BAB 4
IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi

Tahap implementasi sistem merupakan tahap penerjemahan perancangan berdasarkan hasil analisis ke dalam suatu bahasa pemrograman yang digunakan serta penerapan perangkat lunak yang dibangun pada lingkungan yang sesungguhnya. Tujuan implementasi adalah untuk menerapkan perancangan yang telah dilakukan terhadap sistem, sehingga pengguna dapat memberi masukan demi berkembangnya sistem yang telah dibangun. Adapun pembahasan untuk implementasi sistem meliputi kebutuhan perangkat keras, perangkat lunak, pengujian *alpha*, dan pengujian *beta*.

4.1.1 Implementasi Perangkat Pendukung yang digunakan

Perangkat keras dan perangkat lunak yang digunakan untuk melakukan implementasikan dan pengujian dari aplikasi adalah sebagai berikut :

4.1.1.1 Implementasi Perangkat Keras

Spesifikasi perangkat keras yang digunakan untuk menjalankan Aplikasi Pembelajaran Pengenalan dan Perakitan Komputer adalah sebagai berikut :

Tabel 4.1 Kebutuhan Perangkat Keras

No	Perangkat Keras	Spesifikasi
1	*Processor*	*Intel Core i3 2.2 Ghz*
2	*Harddisk*	500 Gb
3	*RAM*	4 Gb
4	*VGA*	1 Gb
5	*Monitor*	*Monitor* dengan resolusi 1366 x 768 *pixel*
6	*Mouse*	*Logitech*
7	*Keyboard*	Ya
8	*Speaker*	Ya

4.1.1.2 Implementasi Kebutuhan Perangkat Lunak

Spesifikasi kebutuhan perangkat lunak yang digunakan untuk membangun Aplikasi ini adalah sebagai berikut :

Tabel 4.2 Kebutuhan Perangkat Lunak

No	Perangkat Lunak	Keterangan
1	*Microsoft Window 7*	Digunakan sebagai sistem operasi
2	*Adobe Flash CS6*	Digunakan sebagai *tools* pembangun
3	*Adobe Photoshop CS6*	Digunakan sebagai *tools desain*
4	*Adobe Flash Player 10*	Digunakan sebagai *tools compiler*

4.1.2 Implementasi Aplikasi

Aplikasi Pembelajaran Pengenalan dan Perakitan Komputer merupakan sebuah aplikasi untuk membantu proses belajar siswa di SMAN 27 Bandung mengenai materi pengenalan dan perakitan komputer. Dalam aplikasi ini diberikan berbagai macam materi pengenalan komputer secara lengkap baik *hardware* maupun *software*, juga diberikan tutorial mengenai perakitan beserta instalasi komputer yang diberikan dalam bentuk video, selain itu para siswa juga diberikan pelatihan perakitan komputer secara simulasi serta diberikan pula soal latihan sebagai bahan evaluasi dari hasil pembelajaran.

Aplikasi ini dibangun dengan basis *desktop* dan menggunakan *tools Adobe Flash CS6* dengan bahasa pemrograman *Action Script 2.0*.

4.1.3 Implementasi Antarmuka

Adapun implementasi antarmuka dari Aplikasi Pembelajaran Pengenalan dan Perakitan Komputer adalah sebagai berikut :

1. Tampilan Halaman Utama

   Antarmuka halaman utama aplikasi merupakan halaman yang akan muncul ketika aplikasi dijalankan. Terdapat 6 menu yang dapat *user* pilih yaitu materi, tutorial, simulasi, latihan, petunjuk dan menu untuk keluar aplikasi.

2. Tampilan Pilih Menu Materi

   Tampilan pilih menu materi menupakan sub menu dari materi materi, dalam menu materi terbagi menjadi 2 sub menu yaitu materi *hardware* dan materi *software*.

3. Tampilan Pilih Menu Materi *Hardware*

   Menu materi *hardware* merupakan menu yang akan menampilkan materi pengenalan lengkap mengenai komponen *hardware* komputer, pengenalan komponen yang diberikan yaitu *motherboard*, *processor*, RAM, VGA, *harddisk, optical drives* dan lain-lain yang dibahas secara *detail*.

4. Tampilan Pilih Menu Materi *Software*

   Selain materi *hardware*, dalam aplikasi ini juga diberikan materi mengenai pengenalan *software* yang umumnya digunakan dalam mengoperasikan komputer seperti pengenalan sistem operasi serta *software-software* lainnya.

5. Tampilan Pilih Menu Tutorial

   Menu ini merupakan menu pilihan yang akan menampilkan tutorial. Menu ini memiliki 2 sub menu didalamnya yaitu tutorial perakitan dan tutorial instalasi yang diberikan dalam bentuk video agar lebih terlihat variatif.

6. Tampilan Menu Tutorial Perakitan

   Pada menu tutorial perakitan akan menampilkan video mengenai perakitan komputer, mulai dari merakit *motherboard* hingga menjadi sebuah *PC*.

7. Tampilan Menu Tutorial Instalasi

   Selain tutorial perakitan, dalam aplikasi ini juga diberikan contoh mengenai cara instalasi dari sebuah sistem operasi yaitu instalasi *windows 7*.

8. Tampilan Pilih Menu Simulasi

   Pada menu ini *user* dapat melakukan proses simulasi perkitan komputer. Sebelum melakukan simulasi, *user* diberikan petunjuk untuk melakukan simulasi kemudian pilih tombol panah untuk melanjutkan ke tahap simulasi.

9. Tampilan Pilih Menu Latihan Simulasi

   Pada menu ini *user* akan diberikan latihan simulasi perakitan sebelum ia benar-benar melakukan simulasi perakitan komputer.

10. Tampilan Latihan Simulasi *Power Supply*

    Tahap pertama yang harus *user* lakukan adalah dengan memasang *power supply* terlebih dahulu pada *casing* komputer, kemudian dilanjutkan dengan memasang baud-baud agar kokoh menempel pada *casing*.

11. Tampilan Latihan Simulai *Motherboard*

    Tahap selanjutnya adalah dengan merakit *motherboar*d. Adapun komponen yang dipasangkan pada *motherboard* yaitu *Processor*, *RAM*, *Pasta Processor* dan *Heatsink.*

12. Tampilan Latihan Simulasi *Eksternal Card*

    Eksternal Card merupakan kartu tambahan yang biasa dipasangakn pada komputer. Kartu-kartu tersebut meliputi *NIC, Wireless NIC* dan *VGA.*

13. Tampilan Latihan Simulasi *Eksternal Drives*

    Merupakan tampilan dimana *user* akan melakukan latihan simulasi memasang *drive* tambahan seperti *Harddisk*, DVD *ROM* dan *Floppy*.

14. Tampilan Latihan Simulasi *Internal Cable*

    Setelah komponen-komponen tadi dipasangkan, maka tahap selanjutnya adalah memasangkan kabel untuk menghubungkan antara satu komponen dengan komponen lain agar komputer dapat beroperasi. Adapun *internal cabel* yang akan dipasangkan adalah kabel *optical drives*, kabel *floppy* dan kabel *harddisk*.

15. Tampilan Latihan Simulasi *Eksternal Cable*

    Selain kabel *internal* yang fungsinya untuk menghubungkan satu komponen dengan komponen lainnya didalan komputer, juga terdapat kabel *eksternal* yang dapat menghubungkan komputer dengan *device* lain seperti *monitor, keyboard, mouse* dan lain-lain.

16. Tampilan Tes Simulasi

    Tampilan yang akan menampilkan proses dari simulasi dari awal hingga akhir pada setiap komponennya.

17. Tampilan Hasil Simulasi Perakitan

    Halaman ini akan menampilkan hasil dari simulasi yang telah dikerjakan.

18. Tampilan Menu Latihan Soal

    Menu latihan merupakan menu yang menampilkan daftar soal latihan yang harus *user* selesaikan sebagai bahan evaluasi atas materi yang telah diberikan. Selain itu, setelah semua soal dijawab, *user* juga dapat melihat langsung nilai dari hasil latihan soal yang dilakukan setelah menjawab semua pertanyaan yang diberikan, juga *user* dapat mengulang menjawab semua pertanyaan dengan memilih tombol coba lagi pada tampilan nilai akhir.

19. Tampilan Pesan Kembali ke Halaman Utama

    Tampilan yang akan muncul ketika tombol *home* di klik.

20. Tampilan Pesan Kembali Ke Menu Sebelumnya

    Tampilan pesan ketika user hendak kembali ke menu sebelumnya.

21. Tampilan Antarmuka Pesan Mengulang Soal

    Tampilan yang akan muncul ketika *user* memilih tombol coba lagi pada tampilan nilai akhir.

4.2 Pengujian Sistem

Pengujian sistem dilakukan untuk menentukan kesalahan atau kekurangan pada perangkat lunak yang diuji. Pengujian bermaksud untuk mengetahui perangkat lunak yang dibangun sudah sesuai dengan perancangan yang dibuat dan memenuhi kebutuhan *user*.

Pengujian yang dilakukan yaitu pengujian *alpha* dan pengujian *beta*. Pengujian *alpha* dilakukan dengan pengujian *black box* dan *white box*. Pengujian *black box* digunakan untuk mengetahui kesalahan proses secara fungsional sedangkan *white box* digunakan untuk menguji performa dari algoritma dari metode yang digunakan. Pengujian *beta* digunakan untuk mengetahui tanggapan pengguna terhadap aplikasi dengan menggunakan kuesioner.

4.2.1 Pengujian *Alpha*

Pengujian fungsional yang digunakan untuk menguji sistem yang baru adalah metode pengujian *alpha*. Pengujian *alpha* dilakukan dengan menggunakan metode *black box* dan *white box*. Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak sedangkan *white box* berfokus pada algoritma dari metode yang digunakan. Skenario pengujian dari aplikasi pembelajaran pengenalan dan perakitan komputer dapat dilihat seperti dibawah ini :

Tabel 4.3 Skenario Pengujian *Alpha*

No	Komponen Yang Di Uji	Skenario Pengujian	Jenis Pengujian
1	Menu Utama	Memilih Menu Materi	*Black box*
		Memilih Menu Tutorial	*Black box*
		Memilih Menu Simulasi	*Black box*
		Memilih Menu Latihan	*Black box*
		Memilih Menu Petunjuk	*Black box*
		Memilih Menu Keluar	*Black box*
2	Menu Materi	Memilih Menu Materi *Hardware*	*Black box*
		Memilih Menu Materi *Software*	*Black box*
3	Menu Materi *Hardware*	Memilih Tombol *Home*	*Black box*
		Memilih Tombol X	*Balck box*
		Memilih Tombol *Next*	*Black box*
		Memilih Tombol *Back*	*Black box*
4	Menu Materi *Software*	Memilih Tombol *Home*	*Black box*
		Memilih Tombol X	*Black box*
		Memilih Tombol *Next*	*Black box*
		Memilih Tombol *Back*	*Black box*

5	Menu Tutorial	Memilih Menu Tutorial Perakitan	*Black box*
		Memilih Menu Tutorial Instalasi	*Black box*
6	Menu Tutorial Perakitan	Memilih Tombol *Home*	*Black box*
		Memilih Tombol X	*Black box*
		Memilih Tombol *Play*	*Black box*
		Memilih Tombol *Stop*	*Black box*
		Memilih Tombol *Rewind*	*Black box*
		Memilih Tombol *Forward*	*Black box*
		Memilih Tombol Suara	*Black box*
7	Menu Tutorial Instalasi	Memilih Tombol *Home*	*Black box*
		Memilih Tombol X	*Black box*
		Memilih Tombol *Play*	*Black box*
		Memilih Tombol *Stop*	*Black box*
		Memilih Tombol *Rewind*	*Black box*
		Memilih Tombol *Forward*	*Black box*
		Memilih Tombol Suara	*Black box*
8	Menu Simulasi	Memilih Menu Latihan Simulasi	*Black box*
		Memilih Menu Tes Simulasi	*Black box*
		Memilih Tombol *Home*	*Black box*
		Memilih Tombol *Next*	*Black box*
		Memilih Tombol cek	*Black box*
9	Menu Latihan Simulasi	Memilih Menu *Powersupply*	*Black box*
		Memilih Menu *Motherboard*	*Black box*
		Memilih Menu *Internal Card*	*Black box*
		Memilih Menu *Eksternal Drives*	*Black box*
		Memilih Menu *Internal Cable*	*Black box*
		Memilih Menu *Eksternal Cable*	*Black box*
10	Menu Tes Simulasi	Memilih Menu Cek	*Black box*

11	Menu Latihan	Memilih Tombol *Home*	*Black box*
		Memilih Tombol X	*Black box*
		Memilih Pilihan Jawaban A	*Black box*
		Memilih Pilihan Jawaban B	*Black box*
		Memilih Pilihan jawaban C	*Black box*
		Memilih Pilihan Jawaban D	*Black box*
		Memilih Tombol Coba Lagi	*Black box*
12	Menu Petunjuk	Memilih Menu Petunjuk	*Black box*
13	Pesan	Memilih Tombol Ya	*Black box*
		Memilih Tombol Tidak	*Black box*
14	Pengecekan kesalahan	Metode *Decision Tree*	*White box*

4.2.1.1 Kasus dan Hasil Pengujian *Black Box*

Pengujian perangkat lunak yang sudah dibangun dilakukan dengan menggunakan metode *black box*, berdasarkan pengamatan yang sudah dilakukan maka dihasilkan kesimpulan dari pengujian sebagai berikut :

Tabel 4.4 Pengujian *Black Box*

No	Kasus Yang akan Diuji	Skenario dan Hasil Uji		
		Skenario Uji	Hasil yang diharapkan	kesimpulan
1	Menu Utama	Memilih Menu Materi	Sistem menampilkan pilihan menu materi	[√] Berhasil [ ] Tidak Berhasil
		Memilih Menu Tutorial	Sistem menampilkan pilihan menu tutorial	[√] Berhasil [ ] Tidak Berhasil
		Memilih Menu Simulasi	Sistem menampilkan halaman simulasi	[√] Berhasil [ ] Tidak Berhasil

		Memilih Menu latihan	Sistem menampilkan daftar soal latihan	[√] Berhasil [ ] Tidak Berhasil
2	Menu Materi	Memilih Menu Materi *Hardware*	Sistem menampilkan materi pengenalan *hardware*	[√] Berhasil [ ] Tidak Berhasil
		Memilih Menu Materi *Software*	Sistem menampilkan materi pengenalan *software*	[√] Berhasil [ ] Tidak Berhasil
3	Menu Materi *Hardware*	Memilih Tombol *Home*	Sistem menmpilkan halaman utaman aplikasi	[√] Berhasil [ ] Tidak Berhasil
		Memilih Tombol X	Sistem menampilkan halaman pilih menu	[√] Berhasil [ ] Tidak Berhasil
		Memilih Tombol *Next*	Sistem menuju ke halaman selanjutnya	[√] Berhasil [ ] Tidak Berhasil
		Memilih Tombol *Back*	Sistem mengembalikan ke halaman yang sebelumnya	[√] Berhasil [ ] Tidak Berhasil
4	Menu Materi Software	Memilih Tombol *Home*	Sistem menampilkan halaman utama	[√] Berhasil [ ] Tidak Berhasil
		Memilih Tombol X	Sistem menampilkan halaman pilih menu	[√] Berhasil [ ] Tidak Berhasil
		Memilih Tombol *Next*	Sistem menju ke halaman selanjutnya	[√] Berhasil [ ] Tidak Berhasil
		Memilih Tombol *Back*	Sistem mengembalikan ke halaman sebelumnya	[√] Berhasil [ ] Tidak Berhasil

5	Menu Tutorial	Memilih Menu Tutorial Perakitan	Sistem menampilkan video tutorial perakitan	[√] Berhasil [ ] Tidak Berhasil
		Memilih Menu Tutorail Instalasi	Sistem menampilkan video tutorial instalasi	[√] Berhasil [ ] Tidak Berhasil
6	Menu Tutorial Perakitan	Memilih Tombol *Home*	Sistem menampilkan halaman utama	[√] Berhasil [ ] Tidak Berhasil
		Memilih Tombol X	Sistem kembali ke halaman pilih menu	[√] Berhasil [ ] Tidak Berhasil
		Memilih Tombol *Play*	Sistem memutar video tutorial	[√] Berhasil [ ] Tidak Berhasil
		Memilih Tombol *Stop*	Sistem menghentikan pemutaran video tutorial	[√] Berhasil [ ] Tidak Berhasil
		Memilih Tombol *Rewind*	Sistem mengulang pemutaran video	[√] Berhasil [ ] Tidak Berhasil
		Memilih Tombol *Forward*	Sistem mempercepat pemutaran video	[√] Berhasil [ ] Tidak Berhasil
		Memilih Tombol Suara	Sistem menampilkan pengaturan *volume* suara	[√] Berhasil [ ] Tidak Berhasil
7	Menu Tutorial Instalasi	Memilih Tombol *Home*	System kembali ke halaman utama	[√] Berhasil [ ] Tidak Berhasil
		Memilih Tombol X	Sistem kembali ke halaman pilih menu	[√] Berhasil [ ] Tidak Berhasil
		Memilih Tombol *Play*	Sistem mulai memutar video tutorial	[√] Berhasil [ ] Tidak Berhasil
		Memilih Tombol *Stop*	Sistem menghentikan pemutaran video tutorial	[√] Berhasil [ ] Tidak Berhasil

		Memilih Tombol *Rewind*	Sistem mengulang pemutaran video tutorial	[√] Berhasil [ ] Tidak Berhasil
		Memilih Tombol *Forward*	Sistem mempercepat pemutaran video tutorial	[√] Berhasil [ ] Tidak Berhasil
		Memilih Tombol Suara	Sistem menampilkan pengaturan *volume* suara	[√] Berhasil [ ] Tidak Berhasil
8	Menu Simulasi	Memilih Tombol *Home*	Siatem manampilkan halaman utama	[√] Berhasil [ ] Tidak Berhasil
		Memilih Tombol *Next*	Sistem menuju ke halaman selanjutnya	[√] Berhasil [ ] Tidak Berhasil
		Memilih Tombol Cek	Sistem menuju ke halaman pengecekan	[√] Berhasil [ ] Tidak Berhasil
		Memilih Tombol Latihan Simulasi	Sistem menampilkan pilihan menu latihan yang akan dilakukan	[√] Berhasil [ ] Tidak Berhasil
		Memilih Tombol Tes Simulasi	Sistem menuju ke halaman tes simulasi	[√] Berhasil [ ] Tidak Berhasil
9	Menu Latihan Simulasi	Memilih Menu *Powersupply*	Sistem akan menuju ke halaman latihan simulasi *powersupply*	[√] Berhasil [ ] Tidak Berhasil
		Memilih Menu *Motherboard*	Sistem akan menampilkan halaman latihan simulasi *motherboard*	[√] Berhasil [ ] Tidak Berhasil
		Memilih Menu *Internal Card*	Sistem akan menampilkan halaman	[√] Berhasil [ ] Tidak Berhasil

			latihan simulasi *internal card*	
		Memilih Menu *Eksternal Drives*	System akan menampilkan halaman latihan perakitan *eksternal drives*	[√] Berhasil [ ] Tidak Berhasil
		Memilih Menu *Internal Cable*	System akan menuju kehalaman latihan simulasi *internal cable*	[√] Berhasil [ ] Tidak Berhasil
		Memilih Menu *Eksternal Cable*	Sistem akan menampilkan halaman simulasi *ekstrenal cable*	[√] Berhasil [ ] Tidak Berhasil
10	Menu Tes Simulasi	Memilih Tombol Cek	Sistem akan menampilkan halaman informasi simulasi	[√] Berhasil [ ] Tidak Berhasil
11	Menu Latihan	Memilih Tombol *Home*	Sistem kembali ke halaman utama	[√] Berhasil [ ] Tidak Berhasil
		Memilih Tombol X	Sistem kembali ke halaman pilih menu	[√] Berhasil [ ] Tidak Berhasil
		Memilih Pilihan Jawaban A	Sistem menampilkan dan menyimpan hasil jawaban untuk jawaban A	[√] Berhasil [ ] Tidak Berhasil
		Memilih Pilihan Jawaban B	Sistem menampilkan dan menyimpan hasil jawaban untuk jawaban B	[√] Berhasil [ ] Tidak Berhasil
		Memilih Pilihan jawaban C	Sistem menampilkan dan menyimpan hasil	[√] Berhasil [ ] Tidak Berhasil

			jawaban untuk jawaban C	
		Memilih Pilihan Jawaban D	Sistem menampilkan dan menyimpan hasil jawaban untuk jawaban D	[√] Berhasil [ ] Tidak Berhasil
		Memilih Tombol Coba Lagi	Sistem menampilkan pesan	[√] Berhasil [ ] Tidak Berhasil
13	Menu Petunjuk	Memilih Menu Petunjuk	System akan menampilkan halaman petunjuk penggunaan aplikasi	[√] Berhasil [ ] Tidak Berhasil
14	Pesan	Memilih Tombol Ya	Sistem menampilkan halaman daftar soal	[√] Berhasil [ ] Tidak Berhasil
		Memilih Tombol Tidak	Sistem keluar dari aplikasi	[√] Berhasil [ ] Tidak Berhasil

4.2.1.2 Kasus dan Hasil Pengujian *White Box*

Berikut ini adalah kasus menguji perangkat lunak yang telah dibangun dengan menggunakan metode *white box*.

Tabel 4.5 Pengujian *White Box*

No	Coding

1	`komponen.onPress=function(){`
	`this.startDrag()`
	`}`
	`komponen.onRelease = function(){`
	`if (eval(this._droptarget) == target)`
	`this.stopDrag();`

2	`Array.prototype.status = function() {`
	`for(i=0;i<jml_komponen;i++){`
	`for (j=0;j>jml_target;j++){`
	`if (komponen[i]=target[j]) {`
	`Status = ('Terpasang');`
	`Nilai = 1`
	`}else{`

	Status = ('Belum Terpasang'); Nilai = 0;
3	function hitung_nilai_status_terpasang():Void{ Nilai_status_terpasang = Math(jml_komponen*nilai);}
4	function hitung_nilai_status_belumterpasang(): Void{ Nilai_status_belumterpasang = Math(jml_komponen-nilai_status_terpasang);}
5	function hitung_entropy_status_terpasang(): Void{ Entropy_status_terpasang = Math.log((2)-nilai_status_terpasang/2 + nilai_status_terpasang/2);}
6	function hitung_entropy_status_belumterpasang(): Void{ Entropy_status_terpasang = Math.log((2)-nilai_status_belumterpasang/2 + nilai_status_belumterpasang/2);}
7	function hitung_nilai_entropy_status(): Void{ Nilai_entropy_status = Math(2/jml_komponen*entopy_statusterpasang)+ (2/jml_komponen*entopy_status_belumterpasang);}

8	Array.prototype.posisi = function() {
	for (i=0;i<jml_komponen;i++){
	for (j=0;j>jml_target;j++){
	if (komponen[i]=target[j]){
	Posisi = ('Benar');
	Nilai = 1
	}else{
	Posisi = ('Salah')
	Nilai = 0;

9	function hitung_nilai_posisi_benar(): Void{ Nilai_posisi_benar = Math(jml_komponen*nilai);}
10	function hitung_nilai_posisi_salah(): Void{ Nilai_posisi_salah = Math(jml_komponen-nilai_posisi_benar);}
11	function hitung_entropy_posisibenar(): Void{ Entropy_posisi_benar = Math.log(2)* (-nilai_posisi_benar/2 + nilai_posisi_benar/2);}
12	function hitung_entropy_posisisalah(): Void{ Entropy_posisi_salah = Math.log(2)* (-nilai_posisi_salah/2 + nilai_posisi_salah/2);}
13	function hitung_nilai_entropy_posisi(): Void{ Nilai_entropy_posisi = Math (2/jml_komponen*entopy_posisibenar)+ (2/jml_komponen*entopy_posisisalah);}

14	Array.prototype.kondisi = function() {
	for (i=0;i<jml_komponen;i++){
	for (j=0;j>jml_target;j++){
	if (komponen[i]=target[j]){
	kondisi = ('Baik');
	Nilai = 1
	}else{
	kondisi = ('Rusak')
	Nilai = 0;

15	function hitung_nilai_kondisi_baik(): Void{ Nilai_kondisi_baik = Math(jml_komponen*nilai);}
16	function hitung_nilai_kondisi_rusak(): Void{

	Nilai_kondisi_rusak = Math(jml_komponen-nilai_kondisi_benar);}
17	function hitung_entropy_kondisibaik(): Void{ Entropy_kondisi_baik =Math.log(2)* (-nilai_kondisi_beaik/2 + nilai_kondisi_baik/2);}
18	function hitung_entropy_kondisirusak(): Void{ Entropy_kondisi_rusak = Math.log(2)* (-nilai_kondisi_rusak/2 + nilai_kondisi_rusak/2);}
19	function hitung_nilai_entropy_kondisi(): Void{ Nilai_entropy_kondisi=Math (2/jml_komponen*entopy_kondisibaik)+ (2/jml_komponen*entopy_kondisirusak);}
20	BinTree{ Create Node.kondisi *root;
21	boolean isEmpty(){ return (root == null);}

22	void insert(Node.posisi){
	Node temp = Node.posisi);
	if (isEmpty())
	root = temp;
	else {
	Node cursor = root,
	parent = null;
	while (cursor != null){
	parent = cursor;
	if (Node.posisi < cursor.kondisi)
	cursor = cursor.left;
	else
	cursor = cursor.right;
	}

23	if (Node.posisi < parent.kondisi){
	parent.left = temp;
	return (posisi);
	}
	else {
	parent.right = temp;
	return (error);

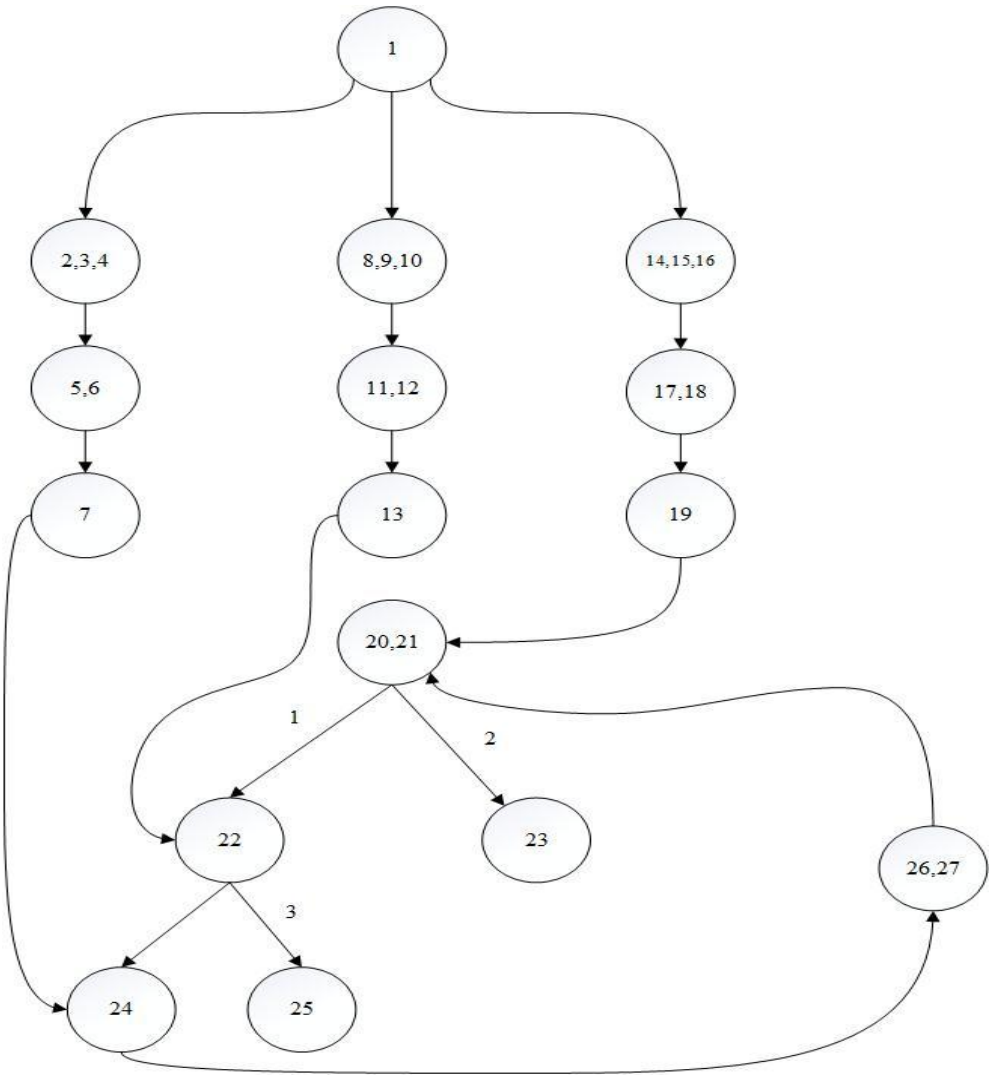
24	void insert(Node.status){
	Node temp = Node.status);
	if (isEmpty())
	root = temp;
	else {
	Node cursor = Node.posisi,
	parent = null;
	while (cursor != null){
	parent = cursor;
	cursor = cursor.left;
	}

25	Node.status < parent.posisi){
	parent.left = temp;
	return (status);
	else {
	parent.right = temp;
	return (error);
	}

26	`void traverse(int tipe){`
	`switch (tipe){`
	`case 1:`
	`trace('Preorder traversal:');`
	`preOrder(root);`
	`break;`
	`case 2:`
	`trace('Inorder traversal:');`
	`inOrder(root);`
	`break;`
	`case 3:`
	`trace('Postorder traversal:');`
	`postOrder(root);`
	`break;`
	`}`
	
	`}`

27	`void preOrder(Node.kondisi Root){`
	`if (Root == null) return;`
	`output(Root.kondisi+" ");`
	`preOrder(Root.left);`
	`preOrder(Root.right);`
	`}`
	`void inOrder(Node.kondisi Root){`
	`if (Root == null) return;`
	`inOrder(Root.left);`
	`output(Root.kondisi+" ");`
	`inOrder(Root.right);`
	`}`
	`void postOrder(Node.kondisi Root){`
	`if (Root == null) return;`
	`postOrder(Root.left);`
	`postOrder(Root.right);`
	`output(Root.kondisi+" ");`
	`}`
	
	`}`

a. *Flow Graph* Penelusuran Algoritma *Decision Tree*



Gambar 4.1 *Flow Graf* Algoritma *Decision Tree*

Keterangan :

◯ = Menggambarkan kondisi

→ = Menggambarkan aksi

Dari gambar 4.1 dapat dihitung *cyclomatic complexity* sebagai berikut :

Region = 3

V(G) = 17 *edge* – 16 *node* + 2 = 3

Jadi, *cyclomatic complexity* untuk gambar 4.2 adalah 3. Berdasarkan *cyclomatic*

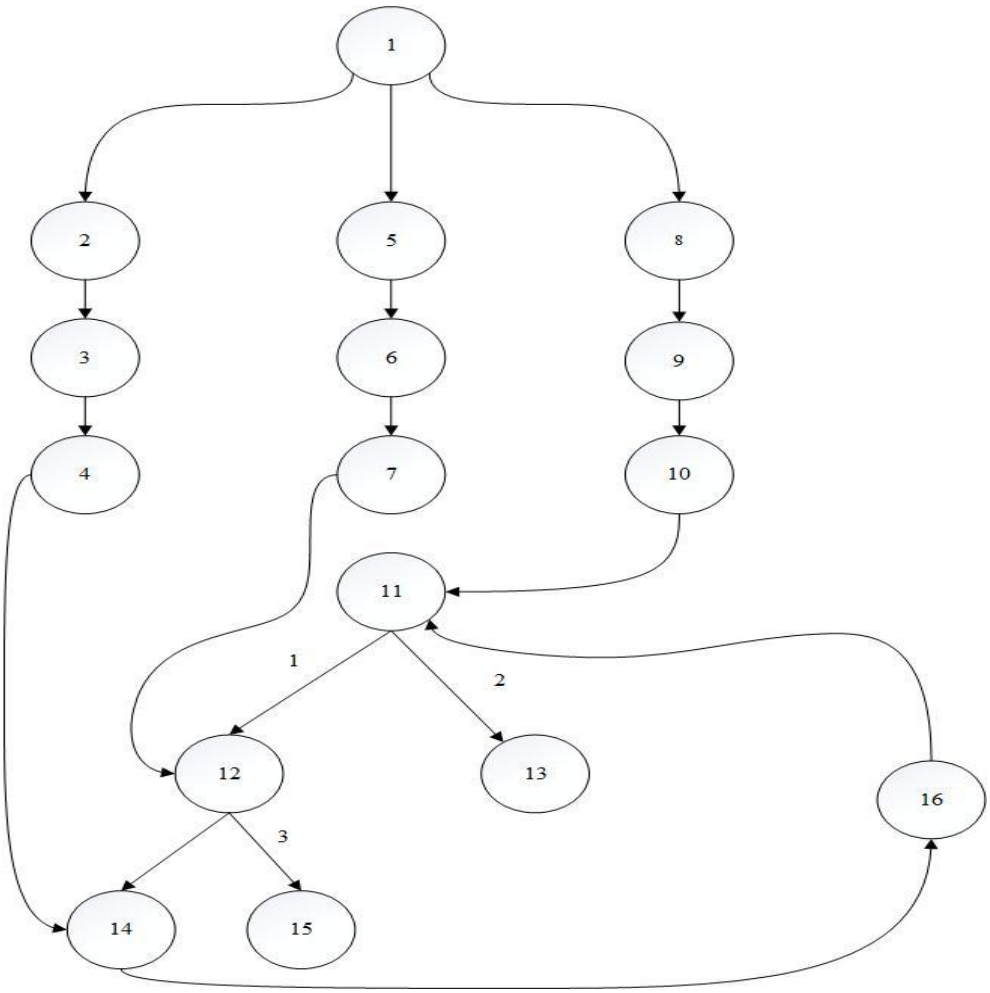
complexity tersebut, maka terdapat 3 *path* yang terdiri dari :

Path 1 : 1-2-3-4-5-6-7-8-9-10-11-12-13-14-15-16-17-18-19-20-21-22-24-26,27

Path 2 : 1-2-3-4-5-6-7-8-9-10-11-12-13-14-15-16-17-18-19-20-21-23

Path 3 : 1-2-3-4-5-6-7-8-9-10-11-12-13-14-15-16-17-18-19-20-21-22-25

Maka gambar 4.2 disederhanakan berdasarkan kondisi atau simpul sebagai berikut:



Gambar 4.2 Hasil Penyederhanaan *Flow Graf* Algoritma *Decision Tree*

b. *Graf Matriks* dari metode *Decision Tree*

Tabel 4.6 *Graf Matriks* dari Metode *Decision Tree*

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	E
1		1			1			1									0
2			1														0
3				1													0
4														1			0
5						1											0
6							1										0
7												1					0
8									1								0
9										1							0
10											1						0
11												1	1				1
12														1	1		1
13																	0
14																1	0
15																	0
16											1						0
	SUM (E) + 1																3

Berdasarkan hasil pengujian diatas maka dapat disimpulkan bahwa hasil pengujian *white box* adalah sebagai berikut :

Tabel 4.7 Kesimpulan Pengujian *White Box*

No	Butir Uji	Hasil yang diharapkan	Hasil sesuai kasus uji	Kesimpulan
1	`komponen.onPress=function(){ this.startDrag() } komponen.onRelease = function(){ if (eval(this._droptarget) == target) this.stopDrag();`	Proses pemasangan komponen	Proses pemasangan komponen	Sesuai dengan yang diharapkan
2	`Array.prototype.status = function() { for(i=0;i<jml_komponen;i++){ for (j=0;j>jml_target;j++) { if (komponen[i]=target[j]) { Status = ('Terpasang'); Nilai = 1 }else{ Status = ('Belum Terpasang'); Nilai = 0;`	Menghitung jumlah komponen yang digunakan untuk atribut status	Menghitung jumlah komponen yang digunakan untuk atribut status	Sesuai dengan yang diharapkan
3	`function hitung_nilai_entropy_status(): Void{ Nilai_entropy_status = Math(2/jml_komponen*entopy_statusterpasang)+ (2/jml_komponen*entopy_status_belumterpasang); }`	Menghitung nilai *entropy* status	Menghitung nilai *entropy* status	Sesuai dengan yang diharapkan
4	`Array.prototype.posisi = function() { for (i=0;i<jml_komponen;i++){`	Menghitung jumlah komponen yang	Menghitung jumlah komponen	Sesuai dengan yang diharapkan

	for (j=0;j>jml_target;j++){ if (komponen[i]=target[j]){ Posisi = ('Benar'); Nilai = 1 }else{ Posisi = ('Salah') Nilai = 0; } } } }	digunakan untuk atribut posisi	yang digunakan untuk atribut posisi	
5	function hitung_nilai_entropy_ posisi(): Void{ Nilai_entropy_posisi = Math (2/jml_komponen*entop y_posisibenar)+ (2/jml_komponen*entop y_posisisalah); }	Menghitung nilai *entropy* posisi	Menghitung nilai *entropy* posisi	Sesuai dengan yang diharapkan
6	Array.prototype.kondi si = function() { for (i=0;i<jml_komponen;i ++){ for (j=0;j>jml_target;j++){ if (komponen[i]=target[j]){ kondisi = ('Baik'); Nilai = 1 }else{ kondisi = ('Rusak') Nilai = 0; } } } }	Menghitung jumlah komponen yang digunakan untuk atribut kondisi	Menghitung jumlah komponen yang digunakan untuk atribut kondisi	Sesuai dengan yang diharapkan
7	function hitung_nilai_entropy_ kondisi(): Void{ Nilai_entropy_kondisi =Math (2/jml_komponen*entop y_kondisibaik)+	Menghitung nilai *entropy* kondisi	Menghitung nilai *entropy* kondisi	Sesuai dengan yang diharapkan

	(2/jml_komponen*entopy_kondisirusak); }			
8	BinTree{ Create Node.kondisi *root;	Membuat *node* awal (*root*)	Membuat *node* awal (*root*)	Sesuai dengan yang diharapkan

9	void insert(Node.posisi){
	Node temp = Node.posisi);
	if (isEmpty())
	root = temp;
	else {
	Node cursor = root,
	parent = null;
	while (cursor != null){
	parent = cursor;
	if (Node.posisi < cursor.kondisi)
	cursor = cursor.left;
	else
	cursor = cursor.right;
	}
		Memasukkan suatu nilai ke dalam *tree* sebagai *leaf node*	Memasukkan suatu nilai ke dalam *tree* sebagai *leaf node*	Sesuai dengan yang diharapkan

10	if (Node.posisi < parent.kondisi){
	parent.left = temp;
	return;
	}
	else {
	parent.right = temp;
	return;
	}
	}
	}
		Menambahkan *Node* baru pada kiri/kanan *Node parent*	Menambahkan *Node* baru pada kiri/kanan *Node parent*	Sesuai dengan yang diharapkan

11	void insert(Node.status){
	Node temp = Node.status);
	if (isEmpty())
	root = temp;
	else {
	Node cursor = Node.posisi,
	parent = null;
	while (cursor != null){
	parent = cursor;
	cursor = cursor.left;
	}
		Memasukkan suatu nilai ke dalam *tree* sebagai *leaf leaf node*	Memasukkan suatu nilai ke dalam *tree* sebagai *leaf leaf node.*	Sesuai dengan yang diharapkan

12	void insert(Node.status){
	Node temp = Node.status);
	if (isEmpty())
	root = temp;
	else {
	Node cursor = Node.posisi,
	parent = null;
	while (cursor != null){
	parent = cursor;
	cursor = cursor.left;
	}
		Memasukkan suatu nilai ke dalam *tree* sebagai *leaf leaf node*	Memasukkan suatu nilai ke dalam *tree* sebagai *leaf leaf node*	Sesuai dengan yang diharapkan

13	void traverse(int tipe){
	switch (tipe){
	case 1:
	trace('Preorder traversal:');
	preOrder(root);
		Method traverse untuk mengelilingi *Node-Node* dalam *tree*	*Method traverse* untuk mengelilingi *Node-Node* dalam *tree*	Sesuai dengan yang diharapkan

```
break;
            case 2:

      trace('Inorder
traversal:');

inOrder(root);

break;
            case 3:

      trace('Postorder
traversal:');

postOrder(root);

break;
            }

      }
void
preOrder(Node.kondisi
Root){
            if    (Root
== null) return;

      output(Root.kond
isi+" ");

      preOrder(Root.le
ft);

      preOrder(Root.ri
ght);
      }
      void
inOrder(Node.kondisi
Root){
            if    (Root
== null) return;

      inOrder(Root.lef
t);

      output(Root.kond
isi+" ");

      inOrder(Root.rig
ht);
      }
      void
postOrder(Node.kondis
i Root){
            if    (Root
== null) return;
```

	postOrder(Root.left);
	
	postOrder(Root.right);
	
	output(Root.kondisi+" ");
	}
	
	}			

4.2.1.3 Kesimpulan Pengujian *Alpha*

Berdasarkan hasil pengujian *alpha* dengan menggunakan metode *white box* dan *black box* dengan kasus uji diatas maka dapat disimpulkan bahwa Aplikasi Pembelajaran Pengenalan dan Perakitan Komputer ini tidak memiliki kesalahan proses dan secara fungsional pun menghasilkan aplikasi yang sudah sesuai dengan perancangan dan kebutuhan *user*.

4.2.2 Pengujian *Beta*

Pengujian *beta* merupakan pengujian yang dilakukan secara objektif yang mana melakukan pengujian secara langsung ke lapangan, dengan tujuan untuk mengetahui sejauh mana kualitas dari sistem. Pengujian *beta* dilakukan dengan penelitian terhadap responden atau calon pengguna sistem untuk mengetahui penilaian *user* terhadap aplikasi. Teknik pengumpulan data dilakukan melalui pengisian kuesioner oleh *user* yang terdiri dari 5 orang guru dan 37 orang siswa di SMAN 27 Bandung.

Untuk megetahui tanggapan dan penilaian *user* terhadap aplikasi pembelajaran pengenalan dan perakitan komputer ini, telah disebar kuesioner kepada responden berdasarkan target *user*, kemudian dilakukan penghitungan dengan menggunakan skala kontinum dimana data dianalisis dengan menghitung rata-rata jawaban berdasarkan skoring setiap jawaban dari responden yang kemudian berdasarkan rata-rata jawaban dari responden lalu kemudian dijumlahkan.

4.2.2.1 Data Kuesioner

Adapun data kuesioner dari aplikasi pembelajaran pengenalan dan perakitan komputer adalah sebagai berikut :

1. Jumlah responden yang terdiri dari 37 orang siswa dan 5 orang guru di SMAN 27 Bandung.
2. *Point* yang menjadi bahan penilaian adalah :

a. Membantu proses belajar
b. Memberikan bentuk pembelajaran yang lebih variatif
c. Memberikan penjelasan materi yang lengkap
d. Menarik minat belajar
e. Memberikan bentuk pembelajaran simulasi
f. Memenuhi kebutuhan yang diharapkan

3. Berikut adalah daftar pertanyaan kepada masing-masing *user*.

Tabel 4.8 Daftar Pertanyaan Kuesioner

User	Pertanyaan
Guru	Apakah aplikasi ini sudah membantu dalam proses pengajaran?
	Apakah aplikasi ini mampu memberikan bentuk pembelajaran yang lebih variatif?
	Apakah aplikasi ini memberikan penjelasan materi pengenalan komputer secara lengkap?
	Apakah aplikasi ini sudah sesuai dengan kebutuhan?
	Apakah dengan aplikasi ini sudah dapat mensimulasikan perakitan komputer?
Siswa	Apakah aplikasi ini sudah membantu proses belajar?
	Apakah aplikasi ini mampu memberikan bentuk pembelajaran yang lebih variatif?
	Apakah aplikasi ini memberikan penjelasan materi pengenalan komputer secara lengkap?

	Apakah aplikasi ini dapat meningkatkan minat belajar?
	Apakah dengan aplikasi ini sudah dapat mensimulasikan perakitan komputer?
	Apakah aplikasi ini mampu mengurangi kejenuhan dalam belajar materi pengenalan komputer?

4.2.2.2 Pengolahan Data Kuesioner

Adapun skala jawaban dari kuesioner adalah sebagai berikut :

Tabel 4.9 Skala Jawaban Kuesioner

Jawaban	Skor
SS = Sangat Setuju	5
S = Setuju	4
R = Ragu-ragu	3
KS = Kurang Setuju	2
TS = Tidak Setuju	1

Berikut adalah hasil pengumpulan data dari masing-masing jawaban untuk setiap *user*. Jawaban dari masing-masing responden dapat dilihat pada tabel dibawah ini :

Tabel 4.10 Jawaban Kuesioner Guru

No	**Pertanyaan**	**Jawaban**					**Total**
		SS	**S**	**R**	**KS**	**TS**	
1	Apakah aplikasi ini sudah membantu dalam proses pengajaran?	1	3	1			**5**
2	Apakah aplikasi ini mampu memberikan bentuk	2	2	1			**5**

	pembelajaran yang lebih variatif?						
3	Apakah aplikasi ini memberikan penjelasan materi pengenalan komputer secara lengkap?	3	2				**5**
4	Apakah aplikasi ini sudah sesuai dengan kebutuhan?	2	3				**5**
5	Apakah dengan aplikasi ini sudah dapat mensimulasikan perakitan komputer?	1	2	2			**5**

Tabel 4.11 Jawaban Kuesioner Siswa

No	**Pertanyaan**	**Jawaban**					**Total**
		SS	**S**	**R**	**KS**	**TS**	
1	Apakah aplikasi ini sudah membantu proses belajar?	20	12	5			**37**
2	Apakah aplikasi ini mampu memberikan bentuk pembelajaran yang lebih variatif?	25	9	3			**37**
3	Apakah aplikasi ini memberikan penjelasan materi pengenalan komputer secara lengkap?	22	10	5			**37**
4	Apakah aplikasi ini dapat meningkatkan minat belajar?	15	15	7			**37**

5	Apakah dengan aplikasi ini sudah dapat mensimulasikan perakitan komputer?	17	10	10			**37**
6	Apakah aplikasi ini mampu mengurangi kejenuhan dalam belajar materi pengenalan komputer?	19	7	11			**37**

4.2.2.3 Hasil Pengolahan Kuesioner Guru

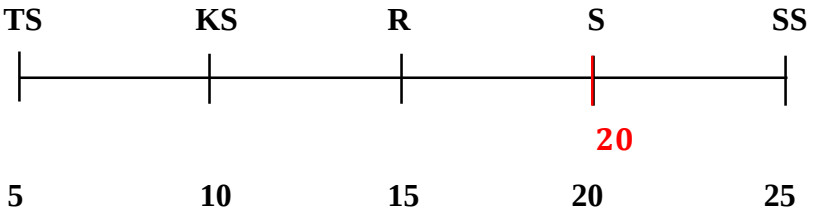
Data dianalisis dengan menghitung rata-rata berdasarkan skoring setiap jawaban dari responden kemudian dijumlahkan, lalu digambarkan secara kontinum. Adapun pengolahan data kuesioner untuk guru adalah sebagai berikut :

Tabel 4.12 Hasil Pengolahan Data Kuesioner Guru Pertanyaan (1)

Jawaban	**Skor**	**Jumlah Responden**	**Jawaban Skor (Skor x Jumlah Responden)**
SS = Sangat Setuju	5	1	5
S = Setuju	4	3	12
R = Ragu-ragu	3	1	3
KS = Kurang Setuju	2	0	0
TS = Tidak Setuju	1	0	0
		Total	20

Jika semua responden (5 orang responden) menjawab SS, maka jumlah skor ideal (kriterium) untuk seluruh item adalah 25 (diperoleh dari 5 x 5). Berdasarkan hasil penelitian, jumlah skor yang diperoleh adalah 20. Jadi berdasarkan data tersebut, maka tingkat persetujuan responden terhadap pertanyaan nomor (1) adalah (20 : 25) x 100% = 80% dari yang diharapkan (100%). Jadi berdasarkan pengolahan

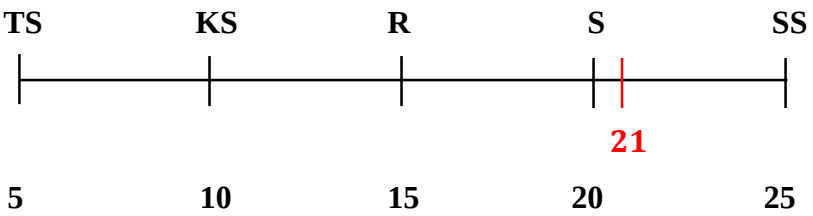
data yang diperoleh dari 5 responden, maka rata-rata 20 terletak pada daerah setuju. Secara kontinum dapat digambarkan sebagai berikut :



Tabel 4.13 Hasil Pengolahan Data Kuesioner Guru Pertanyaan (2)

Jawaban	Skor	Jumlah Responden	Jawaban Skor (Skor x Jumlah Responden)
SS = Sangat Setuju	5	2	10
S = Setuju	4	2	8
R = Ragu-ragu	3	1	3
KS = Kurang Setuju	2	0	0
TS = Tidak Setuju	1	0	0
		Total	21

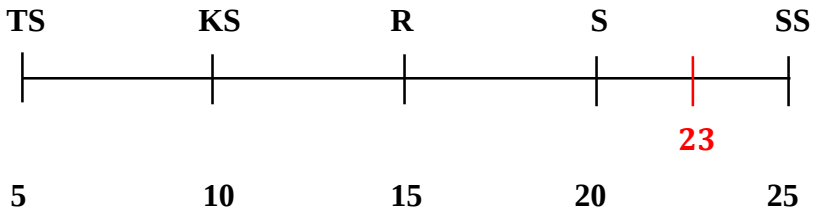
Jika semua responden (5 orang responden) menjawab SS, maka jumlah skor ideal (kriterium) untuk seluruh item adalah 25 (diperoleh dari 5 x 5). Berdasarkan hasil penelitian, jumlah skor yang diperoleh adalah 21. Jadi berdasarkan data tersebut, maka tingkat persetujuan responden terhadap pertanyaan nomor (2) adalah (21 : 25) x 100% = 84% dari yang diharapkan (100%). Jadi berdasarkan pengolahan data yang diperoleh dari 5 responden, maka rata-rata 21 terletak pada daerah setuju. Secara kontinum dapat digambarkan sebagai berikut :



Tabel 4.14 Hasil Pengolahan Data Kuesioner Guru Pertanyaan (3)

Jawaban	Skor	Jumlah Responden	Jawaban Skor (Skor x Jumlah Responden)
SS = Sangat Setuju	5	3	15
S = Setuju	4	2	8
R = Ragu-ragu	3	0	0
KS = Kurang Setuju	2	0	0
TS = Tidak Setuju	1	0	0
		Total	23

Jika semua responden (5 orang responden) menjawab SS, maka jumlah skor ideal (kriterium) untuk seluruh item adalah 25 (diperoleh dari 5 x 5). Berdasarkan hasil penelitian, jumlah skor yang diperoleh adalah 23. Jadi berdasarkan data tersebut, maka tingkat persetujuan responden terhadap pertanyaan nomor (3) adalah (23 : 25) x 100% = 92% dari yang diharapkan (100%). Jadi berdasarkan pengolahan data yang diperoleh dari 5 responden, maka rata-rata 23 terletak pada daerah setuju. Secara kontinum dapat digambarkan sebagai berikut :

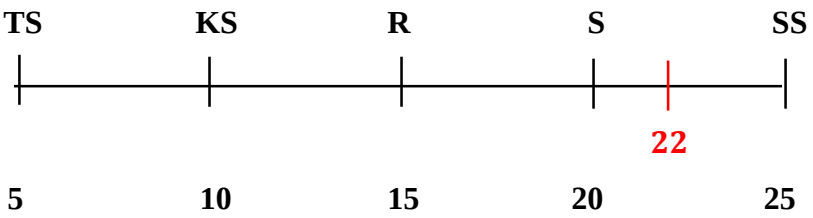


Tabel 4.15 Hasil Pengolahan Data Kuesioner Guru Pertanyaan (4)

Jawaban	Skor	Jumlah Responden	Jawaban Skor (Skor x Jumlah Responden)
SS = Sangat Setuju	5	2	10
S = Setuju	4	3	12

R = Ragu-ragu	3	0	0
KS = Kurang Setuju	2	0	0
TS = Tidak Setuju	1	0	0
		Total	22

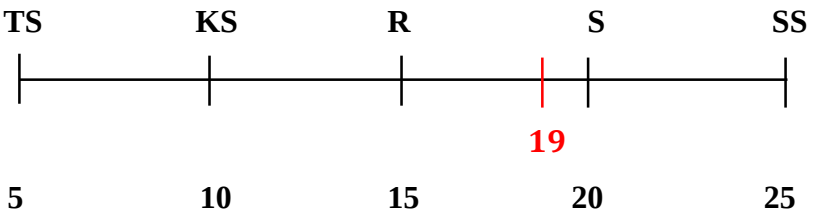
Jika semua responden (5 orang responden) menjawab SS, maka jumlah skor ideal (kriterium) untuk seluruh item adalah 25 (diperoleh dari 5 x 5). Berdasarkan hasil penelitian, jumlah skor yang diperoleh adalah 22. Jadi berdasarkan data tersebut, maka tingkat persetujuan responden terhadap pertanyaan nomor (4) adalah (22 : 25) x 100% = 88% dari yang diharapkan (100%). Jadi berdasarkan pengolahan data yang diperoleh dari 5 responden, maka rata-rata 22 terletak pada daerah setuju. Secara kontinum dapat digambarkan sebagai berikut :



Tabel 4.16 Hasil Pengolahan Data Kuesioner Guru Pertanyaan (5)

Jawaban	**Skor**	**Jumlah Responden**	**Jawaban Skor (Skor x Jumlah Responden)**
SS = Sangat Setuju	5	1	5
S = Setuju	4	2	8
R = Ragu-ragu	3	2	6
KS = Kurang Setuju	2	0	0
TS = Tidak Setuju	1	0	0
Total			19

Jika semua responden (5 orang responden) menjawab SS, maka jumlah skor ideal (kriterium) untuk seluruh item adalah 25 (diperoleh dari 5 x 5). Berdasarkan hasil penelitian, jumlah skor yang diperoleh adalah 19. Jadi berdasarkan data tersebut, maka tingkat persetujuan responden terhadap pertanyaan nomor (5) adalah (19 : 25) x 100% = 76 % dari yang diharapkan (100%). Jadi berdasarkan pengolahan data yang diperoleh dari 5 responden, maka rata-rata 19 terletak pada daerah setuju. Secara kontinum dapat digambarkan sebagai berikut :



4.2.2.4 Hasil Pengolahan Kuesioner Siswa

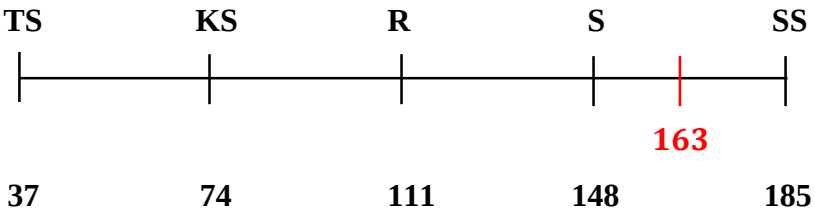
Berikut merupakan data pengolahan kuesioner untuk siswa seperti dibawah ini:

Tabel 4.17 Hasil Pengolahan Data Kuesioner Siswa Pertanyaan (1)

Jawaban	Skor	Jumlah Responden	Jawaban Skor (Skor x Jumlah Responden)
SS = Sangat Setuju	5	20	100
S = Setuju	4	12	48
R = Ragu-ragu	3	5	15
KS = Kurang Setuju	2	0	0
TS = Tidak Setuju	1	0	0
		Total	163

Jika semua responden (37 orang responden) menjawab SS, maka jumlah skor ideal (kriterium) untuk seluruh item adalah 185 (diperoleh dari 5 x 37). Berdasarkan hasil penelitian, jumlah skor yang diperoleh adalah 163. Jadi

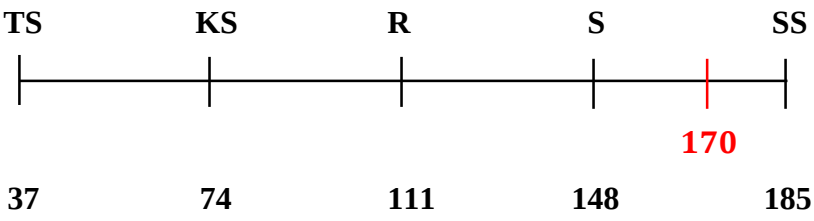
berdasarkan data tersebut, maka tingkat persetujuan responden terhadap pertanyaan nomor (1) adalah (163 : 185) x 100% = 88.11% dari yang diharapkan (100%). Jadi berdasarkan pengolahan data yang diperoleh dari 37 responden, maka rata-rata 163 terletak pada daerah setuju. Secara kontinum dapat digambarkan sebagai berikut :



Tabel 4.18 Hasil Pengolahan Data Kuesioner Siswa Pertanyaan (2)

Jawaban	**Skor**	**Jumlah Responden**	**Jawaban Skor (Skor x Jumlah Responden)**
SS = Sangat Setuju	5	25	125
S = Setuju	4	9	36
R = Ragu-ragu	3	3	9
KS = Kurang Setuju	2	0	0
TS = Tidak Setuju	1	0	0
		Total	170

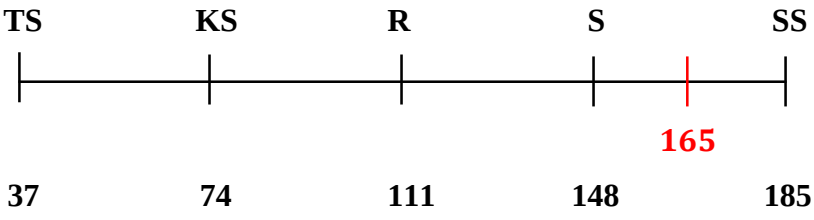
Jika semua responden (37 orang responden) menjawab SS, maka jumlah skor ideal (kriterium) untuk seluruh item adalah 185 (diperoleh dari 5 x 37). Berdasarkan hasil penelitian, jumlah skor yang diperoleh adalah 170. Jadi berdasarkan data tersebut, maka tingkat persetujuan responden terhadap pertanyaan nomor (2) adalah (170 : 185) x 100% = 91.89% dari yang diharapkan (100%). Jadi berdasarkan pengolahan data yang diperoleh dari 37 responden, maka rata-rata 170 terletak pada daerah setuju. Secara kontinum dapat digambarkan sebagai berikut :



Tabel 4.19 Hasil Pengolahan Data Kuesioner Siswa Pertanyaan (3)

Jawaban	Skor	Jumlah Responden	Jawaban Skor (Skor x Jumlah Responden)
SS = Sangat Setuju	5	22	110
S = Setuju	4	10	40
R = Ragu-ragu	3	5	15
KS = Kurang Setuju	2	0	0
TS = Tidak Setuju	1	0	0
		Total	165

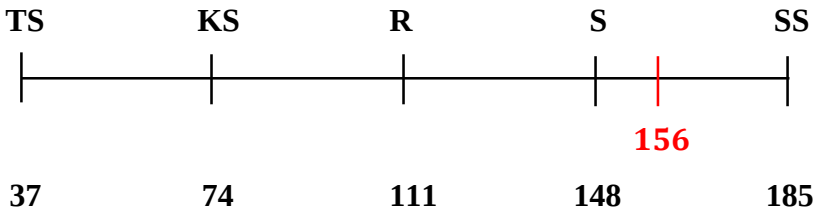
Jika semua responden (37 orang responden) menjawab SS, maka jumlah skor ideal (kriterium) untuk seluruh item adalah 185 (diperoleh dari 5 x 37). Berdasarkan hasil penelitian, jumlah skor yang diperoleh adalah 165. Jadi berdasarkan data tersebut, maka tingkat persetujuan responden terhadap pertanyaan nomor (3) adalah (165 : 185) x 100% = 89.19% dari yang diharapkan (100%). Jadi berdasarkan pengolahan data yang diperoleh dari 37 responden, maka rata-rata 165 terletak pada daerah setuju. Secara kontinum dapat digambarkan sebagai berikut :



Tabel 4.20 Hasil Pengolahan Data Kuesioner Siswa Pertanyaan (4)

Jawaban	Skor	Jumlah Responden	Jawaban Skor (Skor x Jumlah Responden)
SS = Sangat Setuju	5	15	75
S = Setuju	4	15	60
R = Ragu-ragu	3	7	21
KS = Kurang Setuju	2	0	0
TS = Tidak Setuju	1	0	0
		Total	156

Jika semua responden (37 orang responden) menjawab SS, maka jumlah skor ideal (kriterium) untuk seluruh item adalah 185 (diperoleh dari 5 x 37). Berdasarkan hasil penelitian, jumlah skor yang diperoleh adalah 156. Jadi berdasarkan data tersebut, maka tingkat persetujuan responden terhadap pertanyaan nomor (4) adalah (156 : 185) x 100% = 84.32% dari yang diharapkan (100%). Jadi berdasarkan pengolahan data yang diperoleh dari 37 responden, maka rata-rata 156 terletak pada daerah setuju. Secara kontinum dapat digambarkan sebagai berikut :

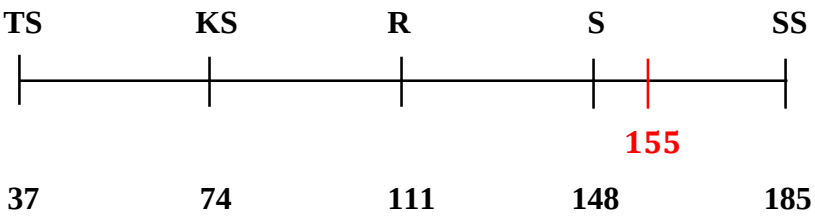


Tabel 4.21 Hasil Pengolahan Data Kuesioner Siswa Pertanyaan (5)

Jawaban	Skor	Jumlah Responden	Jawaban Skor (Skor x Jumlah Responden)
SS = Sangat Setuju	5	17	85
S = Setuju	4	10	40

R = Ragu-ragu	3	10	30
KS = Kurang Setuju	2	0	0
TS = Tidak Setuju	1	0	0
		Total	155

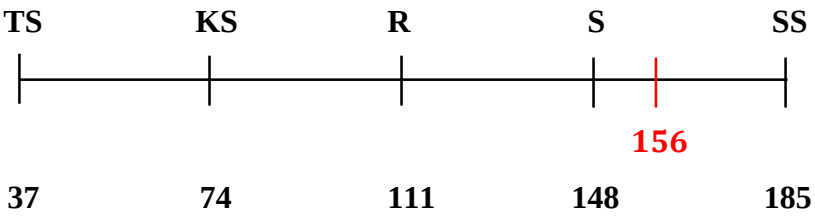
Jika semua responden (37 orang responden) menjawab SS, maka jumlah skor ideal (kriterium) untuk seluruh item adalah 185 (diperoleh dari 5 x 37). Berdasarkan hasil penelitian, jumlah skor yang diperoleh adalah 155. Jadi berdasarkan data tersebut, maka tingkat persetujuan responden terhadap pertanyaan nomor (5) adalah (155 : 185) x 100% = 83.78% dari yang diharapkan (100%). Jadi berdasarkan pengolahan data yang diperoleh dari 37 responden, maka rata-rata 163 terletak pada daerah setuju. Secara kontinum dapat digambarkan sebagai berikut :



Tabel 4.22 Hasil Pengolahan Data Kuesioner Siswa Pertanyaan (6)

Jawaban	**Skor**	**Jumlah Responden**	**Jawaban Skor (Skor x Jumlah Respoinden)**
SS = Sangat Setuju	5	19	95
S = Setuju	4	7	28
R = Ragu-ragu	3	11	33
KS = Kurang Setuju	2	0	0
TS = Tidak Setuju	1	0	0
		Total	156

Jika semua responden (37 orang responden) menjawab SS, maka jumlah skor ideal (kriterium) untuk seluruh item adalah 185 (diperoleh dari 5 x 37). Berdasarkan hasil penelitian, jumlah skor yang diperoleh adalah 156. Jadi berdasarkan data tersebut, maka tingkat persetujuan responden terhadap pertanyaan nomor (4) adalah (156 : 185) x 100% = 84.32% dari yang diharapkan (100%). Jadi berdasarkan pengolahan data yang diperoleh dari 37 responden, maka rata-rata 156 terletak pada daerah setuju. Secara kontinum dapat digambarkan sebagai berikut :



4.2.2.5 Kesimpulan Pengujian *Beta*

Berdasarkan jawaban hasil kuesioner yang diberikan kepada para responden, maka dapat disimpulkan bahwa :

1. Aplikasi pembelajaran pengenalan dan perakitan komputer ini membantu proses pembalajaran di SMAN 27 Bandung menjadi lebih variatif.
2. Aplikasi ini sudah memberikan materi pengenalan komputer dengan lengkap.
3. Aplikasi ini dapat meningkatkan minat belajar siswa terhadap materi pengenalan komputer sehingga dapat mengurangi tingkat jenuh pada saat mempelajari materi tersebut.
4. Melalui aplikasi ini siswa mengetahui cara merakit sebuah komputer.
5. Aplikasi sudah sesuai dengan kebutuhan guru dan para siswa di SMAN 27 Bandung.