

BAB II LANDASAN TEORI

2.1 Huruf (*Font*)

Huruf atau *font* merupakan salah satu tulisan yang berisi serangkaian ikon, karakter, ataupun simbol data elektronik yang dapat di baca di komputer. Ada tiga jenis dasar dari format data berkas huruf, yaitu : *font bitmap*, *font outline*, *font stroke*.

Huruf merupakan bagian terkecil dari struktur bahasa tulis dan merupakan elemen dasar untuk membangun sebuah kata atau kalimat. Huruf memiliki perpaduan nilai fungsional dan estetik. Jenis *font* yang ada disaat ini adalah TTF (*true type font*), OTF (*open type font*), dan lain sebagainya (Danton Sihombing, 2003).

Huruf adalah sebuah grafem (bentuk, goresan, atau lambang) dari suatu sistem tulisan, misalnya 26 huruf dalam alfabet Latin modern, atau 47 huruf dalam hiragana. Dalam suatu huruf terkandung satu fonem atau lebih (tergantung jenis aksara), dan fonem tersebut membentuk suatu bunyi dari bahasa yang dituturkannya. Setiap aksara memiliki huruf dengan nilai bunyi yang berbeda-beda. Dalam aksara jenis alfabet, abjad, dan abugida, biasanya suatu huruf melambangkan suatu fonem atau bunyi. Berbeda dengan logo gram atau ideogram, yang hurufnya mewakili ungkapan atau makna suatu lambang, misalnya aksara Tionghoa. Dalam aksara jenis silabis atau aksara suku kata, suatu huruf melambangkan suatu suku kata, contohnya adalah Hiragana dan Katakana yang digunakan di Jepang dan aksara Lampung yang digunakan oleh masyarakat Lampung.

Beberapa aksara latin seperti alfabet Yunani dan keturunannya memiliki varian dari satu huruf yang sama disebut dengan istilah huruf besar dan huruf kecil. Huruf besar biasanya dipakai di awal kata, sedangkan huruf kecil ditulis setelahnya.



Gambar 2.1 abjad huruf latin

Menurut Roy Brewer (1971) dalam buku Pengantar Tipografi: “Tipografi dapat memiliki arti luas, yang meliputi penataan dan pola halaman, atau cetakan atau dalam arti yang lebih sempit hanya mencakup pemilihan, pengaturan, dan berbagai hal yang berkaitan dengan pengaturan jalur pengaturan huruf (set), tidak termasuk ilustrasi dan elemen lainnya, bukan surat di halaman dicetak” (Sudiana, 2001: 2). Sedangkan Menurut Dendi sudiana dalam buku Pengantar Tipografi: “Gambar adalah elemen grafis yang paling mudah dibaca tetapi melalui kata-kata yang terdiri dari huruf oleh huruflah memandu pemahaman pembaca pesan atau ide” (Sudiana, 2001: 1).

Sejak masa pra-sejarah tulisan baik dalam bentuk gambar, simbol, angka, huruf sudah menjadi sarana untuk bertukar informasi hingga pada abad ke-7 bangsa Cina menemukan teknik cetak timbul dengan menggunakan tinta yang digoreskan pada lembaran kertas yang terbuat dari tanaman papyrus untuk media bertukar informasi.

Pada tahun 1991 *Apple Computer* dan *Microsoft Corporation* mengeluarkan *true type font*. *PostScript font* dan *true type font* adalah huruf elektronik atau yang sering disebut *Font*. Huruf digital sesungguhnya berupa bahasa komputer yang berfungsi menerjemahkan kode- kode untuk menghasilkan tampilan bentuk huruf yang sempurna baik di layar monitor maupun pada saat pencetakan. Jenis *font* yang ada disaat ini adalah TTF (*true type font*), otf (*open type font*), dan lain sebagainya.

2.2 True Type Font

Merupakan salah satu jenis huruf *digital* yang dapat menuliskan karakter-karakter dan dapat dijalankan pada sistem operasi Windows. *true type font* merupakan salah satu buatan dari Microsoft office sama seperti *open type font* (OTF). Photoshop, Illustrator, Indesign akan membawa paket *font* berjenis *Postscript font* sebagai jenis tulisan *default* untuk menjalankan aplikasi. Adobe pun mempunyai jenis huruf tersendiri yaitu, *adobe type manager* (ATM) yang dimana menjadi jenis huruf standar produk buatan Adobe.

Seiring berkembangnya dunia teknologi, tulisan atau *font face* yang dulunya hanya ada di media cetak kertas ikut berkembang. *Font-font* berkembang menjadi digital dalam dunia percetakan dan komunikasi. Dunia teknologi *font* telah melakukan langkah yang besar dengan bermunculannya desain-desain huruf yang inovatif dan telah memperkaya dunia desain komunikasi visual dalam bentuk digital.

Diperjalanan awal dari teknologi *font* digital, *font* didesain dengan ukuran yang pasti seperti 9, 10, 12, 14, 18 dan 24PT dengan menggunakan standar bitmap layar komputer sehingga memiliki kelemahan ketika *font* harus diperbesar atau diperkecil. Akan tetapi kini dengan kehadiran teknologi vektor dan antialiasing teknologi *font* terus berkembang dengan meninggalkan teknologi bitmap.

Jenis tulisan huruf *true type* lebih populer dibandingkan dengan *Postscript font*. *True type* adalah *font* berbasis informasi *outline* dan kebutuhan ukuran format vektornya bisa disesuaikan dengan dengan akurasi yang tinggi. *True type* menggunakan metode Quadratic B-spline dengan menerapkan titik-titik secara langsung pada garis dan bagian-bagian yang dilengkungkan. *Software* standar dari *true type* dibuat oleh Apple, akan tetapi kini juga bisa digunakan oleh sistem operasi Windows. Kedua sistem tersebut memiliki *True type rasterizer* yang menyediakan informasi untuk penggambaran di layar dan *output* cetak. *True type* dirancang sebagai file tunggal (*suitcase*), yang di dalamnya sudah terdapat keluarga huruf dalam bentuk *plain*, *plain italic*, *bold* dan *bold italic* membuat tipe huruf ini lebih rapih dalam pendataan file dibandingkan dengan *Postscript font*.



Gambar 2.2 bentuk tulisan digital dengan format *truetype*

Unicode adalah *standard* baru untuk menjabarkan *characters set* dalam sebuah sistem, bekerja mirip dengan ASCII (*American Standard Code for Information Interchange*). Biasanya dalam ASCII terdapat 200 *characters set*, tetapi dalam Unicode bisa terdapat 65000 jenis karakter sehingga Unicode sering digunakan dalam informasi digital multi bahasa. Unicode bukanlah sebuah *font encoder* (pembaca sandi-sandi *font*) tetapi hanya sebuah standar dimana informasi *glyph* (*outline* bentuk sebuah *font*) sebuah *font* tersimpan.

Peneliti memilih jenis huruf digital yang berformat *true type font* dikarenakan sistem operasi Android mendukung semua jenis tulisan menggunakan jenis huruf *true type font*. Android secara native mengenal tiga jenis *font* dengan nama singkatan *sans*, *serif* serta *monospace*. Ketiga jenis *font* ini merupakan rangkaian-rangkaian *font droid* yang dibuat untuk *open handset alliance* oleh *ascender*. Android yang *open source* membuat beberapa jenis *font* yang dapat dioperasikan didalam sistem Android seperti *true type font* dan *open type font*. Namun sistem Android tidak sepenuhnya mendukung tulisan dengan format huruf *open type font* melainkan Android sepenuhnya mendukung tulisan huruf dengan format *true type font*. Menurut android pada situs resmi android mereka menyatakan bahwa “sistem operasi Android mendukung tulisan dengan huruf *true type font*”.

2.3 Aksara Lampung

Aksara Lampung atau sering disebut dengan istilah (**kaganga**) merupakan salah satu warisan bangsa yang berupa non-fisik. Aksara Lampung adalah bentuk tulisan yang memiliki hubungan dengan aksara Pallawa dari India Selatan. Ragam tulisannya *fonetik* berjenis suku kata yang merupakan huruf hidup seperti dalam huruf Arab, dengan menggunakan tanda-tanda fathah pada baris atas dan tanda-tanda kasrah pada baris bawah, tetapi tidak menggunakan tanda dammah pada baris depan, melainkan menggunakan tanda di belakang, di mana masing-masing tanda mempunyai nama tersendiri. Had Lampung mempunyai total huruf induk sebanyak 20 bentuk yang berbeda dan mempunyai beberapa anak huruf dan angka serta simbol. Gambar induk huruf aksara Lampung (had Lampung) beserta tanda baca huruf Lampung (kelabai Lampung) dapat dilihat pada gambar 2.3.

No	Aksara	Bunyi
1		Ka
2		Ga
3		Nga
4		Pa
5		Ba
6		Ma
7		Ta
8		Da
9		Na
10		Ca

No	Aksara	Bunyi
11		Ja
12		Nya
13		Ya
14		A
15		La
16		Ra
17		Sa
18		Wa
19		Ha
20		Gha

No	Nama	Bunyi	Aksara
1	Datas	an	
2	Ulan	i	
3	Ulan	e	
4	Bicek	e	
5	Tekelubang	ng	
6	Rejunjung	r	
7	Tekelungau	au	
8	Bitan	u	
9	Bitan	o	
10	Tekelingai	ai	
11	Keleniah	ah	
12	Nengen	mati	
13	Kuma (koma)		
14	Beghadu (titik)		

(a) Gambar induk huruf Lampung.

(b) Gambar anak huruf Lampung.

Gambar 2.3 (a) dan (b) induk beserta anak huruf Lampung.

Fonem – fonem atau kelabai Lampung mempunyai nama tersendiri untuk penyebutan tata letak serta bentuknya. Secara bagian anak huruf Lampung (kelabai Lampung) terbagi menjadi 3 bagian, yaitu anak huruf (kelabai Lampung)

yang terletak diatas induk huruf (had Lampung), anak huruf (kelabai Lampung) yang terletak dibawah induk huruf (had Lampung), serta anak huruf (kelabai Lampung) yang terletak dibelakang induk huruf (had Lampung).

Anak huruf (kelabai Lampung) yang terletak diatas induk huruf Lampung (had Lampung) disebut “datas” dengan bentuk garis lurus dua buah. Anak huruf Lampung (kelabai Lampung) yang terletak diatas induk huruf Lampung (had Lampung) dan berbentuk garis siku keatas dan garis siku kebawah bernama “ulan”. Anak huruf Lampung (kelabai Lampung) yang terletak diatas induk huruf Lampung (had Lampung) berbentuk garis miring disebut dengan “bicek”. Anak huruf Lampung (kelabai Lampung) yang terletak diatas induk huruf Lampung (had Lampung) berbentuk garis zigzag disebut dengan “rejunjung”. Anak huruf Lampung (kelabai Lampung) yang terletak diatas induk huruf Lampung (had Lampung) berbentuk garis miring disebut dengan “bicek”.

Anak huruf lampung yang terletak dibawah induk huruf lampung berbentuk garis siku ke atas bernama “tekelungau”. Anak huruf yang terletak dibawah induk huruf Lampung berbentuk garis lurus bernama “bitan”. Anak huruf yang terletak dibawah induk huruf Lampung yang berbentuk garis miring bernama “bitan”.

Anak huruf yang terletak disamping induk huruf Lampung berbentuk garis lurus bernama “tekelingai”. Anak huruf yang terletak disamping induk huruf Lampung berbentuk garis zigzag bernama “keleniah”. Sedangkan untuk tanda huruf mati berada di samping induk huruf berbentuk garis miring panjang dengan sudut bawah melingkar sedikit bernama “nengen”.

2.4 Keyboard

Keyboard berperan penting dalam hal penulisan (*writing*) baik di *computer* maupun diperangkat *mobile* atau *low device* yang berwujud fisik. *Keyboard* dalam arti bahasa Indonesia merujuk kepada papan tombol jari atau papan *tuts*. Pada perkembangan teknologi, perwujudan *keyboard* fisik tergantikan oleh *virtual* yang kebanyakan tertaman secara *default* di *mobile* bersistem operari Android.

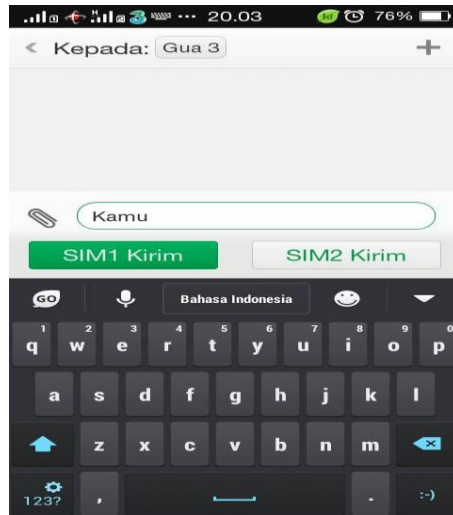
2.5 Virtual Keyboard

Keyboard berperan penting dalam hal penulisan (*writing*) baik di komputer maupun diperangkat *mobile* atau *low device* yang berwujud fisik. *Keyboard* dalam arti bahasa Indonesia merujuk kepada papan tombol jari atau papan *tuts*. Pada perkembangan teknologi, perwujudan *keyboard* fisik tergantikan oleh *virtual* yang kebanyakan tertanam secara *default* di *mobile*.

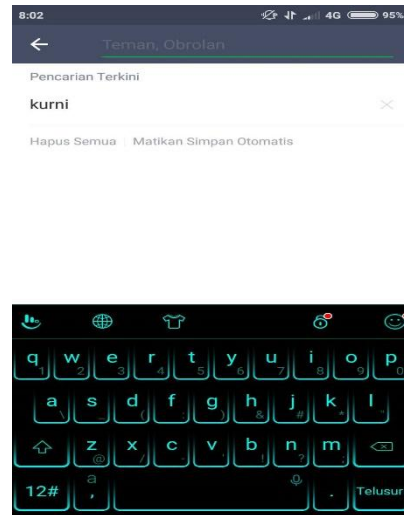
Virtual merujuk kepada arti visual, yang dimana merupakan hayalan atau sebuah kotak (*box*) hayalan yang dimana terlihat nyata dan dapat di sentuh. *Virtual Keyboard* bermula dari ditemukan dan dipatenkannya teknologi optik oleh para insyinyur IBM (*International business machines corporation*) tahun 1992.

Virtual keyboard merupakan inovasi terbaru dari papan ketik (*keyboard*) yang dimana penulisan karakter terdapat di sebuah kotak (*box*) yang dapat di sentuh serta dapat menuliskan karakter-karakter yang tersentuh dilayar.

Sampai saat ini hampir seluruh merek *mobile* yang bersistem Android menggunakan papan ketik berjenis *virtual*. Berbagai nama *virtual keyboard* yang berjalan diatas OS Andorid diantaranya, VIVO Swype, OPPO Swype dan berbagai *virtual keyboard* lainnya dibuat untuk mengantikan *keyboard* fisik. Dan *virtual keyboard* tersebut sudah tertanam (*embedded*) sejak perangkat dibeli. Bahkan *virtual keyboard* yang berjalan di *mobile* berbasis non-Android (Symbian OS/Blackberry OS/Mac OS) pun sudah menanamkan *virtual keyboard* secara *default* seperti gambar 2.4.



(a)



(b)

Gambar 2.4 *Virtual Keyboard* (a) OPPO, (b) Xiaomi

2.6 Translasi Pengkodean Karakter

Translation (convert table ASCII) adalah mengubah *binary code* atau karakter *font* yang telah ditetapkan oleh ASCII (*American Standard code for Information Interchange*). Setiap huruf memiliki nomor sendiri, *input text* untuk mengkonversikan ke nomor ASCII ini disebut *translation*.

Character encoding atau penyandian karakter terdiri dari sandi atau kode yang memasang serangkaian aksara berurutan dari suatu kumpulan dengan sesuatu yang lain, seperti urutan bilangan asli (*natural numbers*), perlapanan (*octet*) untuk mempermudah penyimpanan naskah pada komputer, sandi morse yang menyandikan huruf alfabet Latin ke dalam rangkaian tekanan panjang dan pendek dari kunci telegraf, ASCII yang menyandikan huruf, bilangan, dan simbol-simbol sebagai bilangan bulat dan versi perduaan (*binary*) 7-bit dari bilangan bulat tersebut.

Pada masa-masa awal komputer, pengenalan penyandi aksara seperti ASCII (1963) dan EBCDIC (1964) mengawali proses pembakuan. Keterbatasan kumpulan tersebut mulai tampak dan beberapa metode *seterjadinya* (*ad hoc*) dikembangkan untuk memperluaskannya. Kebutuhan untuk mendukung berbagai macam tata tulisan, termasuk rumpun aksara Cina, Jepang dan Korea dari tulisan

Asia Timur, membutuhkan dukungan untuk jauh lebih banyak aksara dan menuntut pendekatan yang tersusun rapi untuk penyandi aksaraan, dan bukan pendekatan seterjadinya seperti sebelumnya.

2.7 Unicode

Unicode adalah sebuah standar industri komputasi untuk pengkodean secara konsisten yang menangani sebuah tulisan (*font*) dalam bentuk digital. *Unicode consortium* adalah organisasi Nirlaba yang mengkoordinasikan pengembangan standar Unicode. *Unicode consortium* didirikan di California pada tanggal 3 Januari 1991, dan pada bulan Oktober 1991 volume pertama diterbitkan untuk menangani berbagai jenis karakter-karakter yang ada di Dunia dimana pengkodean tabel ASCII tidak sanggup melakukan penulisan karakter-karakter tersebut.

Ada sekitar lima puluh lebih tulisan-tulisan bahasa berbeda diseluruh belahan dunia dan semuanya tidak sanggup ditangani oleh tabel ASCII, misalnya Uni Eropa memerlukan pengkodean yang berbeda untuk mencakup semua bahasanya, bahkan untuk bahasa tunggal seperti bahasa Inggris tidak ada pengkodean tunggal yang memadai untuk semua karakter. Sistem pengkodean tersebut dapat terjadi sebuah korupsi, dimana satu pengkodean tertentu dapat mempunyai dua karakter berbeda atau dua pengkodean mengartikan satu karakter yang sama, sehingga sangat mengganggu proses komputerisasi dari penulisan-penulisan dalam bentuk digital.

Terlepas dari semua itu Unicode mengubah semuanya. Unicode dapat berjalan disemua jenis vendor, bahkan Unicode telah diadopsi oleh perusahaan-perusahaan ternama didunia seperti: Apple, HP, IBM, JustSystems, Microsoft, dan masih banyak lainnya dikarenakan Unicode bisa diadopsi oleh platform-platform berbeda seperti dinyatakan dalam situs resmi Unicode “*Unicode provides a unique number for every character, no matter what the platform, no matter what the program, no matter what the language*” (<http://www.unicode.org/standard/WhatIsUnicode.html>).

Penanganan-penanganan karakter yang dapat di tampung oleh unicode telah diakui oleh tulisan-tulisan yang ada di mancanegara seperti penulisan karakter Arab, Jepang, dan masih banyak lainnya. Bahkan tulisan tulisan daerah yang ada di nusantara pun sudah teridentifikasi di dalam pengkodean Unicode tanpa adanya satu pun yang korupt. Salah satunya aksara Rejang yang ada di Provinsi Bengkulu dan aksara Sunda yang ada di Provinsi Jawa Barat.

Untuk dapat berjalan di atas sistem tertentu, memerlukan pengkodean dari Unicode. Sehingga di saat sebuah mesin memanggil sebuah kode, akan keluar karakter yang dituju tanpa adanya timpang tindih dari pengkodean tersebut. Tabel pengkodean karakter unicode dapat dilihat di gambar 2.5.

A	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
0010	DL	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
0020		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
0030	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
0040	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
0050	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
0060	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
0070	p	q	r	s	t	u	v	w	x	y	z	{		}	~	¸

Gambar 2.5 Pengkodean huruf latin Unicode.

2.8 Android

Android merupakan sistem operasi untuk perangkat *mobile* yang berbasis Linux dan bersifat terbuka atau *open source* dengan lisensi GNU yang dimiliki Google (Wahana, 2013). Awalnya, *Google Inc.* membeli *Android Inc.*, pendatang baru yang membuat peranti lunak untuk ponsel. Kemudian untuk mengembangkan Android, dibentuklah *Open Handset Alliance* konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi.

2.8.1 *The Dalvik Virtual Machine (DVM)*

Salah satu element kunci dari Android adalah *Dalvik Virtual Machine* (DVM). Android berjalan di *Dalvik Virtual Machine* (DVM) bukan di *Java Virtual Machine* (JVM), sebenarnya banyak persamaan dengan *Java Virtual Machine* (JVM) seperti *Java ME (Java Mobile Edition)*, tetapi Android menggunakan *virtual machine* sendiri yang dirancang untuk memastikan beberapa featur-featur berjalan lebih efisien pada perangkat *mobile*.

2.8.2 *Android SDK (Software Development Kit)*

Android SDK adalah *tools API (Application Programming Interface)* yang diperlukan untuk mulai mengembangkan aplikasi pada *plat form* Android menggunakan bahasa pemrograman Java. Android merupakan *subset* perangkat lunak untuk ponsel yang meliputi sistem operasi, *middleware* dan aplikasi kunci yang di-*release* oleh Google. Saat ini disediakan *Android SDK (Software Development Kit)* sebagai alat bantu dan API untuk mulai mengembangkan aplikasi pada *platform Android* menggunakan bahasa pemrograman Java. Sebagai *platform* aplikasi-netral, Android memberi kesempatan untuk membuat aplikasi yang dibutuhkan.

2.8.3 *ADT (Android Development Tools)*


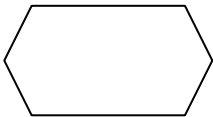


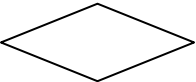

Android development tools adalah *plugin* yang didesain untuk IDE Android Studio yang memberikan kemudahan dalam mengembangkan aplikasi Android dengan menggunakan IDE Android Studio. Dengan menggunakan ADT untuk Android Studio akan memudahkan dalam membuat aplikasi *project* Android, membuat GUI aplikasi, dan menambahkan komponen-komponen yang lainnya.

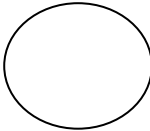
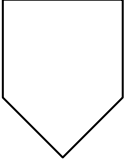
2.9 *Flowchart*

Flowchart merupakan diagram alir yang digunakan untuk menggambarkan algoritma dari program atau aplikasi yang akan dibuat nantinya agar tidak terjadinya masalah pada program atau aplikasi tersebut. Kusrini (2007 : 80) mengungkapkan: “*Flowchart* adalah bagan (*chart*) yang menunjukkan aliran (*flow*) didalam program atau prosedur sistem secara logika, digunakan terutama sebagai alat bantu komunikasi dan untuk dokumentasi”.

Suarga (2006 : 9) mengungkapkan: “*Flowchart* adalah untaian simbol gambar (*chart*) yang menunjukkan aliran (*flow*) dari proses terhadap data”. Menurut Pahlevy (2010) “*Flowchart (bagan alir)* adalah gambaran dalam bentuk diagram alir dari algoritma-algoritma dalam suatu program, yang menyatakan arah alur program tersebut”. Menurut Jogiyanto (2005) “Bagan alir (*flowchart*) adalah bagan (*chart*) yang menunjukkan alir atau arus (*flow*) di dalam program atau prosedur sistem secara logika”. Menurut Indrajani (2011) “*Flowchart* merupakan gambaran secara grafik dari langkah-langkah dan urutan prosedur suatu program”. Menurut Sariadin Siallagan (2009) “*Flowchart* adalah bagan atau suatu diagram alir yang mempergunakan simbol atau tanda untuk menyelesaikan suatu masalah”. Berdasarkan pendapat di atas, dapat disimpulkan bahwa *flowchart* adalah suatu alur atau bagan program yang menggambarkan dan mempresentasikan suatu algoritma yang dibuat dari pembuka, isi, sampai keluar program untuk menyelesaikan suatu permasalahan-permasalahan yang ada.

Tabel 2.1 Simbol *Flowchart* Program (Suarga, 2006 : 10)

Nama Simbol	Simbol	Deskripsi
<i>Terminator</i>		Mulai atau selesai.
<i>Predefined-data</i>		Definisi awal dari variabel atau data.
<i>Proses</i>		Menyatakan proses terhadap data.
<i>Input/output</i>		Menampilkan input atau menampilkan output.
Selesai		Memilih aliran berdasarkan syarat.
<i>Predefine Process</i>		Lambang fungsi atau sub program.

<i>Connector</i>		Penghubung.
<i>Off Page Connector</i>		Penghubung pada halaman yang berbeda.

2.10 Perancangan Pengembangan Perangkat Lunak

Perancangan adalah proses yang mendefinisikan sesuatu yang akan dikerjakan dengan menggunakan teknik yang bervariasi serta didalamnya melibatkan deskripsi mengenai arsitektur serta detail komponen dan juga keterbatasan yang akan dialami dalam proses pengerjaan. Bangun sistem adalah kegiatan menciptakan sistem baru mau pun pengganti atau memperbaiki sistem telah ada secara keseluruhan (Roger S.Pressman, 2002). Rancang bangun berarti mengatur segala sesuatu (sebelum bertindak, mengerjakan, atau melakukan sesuatu), merencanakan (Purwanto, 2008).

Perancangan perangkat lunak sangat penting guna membangun sebuah aplikasi. Perancangan tersebut bertujuan agar menyelesaikan kekacauan yang sering terjadi saat dalam proses pengembangan perangkat lunak. Nogueira dan koleganya menyatakan bahwa sisi kekacauan dapat dijelaskan sebagai “adanya suatu hubungan yang alami di antara urutan dan kekacauan suatu kompromi besar serta hal-hal yang tidak diperkirakan sebelumnya”.

Menurut Takashi Osada dalam buku rekayasa perangkat lunak: “Jika prosesnya benar, maka hasilnya pun akan memuaskan”(Roger S.Pressman, 2002). Salah satu model perancangan pengembangan perangkat lunak adalah metode air terjun (*waterfall*). Waterfall membutuhkan beberapa saat hingga spesifikasi kebutuhan untuk suatu permasalahan dapat dipahami dengan baik saat pekerjaan mengalir secara linier dari komunikasi hingga ke penyerahan sistem (perangkat lunak ke pelanggan).

Waterfall pertama kali diperkenalkan oleh Winston Royce yang memungkinkan adanya lingkaran umpan balik. Model air terjun (*waterfall*) sering juga dinamakan dengan siklus hidup klasik (*classic life cycle*) dimana hal ini menyatakan pendekatan yang sistematis dan berurutan (sekuensial).

Dalam suatu analisis Brandac menyatakan bahwa “adanya tahapan-tahapan dalam siklus hidup klasik cenderung terkunci sehingga berpacu dengan waktu dan subjek dari aliran perubahan-perubahan yang tiada akhirnya”. Meski demikian siklus hidup klasik dapat bertindak sebagai model proses perangkat lunak yang sangat bermanfaat dalam situasi-situasi dimana spesifikasi-spesifikasi kebutuhan telah jelas dan pekerjaan untuk menyelesaikan proyek perangkat lunak bersifat linier (Roger S.Pressman, 2001).

2.11 Android Studio



Gambar 2.6 Logo Android Studio

Android Studio adalah sebuah IDE (*Integrated Development Environment*) untuk *Android development* yang diperkenalkan oleh Google. Android Studio merupakan pengembangan dari Eclipse IDE, dan dibuat berdasarkan IDE Java populer, yaitu IntelliJ IDEA. Android studio mempunyai fitur-fitur yang dapat mempermudah penulis dalam penelitiannya.

2.12 Black Box Testing

Pengujian adalah proses pelaksanaan suatu program dengan tujuan menemukan suatu kesalahan. Suatu kasus tes yang baik adalah apabila pengujian tersebut mempunyai kemungkinan menemukan sebuah kesalahan yang tidak terungkap. Suatu pengujian yang sukses adalah bila pengujian tersebut membongkar suatu kesalahan yang awalnya tidak ditemukan. Salah satu dari jenis pengujian yang ada adalah *black box testing*.

Pengujian *software* sangat diperlukan untuk memastikan *software* atau aplikasi yang sudah atau sedang dibuat dapat berjalan sesuai dengan fungsionalitas yang diharapkan. Pengembang atau penguji *software* harus menyiapkan sesi khusus untuk menguji program yang sudah dibuat agar kesalahan ataupun kekurangan dapat dideteksi sejak awal dan dikoreksi secepatnya. Pengujian atau *testing* sendiri merupakan elemen kritis dari jaminan kualitas perangkat lunak dan merupakan bagian yang tidak terpisahkan dari siklus hidup pengembangan *software* seperti halnya analisis, desain, dan pengkodean (Shi, 2010).

Pengujian *software* haruslah dilakukan dalam proses rekayasa perangkat lunak atau *software engineering*. Sejumlah strategi pengujian *software* telah diusulkan dalam literatur. Semuanya menyediakan template untuk pengujian bagi pembuat *software*. Dalam hal ini, semuanya harus memiliki karakteristik umum berupa (Bhat and Quadri, 2015) :

1. *Testing* dimulai pada level modul dan bekerja keluar ke arah integrasi pada sistem berbasis komputer .
2. Teknik *testing* yang berbeda sesuai dengan poin-poin yang berbeda pada waktunya.
3. *Testing* diadakan oleh pembuat atau pengembang *software* dan untuk proyek yang besar oleh group *testing* yang independent.
4. *Testing* dan *Debugging* adalah aktivitas yang berbeda tetapi *debugging* harus diakomodasikan pada setiap strategi *testing*.

Black box testing berfokus pada spesifikasi fungsional dari perangkat lunak. *Tester* dapat mendefinisikan kumpulan kondisi *input* dan melakukan pengetesan pada spesifikasi fungsional program. *Black box testing* bukanlah solusi alternatif dari *white box testing* tapi lebih merupakan pelengkap untuk menguji hal-hal yang tidak dicakup oleh *white box testing*. *Black box testing* cenderung untuk menemukan hal-hal berikut:

1. Fungsi yang tidak benar atau tidak ada.
2. Kesalahan antarmuka (*interface errors*).
3. Kesalahan pada struktur data dan akses basis data.
4. Kesalahan performansi (*performance errors*)

5. Kesalahan inisialisasi dan terminasi.

Black box testing merupakan pengujian yang berfokus pada spesifikasi fungsional dari perangkat lunak, *tester* dapat mendefinisikan kumpulan kondisi *input* dan melakukan pengetesan pada spesifikasi fungsional program (Khan, 201