

BAB II

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Tinjauan Pustaka

Penelitian mengenai pembuatan *remastering* sistem operasi (OS) linux sudah banyak dilakukan. Terdapat beberapa judul penelitian yang berhubungan dengan *remastering* sistem operasi (OS) linux. Penelitian ini juga dapat digunakan sebagai bahan perbandingan dan referensi dalam menulis.

Muhammad Yandi (2010) membuat distro “*Remastering Linux Ubuntu 9.10 Untuk Kegiatan Mata Kuliah Multimedia*”. Pada pembuatan dan perancangan sebuah sistem operasi yang memiliki basis berupa open source akan memampatkan semua aplikasi yang dibutuhkan ke dalam satu wadah cd/dvd dimana isinya adalah aplikasi - aplikasi multimedia yang nantinya digunakan pada mata kuliah praktikum multimedia di laboratorium komputer. Sehingga Dosen maupun mahasiswa dapat menggunakan sistem operasi ini, tanpa harus adanya rasa khawatir atau menggunakan sistem operasi yang ilegal.

Muhammad Asef Yusriyadi (2010) melakukan penelitian mengenai “*Mandriva 2010 Remaster Power Packflash For Distro Linux Desktop*”. Pada penelitian ini membahas pembuatan dan perancangan sebuah distro sistem operasi yang mempunyai basis linux Mandriva PowerPack, dimana di dalam distro ini terdapat berbagai aplikasi standar yang dapat dipergunakan sesuai kegunaannya. Sistem operasi ini menggunakan *remastersys*.

Andreas Setya dan dkk (2012) membuat sebuah distro “*Pembuatan Distro Linux Edukasi untuk Anak-Anak Sekolah Dasar dalam Proses Pengenalan Operating System Open Source dengan Metode Remastering*”. Pada pembuatan dan perancangan sebuah distro linux yang memiliki basis berupa open source ini adalah bertujuan untuk memperkenalkan sebuah sistem operasi kepada anak-anak serta dibuatnya sebuah distro yang baru yang nantinya akan dipergunakan di dunia pendidikan sekolah dasar.

Ade dan Nono Heryana (2014) melakukan penelitian mengenai “*Pengembangan Sistem Operasi Berbasis Open Source Dalam Pemilihan Paket Distribusi Untuk Menunjang Kegiatan Akademik Dilingkungan Unsika*”. Pada penelitian ini membahas pengembangan distro Linux Unsika dengan teknik *remastering*. Linux Unsika dikembangkan dari Ubuntu Linux. Sehingga setelah distro tersebut terinstal di dalam komputer dapat langsung dijalankan tanpa harus menginstal aplikasi-aplikasi tambahan dari *repository* yang ada. Hasil Penelitian yang didapat dari pengembangan distro linux Unsika adalah sebuah sistem operasi open source yang sesuai dengan kebutuhan yang ada di Unsika dan dapat mengimplementasikan perangkat lunak nya dalam menunjang kegiatan akademik di Unsika.

2.2 Dasar Teori

2.2.1 Sistem Operasi

Sistem Operasi atau bisa disebut dengan OS (*Operating System*) merupakan sebuah perangkat lunak (*Software*) yang mempunyai fungsi sebagai sistem yang dapat mengontrol serta memanajemen suatu perangkat keras (*Hardware*) dan operasi dasar sistem yang ada didalamnya. Selain itu juga dapat mengoperasikan aplikasi *software* yang mana sudah diinstall sebelumnya, seperti program pengolah data atau kata dan web browser.

Secara garis besar, sebuah *hardware* yaitu berupa komputer dan disaat itu dinyalakan serta dioperasikan maka keberadaan suatu sistem operasi yang ada didalamnya tersebut berada pada tingkat atau lapisan paling utama dan ditempatkan pada memori komputer nya. Dan *software* lainnya yang telah diinstall akan dapat dioperasikan setelah sistem operasi sudah berjalan terlebih dahulu.

Jika suatu sistem komputer terdiri dari berbagai lapisan dan tingkatan, maka sistem operasi memiliki tugas utama yaitu berupa menghubungkan sebuah perangkat keras dengan perangkat lunak nya. Selain memiliki tugas utama nya sistem operasi ini juga mempunyai tugas-tugas lain nya yaitu menjamin akan kelancaran beberapa aplikasi-aplikasi berbeda yang berjalan dan beroperasi secara

bersamaan dengan baik dan lancar. Disaat aplikasi-aplikasi berbeda tersebut sedang beroperasi secara bersamaan maka secara otomatis sistem operasi melakukan tugas lain nya berupa mengatur dan memanajemen waktu penggunaan memory, sehingga semua kegiatan dan proses nantinya tidak akan saling terganggu satu sama lain nya. Hal ini dilakukan oleh sebuah prosesor (CPU) yang telah dipasang.

Berikut ini adalah bagian-bagian dari sistem operasi yaitu :

1. Mekanisme *Booting* adalah memasukkan kernel ke memory
2. Kernel adalah sebuah program inti yang dimiliki oleh sistem operasi
3. *Command Interpreter* adalah program yang memiliki tugas untuk membaca perintah berupa input dari sang pengguna
4. Pustaka adalah program yang berisi fungsi-fungsi dasar dan standar
5. *Driver* adalah sebuah komponen perangkat lunak yang memiliki tugas sebagai penghubung antara sistem operasi dengan *hardware*

Sistem operasi juga memiliki beberapa jenis atau tipe yaitu jenis yang pertama bernama *desktop OS* merupakan sistem operasi yang memiliki ukuran kapasitas besar serta kompleks sedangkan sistem operasi yang memiliki ukuran kapasitas kecil disebut dengan *real-time OS*. Contoh sistem operasi nya adalah seperti windows, symbian, linux, solaris dan lain-lain nya.

1.2.2 Open Source Software

Eric Raymond merupakan orang yang pertama kali memperkenalkan dan mempopulerkan sistem *Open Source Software*. Arti dari kata *Open Source* didalam sistem ini adalah kode programnya dapat diakses oleh setiap orang, jadi setiap orang dapat melihat kelebihan dan kekurangan dari sistem tersebut serta jikalau mampu, akan dapat dikembangkan dan diperbaiki tanpa adanya pembayaran.

Linux merupakan salah satu perangkat lunak yang dikembangkan melalui sebuah sistem yang berbasis *open source. Software* ini juga bisa disebut dengan *freeSoftware*. Artinya setiap orang dapat mengakses, memiliki, mengoperasikannya dan dapat dikembangkan tanpa adanya pembayaran apapun. Asalkan tidak menghilangkan atau masih tetap menyertakan kode asal program *software* tersebut.

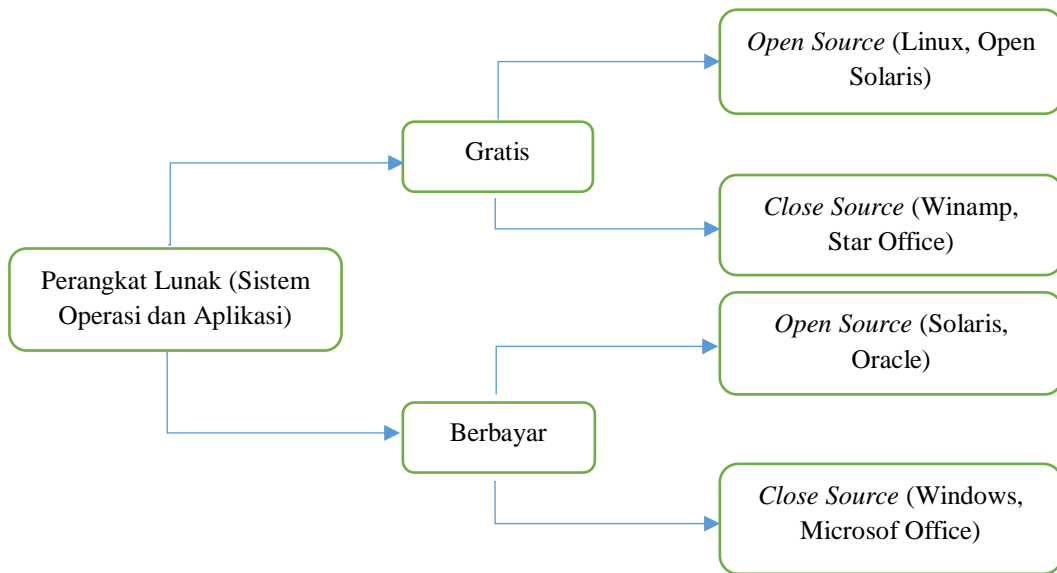
Free software adalah salah satu implementasi dari konsep *open source*. Banyak orang yang salah mengartikan bahwa *free software* itu adalah *software* yang gratisan. Tetapi menurut *Free Software Foundation*, arti dari *free* disini memiliki 4 kebebasan, yaitu :

1. Kebebasan dalam mengoperasikan *software* guna keperluan apapun.
2. Kebebasan dalam menganalisis program akan berjalan, dan mengadaptasinya untuk keperluan sendiri.
3. Kebebasan untuk mendistribusikan kembali kepada orang lain.
4. Kebebasan untuk memperbaiki program, dan menerbitkan perbaikan tersebut secara luas.

Jadi, arti konsep *free* disini bukanlah berarti gratis melainkan bebas. Ada banyak vendor yang mengkomersilkan *free software* mereka, tetapi banyak juga yang benar-benar menggratiskannya.

1.2.3 Free Open Source Software

Menurut David Wheeler (2000), secara umum program yang dinamakan *Free Open Source Software* (FOSS) merupakan suatu program sistem yang memiliki lisensi dan lisensinya itu memberikan sang pengguna atau *user* dapat bebas dalam mengoperasikan, menganalisis, dan dapat mengembangkannya lebih baik lagi tanpa adanya pembayaran *royalty*. Serta dapat bertolak belakang dengan *Close Source* yang menutup kode sumber untuk diketahui masyarakat dan harus membayar *royalty* kepada penggunanya. Berikut gambar perbandingan sistem operasi dan aplikasi *Open Source* dan *Close Source* :



Gambar 2.1 Perbandingan *Open Source* dan *Close Source*

Pengertian dari *open source software* itu sendiri harus mempunyai standard antara lain :

1. Pembagian dari sebuah program sistem perangkat lunak yang secara bebas, artinya didapatkan secara gratis atau tidak bayar.
2. Kode sumber atau *source code* dari sebuah program *software* harus dapat dicantumkan ditempat yang nantinya dapat diakses setiap orang, misalnya melalui jaringan internet dimana setiap orang dapat mengunduh program tanpa dikenakan biaya.
3. Dalam memelihara sebuah karakter dari suatu *source code* yang dimiliki oleh sipembuat *software* maka lisensi yang digunakan pada program dapat melarang pendistribusian *source code* yang telah dimodifikasi, kecuali lisensi itu mengizinkan pendistribusian *patch file* (potongan file program) yang bertujuan memodifikasi program tersebut dengan disertakan *source code* dari program asal.
4. Sebuah lisensi yang berbasis *open source* dilarang untuk membuat suatu diskriminasi baik itu kelompok maupun individu.
5. Dilarang menghalangi pengguna untuk memanfaatkan fasilitas *open source* dalam suatu bidang tertentu. Sebagai contoh, tidak ada

pembatasan program tersebut terhadap penggunaan dalam bidang bisnis, atau terhadap pemanfaatan dalam bidang riset genetic.

6. Sebuah lisensi dilarang memproduksi suatu program yang mana program *software* tersebut hanya memiliki kepentingan dan keuntungan untuk sendiri. Hak-hak yang tercantum pada suatu program tidak boleh tergantung pada apakah program tersebut merupakan bagian dari satu distribusi perangkat lunak tertentu atau tidak. Sekalipun program diambil dari distribusi tersebut dan digunakan atau didistribusikan selaras dengan lisensi program itu.

Dengan demikian, ada satu hal yang perlu kita garis bawahi yaitu mengenai pengertian dari *free* ini, dimana arti *free* dapat dideskripsikan melalui lima kegiatan antara lain :

1. Kebebasan dalam mengoperasikan sebuah program.
2. Kebebasan dalam mengetahui suatu *source code* program *software*.
3. Kebebasan dalam mendistribusikan program *software*.
4. Kebebasan dalam mengembangkan program *software*.
5. Kebebasan dalam menjual program *software*.

1.2.4 Sejarah Linux

Profesor Andrew Tanenbaum adalah orang yang pertama kali membuat sebuah sistem operasi dengan nama Minix. Bermula dari sistem operasi Minix inilah sistem operasi Linux terbentuk yaitu gabungan nama antara nama si pencipta dengan nama sistem operasinya UNIX.

Sedangkan Linus B. Torvald merupakan orang yang pertama kali mengoperasikan sistem operasi Minix, ia seorang mahasiswa Universitas Helsinki. Selama dia mengoperasikan sistem operasi Minix ini lalu ia mendapatkan beberapa pandangan mengenai kelebihan dan kekurangan sistem operasi Minix tersebut. Dan ditahun 1991 Linus B. Torvald merancang dan membuat sebuah sistem operasi yang baru dengan nama Linux yang mana turunan dari UNIX. Pada saat itu sistem nya belum stabil, lalu pada Oktober 1991 Linus B. Torvald meminta bantuan

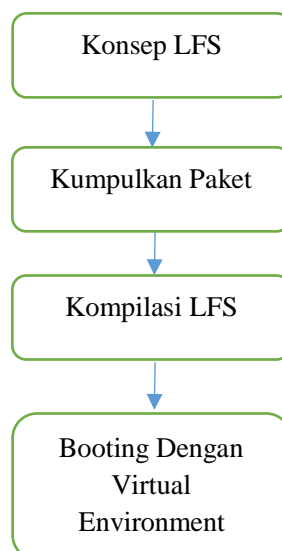
kepada programmer lain untuk dapat mengembangkan sistem operasi Linux ini menjadi lebih stabil dan baik lagi. Dan akhirnya ditahun 1994 Linus B. Torvald meluncurkan sistem operasi Linux versi 1.0, dimana di versi 1.0 ini sistem operasinya sudah dilengkapi dengan *feature symmetric multiprocessing* yaitu kemampuan dalam mengatur dan membagi tugas dari banyaknya CPU serta *feature preemptive multiprocessing* yaitu kemampuan dalam mengatur dan membagi sumber daya CPU saat banyaknya penggunaan aplikasi.

1.2.5 Konsep Pembuatan Distribusi Linux

Dalam pembuatan sebuah distro perangkat lunak harus memiliki tujuan berupa memberikan manfaat yang banyak serta distro tersebut dapat dikembangkan lebih baik lagi. Adapun metode – metode pembuatan distro yaitu :

1.2.5.1 LFS (Linux From Scratch)

Linux From Scratch merupakan salah satu metode pembuatan sebuah distro perangkat lunak secara individu atau mandiri. Arti dari mandiri disini adalah aplikasi-aplikasi yang telah di instal *source code* nya didapat melalui *pristine code* atau kode sumber yang murni. Adapun kekurangan dari metode ini adalah distro yang sudah dibuat sifatnya tidak bisa dikembangkan lagi biasanya digunakan untuk diri sendiri atau pribadi, distro yang nantinya akan diproduksi tidak dapat kerjakan dalam satu ISO yang *installable* dan sangat sulit untuk dapat didistribusikan lagi.



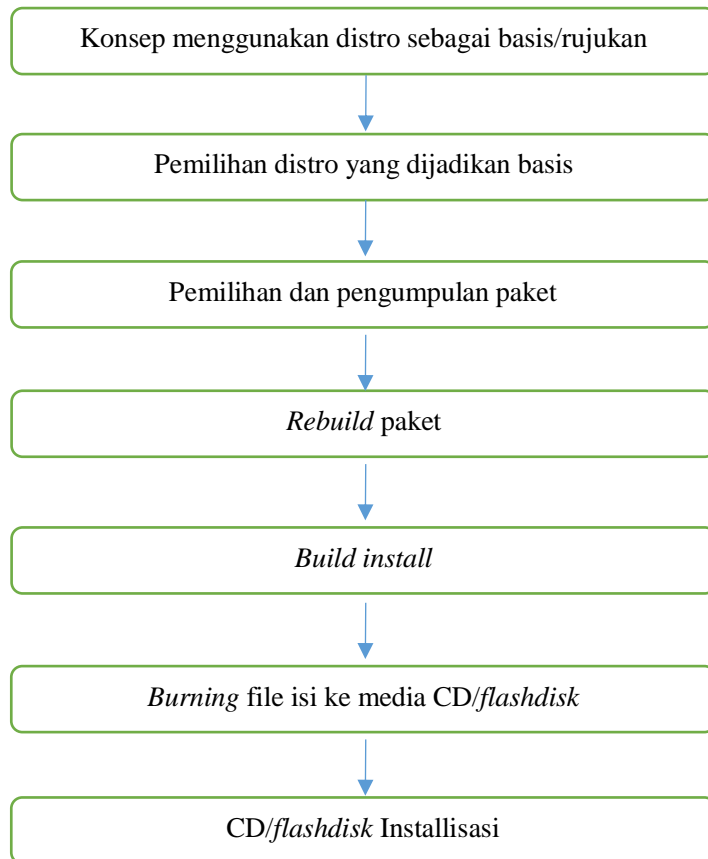
Gambar 2.2 Bagan Sederhana Pembuatan Distro Dengan Konsep LFS

Berikut ini adalah proses - proses pembuatan distro LFS secara singkat :

1. Rancanglah semua paket-paket yang diperlukan.
2. Rancanglah direktori secara statik dengan cara mengkompilasi paket-paket sistem LFS yang telah dirancang sebelumnya sebagai tempat pembuatan file LFS.
3. Selanjutnya adalah merancang sistem-sistem LFS yang telah dikompilasi dan terinstall agar dapat melakukan sebuah *booting*, dengan cara menggunakan sebuah sistem yang bernama virtual environment, maka nanti LFS akan membangun sebuah sistem yang nantinya dapat membantu proses kustomisasi dan kecepatan loading distro tersebut karena telah disesuaikan dengan platform nya dimana distro akan ditempatkan. Maka proses ini akan berjalan dengan memindahkan semua *library* yang terdapat dalam satu file yang diposisikan pada setiap file dengan fungsinya masing-masing. Sehingga nanti setiap paket akan dapat berdiri sendiri tanpa tergantung dengan *library* yang ada, akan tetapi apabila dilakukan suatu proses update maka file yang melakukan perubahan adalah keseluruhan sistem nya. Karena proses kompilasi ini sehingga paket yang ada dapat berdiri sendiri adalah salah satu bagian dari proses pembangunan sistem LFS. Perubahan tersebut nantinya akan menambah waktu pembuatan yang pada dasarnya dapat dimanfaatkan untuk mengoptimalkan paket aplikasi yang ada.

1.2.5.2 Remaster

Metode yang kedua ini merupakan metode pembuatan distro yang banyak orang atau pengembang gunakan, sekitar 80% distro linux yang sudah beredar dan diproduksi yang dibuat menggunakan konsep metode *remaster* ini. Dikarenakan kepopuleran metode ini banyak pengembang sudah menjadikan metode ini sebagai rujukan atau basis dalam membuat sebuah distro perangkat lunak. Kelebihan dari metode ini salah satunya adalah distronya dapat dikembangkan lagi oleh pengembang lain nya untuk jadi lebih baik lagi.



Gambar 2.3 Bagan Sederhana Pembuatan Distro Dengan Konsep *Remaster*

1.2.6 Komponen Sistem Linux

Sistem linux terdiri dari tiga bagian kode penting yaitu:

1. Kernel

Bertanggung jawab untuk mengatur dan menjaga seluruh kerangka utama sebuah inti sari dari suatu sistem operasi, seperti memori virtual dan proses – proses keseluruhan sistem.

2. System Libraries

Memilih beberapa berkas fungsi standar agar aplikasi-aplikasi nya dapat terhubung dengan kernel dan mengimplementasikan nya agar semua fungsi sistem operasi dapat beroperasi dengan baik dan maksimal. *System Libraries* juga mengoleksi serta menjaga suatu fungsi *system call* jikalau nanti dibutuhkan guna mengatur fungsi-fungsi tersebut.

3. System Utilities

Merupakan suatu rancangan sistem yang melakukan pekerjaan manajemen secara individu dan khusus.

1.2.7 Linux Ubuntu 17.10

Ubuntu merupakan sebuah distribusi atau turunan dari Linux yang memiliki basis terhadap Debian dan mempunyai interface desktop. Nama Ubuntu berasal dari sebuah konsep ideologi benua Afrika lebih tepatnya di Afrika Selatan, jadi kata Ubuntu diambil dari sebuah bahasa kuno Afrika, yang memiliki arti rasa berperikemanusiaan kepada manusia lainnya.

Linux Ubuntu telah meluncurkan beberapa versinya salah satu yang terbarunya adalah Linux Ubuntu versi 17.10 yang mana pada versi ini sebelumnya menggunakan sistem Unity lalu diganti dengan menggunakan sistem GNOME, dengan tampilan animasi window-nya lebih halus dan lebih bagus serta yang lebih pentingnya adalah versi ini sistem-nya lebih stabil dibandingkan dengan versi terdahulunya. Adapun kelebihan lain dari versi 17.10 salah satunya adalah dapat secara langsung menginstall program atau aplikasi lainnya secara mudah dikarenakan di versi ini sudah disediakan Apps Store semacam Google Playstore jadi pengguna hanya mencari software yang diinginkan lalu klik install kemudian tunggu dan selesai.

1.2.8 GCompris

GCompris merupakan salah satu *software* aplikasi pendukung distro Linux Ubuntu berbasis *open source* yang isinya berupa program-program kegiatan belajar yang ditujukan kepada anak usia 2 tahun sampai 10 tahun, dimana dalam kegiatan belajarnya dapat dilakukan sambil bermain dikarenakan dasar program dari *software* aplikasinya sendiri adalah *games* atau permainan yang beredukasi. Aplikasi ini memiliki kurang lebih 100 kegiatan edukasi yang dapat dioperasikan, berikut adalah beberapa contoh daftar kegiatan edukasi :

- Membaca : belajar membaca, pengenalan huruf dan kata, dll.
- Matematika : operasi angka (penambahan, pengurangan, perkalian, pembagian), dll.
- Geografi : bagian-bagian negara, daerah, dll.
- Sains : pengetahuan mengenai energi.
- Lainnya : permainan, catur, warna, ingatan, dll.

Aplikasi ini juga dilengkapi dengan visual suara berbahasa Inggris disetiap kegiatan edukasinya, yang sangat dapat membantu sang anak dalam mengucapkan bahasa Inggris secara baik dan benar dalam bentuk huruf, kata, maupun angka. Jadi aplikasi ini sangat memungkinkan dapat menunjang sang anak dalam melakukan kegiatan belajarnya disekolah khususnya lagi dikegiatan belajar matematika.

1.2.9 Childsplay

Childsplay merupakan sebuah *software* yang dapat dioperasikan di *platform* Linux, OSX, maupun Windows dimana aplikasi *software* ini berisikan kumpulan permainan edukasi yang mendidik anak usia 2 tahun hingga 10 tahun. Dengan berbagai macam kegiatan permainan edukasi yang dapat dioperasikan, seorang anak dapat senang serta nyaman saat memainkan aplikasi *software* ini dengan orang tua ataupun guru dapat membantu dan membimbingnya. Dikarenakan tampilan atau *interface* nya yang menarik, lucu dan sangat sesuai dengan kebutuhan sang anak dalam melakukan kegiatan belajarnya baik dilingkungan sekolah maupun dirumah.

1.2.10 Tuxmath

Tuxmath adalah sebuah *software* yang telah dibuat oleh perusahaan yang mana memiliki proyek bernama Tux4kids dengan basis *open source*. Tuxmath juga merupakan aplikasi edukasi khususnya dibidang pembelajaran matematika yaitu pengoperasian hitung penjumlahan, pengurangan, perkalian, dan pembagian dengan metode program permainan *arcade* yang memiliki beberapa tingkatan atau level dalam permainan edukasinya. Aplikasi *software* ini memiliki platform utama dalam cerita permainan *arcade* nya yaitu kita akan berperan sebagai seekor penguin yang memiliki tugas melindungi teman-temannya dari hantaman meteor yang akan berjatuhan.

Untuk menghancurkan hantaman meteor yang berjatuhan, pemain atau penguin peran yang kita mainkan harus dapat menjawab setiap soal yang tertera pada hantaman meteor yang berjatuhan yaitu berupa soal-soal operasi hitungan angka. Jika pemain dapat menjawab soal operasi hitung nya secara terus menerus

maka tingkatan atau level nya pun akan meningkat sejalan dengan pemain dapat atau tidak menjawab soal tersebut.

1.2.11 Komposisi Warna

Dalam merancang sebuah *software* sistem operasi, salah satu aspek yang perlu diperhatikan adalah desain antarmuka *software* tersebut. Untuk dapat membuat sebuah desain yang nyaman dilihat oleh *user*, maka hal yang perlu dilakukan adalah memilih warna yang terbaik. Warna memberikan peran penting untuk kenyamanan interaksi *user* dengan antar muka *software*. Berikut ini merupakan tabel komposisi kombinasi warna pada desain antarmuka, yang terdiri dari komposisi warna terbaik hingga komposisi warna terburuk dalam hubungannya dengan aspek psikologis *user*.

Tabel 2.1 Komposisi Kombinasi Warna Terbaik

Background	Garis dan Teks (Normal)	Garis dan Teks (Tebal)
Putih	Biru (94%), Hitam (63%), Merah (25%)	Hitam (69%), Biru (63%), Merah (31%)
Hitam	Putih (75%), Kuning (63%)	Kuning (69%), Putih (59%), Hijau (25%)
Merah	Kuning (75%), Putih (56%), Hitam (44%)	Hitam (50%), Kuning (44%), Putih (44%), Cyan (31%)
Hijau	Hitam (100%), Biru (56%), Merah (25%)	Hitam (69%), Merah (63%), Biru (31%)
Biru	Putih (81%), Kuning (50%), Cyan (25%)	Kuning (38%), Magenta (31%), Hitam (31%), Cyan (31%), Putih (25%)
Cyan	Biru (69%), Hitam (56%), Merah (37%)	Merah (56%), Biru (50%), Hitam (44%), Magenta (25%)
Magenta	Hitam (63%), Putih (56%), Biru (44%)	Biru (50%), Hitam (44%), Kuning (25%)
Kuning	Merah (63%), Biru (63%), Hitam (56%)	Merah (75%), Biru (63%), Hitam (50%)

Tabel 2.2 Komposisi Kombinasi Warna Terburuk

Background	Garis dan Teks (Normal)	Garis dan Teks (Tebal)
Putih	Kuning (100%), Cyan (94%)	Kuning (94%), Cyan (75%)
Hitam	Biru (87%), Merah (44%), Magenta (25%)	Biru (81%), Magenta (31%)
Merah	Magenta (81%), Biru (44%), Hijau & Cyan (25%)	Magenta (69%), Biru (50%), Hijau (37%), Cyan (25%)
Hijau	Cyan (81%), Magenta (50%), Kuning (37%)	Cyan (81%), Magenta & Kuning (44%)
Biru	Hijau (62%), Merah & Hitam (37%)	Hijau (44%), Merah & Hitam (31%)
Cyan	Hitam (81%), Kuning (75%), Putih (31%)	Kuning (69%), Hijau (62%), Putih (56%)
Magenta	Hijau (75%), Merah (56%), Cyan (44%)	Cyan (81%), Hijau (69%), Merah (44%)
Kuning	Putih & Cyan (81%)	Putih (81%), Cyan (56%), Hijau (25%)

Dari tabel diatas dapat dilihat bahwa hal yang harus diperhatikan dalam melakukan pemilihan warna saat mendesain perangkat lunak antara lain sebagai berikut :

- a. Menghindari perpaduan warna yang membuat mata menjadi lelah, seperti warna cyan, magenta, kuning secara bersama.
- b. Untuk pemakaian pengguna pada usia tua, sebaiknya mempergunakan warna yang tajam.
- c. Menghindari penggunaan warna merah dan hijau dalam skala besar pada tempat berseberangan. Dalam hal ini kombinasi yang terbaik adalah warna biru dengan kuning.
- d. Perpaduan warna hijau dan biru secara psikologis menimbulkan kesan yang tidak baik. Warna berlawanan yang dapat dipadukan adalah warna biru–kuning dan hijau-merah.
- e. Menghindari penggunaan warna tunggal untuk membantu user yang mempunyai kekurangan dalam melihat warna.

1.2.12 Teori Kognitif

Kata kognitif berasal dari bahasa latin yaitu “Cogitare” yang mempunyai makna berfikir dan juga berasal dari bahasa inggris yaitu “Cognitive” yang mempunyai makna mengerti. Sedangkan didalam kamus besar bahasa Indonesia, kata kognitif memiliki arti yaitu segala suatu hal yang berhubungan berdasarkan pengetahuan atau wawasan yang empiris (pengalaman yang diperoleh dari percobaan, penemuan, penelitian yang telah dilakukan).

Dewwy merupakan orang yang pertama kali mengemukakan teori belajar kognitif, kemudian diteruskan lagi oleh Jean Piaget, dan kawan – kawan nya. Jean Piaget juga adalah seorang pelopor yang melopori serta mengembangkan prinsip kognitif kedalam dunia pendidikan yaitu teori belajar kognitif. Teori belajar kognitif adalah teori yang menerangkan segala sesuatu proses belajar baik itu pemikirannya maupun perbedaan dalam kondisi mentalnya dimana teori belajarnya lebih mengutamakan proses belajar sang anak dari pada hasil belajarnya.

Menurut Jean Piaget, ada empat tahapan dalam perkembangan kognitif adalah sebagai berikut :

1. Tahap Sensorimotorik (usia 0-2 tahun)

Tahapan dimana sang anak dapat mengerti, memahami, mengetahui suatu hal tentang dunia yaitu dengan cara memanfaatkan kinerja sensoris dari tubuh sang anak seperti mendengar maupun melihat dan juga dari proses tindakan motorik tubuh sang anak. Jadi melalui gerakan atau suara yang dapat dirasakan oleh alat indranya, sang anak dapat belajar suatu hal yang baru kemudian dapat dikembangkan kemampuannya sedikit demi sedikit.

2. Tahap praoperasional (usia 2-7 tahun)

Tahapan dimana sang anak sudah mulai menggambarkan dunia hal yang baru tersebut melalui kata - kata maupun tingkah lakunya. Akan tetapi sang anak belum mampu mengeksekusi (tindakan mental) dunia hal baru tersebut secara fisik.

3. Tahap operasional konkret (usia 7-11 tahun)

Tahapan dimana sang anak sudah mulai dapat berpikir secara rasional dan logis mengenai tindakan maupun kejadian yang sifatnya nyata atau benar – benar ada (berwujud, dapat dilihat, diraba, dirasakan, dan sebagainya).

4. Tahap operasional formal (11 tahun ke atas)

Tahapan dimana sang anak sudah dapat berpikir secara idelais dan lebih logis. Pada usia ini sang anak mulai mengalami perkembangan penalaran nya, jadi sang anak sudah mulai mampu untuk memecahkan suatu masalah secara logis.