhitungan yang rumit;
3. SUS tersedia secara gratis, tidak membutuhkan biaya tambahan;
4. SUS terbukti *valid* dan *reliable*, walau dengan ukuran sampel yang kecil.

Tabel 1.1 Item Pernyataan System Usability Scale

Kode	Item Pernyataan
R1	Saya akan sering menggunakan/ mengunjungi situs ini
R2	Saya menilai situs ini terlalu kompleks (memuat banyak hal yang tidak

	perlu)
R3	Saya menilai situs ini mudah dijelajahi
R4	Saya membutuhkan bantuan teknis untuk menggunakan/ menjelajahi situs ini
R5	Saya menilai fungsi/ fitur yang disediakan pada situs ini dirancang dan disiapkan dengan baik
R6	Saya menilai terlalu banyak inkonsistensi pada situs ini
R7	Saya merasa kebanyakan orang akan mudah menggunakan/ menjelajahi situs ini dengan cepat
R8	Saya menilai situs ini sangat rumit untuk dijelajahi
R9	Saya merasa sangat percaya diri menjelajahi situs ini
R10	Saya perlu belajar banyak hal sebelum saya dapat menjelajahi situs ini dengan baik

Kuesioner SUS menggunakan 5 poin skala *Likert*. Responden diminta untuk memberikan penilaian “Sangat tidak setuju”, “Tidak setuju”, ”Netral”, ”Setuju”, dan “Sangat setuju” atas 10 item pernyataan SUS sesuai dengan penilaian subyektif. Jika responden merasa tidak menemukan skala respon yang tepat, responden harus mengisi titik tengah skala pengujian. Setiap item pernyataan memilih skor kontribusi. Setiap skor kontribusi item akan berkisar antara 0 - 4. Untuk item 1, 3, 5, 7, dan 9 skor kontribusinya adalah posisi skala dikurangi 1. Untuk item 2, 4, 6, 8 dan 10, skor kontribusinya adalah 5 dikurangi posisi skala. Kalikan jumlah skor kontribusi dengan 2.5 untuk mendapatkan nilai keseluruhan *system usability*. Skor SUS berkisar dari 0 - 100.[12] Berdasarkan skor akhir SUS tersebut akan bisa diketahui seberapa tinggi tingkat *usability* dan akseptabilitas (*acceptable*) desain sistem aplikasi yang dikembangkan. Penilaiannya berdasarkan tiga kategori yaitu *Not Acceptable* dengan rentang skor SUS 0 - 50.9, *Marginal* 51 - 70.9, dan *Acceptable* 71 - 100. Responden diminta menjawab semua butir pertanyaan yang diisi setelah pengguna selesai menggunakan sistem secara keseluruhan.[16] Berikut rumus perhitungan skor SUS[12]:

$$\begin{aligned} \text{Skor SUS} = & ((R1-1)+(5-R2)+(R3-1)+ \\ & (5-R4)+(R5-1)+(5-R6)+ \\ & (R7-1)+(5-R8)+(R9-1)+ \\ & (5-R10))*2.5 \end{aligned}$$

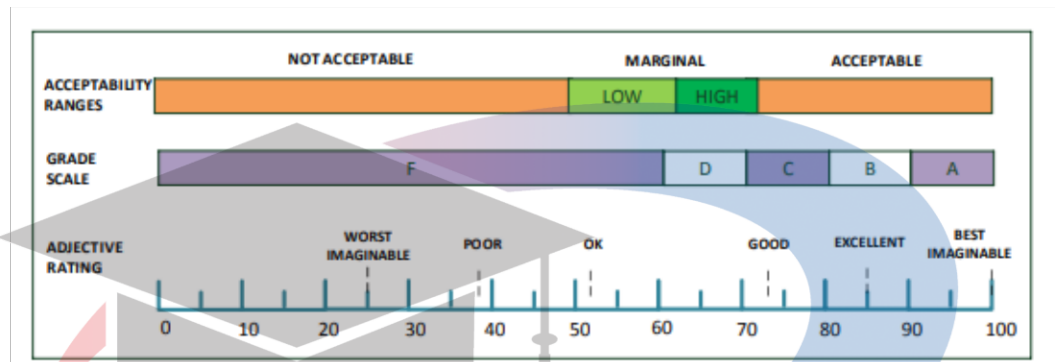
Hasil kuesioner kemudian dihitung dengan rumus yang telah ditentukan untuk mendapat skor SUS. Contoh Hasil penilaian skor SUS di tampilkan pada gambar 2.2 sebagai berikut[12]:

Responden	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	Skor SUS
1	5	2	3	4	5	4	3	2	2	4	55
2	2	3	2	2	3	3	2	4	3	2	45
3	3	2	4	1	3	3	4	1	2	3	65
4	3	3	1	2	4	3	2	2	5	2	57,5
5	3	4	2	3	2	3	2	4	3	4	35
6	5	2	5	2	4	2	4	2	4	2	80
7	4	2	4	1	4	2	5	1	5	3	82,5
8	2	3	3	4	4	3	3	2	2	4	45
9	3	1	2	1	1	3	1	4	3	2	47,5
10	4	5	3	5	3	4	5	5	5	4	42,5
11	3	2	4	1	2	2	4	2	3	2	67,5
12	1	3	3	3	1	2	1	2	1	4	32,5
13	5	3	4	2	4	5	4	3	4	3	62,5
14	1	5	5	2	5	3	2	1	5	2	62,5
15	5	3	5	3	5	2	5	5	5	5	67,5
16	4	3	4	3	4	3	4	3	4	3	62,5
17	5	2	5	4	5	2	3	4	5	2	72,5
18	3	2	3	1	4	2	4	1	4	1	77,5
19	4	2	4	1	4	2	4	1	4	1	82,5
20	4	2	5	5	3	4	4	2	4	4	57,5
21	5	3	3	2	3	3	3	3	3	2	60
22	3	2	4	1	3	3	4	1	2	3	65
23	4	2	4	1	4	2	4	2	4	2	77,5
24	3	3	2	2	2	2	3	3	2	2	50
25	4	3	5	1	4	2	4	2	4	1	80
26	5	4	5	2	4	4	4	2	4	2	70
27	3	4	2	3	5	3	2	4	3	4	42,5
28	2	2	2	1	3	1	3	1	4	1	70
29	3	3	4	1	3	4	2	1	4	1	65
30	2	2	4	2	3	2	3	2	2	2	60
<b>Rata-rata SKOR SUS</b>											<b>61,33</b>

Gambar 2.2 Contoh Hasil Perhitungan Skor SUS

Setelah mendapatkan hasil akhir penilaian responden maka selanjutnya adalah menentukan *grade* hasil penilaian. Untuk menentukan *grade* hasil penilaian ada 2 (dua) cara yang dapat digunakan. Penentuan pertama dilihat dari sisi tingkat

penerimaan pengguna, *grade* skala dan *adjektif rating* yang terdiri dari tingkat penerimaan pengguna terdapat tiga kategori yaitu *not acceptable*, *marginal* dan *acceptable*. Sedangkan dari sisi tingkat *grade* skala terdapat enam skala yaitu A, B, C, D, E dan F. dan dari *adjektif rating* terdiri dari *worst imaginable*, *poor*, *ok*, *good*, *excellent* dan *best imaginable* seperti yang diperlihatkan pada Gambar 2.3.[2]



Gambar 3.3 Penentuan Hasil Penilaian (Bangor, Kortum, Miller, 2009)

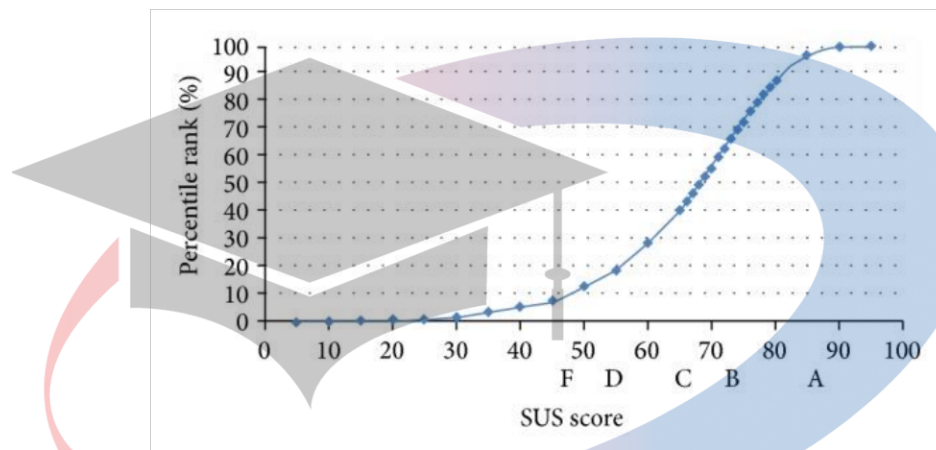
Penentuan yang kedua dilihat dari sisi *percentile range* (SUS skor) yang memiliki *grade* penilaian yang terdiri dari A, B, C, D dan F. Dari dua cara penentuan hasil penilaian tersebut maka dapat dilihat hasil penilaian sebagai berikut[2]:

1. *Acceptability, Grade Scale, Adjective Rating* : Penentuan *Acceptability, grade scale, adjective rating* digunakan untuk melihat sejauh mana *perspective* pengguna terhadap perangkat lunak. Untuk menentukan *Acceptability, grade scale, adjective rating* maka dilakukan perbandingan hasil penilaian rata - rata responden dengan ketentuan penilaian seperti Gambar 2.3.
2. *SUS Skor percentile rank* : Penentuan hasil penilaian dengan cara *SUS skor percentile rank* memiliki perbedaan dengan cara penilaian *Acceptability, grade scale, adjective rating*. Perbedaan yang terjadi pada kategori penilaian, pada *SUS skor percentile rank* dilakukan perbandingan hasil penilaian pengguna secara umum sedangkan pada *Acceptability, grade scale, adjective rating* dibedakan kedalam tiga kategori. Untuk itu dalam menentukan *SUS skor percentile rank* seperti ketentuan sebagai berikut:
  - a. *Grade A* : dengan skor lebih besar atau sama dengan 80,3
  - b. *Grade B* : dengan skor lebih besar sama dengan 74 dan lebih kecil 80,3
  - c. *Grade C* : dengan skor lebih besar 68 dan lebih kecil 74

- d. *Grade D* : dengan skor lebih besar sama dengan 51 dan lebih kecil 68
- e. *Grade F* : dengan skor lebih kecil dari 51.

Pada penelitian ini digunakan rentang hasil penilaian sebagai berikut : [17]

- a. Tidak dapat diterima = 00 – 64.
- b. Dapat diterima = 65 – 84.
- c. Sempurna = 85 – 100.



Gambar 4.4 Kaitan Peringkat Persentil dengan Skor SUS dan Nilai Huruf (Sauro, 2011)

Grafik tersebut merupakan rujukan untuk mengetahui tingkat kegunaan dari suatu produk atau sistem. Apabila nilai hasil evaluasi berada pada skor  $\geq 65$ , maka produk atau sistem tersebut dapat diterima *usability* nya oleh pengguna atau bahkan dalam katagori sempurna. Grafik tersebut dapat digunakan untuk menafsirkan nilai hasil evaluasi menjadi peringkat persentil. [17]

SUS merupakan penilaian global aspek *usability* (efektivitas, efisiensi, dan kepuasan) secara subjektif yang dirasakan oleh pengguna. Skor SUS dapat menunjukkan tingkat penerimaan pengguna. *Acceptability ranges* menilai pada tingkat penerimaan pengguna apakah tidak dapat diterima, tingkat batas sedang terendah /tertinggi, dan dapat diterima. *Grade Scale* menilai efesiensi yang dirasakan pengguna :

- a. Berada di *Grade A* dan *B* maka pengguna sangat menyukai situs tersebut dan akan merekomendasikan kepada pengguna lainnya.
- b. Berada di *Grade C* dan *D* pengguna merasakan situs tersebut baik – baik saja dan masih dapat meningkatkan skor.

- c. Berada di *Grade F* pengguna merasakan perlu perbaikan pada situs tersebut secara cepat.

*Adjective Ranges* menilai rasa kepuasan penggunaan situs tersebut yang dirasakan oleh pemakai yang terdiri dari (*worst imaginable, poor, ok, good, excellent, best imaginable*).

Skor SUS harus bernilai lebih dari 70 agar termasuk ke dalam kategori *Acceptable*. Skor SUS dianggap *Good* apabila bernilai lebih dari 70.4. Penelitian juga menjelaskan kategori skor SUS. Untuk mendapatkan predikat A, skor SUS harus bernilai setidaknya 90. Skor SUS juga dapat menunjukkan kecenderungan untuk menjadi *Net Promoter*. Skor SUS sebesar 82 atau lebih menunjukkan pengguna berpotensi menjadi *Promoter*, sedangkan skor SUS sebesar 67 atau kurang menunjukkan pengguna berpotensi menjadi *Deductor*. [12]

Evaluasi *usability* dilakukan dengan mengukur kualitas komunikasi (interaksi) antara produk teknologi (sistem) dan pengguna teknologi tersebut. Unit pengukuran adalah perilaku pengguna dalam konteks pengguna tertentu. *Usability* mengukur bagaimana pengguna sistem dipandang oleh pengguna. Model pengukuran *usability*, menurut standart ISO 924-11 sebaiknya meliputi aspek-aspek sebagai berikut [12]:

1. Efektivitas : menunjukkan tingkat akurasi dan kesempurnaan yang dicapai pengguna saat menjalankan tugas tertentu.
2. Efisiensi : menunjukkan sumber daya yang digunakan terkait dengan akurasi dan kesempurnaan yang dicapai pengguna dalam menjalankan tugas.
3. Kepuasan : menunjukkan pengguna merasa bebas dari ketidak nyamanan dan menunjukkan perilaku positif terhadap penggunaan produk.

## 2.6 Flowchart






*Flowchart* adalah cara penulisan algoritma dengan menggunakan notasi grafis. *Flowchart* merupakan gambar atau bagan yang memperlihatkan urutan atau langkah-langkah dari suatu program dan hubungan antar proses. Gambar ini dinyatakan dengan simbol. Dengan demikian setiap *symbol* menggambarkan proses tertentu. Sedangkan antara proses digambarkan dengan garis penghubung. Dengan menggunakan *Flowchart* akan memudahkan kita untuk melakukan pemeriksaan bagian-bagian yang terlupakan dalam analisis masalah. Disamping itu *flowchart* juga



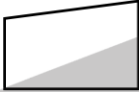
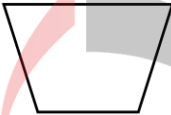




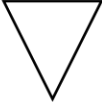
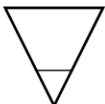
berguna sebagai fasilitas untuk berkomunikasi antara program yang bekerja dalam tim suatu proyek. *Flowchart* menolong dalam menganalisis alternatif-alternatif lain dalam pengoperasian.[14]

Pada dasarnya terdapat berbagai macam *flowchart*, diantaranya yaitu *Flowchart* Sistem (*system flowchart*), *Flowchart paper work/Flowchart* Dokumen (*Document Flowchart*), *Flowchart* Skematik (*Schematic Flowchart*), *Flowchart* program (*Program Flowchart*), *Flowchart* proses (*Process Flowchart*). Dalam pengujian ini peneliti menggunakan *flowchart system*. [14]

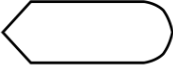
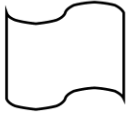


*Flowchart* sistem adalah representasi grafis dari fisik hubungan antar elemen-elemen kunci dari suatu sistem. Elemen-elemen ini dapat mencakup departemen organisasi, kegiatan manual, program manual, catatan akuntansi *hard copy* (dokumen, jurnal, buku besar, dan *file*), dan catatan digital (*file* referensi, *file* transaksi, arsip *file*, dan *file* master). *Flowchart* sistem juga menggambarkan media komputer fisik yang digunakan di sistem seperti pita magnetik, *disk* magnetik, dan terminal. *Flowchart* dalam menggambarkan alur sistem atau program menggunakan suatu gambar yang mempunyai simbol tertentu seperti dibawah ini [14] :

Tabel 2.2 Sistem Flowchart

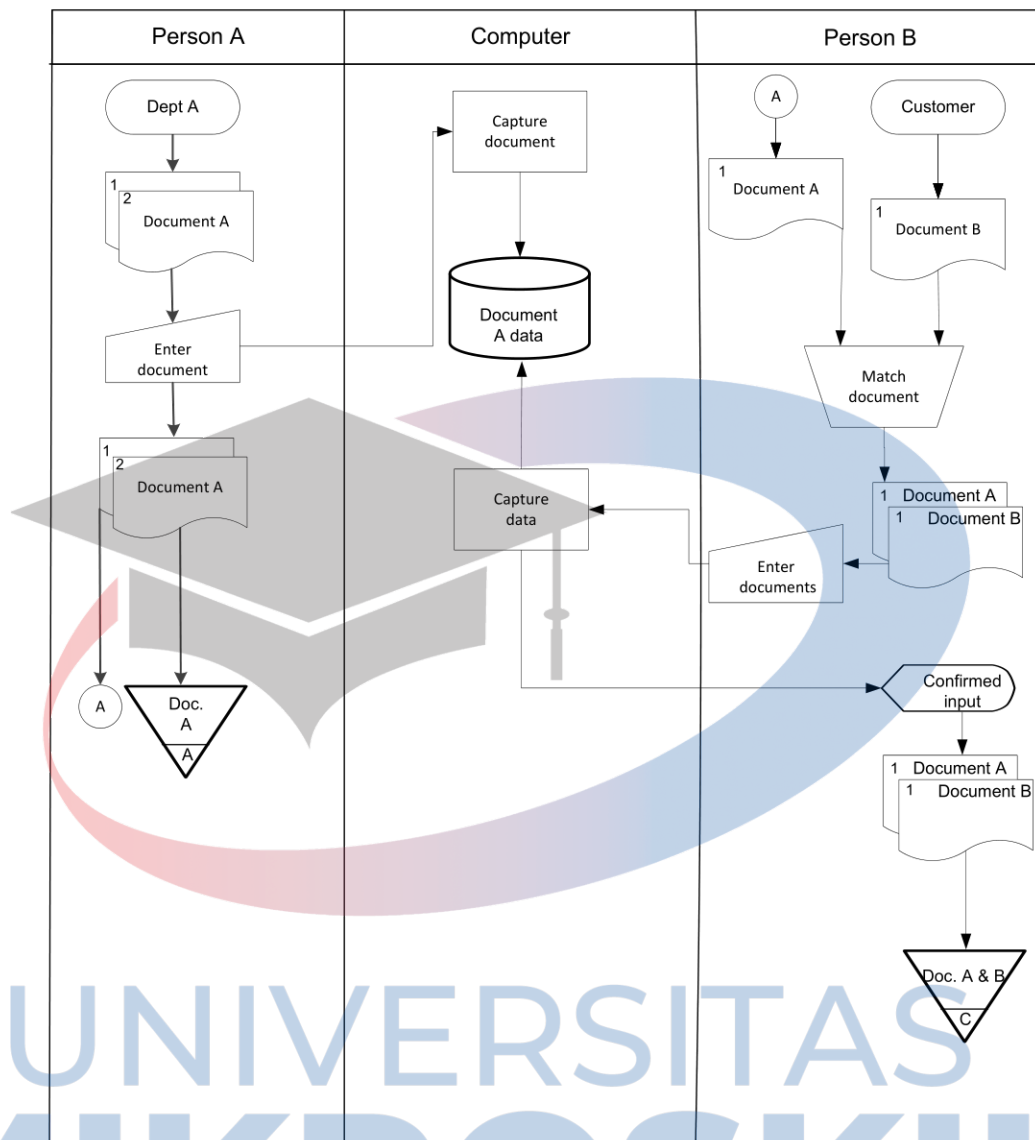
Simbol	Nama	Fungsi
	<i>Start</i> atau <i>Stop</i>	<i>Symbol</i> untuk permulaan ( <i>start</i> ) atau akhir ( <i>stop</i> ) dari suatu kegiatan
	<i>Flow Line</i>	Menunjukkan arus dari proses
	<i>Single Document</i>	Menunjukkan dokumen <i>input</i> dan <i>output</i> baik untuk proses manual, mekanik atau computer
	<i>Computer Proses</i>	Proses pengolahan data
	<i>Multiple Document</i>	Menggambarkan dokumen berserta rangkapannya atau beberapa

		dokumen
	<i>Magnetic Disk Storage</i>	Tempat menyimpan data ( <i>database</i> )
	<i>Tape drive or magnetic (pita magnetik)</i>	Menunjukkan <i>input</i> atau <i>output</i> menggunakan pita <i>magnetic</i>
	<i>Manual Input</i>	Data yang dimasukkan melalui <i>keyboard</i> komputer/ diketikkan oleh <i>user</i>
	<i>Manual Process</i>	Menunjukkan pekerjaan manual
	<i>Offline Process</i>	Menunjukkan kegiatan proses diluar proses operasi komputer.
	<i>On-page Connector</i>	Penghubung <i>flowchart</i> pada aliran yang sama
	<i>Off - page Connector</i>	Penghubung bagian - bagian <i>flowchart</i> yang berada pada halaman berbeda.
	<i>Punch card</i>	Menunjukkan <i>input/ output</i> yang menggunakan kartu plong.
	<i>Temporary paper data store</i>	Arsip <i>file</i> yang bersifat sementara
	<i>Permanent data store (simpanan offline)</i>	<i>File non - komputer</i> yang diarsip



	<i>On-screen display</i>	Menggambarkan informasi yang ditampilkan oleh perangkat <i>outputonline</i> , seperti sebuah terminal, monitor, atau layar.
	<i>General Journal or General Ledger</i>	
	<i>Annotation</i> (penjelasan)	Menunjukkan penjelasan dari suatu proses
	<i>Batch Total</i>	Penyatuan dokumen

# UNIVERSITAS MIKROSKIL



Gambar 5.5 Contoh System Flowchart

Penjelasan contoh gambar *system flowchart* di atas[15] :

1. Departemen A mengirimkan dokumen A sebanyak rangkap 2 kepada “*Person A*”
2. Isi dari dokumen A diinput secara manual kedalam komputer dan hasilnya disimpan pada penyimpanan
  - a. Rangkap pertama diberikan kepada “*Person B*” (menggunakan konektor)
  - b. Rangkap kedua diarsip dan diurutkan secara abjad
3. “*Person B*” menerima dokumen B dari pelanggan, kemudian dokumen B dicocokkan secara manual dengan dokumen A

4. Isi dari dokumen B diinput secara manual kedalam komputer
5. Kemudian komputer melakukan proses pada *input*, dan menyimpan isinya serta kemudian konfirmasinya ditampilkan lewat layar yang digunakan oleh “*Person B*”
6. “*Person B*” kemudian mengarsip dokumen A dan B serta diurutkan berdasarkan kronologi



# UNIVERSITAS MIKROSKIL