BAB 4
IMPLEMENTTASI DAN PENGUJIAN SISTEM

Pada bab ini akan dilakukan implementasi dan pengujian terhadap aplikasi yang dibangun. Tahapan ini dilakukan setelah analisis dan perancangan selesai dilakukan dan selanjutnya akan diimplementasikan ke dalam bahasa pemrograman. Setelah implementasi maka dilakukan pengujian terhadap aplikasi. Aplikasi yang telah dibangun akan dilakukan pengujian untuk mengetahui apakah sistem tersebut dapat berjalan sesuai dengan tujuan atau tidak yang selanjutnya diadakan untuk pengembangan sistem.

4.1 Implementasi Sistem

Implementasi sistem merupakan tahap untuk menerapkan perancangan yang telah dilakukan terhadap sistem sehingga siap untuk dioperasikan. Implementasi sistem meliputi implementasi perangkat keras, implementasi perangkat lunak, implementasi data, implementasi antar muka.

4.1.1 Implementasi Perangkat Keras

Perangkat keras (*hardware*) yang digunakan untuk membangun sistem di *platform mobile* dan *website* untuk aplikasi Bandung Fashion Sale dapat dilihat pada tabel 4.1 dan tabel 4.2.

Tabel 4. 1 Implementasi Perangkat Keras di *Platform Mobile*

Nama Perangkat Keras	Spesifikasi
Proccessor	Quad-core 1.2 GHz
Memory Internal	8 GB
RAM	1 GB
Layar	4.7 inches
Sistem Operasi	Android 4.4.2 Kitkat

Tabel 4. 2 Implementasi Perangkat Keras di *Platform Website*

Nama Perangkat Keras	Spesifikasi
Proccessor	Intel Core I3 – M 370, 2.4GHz
VGA	Intel HD Graphics
Memori	6 GB
Harddisk	320 GB
Koneksi Internet	10mbps

4.1.2 Kebutuhan Perangkat Lunak

Perangkat lunak (*software*) untuk proses implementasi sistem meliputi sistem operasi, IDE, perambah dan text editor untuk menulis kode. Tabel 4.3 merupakan implementasi perangkat lunak yang digunakan untuk membangun aplikasi Bandung Fashion Sale di *platform mobile* dan *website*.

Tabel 4. 3 Implementasi Perangkat Lunak

Nama Perangkat Lunak	Spesifikasi
Sistem Operasi	Windows 7 Ultimate 64 bit
IDE	Android Studio 1.2
Perambah	Google Chrome
Text Editor	Sublime Text 3

4.1.3 Implementasi Data

Pembuatan basis data dilakukan dengan menggunakan aplikasi DBMS MySQL. Tabel 4.4 Implementasi Data dalam bahasa SQL adalah sebagai berikut :

Tabel 4. 4 Implementasi Data

No	Nama Tabel	Perintah SQL
1	t_admin	CREATE TABLE IF NOT EXISTS `t_admin` (`id_admin` int(5) NOT NULL AUTO_INCREMENT, `username` varchar(50) NOT NULL DEFAULT 'NULL', `password` varchar(50) NOT NULL DEFAULT 'NULL',

		`email` varchar(100) NOT NULL DEFAULT 'NULL', `level` varchar(20) NOT NULL, `kunci` varchar(50) DEFAULT NULL, PRIMARY KEY (`id_admin`), UNIQUE KEY `username` (`username`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;
2	t_diskon	CREATE TABLE IF NOT EXISTS `t_diskon` (`id_admin` int(5) NOT NULL, `id_diskon` int(11) NOT NULL AUTO_INCREMENT, `id_toko` int(11) NOT NULL, `gambar_diskon` varchar(100) NOT NULL, `nilai_diskon` varchar(50) NOT NULL, `periode_diskon` varchar(50) NOT NULL, `detail_item` varchar(5000) NOT NULL, `id_kategori` int(10) NOT NULL, PRIMARY KEY (`id_diskon`), KEY `FK_t_diskon` (`id_toko`), KEY `FK_t_kategori` (`id_kategori`), KEY `FK_t_admin` (`id_admin`) FOREIGN KEY (id_toko) REFERENCES t_toko (id_toko) ON DELETE CASCADE ON UPDATE CASCADE FOREIGN KEY (id_kategori) REFERENCES t_diskon (id_kategori) ON DELETE CASCADE ON UPDATE CASCADE FOREIGN KEY (id_admin) REFERENCES t_admin (id_admin) ON DELETE CASCADE ON UPDATE CASCADE) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=383 ;

3	t_kategori	CREATE TABLE IF NOT EXISTS `t_kategori` (`id_kategori` int(10) NOT NULL AUTO_INCREMENT, `jenis_kategori` varchar(50) NOT NULL, PRIMARY KEY (`id_kategori`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=4 ;
4	t_toko	CREATE TABLE IF NOT EXISTS `t_toko` (`id_toko` int(11) NOT NULL AUTO_INCREMENT, `nama_toko` varchar(100) NOT NULL, `alamat_toko` varchar(200) NOT NULL, `lat_toko` double NOT NULL, `long_toko` double NOT NULL, `gambar` varchar(100) NOT NULL, `id_admin` int(5) NOT NULL, PRIMARY KEY (`id_toko`), KEY `FK_t_toko` (`id_admin`) FOREIGN KEY (id_admin) REFERENCES t_admin (id_admin) ON DELETE CASCADE ON UPDATE CASCADE) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=40 ;
5	t_komentar	CREATE TABLE IF NOT EXISTS `t_komentar` (`id_komentar` int(11) NOT NULL AUTO_INCREMENT, `id_diskon` int(11) NOT NULL, `nama_komentar` varchar(50) NOT NULL, `isi_komentar` text NOT NULL, PRIMARY KEY (`id_komentar`) KEY `FK_t_komentar` (`id_komentar`)

		FOREIGN KEY (id_diskon) REFERENCES t_diskon (id_diskon) ON DELETE CASCADE ON UPDATE CASCADE) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=18 ;

4.1.4 Implementasi Antar Muka

Implementasi antarmuka dilakukan terhadap setiap tampilan program yang dibagun dan pengkodeannya dalam bentuk file program. Tabel 4.5 Berikut ini adalah Implementasi Antar Muka di *Platform Mobile* dan tabel 4.6 Implementasi Antar Muka di *Platform Website*.

Tabel 4. 5 Implementasi Antar Muka di *Platform Mobile*

Menu	Deskripsi	Nama File
Get Location	Merupakan pengaturan untuk menggunakan aplikasi ini, dimana GPS harus dinyalakan terlebih dahulu untuk mendeteksi lokasi keberadaan	
Home	Merupakan halaman awal ketika pengguna pertama kali membuka aplikasi Bandung Fashion Sale	app_bar_main.xml
Toko Diskon Terdekat	Merupakan halaman yang menampilkan toko diskon terdekat dengan pengguna	activity_toko_terdekat.xml
Detail Toko	Merupakan halaman yang menampilkan detail informasi toko yang sudah pengguna pilih	activity_detail_toko.xml
Cari Diskon	Merupakan halaman untuk mencari diskon sesuai dengan	activity_cari_diskon.xml

	jumlah diskon yang pengguna inginkan	
Kategori Produk	Merupakan halaman kategori produk-produk diskon	activity_category.xml
Detail Diskon	Merupakan halaman yang menampilkan detail produk yang sudah pengguna pilih	activity_detail_diskon.xml
Timeline Informasi Diskon	Merupakan halaman yang menampilkan produk-produk diskon hasil dari filtering data twitter	item_diskon.xml

Tabel 4. 6 Implementasi Antar Muka di *Platform Website*

Menu	Deskripsi	Nama File
Login	Merupakan halaman untuk *login* admin	*login*.php
Dashboard	Merupakan halaman utama ketika admin sudah *login*	index.php
Form Data Toko	Merupakan halaman yang menampilkan data toko yang sudah terdaftar	toko.php
Form Tambah Data Toko	Merupakan halaman untuk menambah data toko / mendaftarkan toko ke *database*	toko_tambah.php
Form Edit Data Toko	Merupakan halaman untuk mengedit data toko lalu disimpan ke *database*	toko_edit.php
Form Hapus Data Toko	Merupakan halaman untuk menghapus data toko yang telah terdaftar	toko_hapus.php

Form Data Diskon	Merupakan halaman yang menampilkan data diskon yang sudah terdaftar	diskon.php
Form Tambah Data Diskon	Merupakan halaman untuk menambah data diskon / mendaftarkan toko ke *database*	diskon_tambah.php
Form Edit Data Diskon	Merupakan halaman untuk mengedit data diskon lalu disimpan ke *database*	diskon_edit.php
Form Hapus Data Diskon	Merupakan halaman untuk menghapus data diskon yang telah terdaftar	diskon_hapus.php
Form Data Twitter	Merupakan halaman yang menampilkan data Twitter hasil dari proses *filtering* Twitter	index.php (…/pages/twitter)

4.2 Pengujian Sistem

Tahap yang selanjutnya adalah tahap pengujian sistem pada aplikasi Bandung Fashion Sale. Tahap ini merupakaan hal terpenting yang bertujuan untuk menemukan kesalahan – kesalahan ataupun kekurangan – kekurangan pada aplikasi yang dibangun. Pengujian ini bermaksud untuk mengetahui apakah aplikasi yang dibuat telah memenuhi kriteria yang sesuai dengan tujuan perancangan aplikasi atau belum. Pengujian terhadap sistem aplikasi Bandung Fashion Sale akan menggunakan strategi pengujian, pengujian alpha *(black-box* dan *white-box*) dan pengujian beta.

4.2.1 Pengujian Alpha

Pengujian alpha dilakukan dengan menggunakan metode *black-box* yang berfokus pada persyaratan fungsional perangkat lunak.

4.2.1.1 Metode Pengujian

Sebuah aplikasi yang akan diterapkan haruslah bebas dari kesalahan – kesalahan. Oleh karena itu, aplikasi harus diuji coba terlebih dahulu untuk menemukan kesalahan – kesalahan yang mungkin terjadi.

4.2.1.2 Rencana Pengujian

Pengujian program ini menggunakan metode *black-box*. Pengujian *black-box* merupakan pengujian program berdasarkan fungsional dari program. Tujuan dari metode *black-box* ini adalah untuk menemukan kesalahan fungsi pada program. Pengujian dengan metode *black-box* dilakukan dengan cara memberikan sejumlah data masukan pada aplikasi yang kemudian diproses sesuai dengan kebutuhan fungsionalnya untuk melihat apakah aplikasi yang kemudian diproses sesuai dengan kebutuhan fungsionalnya untuk melihat apakah aplikasi menghasilkan keluaran yang digunakan dan sesuai dengan fungsi dari program tersebut. Apabila dari data masukan yang diberikan proses menghasilkan keluaran yang sesuai dengan kebutuhan fungsionalnya, maka aplikasi yang telah dibuat telah benar. Tetapi jika keluaran yang dihasilkan tidak sesuai dengan kebutuhan fungsionalnya, maka masih terdapat kesalahan pada program aplikasi.

Pengujian dilakukan dengan mencoba semua kemungkinan yang terjadi dan pengujian dilakukan berulang-ulang jika dalam pengujian ditemukan kesalahan maka akan dilakukan penelusuran atau perbaikan untuk memperbaiki kesalahan yang terjadi. Jika telah selesai melakukan perbaikan, maka akan dilakukan secara terus menerus sehingga diperoleh hasil yang terbaik. Rencana pengujian alpha yang akan dilakukan pada perangkat lunak ini dapat dilihat pada tabel 4.7 Rencana Pengujian Alpha sebagai berikut:

Tabel 4. 7 Rencana Pengujian Alpha

Pengujian Pada Aplikasi *Mobile*		
Item Uji	**Detail Pengujian**	**Jenis Uji**
Mendeteksi Lokasi Pengguna	1. Sistem memeriksa ketersedian GPS	*Black box*

	2. Sistem mengirimkan permintaan data lokasi ke google maps API 3. Google maps API mengirimkan data lokasi ke system 4. Sistem menyimpan lokasi pengguna	
Menampilkan Informasi Produk Diskon	1. Membuka menu home 2. Sistem mencari informasi produk diskon hasil dari input manual data diskon 3. Data informasi diskon ditampilkan	*Black box*
Mencari Informasi Lokasi Toko Pakaian	1. Pengguna mengetap tombol lokasi toko 2. Sistem menampilkan halaman peta lokasi 3. Pengguna mengetap tombol navigasi 4. Sistem mengirimkan permintaan data lokasi ke google maps API 5. Google maps API melakukan validasi permintaan 6. Goole maps API mengirimkan data lokasi ke sistem 7. Sistem menampilkan halaman peta lokasi	*Black box*

Mendapatkan Rekomendasi Toko Terdekat	1. Mengetap menu toko diskon terdekat 2. Sistem menampilkan data rekomendasi toko pakaian 3. Pengguna memilih data rekomendasi toko pakaian 4. Sistem menampilkan data rekomendasi yang dipilih	
Mencari Infomasi Produk Diskon	1. Pengguna mengetap tombol cari diskon 2. Sistem menampilkan halaman pencarian produk diskon 3. Pengguna mengisi kolom pencarian dengan jumlah diskon produk yang mereka inginkan 4. Pengguna mengetap tombol cari 5. Sistem melakukan pencarian data produk diskon 6. Sistem menampilkan halaman pencarian produk yang telah ditemukan hasil dari input manual data diskon dan pengelolaan filtering data Twitter	*Black box*
Menampilkan Kategori Produk Diskon	1. Pengguna membuka menu kategori	*Black box*

	2. Sistem mencari informasi produk diskon berdasarkan kategori 3. Data kategori diskon ditampilkan	
Mengisi Komentar	1. Pengguna membuka menu detail diskon 2. Sistem mempilkan form komentar 3. Pengguna mengisi form komentar 4. Sistem menyimpan data komentar ke database 5. Sistem menampilkan komentar	*Black box*
Menampilkan Timeline Informasi Produk Diskon	1. Pengguna membuka menu timeline diskon Bandung Fashion Sale 2. Sistem mencari informasi produk diskon hasil dari pengelolaan filtering data Twitter 3. Data timeline informasi diskon ditampilkan	*Black box*
Pengujian Aplikasi *Back end Website*		
Item Uji	**Detail Pengujian**	**Jenis Uji**
Login Admin	1. Memasukkan *username* 2. Memasukkan *password*	*Black box*
Pengelolaan Data Toko	1. Tambah data toko 2. Edit data toko	*Black box*

	3. Hapus data toko	
Pengelolaan Data Diskon	1. Tambah data diskon 2. Edit data diskon 3. Hapus data diskon	*Black box*
Pengelolaan Hasil Filtering Data Twitter	1. Mengambil data Twitter 2. Seleksi data Twitter 3. Menampilkan data Twitter	*Black box*
Penyajian Data Webservice	1. Request data toko 2. Request data diskon 3. Request data lokasi 4. Request data rekomendasi	*Black box*

4.2.1.3 Kasus dan Hasil Pengujian

4.2.1.3.1 Pengujian *Black Box* pada *Platform Mobile*

Pengujian *black box* bertujuan untuk mengukur kinerja dari perangkat lunak yang dibangun, apakah fungsinya dapat berjalan dengan baik atau tidak. Adapun kasus dan hasil pengujian yang dilakukan adalah sebagai berikut:

1. Pengujian Mendeteksi Lokasi Pengguna

Berikut ini pengujian dari mendeteksi lokasi pengguna yang dapat dilihat pada tabel 4.8 Pengujian Mendeteksi Lokasi Pengguna.

Tabel 4. 8 Pengujian *Get Location*

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Membuka aplikasi, contoh masukkan : GPS dinyalakan	Dapat mendeteksi lokasi pengguna	Lokasi pengguna terdeteksi	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Membuka aplikasi, contoh masukkan : GPS dimatikan	Tidak dapat mendeteksi lokasi pengguna	Lokasi pengguna tidak terdeteksi	[√] Diterima [ ] Ditolak

2. Pengujian Menampilkan Informasi Toko Pakaian Diskon

Berikut ini pengujian dari menampilkan informasi toko pakaian diskon yang dapat dilihat pada tabel 4.9 Pengujian Menampilkan Informasi Toko Pakaian Diskon.

Tabel 4. 9 Pengujian Menampilkan Informasi Toko Pakaian Diskon

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Membuka aplikasi	Dapat menampilkan *katalog* produk-produk diskon	Menampilkan *katalog* produk-produk diskon	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Membuka aplikasi	Tidak menampilkan *katalog* produk-produk diskon	Pengguna tidak bisa melihat *katalog* produk-produk diskon	[√] Diterima [ ] Ditolak

3. Pengujian Mencari Informasi Lokasi Toko Pakaian

Berikut ini pengujian dari mendapatkan rekomendasi toko terdekat yang dapat dilihat pada table 4.10 Mencari Informasi Lokasi Toko Pakaian.

Tabel 4. 10 Mencari Informasi Lokasi Toko Pakaian

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Memilih menu toko diskon terdekat	Menampilkan peta menuju lokasi toko pakaian	Tampil peta menuju lokasi toko pakaian	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Memilih menu toko diskon terdekat	Menampilkan halaman kosong	Menampilkan halaman kosong	[√] Diterima [ ] Ditolak

4. Pengujian Mendapatkan Rekomendasi Toko Terdekat

Berikut ini pengujian dari mendapatkan rekomendasi toko terdekat yang dapat dilihat pada table 4.11 Mendapatkan Rekomendasi Toko Terdekat.

Tabel 4. 11 Mendapatkan Rekomendasi Toko Terdekat

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Memilih menu toko diskon terdekat	Pilih menu toko diskon terdekat	Pilih menu toko diskon terdekat	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Memilih menu toko diskon terdekat	Menampilkan halaman kosong	Menampilkan halaman kosong	[√] Diterima [ ] Ditolak

5. Pengujian Mencari Informasi Produk Diskon

Berikut ini pengujian dari mencari informasi produk diskon yang dapat dilihat pada tabel 4.12 Mencari Informasi Produk Diskon.

Tabel 4. 12 Mencari Informasi Produk Diskon

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Pilih menu cari diskon, contoh masukkan : nominal diskon : 50%	Menampilkan hasil pencarian produk diskon	Menampilkan produk diskon dengan nominal diskon 50%	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Pilih menu cari diskon, contoh masukkan nominal diskon : lima puluh persen	Tidak menampilkan hasil pencarian produk diskon	Tidak menampilkan produk diskon yang dicari	[√] Diterima [ ] Ditolak

6. Pengujian Menampilkan Kategori Produk Diskon

Berikut ini pengujian dari menampilkan kategori produk diskon yang dapat dilihat pada tabel 4.13 Menampilkan Kategori Produk Diskon.

Tabel 4. 13 Menampilkan Kategori Produk Diskon

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan

Pilih menu kategori diskon, contoh masukkan : kategori : *All Item*	Dapat menampilkan produk diskon kategori *All Item*	Menampilkan produk diskon kategori *All Item*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Pilih menu cari diskon, contoh masukkan : kategori : *All Item*	Menampilkan halaman kosong	Menampilkan halaman kosong	[√] Diterima [ ] Ditolak

7. Pengujian Mengisi Komentar

Berikut ini pengujian dari mengisi komentar yang dapat dilihat pada tabel 4.14 Mengisi Komentar.

Tabel 4. 14 Mengisi Komentar

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Pilih detail produk diskon, contoh masukkan : Produk yang berkualitas	Mengisi kolom komentar	Kolom komentar terisi	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Pilih detail produk diskon, contoh masukkan : (kosong)	Tidak mengisi kolom komentar	Kolom komentar tidak terisi	[√] Diterima [ ] Ditolak

8. Pengujian Menampilkan Timeline Informasi Diskon

Berikut ini pengujian dari menampilkan timeline informasi diskon yang dapat dilihat pada tabel 4.15 Menampilkan Timeline Informasi Diskon.

Tabel 4. 15 Menampilkan Timeline Informasi Diskon

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Pilih menu *timeline* informasi diskon	Dapat menampilkan informasi produk diskon dari data Twitter	Menampilkan informasi produk diskon dari data Twitter	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Pilih menu *timeline* informasi diskon	Menampilkan halaman kosong	Menampilkan halaman kosong	[√] Diterima [ ] Ditolak

4.2.1.3.2 Pengujian *Black Box* pada *Platform Website*

1. Pengujian *Login* Admin

Berikut ini pengujian dari *Login* Admin yang dapat dilihat pada tabel 4.16 *Login* Admin.

Tabel 4. 16 *Login* Admin

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Username dan *password*, contoh masukkan : *username* : admin *password* : abcd	*Text field username* dan *password* dapat terisi sesuai dengan data yang dimasukkan	Isi *text field username* dan *password* sesuai dengan data yang dimasukkan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Username dan *password* kosong, contoh masukkan : *username* : - (kosong) *password* : - (kosong)	Menampilkan pesan pesan "*Username* dan *password* wajib diisi".	Tampil pesan "*Username* dan *password* wajib diisi".	[√] Diterima [ ] Ditolak
Username dan *password* tidak terdaftar, contoh masukkan : *username* : ^&*() (salah)	Menampilkan pesan "*Username* dan *password* salah / tidak sesuai. Silahkan ulangi kembali"	Tampil pesan "*Username* dan *password* salah / tidak sesuai. Silahkan ulangi kembali".	[√] Diterima [ ] Ditolak

password : !@#$% (salah)			

2. Pengujian Tambah Data Toko

Berikut ini pengujian dari Tambah Data Toko yang dapat dilihat pada tabel 4.17 Tambah Data Toko.

Tabel 4. 17 Tambah Data Toko

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Text field nama toko, alamat, *latitude*, *longitude* dan *browse* gambar, contoh masukkan : nama toko : Linoleum alamat : Jl. Sariasih No. 21 Sarijadi Bandung *latitude* : -6.8756687 *longitude* : 107.5754563 gambar : linoleum.jpg	*Text field* nama toko, alamat, *latitude*, *longitude* dan *browse* gambar sesuai dengan data yang dimasukkan	Isi *text field* nama toko, alamat, *latitude*, *longitude* dan *browse* gambar sesuai dengan data yang dimasukkan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Text field nama toko, alamat, *latitude*, *longitude* dan *browse* gambar yang akan	Menampilkan pesan "Harap isi bidang ini".	Tampil pesan "Harap isi bidang ini".	[√] Diterima [ ] Ditolak

diunggah tidak terisi, contoh masukkan : nama toko : - (kosong) alamat : - (kosong) *latitude* : - (kosong) *longitude* : - (kosong) gambar : - (kosong)			

3. Pengujian Edit Data Toko

Berikut ini pengujian dari Edit Data Toko yang dapat dilihat pada tabel 4.18 Edit Data Toko.

Tabel 4. 18 Edit Data Toko

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Text field nama toko, alamat, *latitude*, *longitude* dan *browse* gambar, contoh masukkan : nama toko : Linoleum alamat : Jl. Sultan Agung No. 9 Bandung *latitude* : -6.9044587 *longitude* : 107.6098383 gambar : linoleum.jpg	*Text field* nama toko, alamat, *latitude*, *longitude* dan *browse* gambar sesuai dengan data yang dimasukkan	Isi *text field* nama toko, alamat, *latitude*, *longitude* dan *browse* gambar sesuai dengan data yang dimasukkan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			

Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Text field nama toko, alamat, *latitude*, *longitude* dan *browse* gambar yang akan diunggah tidak terisi, contoh masukkan : nama toko : - (kosong) alamat : - (kosong) *latitude* : - (kosong) *longitude* : - (kosong) gambar : - (kosong)	Menampilkan pesan "Harap isi bidang ini".	Tampil pesan "Harap isi bidang ini".	[√] Diterima [ ] Ditolak

4. Pengujian Hapus Data Toko

Berikut ini pengujian dari Hapus Data Toko yang dapat dilihat pada tabel 4.19 Hapus Data Toko.

Tabel 4. 19 Hapus Data Toko

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Pilih data toko yang akan dihapus, contoh masukkan : Toko : Linoleum	Menampilkan pesan konfirmasi "Anda yakin akan menghapus data toko dengan id … ?	Menampilkan pesan konfirmasi "Anda yakin akan menghapus data toko dengan id … ?	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			

Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Pilih data toko yang akan dihapus, contoh masukkan : Toko : Linoleum	Tidak menampilkan pesan konfirmasi "Anda yakin akan menghapus data toko dengan id … ?	Tidak menampilkan pesan konfirmasi "Anda yakin akan menghapus data toko dengan id … ?	[√] Diterima [ ] Ditolak

5. Pengujian Tambah Data Diskon

Berikut ini pengujian dari Tambah Data Diskon yang dapat dilihat pada tabel 4.20 Tambah Data Diskon.

Tabel 4. 20 Tambah Data Diskon

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Combo box nama toko, nilai diskon, periode diskon, detail, kategori, gambar, contoh masukkan : nama toko : Arena Experience nilai diskon : 30% periode diskon : 12/08/2015 - 01/11/2016 detail diskon : Go get Discount 30% ALL	*Combo box* nama toko, *text field* nilai diskon, periode diskon, detail, kategori dan *browse* gambar sesuai dengan data yang dimasukkan	Isi *combo box* nama toko, *text field* nilai diskon, periode diskon, detail, kategori dan *browse* gambar sesuai dengan data yang dimasukkan	[√] Diterima [ ] Ditolak

ITEMS At @ArenaExperience Jl. Ambon No.9 - Bandung. Till January, 11th 2016. kategori : *All Item* gambar : CVyKdHgUkAA5gY8.jpg			
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Combo box nama toko, *text field* nilai diskon, periode diskon, detail, kategori dan *browse* gambar yang akan diunggah tidak terisi, contoh masukkan : nama toko : - (kosong) nilai diskon : - (kosong) periode diskon : - (kosong) detail diskon : - (kosong) kategori : - (kosong) gambar : - (kosong)	Menampilkan pesan "Harap isi bidang ini".	Tampil pesan "Harap isi bidang ini".	[√] Diterima [ ] Ditolak

6. Pengujian Edit Data Diskon

Berikut ini pengujian dari Edit Data Diskon yang dapat dilihat pada tabel 4.21 Edit Data Diskon.

Tabel 4. 21 Edit Data Diskon

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Combo box nama toko, nilai diskon, periode diskon, detail, kategori, gambar, contoh masukkan : nama toko : Arena Experience nilai diskon : 30% periode diskon : 12/08/2015 - 01/11/2016 detail diskon : Go get Discount 30% ALL ITEMS At @ArenaExperience Jl. Ambon No.9 - Bandung. Till January, 11th 2016. kategori : *All Item* gambar : images.jpg	*Combo box* nama toko, *text field* nilai diskon, periode diskon, detail, kategori dan *browse* gambar sesuai dengan data yang dimasukkan	Isi *combo box* nama toko, *text field* nilai diskon, periode diskon, detail, kategori dan *browse* gambar sesuai dengan data yang dimasukkan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			

Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Combo box nama toko, *text field* nilai diskon, periode diskon, detail, kategori dan *browse* gambar yang akan diunggah tidak terisi, contoh masukkan : nama toko : Arena Experience nilai diskon : - (kosong) periode diskon : - (kosong) detail diskon : - (kosong) kategori : - (kosong) gambar : - (kosong)	Menampilkan pesan "Harap isi bidang ini"	Tampil pesan "Harap isi bidang ini"	[√] Diterima [ ] Ditolak

7. Pengujian Hapus Data Diskon

Berikut ini pengujian dari Hapus Data Diskon yang dapat dilihat pada tabel 4.22 Hapus Data Diskon.

Tabel 4. 22 Hapus Data Diskon

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Pilih data diskon yang akan dihapus, contoh masukkan : Diskon : Linoleum	Menampilkan pesan konfirmasi "Anda yakin akan menghapus data	Menampilkan pesan konfirmasi "Anda yakin	[√] Diterima [ ] Ditolak

	diskon dengan id … ?	akan menghapus data diskon dengan id … ?	
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Pilih data toko yang akan dihapus, contoh masukkan : Toko : Linoleum	Tidak menampilkan pesan konfirmasi "Anda yakin akan menghapus data toko dengan id … ?	Tidak menampilkan pesan konfirmasi "Anda yakin akan menghapus data toko dengan id … ?	[√] Diterima [ ] Ditolak

8. Pengujian Pengelolaan Hasil *Filtering* Data Twitter

Berikut ini pengujian Pengelolaan Hasil *Filtering* Data Twitter yang dapat dilihat pada tabel 4.23 Hasil *Filtering* Data Twitter.

Tabel 4. 23 Hasil *Filtering* Data Twitter

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Keyword contoh masukkan : *keyword* : discount fashion bandung	Dapat menampilkan hasil *filtering* data Twitter sesuai *keyword* yang dimasukkan	Menampilkan hasil *filtering* data Twitter sesuai *keyword* yang dimasukkan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			

Data Masukkan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Username dan *password* kosong, contoh masukkan : *username* : - (kosong) *password* : - (kosong)	Tidak dapat menampilkan hasil *filtering* data Twitter sesuai *keyword* yang dimasukkan	Tidak menampilkan hasil *filtering* data Twitter sesuai *keyword* yang dimasukkan	[√] Diterima [ ] Ditolak
Keyword contoh masukkan : *keyword* : diskon makanan	Tidak dapat menampilkan hasil *filtering* data Twitter sesuai dengan kebutuhan aplikasi karena *keyword* yang dimasukkan tidak sesuai	Tidak menampilkan hasil *filtering* data Twitter sesuai dengan kebutuhan aplikasi karena *keyword* yang dimasukkan tidak sesuai	[√] Diterima [ ] Ditolak

4.2.1.3.3 Pengujian *White Box*

Pengujian *white-box* atau *white-box testing* merupakan metode perancangan test case yang menggunakan struktur kontrol dari perancangan prosedural dalam mendapatkan test case. Adapun metode yang digunakan dalam pengujian *white-box* ini adalah metode Basis Path. Metode Basis Path mengijinkan pendesain kasus uji untuk membuat perkiraan lojik yang kompleks dari desain prosedural dan menggunakan perkiraan ini untuk mendefinisikan aliran eksekusi. Pengujian *white-box* rekomendasi toko pakaian terdekat menggunakan algoritma *Haversine Formula*.

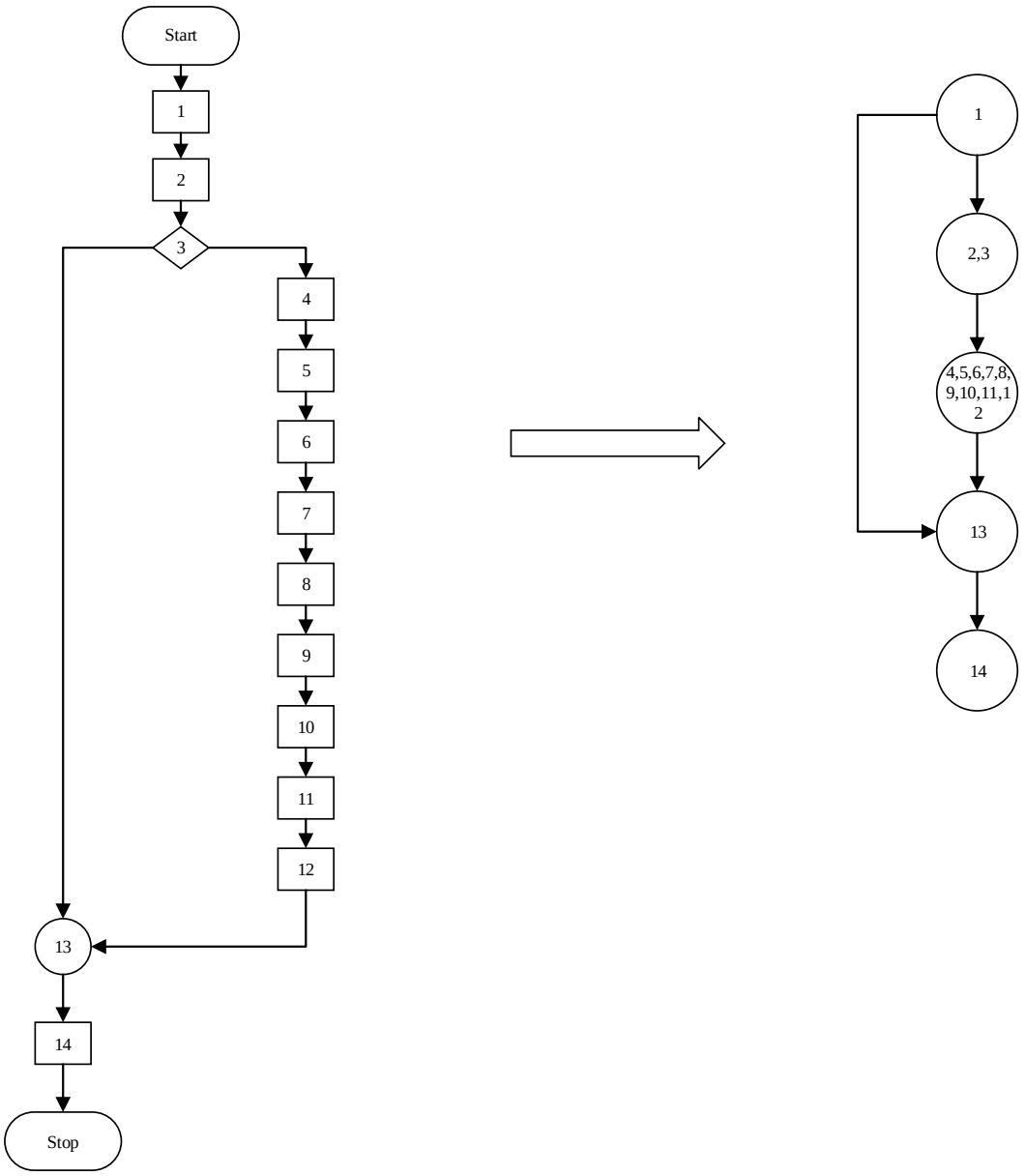
1. Mengubah bentuk *pseucode* menjadi *flowchart* kemudian menjadi *flowgraph*

Berikut pseudocode algoritma *Haversine Formula*

```
$q = mysql_query("SELECT * ,  ROUND(( 6371 * ACOS( COS( RADIANS(
'$latitudetoko' ) ) * COS( RADIANS(  `lat_toko` ) ) * COS( RADIANS(
`long_toko` ) - RADIANS( '$longitudetoko' ) ) + SIN( RADIANS( '$latitudetoko' )
) * SIN( RADIANS(  `lat_toko` ) ) ) ), 2 ) AS distance FROM  `t_toko`  HAVING
distance <10 ORDER BY distance LIMIT 0 , 50");
$response["TOKO_LIST"] = array();
while ($a = mysql_fetch_array($q)){
 $output = array();
 $output["id_toko"] = $a["id_toko"];
 $output["nama_toko"] = $a["nama_toko"];
 $output["alamat_toko"]=$a["alamat_toko"];
 $output["gambar"]=$a["gambar"];
 $output["lat_toko"]=$a["lat_toko"];
 $output["long_toko"]=$a["long_toko"];
 $output["distance"]=$a["distance"];
 array_push($response["TOKO_LIST"], $output);
}
echo json_encode($response);
```

Gambar 4. 1 *Pseudocode* Algoritma *Haversine Formula*

Berdasarkan pseudocode pada gambar 4.1 *Pseudocode* Algoritma *Haversine Formula*, maka dibuat *flowchart* lalu diubah ke dalam bentuk *flowgraph* yang dapat dilihat pada gambar 4.2 Mengubah *flowchart* ke *flowgraph*.



Gambar 4. 2 Mengubah *Flowchart* ke *Flowgraph*

2. Menghitung *Cyclometic Complexity*

   Dari gambar *flowgraph* di atas dapat dihitung *cyclomatic complexity*, yakni:

   1) V(G) = E – N + 2

      V(G) = 5 – 5 + 2

      V(G) = 2

      E = Jumlah busur pada *flowgraph*

      N = Jumlah simpul pada *flowgraph*

2) Menghitung Jumlah Region

Jumlah region = jumlah kurva tertutup + 1 (region terluar)

Jumlah region = 1 + 1

Jumlah region = 2

3) Menghitung Jumlah Predicate Node

Predicate node = 1

V(G) = jumlah predicate node + 1

V(G) = 1 + 1

V(G) = 2

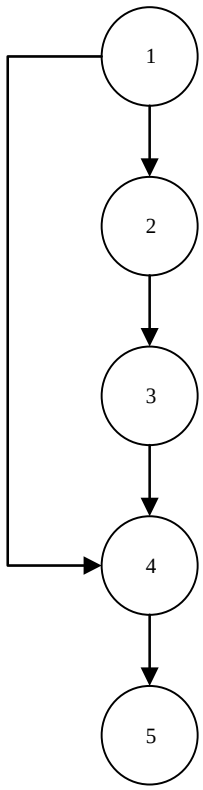
3. Menghitung Independent Path

Dari hasil perhitungan cyclomatic complexity terdapat 2 path independent path yaitu :

Path 1 : 1, 13, 14

Path 2 : 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11 ,12, 13, 14

Untuk melakukan uji coba basis path di atas digunakan graph matrik. Graph matrik merupakan empat persegi yang mempunyai ukuran yang sama dengan jumlah node pada *flowgraph*. Sebelum menghitung menggunakan graph matrik, dilakukan penomoran ulang terhadap *flowgraph* yang dapat dilihat pada gambar 4.3 Penomoran ulang *flowgraph*, dan perhitungan graph matrik dapat dilihat pada tabel 4.24 Graph matrik.



Gambar 4. 3 Penomoran Ulang *Flowgraph*

Tabel 4. 24 Graph Matrik

Node	1	2	3	4	5	Jumlah n(E)-1
1		1		1		2-1=1
2			1			1-1=0
3				1		1-1=0
4					1	1-1=0
5						0
Sum(E)+1						1+1=2

4.2.1.4 Kesimpulan Pengujian Alpha

Berdasarkan pengujian yang telah dilakukan, maka kesimpulan yang didapatkan bahwa poses-proses pada pembangunan perangkat lunak Bandung Fashion Sale telah melalui tahap perbaikan dan sudah dimaksimalkan begitupun secara fungsional sudah menghasilkan output yang diharapkan sehingga sistem sudah layak untuk digunakan dan dari hasil pengujian rekomendasi yang dilakukan, metode *Haversine Formula* yang sudah diimplementasikan menghasilkan rekomendasi yang baik bila mempunyai jarak terdekat dengan lokasi keberadaan pengguna dan menghasilkan rekomendasi yang buruk bila mempunyai jarak yang jauh dengan lokasi keberadaan pengguna. Hasil rekomendasi yang baik inilah yang akan direkomendasikan ke pengguna sebagai rekomendasi yang sesuai dengan jarak lokasi keberadaan pengguna.

4.2.1.4.1 Kesimpulan Pengujian *White Box*

Berdasarkan hasil pengujian yang sudah dilakukan diperoleh jumlah region, *cyclomatic complexity*, dan independent path adalah sama besar yaitu 2, sehingga dapat ditarik kesimpulan bahwa algoritma tersebut sudah benar. Dengan kata lain bahwa sistem rekomendasi toko pakaian terdekat pada aplikasi Bandung Fashin Sale yang dibuat telah layak digunakan.

4.2.1.4.2 Kesimpulan Pengujian *Black Box*

Berdasarkan pengujian yang dilakukan maka dapat disimpulkan bahwa poses-proses pada pembangunan perangkat lunak Bandung Fashion Sale ini telah melalui tahap perbaikan dan sudah dimaksimalkan bergitupun secara *fungsional* sudah menghasilkan *output* yang diharapkan sehingga sistem sudah layak untuk digunakan dan dari hasil pengujian rekomendasi yang dilakukan, metode *haversine formula* yang sudah diimplementasikan menghasilkan rekomendasi yang baik dengan membandingkan toko yang memiliki jarak terdekat dengan keberadaan pengguna. Hasil rekomendasi yang baik inilah yang akan direkomendasikan ke pengguna sebagai rekomendasi yang sesuai dengan jarak terdekat dan produk yang diinginkan.

4.2.2 Pengujian Beta

Pengujian beta merupakan pengujian yang dilakukan secara objektif yang dilakukan secara langsung oleh pengguna yang nantinya akan menggunakan perangkat lunak Bandung Fashion Sale ini diataranya adalah admin Bandung Fashion Sale dan para pencari diskon pakaian di Kota Bandung. Penelitian ini dilakukan di Kota Bandung dengan menggunakan teknik kuesioner untuk admin dan pengguna. Hal ini dilakukan agar dapat mengetahui sejauh mana aplikasi yang dibangun ini dapat membantu dan dapat menyelesaikan masalah yang sudah dijelaskan pada perumusan masalah.

4.2.2.1 Skenario Pengujian Beta

Kuesioner Pengujian merupakan teknik pengolahan data yang merupakan langkah dalam mengolah data yang telah didapat untuk dijadikan hasil penelitian sehinggan dapat ditarik kesimpulan. Berikut ini adalah pertanyaan kuesioner yang akan diajukan kepada pengguna dan pihak toko yang nantinya akan menggunakan aplikasi ini. Pertanyaannya untuk pengguna adalah sebagai berikut:

1. Apakah anda setuju aplikasi Bandung Fashion Sale dapat membantu anda mencari informasi diskon dan toko pakaian?
2. Apakah anda setuju aplikasi Bandung Fashion Sale dapat membantu anda untuk menemukan lokasi toko pakaian terdekat?
3. Apakah anda setuju aplikasi Bandung Fashion Sale dapat membantu anda menemukan toko pakaian yang menyediakan produk dengan jumlah diskon yang anda inginkan?
4. Apakah anda setuju hasil rekomendasi toko pakian terdekat sesuai dengan yang anda harapkan?
5. Apakah anda setuju tampilan/antarmuka aplikasi Bandung Fashion Sale ini mudah dimengerti?
6. Apakah aplikasi Bandung Fashion Sale ini mudah digunakan?
7. Saran

Pertanyaannya untuk pihak toko adalah sebagai berikut:

1. Apakah anda setuju aplikasi Bandung Fashion Sale dapat membantu dalam mempublikasikan produk diskon?
2. Apakah anda setuju tampilan/antarmuka aplikasi Bandung Fashion Sale ini mudah dimengerti?
3. Apakah aplikasi Bandung Fashion Sale ini mudah digunakan?
4. Saran

Jawaban untuk kuesioner :

a. Sangat Setuju
b. Setuju
c. Cukup Setuju
d. Tidak Setuju
e. Sangat Tidak Setuju

Penghitungan kuisioner menggunakan skala Likert. Menurut Sugiyono, skala Likert digunakan untuk mengukur sikap, pendapat dan persepsi seseorang atau kelompok orang tentang fenomena sosial. Fenomena sosial disini adalah tanggapan responden mengenai aplikasi Bandung Fashion Sale yang dibuat untuk membantu pengguna dalam mencari produk pakaian yang sedang diskon. Berikut adalah langkah – langkah penelitian untuk menghitung Hasil Kuesioner.

a. Pada kuisioner ini, memberikan lima kategori nilai pilihan jawaban beserta skor dari masing – masing jawaban yang dapat dilihat pada Tabel 4.25 kategori nilai.

Tabel 4. 25 Kategori Nilai

Kategori	Nilai	Presentase (%)
Sangat Setuju	5	81-100
Setuju	4	61-80
Cukup Setuju	3	41-60
Tidak Setuju	2	21-40
Sangat Tidak Setuju	1	0-20

b. Untuk menghitung jumlah maksimum skor kriterium jumlah pertanyaan = 5, dan jumlah responden = 30 yaitu dengan rumus :

Kriterium = Jumlah soal X Jumlah responden (4.1)

Maka nilai dari kuesioner yang dibuat adalah 5 X 30 = 150

c. Sedangkan untuk mengetahui jumlah jawaban dari responden dalam bentuk presentase, digunaka rumus sebagai berikut.

$$P = \frac{total\ nilai}{skor\ ideal} X\ 100\% \quad (4.2)$$

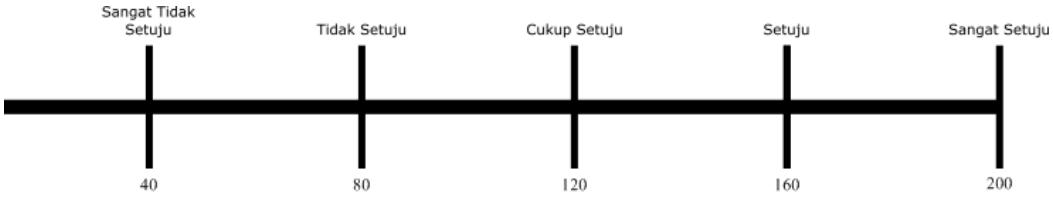
Keterangan:

P : Nilai presentase yang dicari

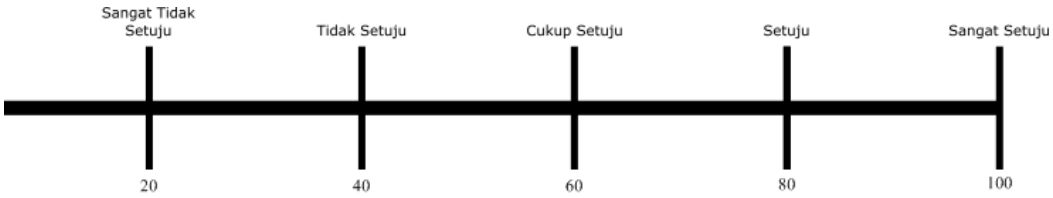
Total nilai : Jumlah frekuensi dikalikan nilai yang ditetapkan jawaban

Skor ideal : Nilai tertinggi dilakukan dengan jumlah sampel

d. Skor yang telah didapat selanjutnya dimasukkan kedalam bentuk *Interval Ratting Scale* untuk pengguna dan toko seperti pada gambar 4.4 dan 4.5 berikut :



Gambar 4. 4 *Interval Ratting Scale* Pengguna



Gambar 4. 5 *Interval Ratting Scale* Toko

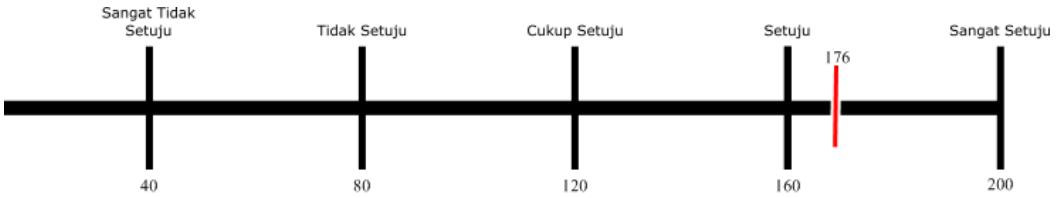
4.2.2.2 Penghitungan Hasil Kuesioner Pengguna

1. Apakah anda setuju aplikasi Bandung Fashion Sale dapat membantu anda mencari informasi diskon dan toko pakaian?

Tabel 4. 26 Kuesioner Pengguna Pertanyaan 1

Kategori Jawaban	Skor	Frekuensi Jawaban	Jumlah Skor	Nilai Presentase (%)
Sangat Setuju	5	13	65	(176 / (43 x 5)) x 100 = 81.86%
Setuju	4	23	92	
Cukup Setuju	3	5	15	
Tidak Setuju	2	2	4	
Sangat Tidak Setuju	1	0	0	
Jumlah		43	176	

Berdasarkan perhitungan diatas, jumlah skor yang didapat adalah sebanyak 176 skor dengan skala kategori jawaban diantara setuju dan sangat setuju. Sedangkan hasil dari nilai presentasi responden adalah 81.86% dari nilai yang diharapkan sebesar 100%. Maka dapat disimpulkan, responden sangat setuju bahwa aplikasi ini dapat membantu mencari informasi diskon dan toko pakaian. Secara kontinium dapat digambarkan seperti pada gambar 4.6.



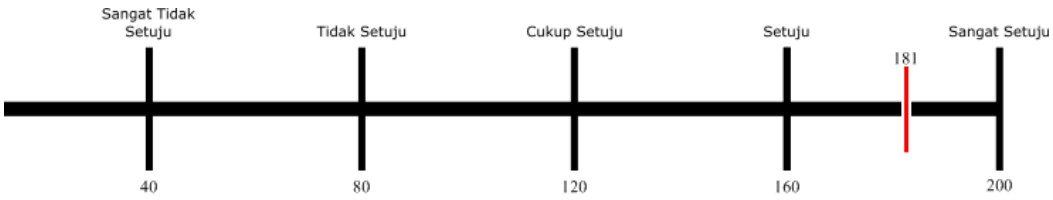
Gambar 4. 6 *Interval Ratting Scale* Kuesioner Pengguna Pertanyaan 1

2. Apakah anda setuju aplikasi Bandung Fashion Sale dapat membantu anda untuk menemukan lokasi toko pakaian terdekat?

Tabel 4. 27 Kuesioner Pengguna Pertanyaan 2

Kategori Jawaban	Skor	Frekuensi Jawaban	Jumlah Skor	Nilai Presentase (%)
Sangat Setuju	5	16	80	(181 / (43 x 5)) x 100 = 80.19%
Setuju	4	20	80	
Cukup Setuju	3	7	21	
Tidak Setuju	2	0	0	
Sangat Tidak Setuju	1	0	0	
Jumlah		43	181	

Berdasarkan perhitungan diatas, jumlah skor yang didapat adalah sebanyak 181 skor dengan skala kategori jawaban diantara setuju dan sangat setuju. Sedangkan hasil dari nilai presentasi responden adalah 80.19% dari nilai yang diharapkan sebesar 100%. Maka dapat disimpulkan, responden sangat setuju bahwa aplikasi ini dapat membantu untuk menemukan lokasi toko pakaian terdekat. Secara kontinium dapat digambarkan seperti pada gambar 4.7.



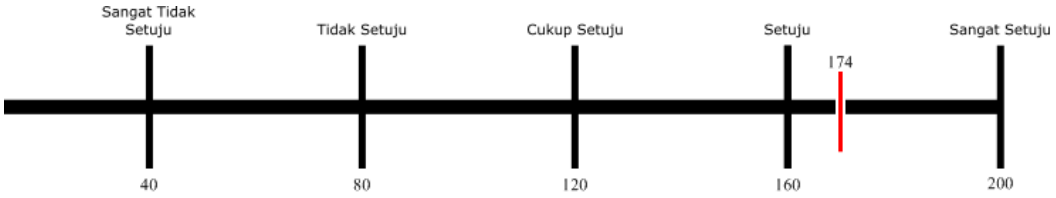
Gambar 4. 7 *Interval Ratting Scale* Kuesioner Pengguna Pertanyaan 2

3. Apakah anda setuju aplikasi Bandung Fashion Sale dapat membantu anda menemukan toko pakaian yang menyediakan produk dengan jumlah diskon yang anda inginkan?

Tabel 4. 28 Kuesioner Pengguna Pertanyaan 3

Kategori Jawaban	Skor	Frekuensi Jawaban	Jumlah Skor	Nilai Presentase (%)
Sangat Setuju	5	10	50	(174 / (43 x 5)) x 100 = 80.93%
Setuju	4	25	100	
Cukup Setuju	3	8	24	
Tidak Setuju	2	0	0	
Sangat Tidak Setuju	1	0	0	
Jumlah		43	174	

Berdasarkan perhitungan diatas, jumlah skor yang didapat adalah sebanyak 174 skor dengan skala kategori jawaban diantara setuju dan sangat setuju. Sedangkan hasil dari nilai presentasi responden adalah 80.93% dari nilai yang diharapkan sebesar 100%. Maka dapat disimpulkan, responden setuju bahwa aplikasi ini dapat membantu menemukan toko pakaian yang menyediakan produk dengan jumlah diskon yang diinginkan. Secara kontinium dapat digambarkan seperti pada gambar 4.8.



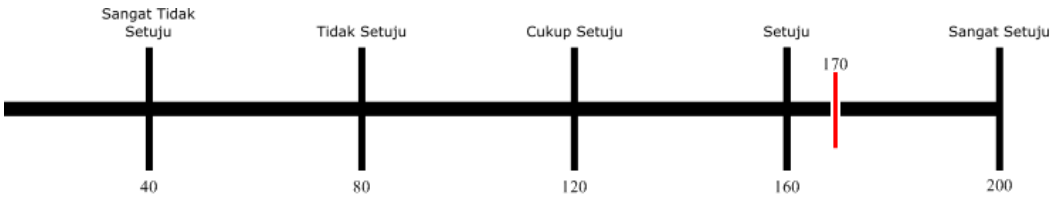
Gambar 4. 8 *Interval Ratting Scale* Kuesioner Pengguna Pertanyaan 3

4. Apakah anda setuju hasil rekomendasi toko pakian terdekat sesuai dengan yang anda harapkan?

Tabel 4. 29 Kuesioner Pengguna Pertanyaan 4

Kategori Jawaban	Skor	Frekuensi Jawaban	Jumlah Skor	Nilai Presentase (%)
Sangat Setuju	5	10	50	(170 / (43 x 5)) x 100 = 79.07%
Setuju	4	24	96	
Cukup Setuju	3	8	24	
Tidak Setuju	2	1	2	
Sangat Tidak Setuju	1	0	0	
Jumlah		43	170	

Berdasarkan perhitungan diatas, jumlah skor yang didapat adalah sebanyak 181 skor dengan skala kategori jawaban diantara setuju dan sangat setuju. Sedangkan hasil dari nilai presentasi responden adalah 79.07% dari nilai yang diharapkan sebesar 100%. Maka dapat disimpulkan, responden setuju bahwa aplikasi ini dapat merekomendasi toko pakian terdekat sesuai dengan yang diharapkan. Secara kontinium dapat digambarkan seperti pada gambar 4.9.



Gambar 4. 9 *Interval Ratting Scale* Kuesioner Pengguna Pertanyaan 4

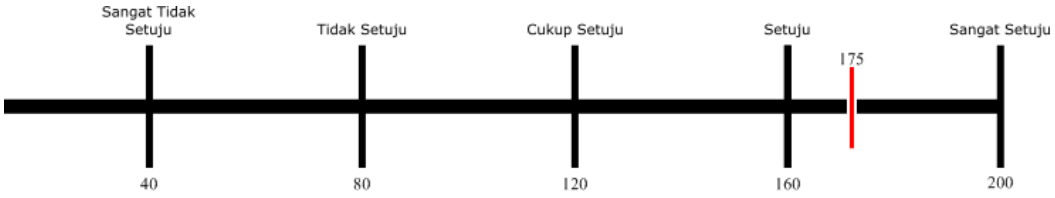
5. Apakah anda setuju tampilan/antarmuka perangkat aplikasi Bandung Fashion Sale ini mudah dimengerti?

Tabel 4. 30 Kuesioner Pengguna Pertanyaan 5

Kategori Jawaban	Skor	Frekuensi Jawaban	Jumlah Skor	Nilai Presentase (%)
Sangat Setuju	5	13	65	
Setuju	4	20	80	

Cukup Setuju	3	10	30	(175 / (43 x 5)) x 100 = 81.40%
Tidak Setuju	2	0	0	
Sangat Tidak Setuju	1	0	0	
Jumlah		43	175	

Berdasarkan perhitungan diatas, jumlah skor yang didapat adalah sebanyak 181 skor dengan skala kategori jawaban diantara setuju dan sangat setuju. Sedangkan hasil dari nilai presentasi responden adalah 81.40% dari nilai yang diharapkan sebesar 100%. Maka dapat disimpulkan, responden sangat setuju bahwa tampilan/antarmuka perangkat aplikasi Bandung Fashion Sale ini mudah dimengerti. Secara kontinium dapat digambarkan seperti pada gambar 4.10.



Gambar 4. 10 *Interval Ratting Scale* Kuesioner Pengguna Pertanyaan 5

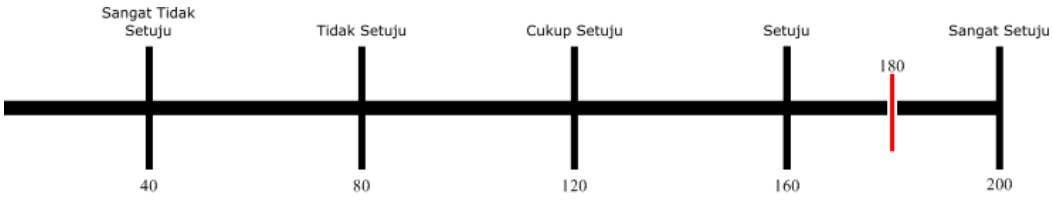
6. Apakah aplikasi Bandung Fashion Sale ini mudah digunakan?

Tabel 4. 31 Kuesioner Pengguna Pertanyaan 6

Kategori Jawaban	Skor	Frekuensi Jawaban	Jumlah Skor	Nilai Presentase (%)
Sangat Setuju	5	15	75	(180 / (43 x 5)) x 100 = 83.72%
Setuju	4	21	84	
Cukup Setuju	3	7	21	
Tidak Setuju	2	0	0	
Sangat Tidak Setuju	1	0	0	
Jumlah		43	180	

Berdasarkan perhitungan diatas, jumlah skor yang didapat adalah sebanyak 181 skor dengan skala kategori jawaban diantara setuju dan sangat setuju. Sedangkan hasil dari nilai presentasi responden adalah 83.72% dari nilai yang

diharapkan sebesar 100%. Maka dapat disimpulkan, responden sangat setuju bahwa aplikasi Bandung Fashion Sale ini mudah digunakan. Secara kontinium dapat digambarkan seperti pada gambar 4.11.



Gambar 4. 11 *Interval Ratting Scale* Kuesioner Pengguna Pertanyaan 6

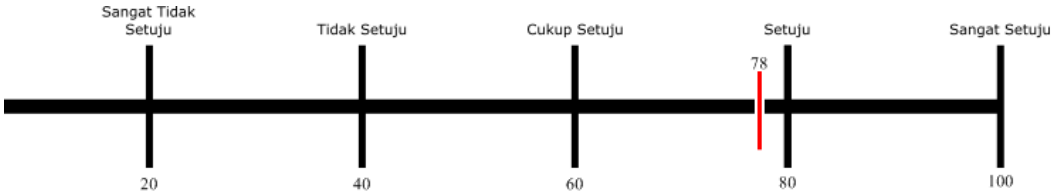
4.2.2.3 Penghitungan Hasil Kuesioner Toko

1. Apakah anda setuju aplikasi Bandung Fashion Sale dapat membantu dalam mempublikasikan produk diskon?

Tabel 4. 32 Kuesioner Toko Pertanyaan 1

Kategori Jawaban	Skor	Frekuensi Jawaban	Jumlah Skor	Nilai Presentase (%)
Sangat Setuju	5	4	20	(78 / (20 x 5)) x 100 = 78%
Setuju	4	10	40	
Cukup Setuju	3	6	18	
Tidak Setuju	2	0	0	
Sangat Tidak Setuju	1	0	0	
Jumlah		20	78	

Berdasarkan perhitungan diatas, jumlah skor yang didapat adalah sebanyak 78 skor dengan skala kategori jawaban diantara setuju dan sangat setuju. Sedangkan hasil dari nilai presentasi responden adalah 78% dari nilai yang diharapkan sebesar 100%. Maka dapat disimpulkan, responden setuju bahwa aplikasi ini dapat membantu mempublikasikan produk diskon. Secara kontinium dapat digambarkan seperti pada gambar 4.12



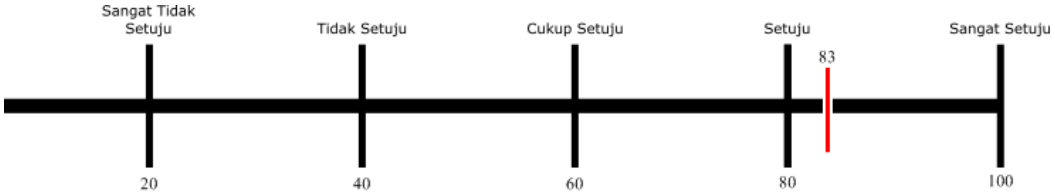
Gambar 4. 12 *Interval Ratting Scale* Kuesioner Toko Pertanyaan 1

2. Apakah anda setuju tampilan/antarmuka perangkat aplikasi Bandung Fashion Sale ini mudah dimengerti?

Tabel 4. 33 Kuesioner Toko Pertanyaan 2

Kategori Jawaban	Skor	Frekuensi Jawaban	Jumlah Skor	Nilai Presentase (%)
Sangat Setuju	5	9	45	(83 / (20 x 5)) x 100 = 83%
Setuju	4	5	20	
Cukup Setuju	3	6	18	
Tidak Setuju	2	0	0	
Sangat Tidak Setuju	1	0	0	
Jumlah		20	83	

Berdasarkan perhitungan diatas, jumlah skor yang didapat adalah sebanyak 83 skor dengan skala kategori jawaban diantara setuju dan sangat setuju. Sedangkan hasil dari nilai presentasi responden adalah 83% dari nilai yang diharapkan sebesar 100%. Maka dapat disimpulkan, responden sangat setuju bahwa tampilan/antarmuka perangkat aplikasi Bandung Fashion Sale ini mudah dimengerti. Secara kontinium dapat digambarkan seperti pada gambar 4.13



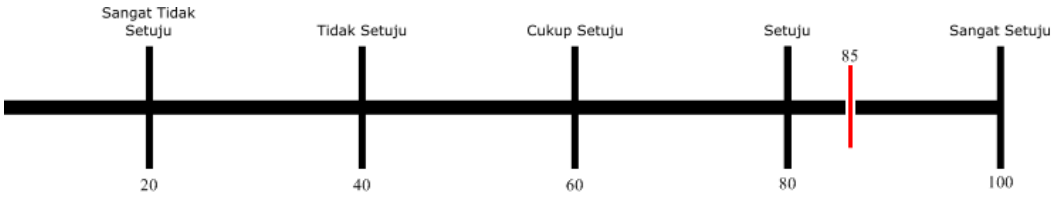
Gambar 4. 13 *Interval Ratting Scale* Kuesioner Toko Pertanyaan 2

3. Apakah aplikasi Bandung Fashion Sale ini mudah digunakan?

Tabel 4. 34 Kuesioner Toko Pertanyaan 3

Kategori Jawaban	Skor	Frekuensi Jawaban	Jumlah Skor	Nilai Presentase (%)
Sangat Setuju	5	10	50	(85 / (20 x 5)) x 100 = 85%
Setuju	4	5	20	
Cukup Setuju	3	5	15	
Tidak Setuju	2	0	0	
Sangat Tidak Setuju	1	0	0	
Jumlah		20	85	

Berdasarkan perhitungan diatas, jumlah skor yang didapat adalah sebanyak 85 skor dengan skala kategori jawaban diantara setuju dan sangat setuju. Sedangkan hasil dari nilai presentasi responden adalah 85% dari nilai yang diharapkan sebesar 100%. Maka dapat disimpulkan, responden sangat setuju bahwa aplikasi Bandung Fashion Sale ini mudah digunakan. Secara kontinium dapat digambarkan seperti pada gambar 4.14



Gambar 4. 14 *Interval Ratting Scale* Kuesioner Toko Pertanyaan 3

4.2.2.4 Kesimpulan Hasil Pengujian Beta

Berdasarkan hasil pengujian beta, dapat diambil kesimpulan bahwa :

1. Aplikasi Bandung Fashion Sale dapat membantu mencari informasi diskon dan toko pakaian.
2. Aplikasi Bandung Fashion Sale dapat membantu untuk menemukan lokasi toko pakaian terdekat.
3. Aplikasi Bandung Fashion Sale dapat membantu menemukan toko pakaian yang menyediakan produk dengan jumlah diskon yang diinginkan.
4. Hasil rekomendasi toko pakian terdekat sesuai dengan yang diharapkan.
5. Tampilan/antarmuka perangkat aplikasi Bandung Fashion Sale mudah dimengerti.
6. Aplikasi Bandung Fashion Sale mudah digunakan.