



### **Hak cipta dan penggunaan kembali:**

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

### **Copyright and reuse:**

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

## **BAB II**

### **LANDASAN TEORI**

#### **2.1. *Smartphone***

Menurut Williams & Sawyer (2011), *smartphone* adalah telepon selular dengan mikroprosesor, memori, layar dan modem bawaan.

Menurut Shelly, Cahsman, & Vermaat (2012) *smartphone* adalah telepon yang *internet-enabled* sehingga memungkinkan pengguna untuk menerima dan mengirim *email* serta mengakses web. Ponsel pintar juga menyediakan fungsi-fungsi manajemen informasi pribadi seperti kalender, buku agenda, buku alamat, kalkulator, dan catatan. Beberapa ponsel pintar juga menawarkan berbagai macam aplikasi seperti *word processing* dan *spreadsheet* untuk membantu pengguna dalam mengerjakan suatu dokumen.

Dari dua pendapat tersebut dapat disimpulkan bahwa *smartphone* merupakan ponsel pintar yang memiliki mikroprosesor, memori, layar, dan modem bawaan dan fitur-fitur lebih canggih dibanding ponsel biasa sehingga fungsi ponsel tidak lagi hanya sekedar alat komunikasi, namun juga sebagai salah satu alat bantu manusia dalam mengerjakan pekerjaan mereka, mengakses informasi, dan terhubung dengan orang banyak secara global.

#### **2.2. *Aplikasi Mobile***

Aplikasi adalah satu unit perangkat lunak yang dibuat untuk melayani kebutuhan akan beberapa aktivitas seperti sistem perniagaan,

permainan, pelayanan masyarakat, periklanan, atau semua proses yang hampir dilakukan oleh manusia. (Hengky, 2006). Sedangkan *mobile* dapat diartikan sebagai perpindahan yang mudah dari satu tempat ke tempat lain, atau dapat diartikan sebagai *mobile phone* (telepon seluler). Sehingga dapat dikatakan bahwa aplikasi *mobile* adalah sistem perangkat lunak yang dapat digunakan *user* meskipun *user* berpindah-pindah tempat dengan menggunakan perlengkapan seperti telepon seluler, PDA (*personal digital assistant*), atau tablet.

## **2.3. Android**

### **2.3.1. Pengertian Android**

Android adalah sistem operasi untuk telepon seluler yang berbasis Linux. Android menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam piranti bergerak. (Arifianto, 2011).

### **2.3.2. Sejarah Android**

Android pertama kali diluncurkan pada 5 November 2007. *Smartphone* pertama yang menggunakan sistem operasi Android dikeluarkan oleh T-Mobile dengan sebutan G1 pada bulan September 2008. Hingga saat ini Android telah merilis beberapa versi Android untuk menyempurnakan versi sebelumnya. Berikut adalah beberapa versi Android yang telah dirilis menurut data dari Wikipedia.

1. Android versi 1.5 (Cupcake) pada bulan April 2009

2. Android versi 1.6 (Donut) dirilis pada bulan September 2009
3. Android versi 2.0/2.1 (Éclair) dirilis pada bulan Oktober 2009
4. Android versi 2.2 (Froyo: Frozen Yoghurt) dirilis pada bulan Mei 2010
5. Android versi 2.3 (Gingerbread) dirilis pada bulan Desember 2010
6. Android versi 3.0 (Honeycomb) dirilis pada bulan Februari 2011
7. Android versi 4.0 (Ice Cream Sandwich) dirilis pada bulan Oktober 2011
8. Android versi 4.2 (Jelly Bean) yang dirilis pada bulan Juli 2012
9. Android versi 4.4 (KitKat) yang dirilis pada bulan Oktober 2013
10. Android versi 5.0 (Lollipop) yang dirilis pada bulan Juni 2014
11. Android versi 6.0 (Marshmallow) yang dirilis pada bulan September 2015
12. Android versi 7.0 (Nougat) yang dirilis pada Juni 2016.

#### **2.4. User Interface**

Antarmuka pemakai atau biasa disebut *user interface* merupakan bagian dari sistem informasi yang membutuhkan interaksi pengguna untuk membuat *input* dan *output* (Satzinger, 2010). *User interface* dapat menerima informasi dari pengguna dan memberikan informasi kepada pengguna yang bertujuan untuk membantu dalam mengarahkan alur navigasi sampai pengguna menemukan solusi masalah yang dicari. Beberapa hal yang perlu dipertimbangkan dan diterapkan pada saat merancang desain *user interface*

agar dapat berfungsi dengan baik dan menarik adalah bentuk, ukuran, nilai, warna, tata letak *layout*, dan ikon sebagai navigasi dan *typografi*.

## 2.5. *Flowchart*

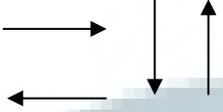
Diagram alir (*flowchart*) dibutuhkan untuk menggambarkan sebuah algoritma yang terstruktur dan mudah dipahami oleh orang lain. *Flowchart* merupakan langkah-langkah penyelesaian masalah yang dituliskan dalam simbol-simbol tertentu dengan menggambarkan urutan logika dari suatu prosedur pemecahan masalah. Selain dibutuhkan sebagai alat komunikasi, diagram alir (*flowchart*) juga diperlukan sebagai dokumentasi. Tujuannya yaitu untuk menggambarkan suatu tahapan penyelesaian masalah secara sederhana, terurai rapi dan jelas menggunakan simbol-simbol yang standar (Sitorus, 2015).

Simbol-simbol yang dipake di dalam *flowchart* dibagi menjadi tiga kelompok, yaitu.

1. *Flow direction symbols*

Simbol yang dipakai untuk menghubungkan antara simbol yang satu dengan simbol lainnya atau disebut juga *connecting line*, seperti yang ditunjukkan pada Tabel 2.1

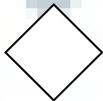
Tabel 2. 1 *Flow direction symbols*

<b>Simbol</b>	<b>Nama</b>	<b>Fungsi</b>
	<i>Arus / Flow</i>	Penghubung antara prosedur / proses
	<i>Connector</i>	Simbol keluar / masuk prosedur atau proses dalam lembar / halaman yang sama
	<i>Off-line Connector</i>	Simbol keluar / masuk prosedur atau proses dalam lembar / halaman yang lain

## 2. *Processing symbols*

Simbol yang menunjukkan jenis operasi pengolahan dalam suatu prosedur seperti pada Tabel 2.2.

Tabel 2. 2 *Processing symbols*

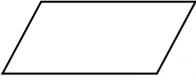
<b>Simbol</b>	<b>Nama</b>	<b>Fungsi</b>
	<i>Process</i>	Simbol yang menunjukkan pengolahan yang dilakukan Komputer
	<i>Decision</i>	Simbol untuk kondisi yang akan menghasilkan

Simbol	Nama	Fungsi
		beberapa kemungkinan jawaban / aksi
	<i>Predefined Process</i>	Simbol untuk mempersiapkan penyimpanan yang akan digunakan sebagai tempat pengolahan didalam <i>storage</i>
	<i>Terminal</i>	Simbol untuk permulaan atau akhir dari suatu program
	<i>Manual Input</i>	Simbol untuk memasukkan data secara <i>manual on-line keyboard</i>
	<i>Predefined Process (Sub Program)</i>	Permulaan sub program/proses menjalankan sub program

### 3. *Input/output symbols*

Simbol yang dipakai untuk menyatakan jenis peralatan yang digunakan sebagai media *input* atau *output* seperti pada Tabel

2.3.

Simbol	Nama	Fungsi
	<i>Input-Output</i>	Simbol yang menyatakan proses input dan output tanpa tergantung dengan jenis peralatannya
	<i>Document</i>	Simbol yang menyatakan input berasal dari dokumen dalam bentuk kertas atau output di cetak dikertas
	<i>Disk and On-line Storage</i>	Simbol untuk menyatakan input berasal dari disk atau output di simpan ke disk

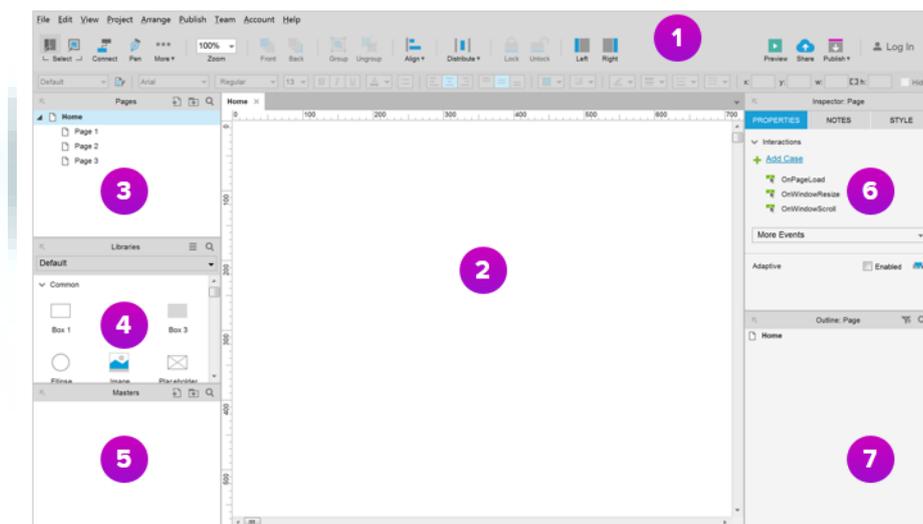
## 2.6. Axure

Axure adalah sebuah perusahaan yang didirikan pada Mei 2002 oleh Victor Hsu dan Martin Smith di Berkeley, CA. Versi Pilot dari Axure RP dirilis pada tahun 2003 dan merupakan salah satu desain yang dispesialisasi dalam perancangan aplikasi berbasis *browser*. Setelah itu, 12 tahun kemudian Axure RP dikenal sebagai sebuah standar dalam pembuatan perancangan sebuah aplikasi.

Axure RP merupakan aplikasi desktop yang digunakan untuk membuat *wireframe*, *prototype* interaktif, *flow diagrams*, dan dokumentasi untuk aplikasi bisnis, *website* dan aplikasi *mobile*. Axure berfokus pada

fleksibilitas, menyediakan fitur untuk membuat tampilan visual yang bagus dan membantu *non-coders* (orang yang tidak dapat melakukan *coding*), serta memberikan fitur yang mudah dimengerti oleh *user* yaitu dengan *drag and drop*. Axure RP juga dapat diexport menjadi file HTML agar dapat dilihat melalui *web browser*.

Pada Gambar 2.1 ditampilkan halaman aplikasi Axure dan berikut ini adalah penjelasan dari setiap nomor yang tertera di gambar.



Gambar 2. 1 Tampilan halaman Axure

- 1. Main Menu dan Main Toolbar:** untuk melakukan operasi dasar seperti membuka, menyimpan *file*, *zoom in* dan *zoom out*, *publish*. Selain itu fungsinya juga untuk melakukan *edit*
- 2. Canvas:** lokasi untuk *edit widget*, selain itu juga untuk menarik penggaris (*ruler*) untuk mempermudah dalam merapihkan *widget*.
- 3. Pages:** untuk menambahkan, menghapus, mengganti nama, dan mengatur halaman.

4. **Widgets:** terdapat *widget* yang dibutuhkan seperti *label*, *textfield*, *button*, *rectangle* dan lainnya. Cara menggunakannya adalah dengan menarik *widget* yang diinginkan dan meletakkannya pada *canvas*. Selain itu juga pada bagian ini dapat digunakan untuk *download libraries* dan membuat *custom libraries*.
5. **Masters:** menambahkan, menghapus, mengganti nama dan mengatur *master*. *Master* adalah kumpulan *widget* yang dapat digunakan berulang-ulang pada satu *file*.
6. **Inspector:** apapun yang dipilih di *canvas*, baik itu satu *widget* atau beberapa *widget*, akan bisa dikonfigurasi melalui *Inspector* ini. Namun, apabila tidak ada yang dipilih di *canvas*, maka *Inspector* akan menampilkan *page-level controls*. Pada bagian *inspector* ini terdapat tiga *tab*, yaitu *tab properties*, *notes*, dan *style*. Berikut adalah penjelasan singkat mengenai ketiga *tab* tersebut.
  - a. *Properties:* menambahkan dan mengubah interaksi (yang menentukan perilaku dinamis *widget* atau halaman) dan *interactive properties* lain
  - b. *Notes:* menambahkan dan mengubah *widget* dan halaman *notes*
  - c. *Style:* mengubah *style widget* dan halaman.
7. **Outline:** melihat daftar yang dapat dicari, diurutkan, dan disaring dari semua *widget*, *master*, dan *panel* dinamis pada diagram saat ini.

## 2.7. TCSD (*Task Center System Design*)

Metode *Task Centered System Design* (TCSD) adalah proses di mana desainer (Greenberg, 2004):

- 1) Diartikulasi menjadi suatu deskripsi konkrit dalam bentuk dunia nyata seperti mengerjakan setiap tugas mereka dalam kehidupan sehari-hari.
- 2) Menggunakan deskripsi untuk menentukan pengguna dan tugas yang mana yang seharusnya didukung oleh sistem.
- 3) Suatu prototipe dari sebuah interface yang memenuhi kebutuhan sistem.
- 4) Melakukan evaluasi interface dengan melakukan sebuah task-centered walkthrough.

Metode TCSD menurut (Greenberg, 2004) terdiri dari 4 tahap, yaitu:

### 1) *Identification*

Tugas berpusat pada pengidentifikasi masalah dari pengguna sistem dan mengartikulasikan tugas-tugas yang realistis.

### 2) *User-Centered Requirements Analysis*

Menganalisis permasalahan yang ada serta untuk memutuskan apakah hasil analisis disertakan atau dikecualikan dari desain.

### 3) *Design as Scenario*

Menentukan desain sistem proses dan data yang diperlukan oleh sistem dan mengembangkan desain untuk menyesuaikan dengan kebutuhan *user* dan kebutuhan *task*.

### 4) *Walkthrough Evaluate*

Pada tahap ini dilakukan evaluasi akhir terhadap desain sistem.

## 2.8. *Post-Study Sistem Usability Questionnaire (PSSUQ)*

Post-Study Sistem Usability Questionnaire (PSSUQ) merupakan instrumen penelitian yang dikembangkan untuk digunakan dalam evaluasi *usability* di IBM. PSSUQ terdiri dari 19 item yang ditujukan untuk menilai lima sistem karakteristik *usability*. Berikut paket kuesioner PSSUQ (*Post-Study System Usability Questionnaire*) selengkapnya sebagai berikut.

1. Secara keseluruhan, saya puas dengan kemudahan penggunaan *aplikasi mobile* ini
2. *Aplikasi mobile* mudah untuk digunakan
3. Saya secara efektif dapat menyelesaikan tugastugas dan skenario menggunakan *aplikasi mobile* ini
4. Saya bisa menyelesaikan tugas dan skenario dengan cepat menggunakan *aplikasi mobile* ini
5. Saya dengan efisien dapat menyelesaikan tugas dan skenario menggunakan *aplikasi mobile* ini
6. Saya merasa nyaman menggunakan *aplikasi mobile* ini
7. Mudah untuk belajar menggunakan *aplikasi mobile* ini
8. Saya percaya saya bisa menjadi produktif dengan cepat menggunakan *aplikasi mobile* ini
9. *Aplikasi mobile* memberikan pesan kesalahan yang jelas memberitahu saya bagaimana untuk memperbaiki masalah
10. Setiap kali saya melakukan kesalahan dengan menggunakan *aplikasi mobile*, saya bisa pulih dengan mudah dan cepat

11. Informasi (seperti pesan bantuan *online*, pesan yang muncul pada layar, dan dokumentasi lainnya) disediakan dengan jelas oleh aplikasi *mobile* ini
12. Mudah untuk menemukan informasi yang saya butuhkan
13. Informasi yang disediakan untuk aplikasi *mobile* ini mudah untuk dipahami
14. Informasi tersebut efektif dalam membantu saya menyelesaikan tugas dan skenario
15. Peletakan informasi pada layar aplikasi *mobile* ini jelas
16. Tampilan antarmuka aplikasi *mobile* ini menyenangkan
17. Saya suka menggunakan antar muka aplikasi *mobile* ini
18. Aplikasi *mobile* ini memiliki semua fungsi dan kemampuan yang saya harapkan untuk ada
19. Secara keseluruhan, saya merasa puas dengan aplikasi *mobile* ini.

Dari 19 item kuesioner dapat dikelompokkan menjadi empat tanggapan PSSUQ yaitu : Skor kepuasan secara keseluruhan (*OVERALL*), kegunaan sistem (*SYSUSE*), kualitas informasi (*INFOQUAL*) dan kualitas antarmuka (*INTERQUAL*). Untuk mengukur tingkat persetujuan *user* terhadap item-item kuesioner digunakan bentuk skor tujuh poin dengan model skala Likert. Hasil pengukuran kemudian diolah dengan metode

statistik deskriptif dan dilakukan analisis baik terhadap masing-masing parameter atau terhadap keseluruhan parameter (Lestari, S., 2014).

## 2.9. Wawancara

Menurut Nazir (1988), wawancara adalah proses memperoleh keterangan untuk tujuan penelitian dengan cara tanya jawab sambil bertatap muka antara si penanya atau pewawancara dengan si penjawab atau responden dengan menggunakan alat yang dinamakan *interview guide* (panduan wawancara). Sedangkan Charles Stewart dan W. B. Cash (2000) mendefinisikan wawancara sebagai sebuah proses komunikasi berpasangan dengan suatu tujuan yang serius dan telah ditetapkan sebelumnya yang dirancang untuk bertukar perilaku dan melibatkan tanya-jawab.

Dari kedua pendapat ahli tersebut dapat disimpulkan bahwa wawancara merupakan sebuah proses interaksi tanya jawab antara pewawancara dengan narasumber berdasarkan pertanyaan yang telah disiapkan sebelumnya oleh pewawancara untuk mendapatkan tujuan yang diinginkan.

## 2.10. Prinsip 8 *Golden Rules*

Menurut Ben Shneiderman, *human-Computer Interaction* (HCI) adalah disiplin ilmu yang berhubungan dengan perancangan, evaluasi, dan implementasi sistem komputer interaktif untuk digunakan oleh manusia, serta studi fenomena-fenomena besar yang berhubungan dengannya. Fokus pada HCI adalah perancangan dan evaluasi antarmuka pemakai (*user interface*). Prinsip 8 *golden rules* merupakan sebuah aturan yang

menetapkan *interface* agar dapat menyempurnakan implementasi dalam hal berinteraksi pada manusia. Beberapa poin-poin yang terdapat dalam 8 *golden rules* diantaranya sebagai berikut:

a. *Strive for consistency*

Tampilan aplikasi antar halaman dalam satu aplikasi ataupun antara aplikasi yang masih berhubungan harus dibuat konsisten agar *user* terutama yang tergolong *user* pemula tetap dapat mengenali bahwa halaman yang dilihatnya masih termasuk aplikasi yang digunakan. Tampilan yang dibuat juga tidak seharusnya rumit dan tidak menggunakan warna yang terlalu banyak.

b. *Cater to universal usability*

Mengenali kebutuhan *user* yang beragam dan memudahkan *user* dalam melakukan perubahan konten. Desainer harus merancang tampilan antarmuka yang sesuai dengan kategori *user* yang menggunakan aplikasi tersebut dengan mencari perbedaan antara *user* pemula dan *user* ahli, rentang usia, keterbatasan, kemampuan fisik, dan perbedaan teknologi sehingga dapat disesuaikan juga dengan tingkat pemahaman *user* agar mereka mudah dalam menggunakan aplikasi tersebut.

c. *Offer informative feedback*

Aplikasi yang baik selalu memberikan timbal balik (*feedback*) ketika terjadi sesuatu di dalam aplikasi itu sendiri.

Sehingga dapat memudahkan *user* untuk mengetahui apa yang harus dilakukan. *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.

d. *Design dialogs to yield closure*

Prinsip ini merupakan bagian dari prinsip ketiga. Aplikasi yang baik juga harus mempunyai suatu perbedaan pada tampilan. Seperti adanya dialog yang muncul untuk memberi tahu *user* bahwa proses yang dijalankan sudah selesai.

e. *Prevent errors*

Hal ini diperlukan untuk menjaga agar *user* tidak melakukan kesalahan dalam menjalankan proses. Terjadinya kesalahan yang dilakukan oleh *user* harus cepat ditanggapi oleh aplikasi dan memberikan solusi kepada para pengunjung untuk mengatasi kesalahannya tersebut. Contohnya adanya petunjuk pengisian formulir sesuai format yang diterima oleh aplikasi, sehingga *user* dapat mengisi formulir dengan tepat pada percobaan pertama.

f. *Permit easy reversal of actions*

Di dalam sebuah aplikasi harus ada fasilitas untuk *user* bisa mengubah atau menghapus tindakan yang sudah dilakukan sebelumnya untuk mengantisipasi *human error*.

g. *Support internal locus of control*

Aplikasi perlu memiliki fasilitas untuk pengguna atau *user* agar dapat dengan bebas bernavigasi dan mengubah informasi akun yang dimilikinya sesuai dengan yang dikehendaki.

h. *Reduce short term memory load*

Pada poin ini biasanya orang lebih memusatkan pada desain tata letak menu dan tombol. Keterbatasan manusia dalam mengolah informasi dalam memori jangka pendek membutuhkan antarmuka (*interface*) yang sederhana dan mengurangi pergerakan *window*. Yang dimaksud pergerakan *window* seperti ketika *user* harus mengingat suatu informasi pada satu layar dan informasi tersebut digunakan pada layar lain.

UMMN