BAB IV

IMPLEMENTASI DAN PENGUJIAN SISTEM

Pada bab ini akan dilakukan implementasi dan pengujian terhadap aplikasi *tracking* penjualan produk Maicih pada *mobile* dan *web* yang dibangun. Tahapan ini dilakukan setelah perancangan selesai dilakukan dan selanjutnya akan di implementasikan ke dalam bahasa pemrograman. Setelah implementasi maka dilakukan pengujian terhadap sistem yang baru akan dilihat kekurangan-kekurangan pada aplikasi yang baru untuk pengembangan sistem selanjutnya.

4.1 Implementasi Sistem

Tahapan implementasi merupakan tahap penciptaan perangkat lunak, tahap kelanjutan dari kegiatan perancangan sistem. Tahap ini merupakan tahap dimana sistem siap untuk dioperasikan, yang terdiri dari penjelasan mengenai lingkungan implementasi, dan implementasi program. Untuk mendukung aplikasi yang diterapkan pada PT Maicih Inti Synergi, maka dalam hal ini menggunakan perangkat keras dan perangkat lunak yang menunjang dalam pembangunan aplikasi *tracking* berbasis *mobile* dan *web*.

4.1.1 Perangkat Keras Yang Digunakan

Spesifikasi perangkat keras yang digunakan dalam pembangunan aplikasi tracking berbasis *mobile* dan *web* pada PT Maicih Inti Synergi adalah tercantum pada tabel 4.1 sebagai berikut :

Tabel 4.1 Perangkat Keras Yang Digunakan

No.	Perangkat Keras	Spesifikasi
1.	Processor	Kecepatan 2.0 GHz
2.	RAM	1 Gb
3.	Harddisk	80 Gb
4.	Monitor	Monitor 15 Inch
5.	VGA	VGA On-Board 64 Mb
6.	Lan Card	Standar
7.	Koneksi Internet	256 Kbps
8.	Printer	EPSON C45

4.1.2 Perangkat Lunak Yang Digunakan

Spesifikasi perangkat keras yang digunakan dalam pembangunan aplikasi tracking berbasis *mobile* dan *web* pada PT Maicih Inti Synergi adalah tercantum pada table 4.2 sebagai berikut :

Tabel 4.2 Perangkat Lunak Yang Digunakan

No.	Perangkat Lunak	Spesifikasi
1	Sistem Operasi	Windows XP Professional (5.1, Build 2000) Android 2.1 (untuk telepon genggam)
2	Web Browser	Mozilla Firefox, Google Chrome
3	Bahasa Pemrograman	PHP 5.3.1, Android
4	DBMS	MySQL 5.1.41
5	Code Editor	Adobe Dreamweaver CS5
6	UML Modeler	Astah, Microsoft Visio 2007
7	Web Server	Apache 2.2.14

4.1.3 Implementasi Database

Pembuatan database dilakukan dengan menggunakan aplikasi DBMS MySQL. Implementasi database dalam bahasa SQL dijelaskan pada tabel 4.3.

Tabel 4.3 Implementasi Tabel Admin

#	Column	Type	Collation	Attributes	Null	Default	Extra
1	id_admin	int(11)			No	None	AUTO_INCREMENT
2	nama_admin	varchar(50)	latin1_swedish_ci		No	None	
3	email_admin	varchar(50)	latin1_swedish_ci		No	None	
4	telp_admin	varchar(20)	latin1_swedish_ci		No	None	
5	password_admin	varchar(32)	latin1_swedish_ci		No	None	
6	kode_aktivasi	varchar(50)	latin1_swedish_ci		No	None	
7	status	enum('1', '0')	latin1_swedish_ci		No	None	

```
CREATE  TABLE IF NOT EXISTS `appmhcom_maicih`.`admin` (
  `id_admin` INT(11) NOT NULL AUTO_INCREMENT ,
  `nama_admin` VARCHAR(50) NOT NULL ,
  `email_admin` VARCHAR(50) NOT NULL ,
  `telp_admin` VARCHAR(20) NOT NULL ,
  `password_admin` VARCHAR(32) NOT NULL ,
  `kode_aktivasi` VARCHAR(50) NOT NULL ,
  `status` ENUM('1','0') NOT NULL ,
  PRIMARY KEY (`id_admin`) ,
  UNIQUE INDEX `email_admin` (`email_admin` ASC) )
ENGINE = InnoDB
DEFAULT CHARACTER SET = latin1
```

Implementasi dari tabel detailpenjualan seperti tabel 4.4 berikut

Tabel 4.4 Implementasi Tabel Detail Penjualan

#	Column	Type	Collation	Attributes	Null	Default	Extra
1	id_detaildp	int(11)			No	None	AUTO_INCREMENT
2	id_penjualan	int(11)			No	None	
3	id_produk	int(11)			No	None	
4	stok_awal	int(11)			No	None	
5	stok_akhir	int(11)			No	None	

```
CREATE  TABLE IF NOT EXISTS `appmhcom_maicih`.`detailpenjualan` (
  `id_detaildp` INT(11) NOT NULL AUTO_INCREMENT ,
  `id_penjualan` INT(11) NOT NULL ,
  `id_produk` INT(11) NOT NULL ,
  `stok_awal` INT(11) NOT NULL ,
  `stok_akhir` INT(11) NOT NULL ,
  PRIMARY KEY (`id_detaildp`) ,
  INDEX `id_produk` (`id_produk` ASC) ,
```

```
  INDEX `detailpenjualan_ibfk_2` (`id_penjualan` ASC) ,
  CONSTRAINT `detailpenjualan_ibfk_1`
    FOREIGN KEY (`id_produk` )
    REFERENCES `appmhcom_maicih`.`produk` (`id_produk` )
    ON DELETE RESTRICT
    ON UPDATE CASCADE,
  CONSTRAINT `detailpenjualan_ibfk_2`
    FOREIGN KEY (`id_penjualan` )
    REFERENCES `appmhcom_maicih`.`penjualan` (`id_penjualan` )
    ON DELETE RESTRICT
    ON UPDATE CASCADE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = latin1;
```

Implementasi dari tabel Jendral seperti tabel 4.5 berikut

Tabel 4.5 Implementasi Tabel Jendral

	#	Column	Type	Collation	Attributes	Null	Default	Extra
☐	1	**username**	varchar(50)	latin1_swedish_ci		No	*None*	
☐	2	**nama_jendral**	varchar(50)	latin1_swedish_ci		No	*None*	
☐	3	**alamat_jendral**	text	latin1_swedish_ci		No	*None*	
☐	4	**id_kota**	int(11)			No	*None*	
☐	5	**pin_jendral**	varchar(8)	latin1_swedish_ci		No	*None*	
☐	6	**telp_jendral**	varchar(19)	latin1_swedish_ci		No	*None*	
☐	7	**twitter_jendral**	varchar(20)	latin1_swedish_ci		No	*None*	
☐	8	**kend_jendral**	varchar(20)	latin1_swedish_ci		No	*None*	
☐	9	**plat_jendral**	varchar(20)	latin1_swedish_ci		No	*None*	

```
CREATE  TABLE IF NOT EXISTS `appmhcom_maicih`.`jendral` (
  `username` VARCHAR(50) NOT NULL ,
  `nama_jendral` VARCHAR(50) NOT NULL ,
  `alamat_jendral` TEXT NOT NULL ,
  `id_kota` INT(11) NOT NULL ,
  `pin_jendral` VARCHAR(8) NOT NULL ,
  `telp_jendral` VARCHAR(19) NOT NULL ,
  `twitter_jendral` VARCHAR(20) NOT NULL ,
  `kend_jendral` VARCHAR(20) NOT NULL ,
  `plat_jendral` VARCHAR(20) NOT NULL ,
  PRIMARY KEY (`username`) ,
  INDEX `id_kota` (`id_kota` ASC) ,
  CONSTRAINT `jendral_ibfk_3`
    FOREIGN KEY (`username` )
    REFERENCES `appmhcom_maicih`.`pengguna` (`username` )
```

```
  ON UPDATE CASCADE,
 CONSTRAINT `jendral_ibfk_4`
  FOREIGN KEY (`id_kota` )
  REFERENCES `appmhcom_maicih`.`kota` (`id_kota` )
  ON UPDATE CASCADE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = latin1
```

Implementasi dari tabel jenis seperti tabel 4.6 berikut

Tabel 4.6 Implementasi Tabel Jenis

#	Column	Type	Collation	Attributes	Null	Default	Extra
1	**id_jenis**	int(11)			No	*None*	AUTO_INCREMENT
2	**nama_jenis**	varchar(50)	latin1_swedish_ci		No	*None*	

```
CREATE  TABLE IF NOT EXISTS `appmhcom_maicih`.`jenis` (
 `id_jenis` INT(11) NOT NULL AUTO_INCREMENT ,
 `nama_jenis` VARCHAR(50) NOT NULL ,
 PRIMARY KEY (`id_jenis`) )
ENGINE = InnoDB
DEFAULT CHARACTER SET = latin1;
```

Implementasi dari tabel komentar seperti tabel 4.7 berikut

Tabel 4.7 Implementasi Tabel komentar

#	Column	Type	Collation	Attributes	Null	Default	Extra
1	**id_komentar**	int(11)			No	*None*	AUTO_INCREMENT
2	**tanggal_komentar**	datetime			No	*None*	
3	**username**	varchar(52)	latin1_swedish_ci		No	*None*	
4	**isi_komentar**	text	latin1_swedish_ci		No	*None*	

```
CREATE  TABLE IF NOT EXISTS `appmhcom_maicih`.`komentar` (
 `id_komentar` INT(11) NOT NULL AUTO_INCREMENT ,
 `tanggal_komentar` DATETIME NOT NULL ,
 `username` VARCHAR(52) NOT NULL ,
 `isi_komentar` TEXT NOT NULL ,
 PRIMARY KEY (`id_komentar`) ,
 INDEX `username` (`username` ASC) ,
 CONSTRAINT `komentar_ibfk_1`
  FOREIGN KEY (`username` )
  REFERENCES `appmhcom_maicih`.`pelanggan` (`username` )
  ON UPDATE CASCADE)
```

```
DEFAULT CHARACTER SET = latin1;
```

Implementasi dari tabel kota seperti tabel 4.8 berikut

Tabel 4.8 Implementasi Tabel Kota

#	Column	Type	Collation	Attributes	Null	Default	Extra
1	**id_kota**	int(10)			No	*None*	AUTO_INCREMENT
2	**id_provinsi**	int(10)			Yes	*NULL*	
3	**nama_kota**	varchar(50)	latin1_swedish_ci		Yes	*NULL*	
4	**kabkota**	varchar(20)	latin1_swedish_ci		Yes	*NULL*	

```
CREATE  TABLE IF NOT EXISTS `appmhcom_maicih`.`kota` (
  `id_kota` INT(10) NOT NULL AUTO_INCREMENT ,
  `id_provinsi` INT(10) NULL DEFAULT NULL ,
  `nama_kota` VARCHAR(50) NULL DEFAULT NULL ,
  `kabkota` VARCHAR(20) NULL DEFAULT NULL ,
  PRIMARY KEY (`id_kota`) ,
  INDEX `id_provinsi` (`id_provinsi` ASC) ,
  CONSTRAINT `kota_ibfk_1`
    FOREIGN KEY (`id_provinsi` )
    REFERENCES `appmhcom_maicih`.`provinsi` (`id_provinsi` )
    ON DELETE RESTRICT
    ON UPDATE CASCADE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = latin1;
```

Implementasi dari tabel level seperti tabel 4.9 berikut

Tabel 4.9 Implementasi Tabel Level

#	Column	Type	Collation	Attributes	Null	Default	Extra
1	**id_level**	int(11)			No	*None*	AUTO_INCREMENT
2	**nama_level**	varchar(30)	latin1_swedish_ci		Yes	*NULL*	

```
CREATE  TABLE IF NOT EXISTS `appmhcom_maicih`.`level` (
  `id_level` INT(11) NOT NULL AUTO_INCREMENT ,
  `nama_level` VARCHAR(30) NULL DEFAULT NULL ,
  PRIMARY KEY (`id_level`) )
ENGINE = InnoDB
DEFAULT CHARACTER SET = latin1;
```

Implementasi dari tabel pelanggan seperti tabel 4.10 berikut

Tabel 4.10 Implementasi Tabel Pelanggan

#	Column	Type	Collation	Attributes	Null	Default	Extra
1	username	varchar(50)	latin1_swedish_ci		No	None	
2	nama_pelanggan	varchar(50)	latin1_swedish_ci		No	None	
3	alamat_pelanggan	text	latin1_swedish_ci		No	None	
4	id_kota	int(11)			No	None	
5	jkelamin	enum('l', 'p')	latin1_swedish_ci		No	None	
6	telp_pelanggan	varchar(19)	latin1_swedish_ci		No	None	

```
CREATE  TABLE IF NOT EXISTS `appmhcom_maicih`.`pelanggan` (
  `username` VARCHAR(50) NOT NULL ,
  `nama_pelanggan` VARCHAR(50) NOT NULL ,
  `alamat_pelanggan` TEXT NOT NULL ,
  `id_kota` INT(11) NOT NULL ,
  `jkelamin` ENUM('l','p') NOT NULL ,
  `telp_pelanggan` VARCHAR(19) NOT NULL ,
  PRIMARY KEY (`username`) ,
  INDEX `id_kota` (`id_kota` ASC) ,
  CONSTRAINT `pelanggan_ibfk_1`
    FOREIGN KEY (`username` )
    REFERENCES `appmhcom_maicih`.`pengguna` (`username` )
    ON UPDATE CASCADE,
  CONSTRAINT `pelanggan_ibfk_2`
    FOREIGN KEY (`id_kota` )
    REFERENCES `appmhcom_maicih`.`kota` (`id_kota` )
    ON UPDATE CASCADE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = latin1
COMMENT = 'Tabel Member';
```

Implementasi dari tabel pengguna seperti tabel 4.11 berikut

Tabel 4.11 Implementasi Tabel Pengguna

#	Column	Type	Collation	Attributes	Null	Default	Extra
1	username	varchar(50)	latin1_swedish_ci		No	None	
2	password	varchar(50)	latin1_swedish_ci		No	None	
3	email	varchar(50)	latin1_swedish_ci		No	None	
4	kode_aktivasi	varchar(50)	latin1_swedish_ci		No	None	
5	status	enum('1', '0')	latin1_swedish_ci		No	None	

```
CREATE  TABLE IF NOT EXISTS `appmhcom_maicih`.`pengguna` (
  `username` VARCHAR(50) NOT NULL ,
  `password` VARCHAR(50) NOT NULL ,
  `email` VARCHAR(50) NOT NULL ,
  `kode_aktivasi` VARCHAR(50) NOT NULL ,
  `status` ENUM('1','0') NOT NULL ,
  PRIMARY KEY (`username`) )
ENGINE = InnoDB
DEFAULT CHARACTER SET = latin1;
```

Implementasi dari tabel penjualan seperti tabel 4.12 berikut

Tabel 4.12 Implementasi Tabel Penjualan

#	Column	Type	Collation	Attributes	Null	Default	Extra
1	**id_penjualan**	int(11)			No	0	
2	**username**	varchar(50)	latin1_swedish_ci		No	*None*	
3	**tanggal_penjualan**	datetime			No	*None*	

```
CREATE  TABLE IF NOT EXISTS `appmhcom_maicih`.`penjualan` (
  `id_penjualan` INT(11) NOT NULL DEFAULT '0' ,
  `username` VARCHAR(50) NOT NULL ,
  `tanggal_penjualan` DATETIME NOT NULL ,
  PRIMARY KEY (`id_penjualan`) ,
  INDEX `username` (`username` ASC) ,
  CONSTRAINT `penjualan_ibfk_1`
    FOREIGN KEY (`username` )
    REFERENCES `appmhcom_maicih`.`jendral` (`username` )
    ON UPDATE CASCADE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = latin1;
```

Implementasi dari tabel posisi seperti tabel 4.13 berikut

Tabel 4.13 Implementasi Tabel Posisi

#	Column	Type	Collation	Attributes	Null	Default	Extra
1	**id_posisi**	int(11)			No	*None*	AUTO_INCREMENT
2	**username**	varchar(52)	latin1_swedish_ci		No	*None*	
3	**longitude**	double			No	*None*	
4	**latitude**	double			No	*None*	
5	**mulai**	datetime			No	*None*	
6	**berakhir**	datetime			No	*None*	

```
CREATE  TABLE IF NOT EXISTS `appmhcom_maicih`.`posisi` (
  `id_posisi` INT(11) NOT NULL AUTO_INCREMENT ,
  `username` VARCHAR(52) NOT NULL ,
  `longitude` DOUBLE NOT NULL ,
  `latitude` DOUBLE NOT NULL ,
  `mulai` DATETIME NOT NULL ,
  `berakhir` DATETIME NOT NULL ,
  PRIMARY KEY (`id_posisi`) ,
  INDEX `username` (`username` ASC) ,
  CONSTRAINT `posisi_ibfk_1`
    FOREIGN KEY (`username` )
    REFERENCES `appmhcom_maicih`.`pengguna` (`username` )
    ON UPDATE CASCADE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = latin1;
```

Implementasi dari tabel produk seperti tabel 4.14 berikut

Tabel 4.14 Implementasi Tabel Produk

#	Column	Type	Collation	Attributes	Null	Default	Extra
1	id_produk	int(11)			No	None	AUTO_INCREMENT
2	id_jenis	int(11)			No	None	
3	id_level	int(11)			No	None	
4	deskripsi_produk	text	latin1_swedish_ci		No	None	
5	harga	double			No	None	

```
CREATE  TABLE IF NOT EXISTS `appmhcom_maicih`.`produk` (
  `id_produk` INT(11) NOT NULL AUTO_INCREMENT ,
  `id_jenis` INT(11) NOT NULL ,
  `id_level` INT(11) NOT NULL ,
  `deskripsi_produk` TEXT NOT NULL ,
  `harga` DOUBLE NOT NULL ,
  PRIMARY KEY (`id_produk`) ,
  INDEX `id_jenis` (`id_jenis` ASC) ,
  INDEX `id_level` (`id_level` ASC) ,
  CONSTRAINT `produk_ibfk_5`
    FOREIGN KEY (`id_jenis` )
    REFERENCES `appmhcom_maicih`.`jenis` (`id_jenis` )
    ON UPDATE CASCADE,
  CONSTRAINT `produk_ibfk_6`
    FOREIGN KEY (`id_level` )
    REFERENCES `appmhcom_maicih`.`level` (`id_level` )
    ON UPDATE CASCADE)
```

```
ENGINE = InnoDB
DEFAULT CHARACTER SET = latin1;
```

Implementasi dari tabel provinsi seperti tabel 4.15 berikut

Tabel 4.15 Tabel Implentasi Tabel Provinsi

#	Column	Type	Collation	Attributes	Null	Default	Extra
1	**id_provinsi**	int(10)			No	*None*	AUTO_INCREMENT
2	**nama_provinsi**	varchar(30)	latin1_swedish_ci		Yes	*NULL*	

```
CREATE  TABLE IF NOT EXISTS `appmhcom_maicih`.`provinsi` (
  `id_provinsi` INT(10) NOT NULL AUTO_INCREMENT ,
  `nama_provinsi` VARCHAR(30) NULL DEFAULT NULL ,
  PRIMARY KEY (`id_provinsi`) )
ENGINE = InnoDB
DEFAULT CHARACTER SET = latin1;
```

Implementasi dari tabel bank seperti tabel 4.16 berikut

Tabel 4.16 Tabel Implentasi Tabel Bank

	#	Column	Type	Collation	Attributes	Null	Default	Extra
☐	1	**id_bank**	int(11)			No	*None*	AUTO_INCREMENT
☐	2	**nama_bank**	varchar(25)	latin1_swedish_ci		No	*None*	

```
) CREATE  TABLE IF NOT EXISTS `appmhcom_maicih`.`bank` (
  `id_bank` INT(11) NOT NULL AUTO_INCREMENT ,
  `nama_bank` VARCHAR(25) NOT NULL ,
  PRIMARY KEY (`id_bank`) )
ENGINE = InnoDB
DEFAULT CHARACTER SET = latin1;
```

Implementasi dari tabel pembayaran seperti tabel 4.17 berikut

Tabel 4.17 Tabel Implentasi Tabel pembayaran

#	Column	Type	Collation	Attributes	Null	Default	Extra
1	id_penjualan	int(11)			No	None	
2	status_bayar	enum('0', '1', '2')	latin1_swedish_ci		No	None	
3	jenis_bayar	enum('l', 't')	latin1_swedish_ci		No	None	
4	id_bank	int(11)			No	None	
5	no_rekening	varchar(25)	latin1_swedish_ci		No	None	
6	atas_nama	varchar(25)	latin1_swedish_ci		No	None	
7	total	double			No	None	
8	tgl_bayar	datetime			No	None	
9	keterangan	text	latin1_swedish_ci		No	None	
10	id_admin	int(11)			No	None	

```
CREATE  TABLE IF NOT EXISTS `appmhcom_maicih`.`pembayaran` (
  `id_penjualan` INT(11) NOT NULL ,
  `status_bayar` ENUM('0','1','2') NOT NULL ,
  `jenis_bayar` ENUM('l','t') NOT NULL ,
  `id_bank` INT(11) NOT NULL ,
  `no_rekening` VARCHAR(25) NOT NULL ,
 `atas_nama` VARCHAR(25) NOT NULL ,
  `total` DOUBLE NOT NULL ,
  `tgl_bayar` DATETIME NOT NULL ,
  `keterangan` TEXT NOT NULL ,
  `id_admin` INT(11) NOT NULL ,
  PRIMARY KEY (`id_penjualan`) ,
  INDEX `id_admin` (`id_admin` ASC) ,
  INDEX `id_bank` (`id_bank` ASC) ,
  CONSTRAINT `pembayaran_ibfk_16`
    FOREIGN KEY (`id_admin` )
    REFERENCES `appmhcom_maicih`.`admin` (`id_admin` )
    ON UPDATE CASCADE,
  CONSTRAINT `pembayaran_ibfk_14`
    FOREIGN KEY (`id_penjualan` )
    REFERENCES `appmhcom_maicih`.`penjualan` (`id_penjualan` )
    ON UPDATE CASCADE,
  CONSTRAINT `pembayaran_ibfk_15`
    FOREIGN KEY (`id_bank` )
    REFERENCES `appmhcom_maicih`.`bank` (`id_bank` )
    ON UPDATE CASCADE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = latin1;
```

Implementasi dari tabel detailpembayaran seperti tabel 4.18 berikut

Tabel 4.18 Tabel Implentasi Tabel DetailPembayaran

#	Column	Type	Collation	Attributes	Null	Default	Extra
1	id_detailpembayaran	int(11)			No	None	AUTO_INCREMENT
2	id_penjualan	int(11)			No	None	
3	id_produk	int(11)			No	None	
4	stok	int(11)			No	None	

```
CREATE  TABLE IF NOT EXISTS `appmhcom_maicih`.`detailpembayaran` (
  `id_detailpembayaran` INT(11) NOT NULL AUTO_INCREMENT ,
  `id_penjualan` INT(11) NOT NULL ,
  `id_produk` INT(11) NOT NULL ,
  `stok` INT(11) NOT NULL ,
  PRIMARY KEY (`id_detailpembayaran`) ,
  INDEX `id_produk` (`id_produk` ASC) ,
  INDEX `detailpembayaran_ibfk_2` (`id_penjualan` ASC) ,
  CONSTRAINT `detailpembayaran_ibfk_1`
    FOREIGN KEY (`id_produk` )
    REFERENCES `appmhcom_maicih`.`produk` (`id_produk` )
    ON DELETE RESTRICT
    ON UPDATE CASCADE,
  CONSTRAINT `detailpembayaran_ibfk_2`
    FOREIGN KEY (`id_penjualan` )
    REFERENCES `appmhcom_maicih`.`pembayaran` (`id_penjualan` )
    ON DELETE RESTRICT
    ON UPDATE CASCADE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = latin1;
```

4.2 Implementasi Antar Muka

Implementasi antar muka dilakukan dengan setiap tampilan program yang dibangun dan pengkodeannya dalam bentuk file program. Berikut ini adalah implementasi antarmuka untuk pengunjung dan pelanggan, jendral, dan administrator. Adapun untuk tampilan implementasi pembangunan aplikasi

tracking berbasis *mobile* dan *web* di PT Maicih Inti Synergi akan dilampirkan pada halaman lampiran.

Tabel 4.19 Implementasi Antar Muka Pengunjung dan Pelanggan

No.	Menu	Deskripsi	Nama *File*
1	Login Pelanggan	Digunakan untuk login pelanggan	Login.java
2	Menu Pelanggan	Digunakan sebagai halaman pelanggan	HalamanUtama.java
4	Update Posisi	Digunakan pelanggan untuk mengupdate lokasi terbaru	UpdatePosisi.java
5	Lihat Posisi Jendral	Digunakan pelanggan untuk melihat posisi lokasi Jendral	PosisiMapJendral.java
6	Tambah Data Komentar	Digunakan untuk menambah data komentar pelanggan	Komentar.java
7	Tampil saran Lokasi	Digunakan untuk melihat permintaan lokasi pelanggan	PosisimapPelanggan.java
8	Akun Saya	Digunakan untuk melihat, mengubah profile pelanggan	AkunSaya.java
9	Ubah Password	Digunakan untuk mengubah password	UbahPass.java
10	Registrasi	Digunakan untuk mendaftarkan sebagai pelanggan	Registrasi.java

Tabel 4.20 Implementasi Antar Muka Jendral

No.	Menu	Deskripsi	Nama *File*
1	Login Jendral	Digunakan untuk login Jendral	Login.java
2	Menu Jendral	Digunakan sebagai halaman Jendral	HalamanUtama.java
3	Update Posisi	Digunakan Jendral untuk mengupdate lokasi terbaru	PilihLokasi.java
4	Lihat Posisi Jendral	Digunakan pelanggan untuk melihat posisi lokasi Jendral	PosisiMapJendral..java
5	Permintaan Lokasi	Digunakan untuk melihat saran lokasi pelanggan	PosisiMapPelalanggan..java

6	Akun Saya	Digunakan untuk melihat, mengubah profile Jendral	AkunSaya.java
7	Ubah Password	Digunakan untuk mengubah password	UbahPass.java

Tabel 4.21 Implementasi Antar Muka Admin

No.	Menu	Deskripsi	Nama *File*
1.	Login Admin	Digunakan sebagai halaman login admin	login.php
2.	Menu Admin	Digunakan sebagai halaman admin	home.php
3.	Pengolahan Data Detail Penjualan	Digunakan untuk melihat, menambah dan merubah data penjualan	detailpenjualan.php
4.	Pengolahan Data Jendral	Digunakan untuk melihat, menambah dan merubah data Jendral	Jendral.php
5.	Pengolahan Data Jenis	Digunakan untuk melihat, menambah dan merubah data jenis	jenis.php
6.	Pengolahan Data Kategori	Digunakan untuk melihat, menambah dan merubah data kategori	kategori.php
7.	Pengolahan Komentar	Digunakan untuk melihat, menambah dan merubah data komentar	komentar.php
8.	Pengolahan Data Kota	Digunakan untuk melihat, menambah dan merubah data kota	kota.php
9.	Pengolahan Data Level	Digunakan untuk melihat, menambah dan merubah data level	level.php
10.	Logout Admin	Digunakan admin untuk keluar dari halaman admin	logout.php
11	Pengolahan Data Pelanggan	Digunakan untuk melihat data pelanggan	pelanggan.php
12	Pengolahan Data Penjualan	Digunakan untuk melihat, dan merubah data penjualan	penjualan.php
13	Pengolahan Data Produk	Digunakan untuk melihat, menambah dan merubah data produk	produk.php
14	Pengolahan Data Provinsi	Digunakan untuk melihat, menambah dan merubah data provinsi	provinsi.php
15	Pengolahan Data	Digunakan untuk melihat,	tambahproduk.php

	Produk	menambah	
16	Ubah Password	Digunakan untuk mengubah data password	ubahpassword.php
17	Ubah Profile	Digunakan untuk mengubah data profile admin	ubahprofile.php

4.3 Pengujian Perangkat Lunak

Pengujian sistem merupakan hal terpenting yang bertujuan untuk menemukan kesalahan-kesalahan dan kekurangan-kekurangan pada perangkat lunak yang diuji. Pengujian bermaksud untuk mengetahui perangkat lunak yang dibuat sudah memenuhi kreteria yang sesuai dengan tujuan perancangan perangkat lunak tersebut. Pengujian perangkat lunak ini menggunakan pengujian *black box*. Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak.

4.4 Pengujian Alpa

Pengujian alpa dilakukan menggunakan metode *black box*. Untuk menentukan pengujian alpa ini dibagi menjadi tiga, yaitu skenario pengujian halaman pelanggan, skenario pengujian halaman Jendral dan skenario pengujian halaman administrator.

4.4.1 Skenario Pengujian Alpa

Untuk penjelasan lebih lanjut tentang skenario pengujian halaman pelanggan maka dapat dilihat pada Tabel 4.22 sebagai berikut :

Tabel 4.22 Pengujian Halaman Pelanggan

Kelas Uji	Butir Uji	Jenis Pengujian
Login Pelanggan	Isi data login	*Black Box*
	Verifikasi e-mail	*Black Box*

	Verifikasi Password	*Black Box*
Saran Lokasi	Ketersediaan GPS	*Black Box*
	Ketersediaan koneksi internet	*Black Box*
	Menampilkan Lokasi Terbaru	*Black Box*
Posisi Jendral	Menampilkan lokasi Jendral	*Black Box*
Saran Pelanggan	Menampilkan lokasi saran pelanggan	*Black Box*
Data Pelanggan	Menampilkan Data Jendral	*Black Box*
	Ubah data Jendral	*Black Box*
Ubah password	Menampilkan form ganti password	*Black Box*
	Isi form ganti password	*Black Box*
	Simpan data baru	*Black Box*
Keluar	Menghapus session Jendral	*Black Box*

Untuk penjelasan lebih lanjut tentang skenario pengujian halaman Jendral maka dapat dilihat pada Tabel 4.23 sebagai berikut :

Tabel 4.23 Pengujian Halaman Jendral

Kelas Uji	Butir Uji	Jenis Pengujian
Login Jendral	Isi data login	*Black Box*
	Verifikasi e-mail	*Black Box*
	Verifikasi Password	*Black Box*
Update Lokasi	Ketesediaan GPS	*Black Box*
	Ketersediaan koneksi internet	*Black Box*
	Menampilkan Lokasi Terbaru	*Black Box*
Posisi Jendral	Menampilkan lokasi Jendral	*Black Box*
Saran Pelanggan	Menampilkan lokasi Saran Pelanggan	*Black Box*
Data Jendral	Menampilkan Data Jendral	*Black Box*
	Ubah data Jendral	*Black Box*
Ubah password	Menampilkan form ganti password	*Black Box*
	Isi form ganti password	*Black Box*
	Simpan data baru	*Black Box*
Keluar	Menghapus session Jendral	*Black Box*

Untuk penjelasan lebih lanjut tentang skenario pengujian halaman administrator maka dapat dilihat pada Tabel 4.24 sebagai berikut :

Tabel 4.24 Pengujian Halaman Administrator

Kelas Uji	Butir Uji	Jenis Pengujian
Lupa password	Menampilkan form lupa password	*Black Box*
Login	Isi data login	*Black Box*
	Verifikasi e-mail	*Black Box*
	Verifikasi *password*	*Black Box*
Ubah profile	Menampilkan data *profile*	*Black Box*
	Ubah data profile	*Black Box*
Ubah *password*	Menampilkan form ubah *password*	*Black Box*
	Isi form ganti *password*	*Black Box*
	Simpan data baru	*Black Box*
Data Jenis	Menampilkan *form* data jenis	*Black Box*
	Tambah data jenis	*Black Box*
	Ubah data jenis	*Black Box*
	Hapus data jenis	*Black Box*
Data Level	Menampilkan *form* data level	*Black Box*
	Tambah data level	*Black Box*
	Ubah data level	*Black Box*
	Hapus data level	*Black Box*
Data Produk	Menampilkan form data produk	*Black Box*
	Tambah data produk	*Black Box*
	Ubah data produk	*Black Box*
	Hapus data produk	*Black Box*
Data Provinsi	Menampilkan form data provinsi	*Black Box*
	Tambah data provinsi	*Black Box*
	Ubah data provinsi	*Black Box*
	Hapus data provinsi	*Black Box*
Data Kota	Menampilkan form data kota	*Black Box*
	Tambah data kota	*Black Box*
	Ubah data kota	*Black Box*
	Hapus data kota	*Black Box*
Data Jendral	Menampilkan form data Jendral	*Black Box*
	Cari data Jendral	*Black Box*
	Ubah data Jendral	*Black Box*
	Hapus data Jendral	*Black Box*
Data Pelanggan	Menampilkan form data pelanggan	*Black Box*

	Cari data pelanggan	*Black Box*
	Ubah data pelanggan	*Black Box*
	Hapus data pelanggan	*Black Box*
Data Penjualan	Menampilkan form data penjualan	*Black Box*
	Tambah data penjualan	*Black Box*
	Menampilkan detail pelanggan	*Black Box*
	Hapus data penjualan	*Black Box*
Data Komentar	Menampilkan form data pelanggan	*Black Box*
	Cari data pelanggan	*Black Box*
	Ubah data pelanggan	*Black Box*
	Hapus data pelanggan	*Black Box*
Logout	Menghapus *session* admin	*Black Box*

4.4.2 Kasus Dan Hasil Pengujian

Berdasarkan rencana pengujian yang disusun, maka dilakukan pengujian sebagai berikut:

4.4.2.1 Pengujian Login Pelanggan

Berikut dapat dilihat pengujian dari *Login* Pelanggan pada tabel 4.25 di bawah ini :

Tabel 4.25 Pengujian Login Pelanggan

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Username : pel1	Tercantum pada text box *username*	Dapat mengisi field *username* dengan benar	[✓] Diterima [ ] Ditolak
Password : ada	Tercantum pada text box *password*	Dapat mengisi field *password* dengan benar	[✓] Diterima [ ] Ditolak
Klik tombol login	jika sama dengan data login maka admin masuk ke menu utama admin	Tombol login dapat berfungsi	[✓] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Data *username* atau *password* tidak terdaftar	Tidak dapat login	Muncul pesan "maaf, username dan password anda salah."	[✓] Diterima [ ] Ditolak
Data *username* atau *password* kosong	Tidak dapat login	Muncul pesan kesalahan "account tidak ditemukan."	[✓] Diterima [ ] Ditolak
Username tidak diisi	Menampilkan pesan kesalahan "username harus diisi"	Dapat menampilkan pesan kesalahan username harus diisi"	[✓] Diterima [ ] Ditolak
Password tidak diisi	Menampilkan pesan kesalahan "*password* harus diisi"	Dapat menampilkan pesan kesalahan "*password* harus diisi"	[✓] Diterima [ ] Ditolak

4.4.2.2 Pengujian Registrasi Pelanggan

Berikut dapat dilihat pengujian dari registrasi pelanggan pada tabel 4.26 dibawah ini :

Tabel 4.26 Pengujian Registrasi Pelanggan

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik Registrasi	Menampilkan *form* registrasi	Dapat menampilkan *form* registrasi	[✓] Diterima [ ] Ditolak
Mengisi data registrasi Contoh: Nama : refan Alamat : sekeloa utara Kota : Jawa Barat, Bandung Jenis Kelamin ; Laki-laki No Telp: 0856000000	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[✓] Diterima [ ] Ditolak

Email : peys032@yahoo.com Konfirmasi email : peys032@yahoo.com			
Klik Selanjutnya	Menampilkan *form* registrasi	Dapat menampilkan *form* registrasi	[✓] Diterima [ ] Ditolak
Melanjutkan mengisi data registrasi Contoh: Username : peys032 Password : coba Konfirmasi Password : coba	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[✓] Diterima [ ] Ditolak
Klik tombol registrasi	Data registrasi akan tersimpan di dalam *database*	Data registrasi akan tersimpan di dalam *database* dan muncul pesan ”Terima kasih silahkan login kembali”	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Data Nama kosong	Tampilkan pesan kesalahan “nama harus diisi”	Dapat menampilkan pesan kesalahan ” nama harus diisi”	[✓] Diterima [ ] Ditolak
Data Alamat kosong	Tampilkan pesan kesalahan “alamat harus diisi”	Dapat menampilkan pesan kesalahan ” alamat harus diisi”	[✓] Diterima [ ] Ditolak
Data Kota Kosong	Tampilkan pesan kesalahan “kota harus diisi”	Dapat menampilkan pesan kesalahan ” kota harus diisi”	[✓] Diterima [ ] Ditolak
Data Jenis Kelamin Kosong	Tampilkan pesan kesalahan “jenis kelamin harus diisi”	Dapat menampilkan pesan kesalahan ” jenis kelamin	[✓] Diterima [ ] Ditolak

		harus diisi”	
Data No Telp. Kosong	Tampilkan pesan kesalahan “no telp harus diisi”	Dapat menampilkan pesan kesalahan ” no telp harus diisi”	[✓] Diterima [ ] Ditolak
Data Email Kosong	Tampilkan pesan kesalahan “email harus diisi”	Dapat menampilkan pesan kesalahan ” email harus diisi”	[✓] Diterima [ ] Ditolak
Data Konfirmasi Email Kosong	Tampilkan pesan kesalahan “konfirmasi email harus sama”	Dapat menampilkan pesan kesalahan ” konfirmasi email harus sama”	[✓] Diterima [ ] Ditolak
Username tidak diisi	Tampilkan pesan kesalahan “username harus diisi”	Dapat menampilkan pesan kesalahan ” username harus diisi”	[✓] Diterima [ ] Ditolak
Username sudah ada	Tampilkan pesan kesalahan “username telah digunakan”	Dapat menampilkan pesan kesalahan ” nama harus diisi”	[✓] Diterima [ ] Ditolak
Password tidak diisi	Tampilkan pesan kesalahan “password harus diisi”	Dapat menampilkan pesan kesalahan ” nama harus diisi”	[✓] Diterima [ ] Ditolak
Konfirmasi Password tidak diisi	Tampilkan pesan kesalahan “email harus sesuai”	Dapat menampilkan pesan kesalahan ” email harus harus sesuai”	[✓] Diterima [ ] Ditolak

4.4.2.3 Pengujian Saran Lokasi Pelanggan

Berikut dapat dilihat pengujian dari Saran Lokasi Pelanggan pada tabel 4.27 di bawah ini :

Tabel 4.27 Pengujian Saran Lokasi Pelanggan

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik Saran Lokasi	Menampilkan Map saran lokasi pelanggan	Dapat menampilkan map saran lokasi pelanggan	[✓] Diterima [ ] Ditolak
Klik Ubah Komentar	Menampilkan form ubah komentar	Dapat menampilkan ubah komentar	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik Saran Posisi (Tidak ada koneksi internet)	Tidak menampilkan saran posisi Jendral	Tidak menampilkan saran posisi Jendral	[✓] Diterima [ ] Ditolak

4.4.2.4 Pengujian Posisi Jendral

Berikut dapat dilihat pengujian dari Posisi Pelanggan pada tabel 4.28 di bawah ini :

Tabel 4.28 Pengujian Posisi Jendral

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik Posisi Jendral	Menampilkan Map Posisi Jendral	Dapat menampilkan map posisi Jendral	[✓] Diterima [ ] Ditolak
Klik *My Location*	Menampilkan Posisi *Latitude* dan *Longitude*	Dapat menampilkanPosisi *Latitude* dan *Longitude*	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik Posisi Jendral (Tidak ada koneksi internet)	Tidak menampilkan map posisi Jendral	Tidak menampilkan map posisi Jendral	[✓] Diterima [ ] Ditolak

4.4.2.5 Pengujian Tampil Saran Lokasi Pelanggan

Berikut dapat dilihat pengujian dari Permintaan Pelanggan pada tabel 4.29 di bawah ini :

Tabel 4.29 Pengujian Tampil Saran Lokasi

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik Saran Lokasi	Menampilkan Map saran lokasi pelanggan	Dapat menampilkan map saran lokasi pelanggan	[✓] Diterima [ ] Ditolak
Klik Icon Pelanggan	Menampilkan pesan pelanggan	Dapat menampilkan pesan pelanggan	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik Saran Posisi (Tidak ada koneksi internet)	Tidak menampilkan saran posisi Jendral	Tidak menampilkan saran posisi Jendral	[✓] Diterima [ ] Ditolak

4.4.2.6 Pengujian Ubah Data Pelanggan

Berikut dapat dilihat pengujian dari Ubah Data Pelanggan pada tabel 4.30 di bawah ini :

Tabel 4.30 Pengujian Ubah Data Pelanggan

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama*:* refan	Tercantum pada text box *nama*	Dapat mengisi / mengubah field nama dengan benar	[✓] Diterima [ ] Ditolak
No Telp : 0856000000	Tercantum pada text box *no telp*	Dapat mengisi / mengubah field no telp dengan benar	[✓] Diterima [ ] Ditolak
Klik tombol simpan	Data dapat tersimpan ke database	Tombol simpan dapat berfungsi	[✓] Diterima [ ] Ditolak
Klik tombol kembali	Menampilkan ke	Dapat	[✓] Diterima

	halaman sebelumnya	menampilkan halaman sebelumnya	[ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Data nama dan No Telp kosong	Tidak dapat disimpan	Muncul pesan kesalahan "nama dan no telp belum diisi."	[✓] Diterima [ ] Ditolak
Nama tidak diisi	Menampilkan pesan kesalahan "Nama harus diisi"	Dapat menampilkan pesan kesalahan "Nama harus diisi"	[✓] Diterima [ ] Ditolak
No Telp tidak diisi	Menampilkan pesan kesalahan "No Telp harus diisi"	Dapat menampilkan pesan kesalahan "No Telp harus diisi"	[✓] Diterima [ ] Ditolak

4.4.2.7 Pengujian Ubah Password Pelanggan

Berikut dapat dilihat pengujian dari Ubah *Password* Pelanggan pada tabel 4.31 di bawah ini :

Tabel 4.31 Pengujian Ubah Password Pelanggan

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik menu ubah *password*	Menampilkan *form* ubah *password*	Dapat menampilkan *form* ubah *password*	[✓] Diterima [ ] Ditolak
Memasukan data yang benar	*Password* baru berhasil disimpan di *database*	Dapat menyimpan data *password* yang baru dengan muncul pesan " Data berhasil diubah"	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengosongkan	Muncul pesan	Dapat	[✓] Diterima

salah satu *field*	kesalahan *field* tersebut harus diisi	menampilkan pesan kesalahan ”*field* ini harus diisi” bila ada field yang belum diisi	[ ] Ditolak
Memasukan *password* yang salah	muncul pesan "password salah"	Dapat menampilkan pesan kesalahan ”*Password* salah”	[✓] Diterima [ ] Ditolak
Memasukan data *password* baru dan konfirmasi *password* baru yang tidak sama	Muncul pesan "password baru dan konfirmasi password harus sama"	Dapat menampilkan pesan "password baru dan konfirmasi password harus sama"	[✓] Diterima [ ] Ditolak
Mengosongkan semua *field*	Muncul pesan kesalahan *field* harus diisi	Dapat menampilkan pesan kesalahan *field* harus diisi	[✓] Diterima [ ] Ditolak

4.4.2.8 Pengujian Login Jendral

Berikut dapat dilihat pengujian dari *Login* Jendral pada tabel 4.32 di bawah ini :

Tabel 4.32 Pengujian Login Jendral

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Username : User5	Tercantum pada text box *username*	Dapat mengisi field *username* dengan benar	[✓] Diterima [ ] Ditolak
Password : ada	Tercantum pada text box *password*	Dapat mengisi field *password* dengan benar	[✓] Diterima [ ] Ditolak
Klik tombol login	jika sama dengan data login maka admin masuk ke menu utama admin	Tombol login dapat berfungsi	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			

Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Data *username* atau *password* tidak terdaftar	Tidak dapat login	Muncul pesan "maaf account tidak ditemukan."	[✓] Diterima [ ] Ditolak
Data *email* atau *password* kosong	Tidak dapat login	Muncul pesan kesalahan "account tidak ditemukan."	[✓] Diterima [ ] Ditolak
Email tidak diisi	Menampilkan pesan kesalahan "email harus diisi"	Dapat menampilkan pesan kesalahan email harus diisi"	[✓] Diterima [ ] Ditolak
Password tidak diisi	Menampilkan pesan kesalahan "*password* harus diisi"	Dapat menampilkan pesan kesalahan "*password* harus diisi"	[✓] Diterima [ ] Ditolak

4.4.2.9 Pengujian Update Lokasi

Berikut dapat dilihat pengujian dari *Update* Lokasi Jendral pada tabel 4.33 di bawah ini :

Tabel 4.33 Pengujian Update Lokasi

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik Update Lokasi	Menampilkan Map Posisi Jendral Terbaru	Dapat menampilkan map posisi Jendral terbaru	[✓] Diterima [ ] Ditolak
Klik Set Lama Gentayangan	Menampilkan set waktu	Dapat menampilkan set waktu	[✓] Diterima [ ] Ditolak
Klik set waktu	Menampilkan *range* angka jam dan menit	Dapat menampilkan *range* angka jam dan menit	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik Update Lokasi (Tidak ada koneksi internet)	Tidak dapat update lokasi	Tidak dapat update lokasi	[✓] Diterima [ ] Ditolak

4.4.2.10 Pengujian Posisi Jendral

Berikut dapat dilihat pengujian dari Posisi Jendral pada tabel 4.34 di bawah ini :

Tabel 4.34 Pengujian Posisi Jendral

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik Posisi Jendral	Menampilkan Map Posisi Jendral	Dapat menampilkan map posisi Jendral	[✓] Diterima [ ] Ditolak
Klik *My Location*	Menampilkan Posisi *Latitude* dan *Longitude*	Dapat menampilkanPosisi *Latitude* dan *Longitude*	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik Posisi Jendral (Tidak ada koneksi internet)	Tidak menampilkan map posisi Jendral	Tidak menampilkan map posisi Jendral	[✓] Diterima [ ] Ditolak

4.4.2.11 Pengujian Tampil Saran Lokasi Jendral

Berikut dapat dilihat pengujian dari Pengujian Tampil Saran Lokasi Jendral pada tabel 4.35 di bawah ini :

Tabel 4.35 Pengjian Tampil Saran Lokasi Jendral

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik Saran Lokasi	Menampilkan Map saran lokasi pelanggan	Dapat menampilkan map saran lokasi pelanggan	[✓] Diterima [ ] Ditolak
Klik Icon Pelanggan	Menampilkan komentar pelanggan	Dapat menampilkan komentar	[✓] Diterima [ ] Ditolak

		pelanggan	
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik Saran Posisi (Tidak ada koneksi internet)	Tidak menampilkan saran posisi Jendral	Tidak menampilkan saran posisi Jendral	[✓] Diterima [ ] Ditolak

4.4.2.12 Pengujian Ubah Data Jendral

Berikut dapat dilihat pengujian dari Ubah Data Jendral pada tabel 4.36 di bawah ini :

Tabel 4.36 Pengujian Ubah Data Jendral

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama Jendral : User5	Tercantum pada text box Nama Jendral	Dapat mengisi/mengubah field Nama Jendral dengan benar	[✓] Diterima [ ] Ditolak
Twitter : @meals03	Tercantum pada text box *Twitter*	Dapat mengisi/mengubah field twitter dengan benar	[✓] Diterima [ ] Ditolak
Pin Jendral : 2d34567	Tercantum pada text box pin Jendral	Dapat mengisi/mengubah field pin Jendral dengan benar	[✓] Diterima [ ] Ditolak
Telepon Jendral : 1234567	Tercantum pada text box Telepon Jendral	Dapat mengisi/mengubah field telepon Jendral dengan benar	[✓] Diterima [ ] Ditolak
Merk/tipe kendaraan Jendral : Vios Silver	Tercantum pada text box Merk/Tipe Kendaraan	Dapat mengisi/mengubah field merk/tipe kendaraan Jendral dengan benar	[✓] Diterima [ ] Ditolak
Plat No Kendaraan : D 1234 FF	Tercantum pada text box Plat No Kendaraan	Dapat mengisi/mengubah field Plat No Kendaraan dengan	[✓] Diterima [ ] Ditolak

		benar	
Klik tombol simpan	Data dapat tersimpan ke database	Tombol simpan dapat berfungsi	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama Jendral Kosong	Menampilkan pesan kesalahan "Nama Jendral harus diisi"	Dapat menampilkan pesan kesalahan "Nama Jendral harus diisi"	[✓] Diterima [ ] Ditolak
Twitter Kosong	Menampilkan pesan kesalahan "Twitter harus diisi"	Dapat menampilkan pesan kesalahan "Twitter harus diisi"	[✓] Diterima [ ] Ditolak
Pin Jendral Kosong	Menampilkan pesan kesalahan "Pin Jendral harus diisi"	Dapat menampilkan pesan kesalahan "Pin Jendral harus diisi"	[✓] Diterima [ ] Ditolak
Telepon Jendral Kosong	Menampilkan pesan kesalahan "Telepon Jendral harus diisi"	Dapat menampilkan pesan kesalahan "Telepon Jendral harus diisi"	[✓] Diterima [ ] Ditolak
Merk/tipe kendaraan Jendral kosong	Menampilkan pesan kesalahan "Merk/tipe kendaraan Jendral harus diisi"	Dapat menampilkan pesan kesalahan "Merk/tipe kendaraan Jendral harus diisi"	[✓] Diterima [ ] Ditolak
Plat No Kendaraan Kosong	Menampilkan pesan kesalahan "No Kendaraan harus diisi"	Dapat menampilkan pesan kesalahan "Plat No Kendaraan harus diisi"	[✓] Diterima [ ] Ditolak

4.4.2.13 Pengujian Ubah Password Jendral

Berikut dapat dilihat pengujian dari *Password* Jendral pada tabel 4.37 di bawah ini :

Tabel 4.37 Pengujian Ubah Password Jendral

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik menu ubah *password*	Menampilkan *form* ubah *password*	Dapat menampilkan *form* ubah *password*	[✓] Diterima [ ] Ditolak
Memasukan data yang benar	*Password* baru berhasil disimpan di *database*	Dapat menyimpan data *password* yang baru dengan muncul pesan “ Data berhasil diubah”	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengosongkan salah satu *field*	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat menampilkan pesan kesalahan ”*field* ini harus diisi” bila ada field yang belum diisi	[✓] Diterima [ ] Ditolak
Memasukan *password* yang salah	muncul pesan "password salah"	Dapat menampilkan pesan kesalahan ”*Password* salah”	[✓] Diterima [ ] Ditolak
Memasukan data *password* baru dan konfirmasi *password* baru yang tidak sama	Muncul pesan "password baru dan konfirmasi password harus sama"	Dapat menampilkan pesan "password baru dan konfirmasi password harus sama"	[✓] Diterima [ ] Ditolak
Mengosongkan semua *field*	Muncul pesan kesalahan *field* harus diisi	Dapat menampilkan pesan kesalahan *field* harus diisi	[✓] Diterima [ ] Ditolak

4.4.2.14 Pengujian Login Administrator

Berikut dapat dilihat pengujian dari *Login Administrator* pada tabel 4.38 di bawah ini :

Tabel 4.38 Pengujian Login Administrator

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
E-mail : admin@Maicih.com	Tercantum pada text box *username*	Dapat mengisi field *username* dengan benar	[✓] Diterima [ ] Ditolak
Password : reygan	Tercantum pada text box *password*	Dapat mengisi field *password* dengan benar	[✓] Diterima [ ] Ditolak
Klik tombol login	jika sama dengan data login maka admin masuk ke menu utama admin	Tombol login dapat berfungsi	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Data *email* atau *password* tidak terdaftar	Tidak dapat login	Muncul pesan “maaf account tidak ditemukan.”	[✓] Diterima [ ] Ditolak
Data *email* atau *password* kosong	Tidak dapat login	Muncul pesan kesalahan “account tidak ditemukan.”	[✓] Diterima [ ] Ditolak
Email tidak diisi	Menampilkan pesan kesalahan “email harus diisi”	Dapat menampilkan pesan kesalahan email harus diisi”	[✓] Diterima [ ] Ditolak
Password tidak diisi	Menampilkan pesan kesalahan “*password* harus diisi”	Dapat menampilkan pesan kesalahan “*password* harus diisi”	[✓] Diterima [ ] Ditolak

4.4.2.15 Pengujian Ubah Profil Admin

Berikut dapat dilihat pengujian dari ganti profil *administrator* pada tabel 4.39 di bawah ini

Tabel 4.39 Pengujian Ubah Profil Administrator

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik edit profil Administrator	Menampilkan *form* ubah profil	Dapat mengisi field *username* dengan benar	[✓] Diterima [ ] Ditolak
Data profil yang akan diubah	Tercantum pada text box data yang akan diubah	Dapat mengisi *field* data yang akan diubah dengan benar	[✓] Diterima [ ] Ditolak
Klik tombol ubah	Data berhasil diubah di*database*	Tampil pesan "Data berhasil diubah"	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Data masukan ada yang tidak diisi	Menampilkan pesan kesalahan bahwa field tersebut harus terisi.	Dapat menampilkan pesan kesalahan bahwa field tersebut harus terisi.	[✓] Diterima [ ] Ditolak
Data *emaikl* atau *password* kosong	Tidak dapat login	Muncul pesan kesalahan "account tidak ditemukan."	[✓] Diterima [ ] Ditolak
Email tidak diisi	Muncul pesan kesalahan *email* harus diisi	Muncul pesan kesalahan *email* harus diisi	[✓] Diterima [ ] Ditolak
Nama tidak diisi	Muncul pesan kesalahan nama harus diisi	Muncul pesan kesalahan nama harus diisi	[✓] Diterima [ ] Ditolak
Alamat tidak diisi	Muncul pesan kesalahan alamat harus diisi	Muncul pesan kesalahan alamat harus diisi	[✓] Diterima [ ] Ditolak

4.4.2.16 Pengujian Ubah *Password* Admin

Berikut dapat dilihat pengujian dari ubah *password* administrator pada tabel 4.40 di bawah ini :

Tabel 4.40 Pengujian Ubah Password Administrator

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik menu ubah *password*	Menampilkan *form* ubah *password*	Dapat menampilkan *form* ubah *password*	[✓] Diterima [ ] Ditolak
Memasukan data yang benar	*Password* baru berhasil disimpan di *database*	Dapat menyimpan data *password* yang baru dengan muncul pesan “ Data berhasil diubah”	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengosongkan salah satu *field*	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat menampilkan pesan kesalahan ”*field* ini harus diisi” bila ada field yang belum diisi	[✓] Diterima [ ] Ditolak
Memasukan *password* yang salah	muncul pesan "password salah"	Dapat menampilkan pesan kesalahan ”*Password* salah”	[✓] Diterima [ ] Ditolak
Memasukan data *password* baru dan konfirmasi *password* baru yang tidak sama	Muncul pesan "password baru dan konfirmasi password harus sama"	Dapat menampilkan pesan "password baru dan konfirmasi password harus sama"	[✓] Diterima [ ] Ditolak
Mengosongkan semua *field*	Muncul pesan kesalahan *field*	Dapat menampilkan	[✓] Diterima [ ] Ditolak

	harus diisi	pesan kesalahan *field* harus diisi	

4.4.2.17 Pengujian Pengolahan Data Jenis

Berikut dapat dilihat pengujian dari pengolahan data jenis pada tabel 4.41 di bawah ini :

Tabel 4.41 Pengujian Pengolahan Data Jenis

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik menu jenis	Menampilkan *form* lihat jenis	Dapat menampilkan *form* lihat jenis	[✓] Diterima [ ] Ditolak
Klik tombol ubah	Menampilkan *form* ubah jenis	Dapat menampilkan *form* ubah jenis	[✓] Diterima [ ] Ditolak
Klik tombol hapus	Data berhasil dihapus di*database*	Tampil pesan "Data berhasil dhapus"	[✓] Diterima [ ] Ditolak
Klik Tambah Jenis	Menampilkan *form* tambah jenis	Dapat menampilkan *form* tambah jenis	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tidak mengisi salah satu *field*	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[✓] Diterima [ ] Ditolak
Tidak mengisi semua *field*	Tampilkan pesan kesalahan *field* harus diisi	Dapat menampilkan pesan kesalahan *field* harus diisi	[✓] Diterima [ ] Ditolak

4.4.2.18 Pengujian Pengolahan Tambah Jenis

Berikut dapat dilihat pengujian dari tambah jenis pada tabel 4.42 di bawah ini :

Tabel 4.42 Pengujian Tambah Jenis

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik menu Tambah	Menampilkan *form* tambah jenis	Dapat menampilkan *form* tambah jenis	[✓] Diterima [ ] Ditolak
Mengisi data jenis Contoh: Nama jenis: Gurilem	Jika semua field terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses tambah jenis	Jika semua field terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses tambah jenis	[✓] Diterima [ ] Ditolak
Klik tombol simpan	Data jenis akan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	Data jenis akan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	[✓] Diterima [ ] Ditolak
Klik tombol batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Data nama jenis masih kosong	Tampilkan pesan kesalahan “nama jenis harus isi”	Dapat menampilkan pesan kesalahan “nama jenis harus diisi”	[✓] Diterima [ ] Ditolak
Data jenis kosong	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[✓] Diterima [ ] Ditolak

4.4.2.19 Pengujian Pengolahan Data Produk

Berikut dapat dilihat pengujian dari pengolahan data produk pada tabel 4.43 di bawah ini :

Tabel 4.43 Pengujian Pengolahan Data Produk

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik menu produk	Menampilkan *form* lihat produk	Dapat menampilkan *form* lihat produk	[✓] Diterima [ ] Ditolak
Klik tombol ubah	Menampilkan *form* ubah produk	Dapat menampilkan *form* ubah produk	[✓] Diterima [ ] Ditolak
Klik tombol hapus	Data berhasil dihapus di*database*	Tampil pesan "Data berhasil dhapus"	[✓] Diterima [ ] Ditolak
Klik tombol tambah produk	Menampilkan form tambah produk	Dapat menampilkan form tambah produk	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Salah satu *field* kosong	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[✓] Diterima [ ] Ditolak
Data produk kosong	Tampilkan pesan kesalahan *field* harus diisi	Dapat menampilkan pesan kesalahan *field* harus diisi	[✓] Diterima [ ] Ditolak

4.4.2.20 Pengujian Pengolahan Tambah produk

Berikut dapat dilihat pengujian dari tambah produk pada tabel 4.44 di bawah ini

Tabel 4.44 Pengujian Tambah Produk

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik menu Tambah	Menampilkan *form* tambah produk	Dapat menampilkan	[✓] Diterima [ ] Ditolak

		form tambah produk	
Mengisi data produk Contoh: produk: Keripik Singkong Nama : Level 3	Jika semua field terisi semua maka akan muncul pesan " data berhasil ditambah"	Jika semua field terisi semua maka akan muncul pesan " data berhasil ditambah"	[✓] Diterima [ ] Ditolak
Klik tombol simpan	Data kategori akan tersimpan didalam *database* dan muncul pesan "data berhasil di tambah"	Data kategori akan tersimpan didalam *database* dan muncul pesan "data berhasil di tambah"	[✓] Diterima [ ] Ditolak
Klik tombol batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Data produk masih kosong	Tampilkan pesan kesalahan "field tersebut harus diisi"	Dapat menampilkan pesan kesalahan" *field* tersebut harus diisi"	[✓] Diterima [ ] Ditolak

4.4.2.21 Pengujian Pengolahan Data Provinsi

Berikut dapat dilihat pengujian dari pengolahan data provinsi pada tabel 4.45 di bawah ini :

Tabel 4.45 Pengujian Lihat Provinsi

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik menu provinsi	Menampilkan *form* lihat provinsi	Dapat menampilkan *form* lihat provinsi	[✓] Diterima [ ] Ditolak
Klik tombol ubah	Menampilkan *form* ubah provinsi	Dapat menampilkan *form* ubah provinsi	[✓] Diterima [ ] Ditolak

Klik tombol hapus	Data berhasil dihapus dari *database*	Tampil pesan "Data berhasil dihapus"	[✓] Diterima [ ] Ditolak

4.4.2.22 Pengujian Pengolahan Tambah Provinsi

Berikut dapat dilihat pengujian dari tambah provinsi pada tabel 4.46 di bawah ini :

Tabel 4.46 Pengujian Tambah Provinsi

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik tambah provinsi	Menampilkan *form* tambah provinsi	Dapat menampilkan *form* tambah provinsi	[✓] Diterima [ ] Ditolak
Mengisi data provinsi Contoh: Nama Provinsi : Jawa Barat	Jika semua field terisi maka akan muncul pesan "data berhasil ditambah"	Jika semua field terisi maka akan muncul pesan "data berhasil ditambah".	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama provinsi masih kosong	Tampilkan pesan kesalahan "nama provinsi harus diisi"	Dapat menampilkan pesan kesalahan " nama provinsi harus diisi"	[✓] Diterima [ ] Ditolak

4.4.2.23 Pengujian Pengolahan Data Kota

Berikut dapat dilihat pengujian dari pengolahan data kota pada tabel 4.47 di bawah ini :

Tabel 4.47 Pengujian Lihat Kota

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik menu kota	Menampilkan *form* lihat kota	Dapat menampilkan data kota	[✓] Diterima [ ] Ditolak
Klik tombol ubah	Menampilkan *form* ubah kota	Dapat menampilkan *form* ubah kota	[✓] Diterima [ ] Ditolak
Klik tombol hapus	Data berhasil dihapus dari *database*	Tampil pesan "Data berhasil dihapus"	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
salah satu field kosong	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[✓] Diterima [ ] Ditolak
nama kota masih kosong	Tampilkan pesan kesalahan nama kota harus disi	Dapat menampilkan pesan kesalahan nama kota harus disi	[✓] Diterima [ ] Ditolak

4.4.2.24 Pengujian Pengolahan Tambah Kota

Berikut dapat dilihat pengujian dari tambah kota pada tabel 4.48di bawah ini :

Tabel 4.48 Pengujian Tambah Kota

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik tambah kota	Menampilkan *form* tambah kota	Dapat menampilkan *form* tambah kota	[✓] Diterima [ ] Ditolak
Mengisi data kota Contoh: Provinsi : Jawa	Jika salah satu field tidak terisi atau salah dalam	Dapat memunculkan pesan kesalahan	[✓] Diterima [ ] Ditolak

Barat Nama Kota : Cianjur	pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	
Klik tombol simpan	Data pengiriman akan tersimpan didalam *database*	Data pengiriman akan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	[✓] Diterima [ ] Ditolak
Klik tombol batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama provinsi masih kosong	Tampilkan pesan kesalahan “nama kota harus diisi”	Dapat menampilkan pesan kesalahan ” nama kota harus diisi”	[✓] Diterima [ ] Ditolak

4.4.2.25 Pengujian Pengolahan Data Level

Berikut dapat dilihat pengujian dari pengolahan data level pada tabel 4.49 di bawah ini :

Tabel 4.49 Pengujian Pengolahan Data Level

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik menu level	Menampilkan *form* lihat level	Dapat menampilkan data level	[✓] Diterima [ ] Ditolak
Klik tombol ubah	Menampilkan *form* ubah level	Dapat menampilkan *form* ubah level	[✓] Diterima [ ] Ditolak
Klik tombol hapus	Data berhasil dihapus di*database*	Tampil pesan “Data berhasil dhapus”	[✓] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
salah satu field kosong	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[✓] Diterima [ ] Ditolak
nama kota masih kosong	Tampilkan pesan kesalahan nama ukuran harus disi	Dapat menampilkan pesan kesalahan nama ukuran harus disi	[✓] Diterima [ ] Ditolak

4.4.2.26 Pengujian Pengolahan Tambah Level

Berikut dapat dilihat pengujian dari tambah level pada tabel 4.50 di bawah ini :

Tabel 4.50 Pengujian Tambah Level

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data kota Contoh: Nama level: Level 3 Keterangan : Tingkat Kepedasan paling ringan	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[✓] Diterima [ ] Ditolak
Klik tombol simpan	Data pengiriman akan tersimpan didalam *database*	Data pengiriman akan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	[✓] Diterima [ ] Ditolak
Klik tombol batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama level masih kosong	Tampilkan pesan kesalahan “nama	Dapat menampilkan	[✓] Diterima [ ] Ditolak

	level harus diisi”	pesan kesalahan ” nama level harus diisi”	

4.4.2.27 Pengujian Pengolahan Data Jendral

Berikut dapat dilihat pengujian dari pengolahan data Jendral pada tabel 4.51 di bawah ini :

Tabel 4.51 Pengujian Pengolahan Data Jendral

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik pengolahan data Jendral	Menampilkan data Jendral	Dapat menampilkan data Jendral	[✓] Diterima [ ] Ditolak
Klik tombol ubah	Menampilkan *form* ubah data Jendral	Dapat menampilkan *form* ubah Jendral	[✓] Diterima [ ] Ditolak
Klik tombol hapus	Data berhasil dihapus di*database*	Tampil pesan “Data berhasil dhapus”	[✓] Diterima [ ] Ditolak
Klik tombol tambah Jendral	Menampilkan tambah data Jendral	Tampil tambah data Jendral	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
salah satu field kosong	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[✓] Diterima [ ] Ditolak

4.4.2.28 Pengujian Pengolahan Tambah Jendral

Berikut dapat dilihat pengujian dari tambah Jendral pada tabel 4.52 di bawah ini :

Tabel 4.52 Pengujian Tambah Jendral

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik tambah Jendral	Menampilkan *form* tambah	Dapat menampilkan	[✓] Diterima [ ] Ditolak

	Jendral	*form* tambah Jendral	
Mengisi data ongkos kirim Contoh: Username : peys032 Email : peys032@yahoo.com Nama Jendral : Ipey Alamat : Sekeloa Utara Provinsi Jawa Barat Kota : Bandung No. Telp : 08560000000 Pin BB : 5D5SF Nama Kendaraan : Honda Jazz Plat Nomor : F 312 YY ID Twitter : @peys032 Status Aktif : (Check)	Jika semua field terisi atau benar dalam pengisian, maka data-data Jendral akan masuk ke Pengolahan Data Jendral	Jika semua field terisi atau benar dalam pengisian, maka akan muncul di Pengolahan Data Jendral	[✓] Diterima [ ] Ditolak
Klik tombol simpan	Data Jendral akan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	Data Jendral akan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	[✓] Diterima [ ] Ditolak
Klik tombol batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Username Jendral kirim masih kosong	Tampilkan pesan kesalahan “username Jendral harus diisi”	Dapat menampilkan pesan kesalahan ” username Jendral harus diisi”	[✓] Diterima [ ] Ditolak
Email masih	Tampilkan pesan	Dapat	[✓] Diterima

kosong	kesalahan “email Jendral harus diisi”	menampilkan pesan kesalahan ” email Jendral harus diisi”	[ ] Ditolak
Nama Jendral masih kosong	Tampilkan pesan kesalahan “nama Jendral harus diisi”	Dapat menampilkan pesan kesalahan ” nama Jendral harus diisi”	[✓] Diterima [ ] Ditolak
Alamat masih kosong	Tampilkan pesan kesalahan “alamat Jendral harus diisi”	Dapat menampilkan pesan kesalahan ” alamat Jendral harus diisi”	[✓] Diterima [ ] Ditolak
Provinsi masih kosong	Tampilkan pesan kesalahan “provinsi Jendral harus diisi”	Dapat menampilkan provinsi kesalahan ” nama Jendral harus diisi”	[✓] Diterima [ ] Ditolak
Kota masih kosong	Tampilkan pesan kesalahan “kota Jendral harus diisi”	Dapat menampilkan pesan kesalahan ” kota Jendral harus diisi”	[✓] Diterima [ ] Ditolak
No. Telepon masih kosong	Tampilkan pesan kesalahan “No. Telp Jendral harus diisi”	Dapat menampilkan pesan kesalahan ” No. Telp Jendral harus diisi”	[✓] Diterima [ ] Ditolak
Nama kendaraan masih kosong	Tampilkan pesan kesalahan “Nama kendraan Jendral harus diisi”	Dapat menampilkan pesan kesalahan ” Nama Kendaraan Jendral harus diisi”	[✓] Diterima [ ] Ditolak
Plat nomor masih kosong	Tampilkan pesan kesalahan “Plat Nomor Jendral harus diisi”	Dapat menampilkan pesan kesalahan ” Plat Nomor Jendral harus diisi”	[✓] Diterima [ ] Ditolak
ID Twitter masih kosong	Tampilkan pesan kesalahan “ID Twitter Jendral harus diisi”	Dapat menampilkan pesan kesalahan ” ID Twitter	[✓] Diterima [ ] Ditolak

		Jendral harus diisi”	

4.4.2.29 Pengujian Pengolahan Data Pelanggan

Berikut dapat dilihat pengujian dari pengolahan data pelanggan pada tabel 4.53 di bawah ini :

Tabel 4.53 Pengujian Pengolahan Data Pelanggan

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik pelanggan	Menampilkan data pelanggan	Dapat menampilkan data pelanggan	[✓] Diterima [ ] Ditolak
Klik ubah	Memilih data yang akan diubah	Data yang akan diubah telah dipilih	[✓] Diterima [ ] Ditolak
Klik hapus	Data akan dihapus dari *database*. Jika tidak ada data yang dipilih akan muncul pesan kesalahan	Data telah dihapus dari database. Muncul pesan “pilih data yang akan dihapus” bila data belum dipilih	[✓] Diterima [ ] Ditolak

4.4.2.30 Pengujian Pengolahan Data Penjualan

Berikut dapat dilihat pengujian dari pengolahan data penjualan pada tabel 4.54 di bawah ini :

Tabel 4.54 Pengujian Lihat Penjualan

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik pengeolahan data penjualan	Menampilkan *form* data penjualan	Dapat menampilkan *form* lihat penjualan	[✓] Diterima [ ] Ditolak
Klikdetail penjualan	Menampilkan *form* detail	Dapat menampilkan *form*	[✓] Diterima [ ] Ditolak

	penjualan	detail penjualan	
Klik hapus	Data akan dihapus dari *database*. Jika tidak ada data yang dipilih akan muncul pesan kesalahan	Data telah dihapus dari database. Muncul pesan "pilih data yang akan dihapus" bila data belum dipilih	[✓] Diterima [ ] Ditolak
Klik data penjualan	Menampilkan *form* Nama Jendral. Setelah dipilih Jendral maka ditambahkan detail penjualan	Dapat menampilkan dan menambahkan detail penjualan	[✓] Diterima [ ] Ditolak
Klik tombol simpan	Data Penjualan akan tersimpan didalam *database* dan muncul pesan "data berhasil di tambah"	Data Penjualan akan tersimpan didalam *database* dan muncul pesan "data berhasil di tambah"	[✓] Diterima [ ] Ditolak
Klik tombol batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik tabel penjualan gagal	Menampilkan *form* penjualan gagal	Dapat menampilkan *form* penjualan gagal	[✓] Diterima [ ] Ditolak

4.4.2.31 Pengujian Pengolahan Komentar

Berikut dapat dilihat pengujian dari pengolahan komentar pada tabel 4.55 di bawah ini :

Tabel 4.55 Pengujian Pengolahan Komentar

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik pengolahan komentar	Menampilkan komentar	Dapat menampilkan komentar	[✓] Diterima [ ] Ditolak
Klik *check box* pada data yang akan dihapus hapus	Memilih data yang akan dihapus	Data yang akan dihapus telah dipilih	[✓] Diterima [ ] Ditolak

4.4.3 *Cause Effect Testing* Untuk Pelanggan dan Jendral

Teknik ini merupakan suplemen dari untuk *equivalence testing* menyediakan cara untuk memilih kombinasi data input, melibatkan kondisi input *(Cause)* dan kondisi output *(effect)* untuk mencegah pendefinisian kasus uji yang terlalu banyak. Adapun langkah-langkahnya membagi spesifikasi kebutuhan menjadi bagian yang memiliki kemungkinan kerja kemudian definisikan *cause effect* berdasarkan spesifikasi kebutuhan dan membuat tabel keputusan, kolom untuk *test case* dan baris untuk *cause/effect*, symbol 1 untuk mewakili ya symbol 0 untuk mewakili tidak. Berikut ini *cause effect testing* untuk pelanggan:

4.4.3.1 *Cause effect testing* registrasi pelanggan

Cause effect testing registrasi pelanggan menjelaskan pendefinisian kasus uji kondisi input *(Cause)* dan kondisi output *(effect)*yang terjadi pada proses registrasi pelanggan. Berikut *Cause effect testing* untuk registrasi pelanggan :

***Input states* :**

1. Nama memenuhi kondisi yang telah ditentukan
2. Alamat memenuhi kondisi yang telah ditentukan
3. Kota memenuhi kondisi yang telah ditentukan
4. Jenis Kelamin memenuhi kondisi yang telah ditentukan
5. No Telepon memenuhi kondisi yang telah ditentukan
6. Email memenuhi kondisi yang telah ditentukan
7. Konfirmasi Email memenuhi kondisi yang telah ditentukan
8. Username memenuhi kondisi yang telah ditentukan

9. Password memenuhi kondisi yang telah ditentukan
10. Konfirmasi Password memenuhi kondisi yang telah ditentukan

***Output states* :**

A. Login berhasil
B. Menampilkan pesan kesalahan dari Nama yang diinputkan
C. Menampilkan pesan kesalahan dari Alamat yang diinputkan
D. Menampilkan pesan kesalahan dari Kota yang diinputkan
E. Menampilkan pesan kesalahan dari Jenis Kelamin yang diinputkan
F. Menampilkan pesan kesalahan dari No Telepon yang diinputkan
G. Menampilkan pesan kesalahan dari Email yang diinputkan
H. Menampilkan pesan kesalahan dari Konfirmasi Email yang diinputkan
I. Menampilkan pesan kesalahan dari Username yang diinputkan
J. Menampilkan pesan kesalahan dari Password yang diinputkan
K. Menampilkan pesan kesalahan dari Konfirmasi Password yang diinputkan

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari login pelanggan. Tabel *cause effect testing* untuk login pelanggan dapat dilihat pada tabel 4.56

Tabel 4.56 *Cause effect testing* untuk registrasi pelanggan

st cause / *Cause effect*	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**
1	1	0	1	1	1	1	1	1	1	1	1
2	1	1	0	1	1	1	1	1	1	1	1
3	1	1	1	0	1	1	1	1	1	1	1

4	1	1	1	1	0	1	1	1	1	1	1
5	1	1	1	1	1	0	1	1	1	1	1
6	1	1	1	1	1	1	0	1	1	1	1
7	1	1	1	1	1	1	1	0	1	1	1
8	1	1	1	1	1	1	1	1	0	1	1
9	1	1	1	1	1	1	1	1	1	0	1
10	1	1	1	1	1	1	1	1	1	1	0
A	1	0	0	0	0	0	0	0	0	0	0
B	0	1	0	0	0	0	0	0	0	0	0
C	0	0	1	0	0	0	0	0	0	0	0
D	0	0	0	1	0	0	0	0	0	0	0
E	0	0	0	0	1	0	0	0	0	0	0
F	0	0	0	0	0	1	0	0	0	0	0
G	0	0	0	0	0	0	1	0	0	0	0
H	0	0	0	0	0	0	0	1	0	0	0
I	0	0	0	0	0	0	0	0	1	0	0
J	0	0	0	0	0	0	0	0	0	1	0

4.4.3.2 Cause effect testing login pelanggan

Cause effect testing login pelanggan menjelaskan pendefinisian kasus uji kondisi input *(Cause)* dan kondisi output *(effect)*yang terjadi pada proses login pelanggan. Berikut *Cause effect testing* untuk login pelanggan :

***Input states* :**

1. Username memenuhi kondisi yang telah ditentukan
2. Password memenuhi kondisi yang telah ditentukan
3. Type memenuhi kondisi yang telah ditentukan

***Output states* :**

A. Login berhasil

B. Menampilkan pesan kesalahan dari username yang diinputkan

C. Menampilkan pesan kesalahan dari password yang diinputkan

D. Menampilkan pesan kesalahan dari type yang diinputkan atau dipilih

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari login pelanggan. Tabel *cause effect testing* untuk login pelanggan dapat dilihat pada tabel 4.57

Tabel 4.57 *Cause effect testing* untuk login pelanggan

st cause / *Cause effect*	**1**	**2**	**3**	**4**
1	1	0	1	1
2	1	1	0	1
3	1	1	1	0
A	1	0	0	0
B	0	1	0	0
C	0	0	1	0
D	0	0	0	1

4.4.3.3 Cause effect testing Login Jendral

Cause effect testing login Jendral menjelaskan pendefinisian kasus uji kondisi input *(Cause)* dan kondisi output *(effect)*yang terjadi pada proses login Jendral. Berikut *Cause effect testing* untuk login Jendral:

***Input states* :**

4. Username memenuhi kondisi yang telah ditentukan
5. Password memenuhi kondisi yang telah ditentukan
6. Type memenuhi kondisi yang telah ditentukan

***Output states* :**

E. Login berhasil
F. Menampilkan pesan kesalahan dari username yang diinputkan
G. Menampilkan pesan kesalahan dari password yang diinputkan
H. Menampilkan pesan kesalahan dari type yang diinputkan atau dipilih

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari login Jendral. Tabel *cause effect testing* untuk login Jendral dapat dilihat pada tabel 4.58

Tabel 4.58 *Cause effect testing* untuk login Jendral

st cause / *Cause effect*	**1**	**2**	**3**	**4**
1	1	0	1	1
2	1	1	0	1
3	1	1	1	0
A	1	0	0	0
B	0	1	0	0
C	0	0	1	0
D	0	0	0	1

4.4.3.4 *Cause Effect Testing* Ubah Informasi Akun Saya

Cause effect testing ubah informasi akun saya menjelaskan pendefinisian kasus uji kondisi input *(Cause)* dan kondisi output *(effect)*yang terjadi pada proses ubah informasi akun saya. Berikut *Cause effect testing* untuk ubah informasi akun saya :

***Input states* :**

1. Nama Jendral memenuhi kondisi yang telah ditentukan
2. Twitter Jendral memenuhi kondisi yang telah ditentukan
3. Pin Jendral memenuhi kondisi yang telah ditentukan
4. Telepon Jendral memenuhi kondisi yang telah ditentukan
5. Merk/Tipe Kendaraan Jendral memenuhi kondisi yang telah ditentukan
6. Plat No Kendaraan memenuhi kondisi yang telah ditentukan

***Output states* :**

A. Data disimpan kedalam database
B. Menampilkan pesan kesalahan dari Nama Jendral yang diinputkan
C. Menampilkan pesan kesalahan dari Twitter Jendral yang diinputkan
D. Menampilkan pesan kesalahan dari Pin Jendral yang diinputkan
E. Menampilkan pesan kesalahan dari Telepon Jendral yang diinputkan
F. Menampilkan pesan kesalahan dari Merk/Tipe Kendaraan yang diinputkan
G. Menampilkan pesan kesalahan dari Plat No Kendaraan yang diinputkan

Dari input states dan output states yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel cause effect testing untuk

mempermudah membaca hasil dari uji atau testing dari ubah informasi akun saya Jendral. Tabel cause effect testing untuk ubah informasi akun saya Jendral dapat dilihat pada tabel 4.59

Tabel 4.59 *Cause effect testing* Ubah Informasi Akun Saya

st cause / ***Cause effect***	**1**	**2**	**3**	**4**	**5**	**5**	**6**
1	1	0	1	1	1	1	1
2	1	1	0	1	1	1	1
3	1	1	1	0	1	1	1
4	1	1	1	1	0	1	1
5	1	1	1	1	1	0	1
6	1	1	1	1	1	1	0
A	1	0	0	0	0	0	0
B	0	1	0	0	0	0	0
C	0	0	1	0	0	0	0
D	0	0	0	1	0	0	0
E	0	0	0	0	1	0	0
F	0	0	0	0	0	1	0
G	0	0	0	0	0	0	1

4.4.3.5 *Cause Effect Testing* Ubah Password Jendral

Cause effect testing ubah password Jendral menjelaskan pendefinisian kasus uji kondisi input *(Cause)* dan kondisi output *(effect)*yang terjadi pada proses ubah password Jendral. Berikut *Cause effect testing* untuk ubah password Jendral

Input states:

1. Password memenuhi kondisi yang telah ditentukan
2. Konfirmasi Password memenuhi kondisi yang telah ditentukan

Output states :

A. Data disimpan ke dalam database
B. Menampilkan pesan kesalahan dari password yang diinputkan
C. Menampilkan pesan kesalahan dari Konfirmasi Password yang diinputkan

Dari input states dan output states yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari ubah password Jendral. Tabel cause effect testing untuk ubah password Jendral dapat dilihat pada tabel 4.60.

Tabel 4.60 *Cause effect testing* ubah password Jendral

Test cause / *Cause effect*	**1**	**2**	**3**
1	1	0	1
2	1	1	0
A	1	0	0
B	0	1	0
C	0	0	1

4.4.4 Cause Effect Testing Untuk Admin

Teknik ini merupakan suplemen dari untuk *equivalence testing* menyediakan cara untuk memilih kombinasi data input, melibatkan kodisi input *(Cause)* dan kodisi output *(effect)* untuk mencegah pendefinisian kasus uji yang terlalu banyak. Berikut ini *cause effect testing* untuk Admin:

4.4.4.1 Cause effect testing login admin

Cause effect testing login admin menjelaskan pendefinisian kasus uji kondisi input *(Cause)* dan kondisi output *(effect)*yang terjadi pada proses login admin. Berikut *Cause effect testing* untuk login admin :

***Input states* :**

1. Email memenuhi kondisi yang telah ditentukan
2. Password memenuhi kondisi yang telah ditentukan
3. Type memenuhi kondisi yang telah ditentukan

***Output states* :**

A. Login berhasil

B. Menampilkan pesan kesalahan dari email yang diinputkan

C. Menampilkan pesan kesalahan dari password yang diinputkan

D. Menampilkan pesan kesalahan dari type yang diinputkan atau dipilih

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari login admin. Tabel *cause effect testing* untuk login admin dapat dilihat pada tabel 4.61

Tabel 4.61 *Cause effect testing* untuk login admin

st cause / *Cause effect*	**1**	**2**	**3**	**4**
1	1	0	1	1
2	1	1	0	1
3	1	1	1	0

A	1	0	0	0
B	0	1	0	0
C	0	0	1	0
D	0	0	0	1

4.4.4.2 Cause Effect Testing Akun Saya

Cause effect testing Akun Saya menjelaskan pendefinisian kasus uji kondisi input *(Cause)* dan kondisi output *(effect)*yang terjadi pada proses Akun saya . Berikut *Cause effect testing* untuk Akun Saya :

Input states :

1. Email memenuhi kondisi yang telah ditentukan
2. Nama memenuhi kondisi yang telah ditentukan
3. Telp memenuhi kondisi yang telah ditentukan

Output states :

A. Data disimpan kedalam database

B.Menampilkan pesan kesalahan dari email yang diinputkan

C.Menampilkan pesan kesalahan dari nama yang diinputkan

D.Menampilkan pesan kesalahan dari Telp yang diinputkan

Dari input states dan output states yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel cause effect testing untuk mempermudah membaca hasil dari uji atau testing dari edit profil. Tabel cause effect testing untuk edit profil dapat dilihat pada tabel 4.62

Tabel 4.62 *Cause effect testing* Akun Saya

st cause / *Cause effect*	**1**	**2**	**3**	**4**
1	1	0	1	1
2	1	1	0	1
3	1	1	1	0
A	1	0	0	0
B	0	1	0	0
C	0	0	1	0
D	0	0	0	1

4.4.4.3 Cause Effect Testing edit password

Cause effect testing edit assword menjelaskan pendefinisian kasus uji kondisi input *(Cause)* dan kondisi output *(effect)*yang terjadi pada proses edit password . Berikut *Cause effect testing* untuk edit password :

Input states:

1. Password lama memenuhi kondisi yang telah ditentukan
2. Password baru memenuhi kondisi yang telah ditentukan
3. Confirm password memenuhi kondisi yang telah ditentukan

Output states :

A. Data disimpan kedalam database

B. Menampilkan pesan kesalahan dari password lama yang diinputkan

C. Menampilkan pesan kesalahan dari password baru yang diinputkan

D. Menampilkan pesan kesalahan dari confirm yang diinputkan

Dari input states dan output states yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari edit password. Tabel cause effect testing untuk edi password dapat dilihat pada tabel 4.63

Tabel 4.63 *Cause effect* testing edit password

:st cause / *Cause effect*	**1**	**2**	**3**	**4**
1	1	0	1	1
2	1	1	0	1
3	1	1	1	0
A	1	0	0	0
B	0	1	0	0
C	0	0	1	0
D	0	0	0	1

4.4.4.4 Cause Effect Testing Tambah Jenis

Cause effect testing tambah jenis menjelaskan pendefinisian kasus uji kondisi input *(Cause)* dan kondisi output *(effect)*yang terjadi pada proses tambah jenis . Berikut *Cause effect testing* untuk tambah jenis :

Input states :

1. Nama Jenis Produk memenuhi yang telah ditentukan

Output states :

A. Data disimpan ke dalam database

B. Menampilkan pesan kesalahan dari Nama Jenis Produk yang diinputkan

Dari input states dan output states yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel cause effect testing untuk mempermudah membaca hasil dari uji atau testing dari tambah jenis. Tabel cause effect testing untuk tambah jenis dapat dilihat pada tabel 4.64

Tabel 4.64 *Cause Effect Testing* Tambah Jenis

st cause / Cause effect	**1**	**2**
1	1	0
A	1	0
B	0	1

4.4.4.5 Cause Effect Testing Tambah Level

Cause effect testing tambah level menjelaskan pendefinisian kasus uji kondisi input *(Cause)* dan kondisi output *(effect)*yang terjadi pada proses tambah jenis . Berikut *Cause effect testing* untuk tambah level :

Input states :

1. Nama Level memenuhi yang telah ditentukan

Output states :

A. Data disimpan ke dalam database

B. Menampilkan pesan kesalahan dari Nama Jenis Produk yang diinputkan

Dari input states dan output states yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel cause effect testing untuk mempermudah membaca hasil dari uji atau testing dari tambah level. Tabel cause effect testing untuk tambah level dapat dilihat pada tabel 4.65

Tabel 4.65 *Cause effect testing* untuk Tambah Level

:st cause / *Cause effect*	**1**	**2**
1	1	0
A	1	0
B	0	1

4.4.4.6 Cause Effect Testing Tambah Produk

Cause effect testing tambah produk menjelaskan pendefinisian kasus uji kondisi input *(Cause)* dan kondisi output *(effect)*yang terjadi pada proses tambah produk . Berikut *Cause effect testing* untuk tambah produk :

Input states :

1. Jenis Produk memenuhi yang telah ditentukan
2. Level memenuhi yang telah ditentukan
3. Deskripsi produk memenuhi yang telah ditentukan

Output states :

A. Data disimpan ke dalam database

B. Menampilkan pesan kesalahan dari Jenis Produk yang diinputkan

C. Menampilkan pesan kesalahan dari Level yang diinputkan

Dari input states dan output states yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel cause effect testing untuk mempermudah membaca hasil dari uji atau testing dari tambah produk. Tabel cause effect testing untuk tambah produk dapat dilihat pada tabel 4.66

Tabel 4.66 *Cause Effect Testing* Tambah Produk

st cause / Cause effect	**1**	**2**	**3**	**4**
1	1	0	1	1
2	1	1	0	1
3	1	1	1	0
A	1	0	0	1
B	0	1	0	0
C	0	0	1	0
D	0	0	0	0

4.4.4.7 Cause Effect Testing Tambah Provinsi

Cause effect testing tambah provinsi menjelaskan pendefinisian kasus uji kondisi input *(Cause)* dan kondisi output *(effect)*yang terjadi pada proses tambah provinsi . Berikut *Cause effect testing* untuk tambah provinsi :

Input states :

1. Nama Provinsi memenuhi yang telah ditentukan

Output states :

A. Data disimpan ke dalam database

B. Menampilkan pesan kesalahan dari Nama Provinsi yang diinputkan

Dari input states dan output states yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel cause effect testing untuk mempermudah membaca hasil dari uji atau testing dari tambah provinsi. Tabel cause effect testing untuk tambah provinsi dapat dilihat pada tabel 4.67

Tabel 4.67 *Cause effect testing* untuk Tambah Provinsi

:st cause / *Cause effect*	**1**	**2**
1	1	0
A	1	0
B	0	1

4.4.4.8 Cause Effect Testing Tambah Kota

Cause effect testing tambah kota menjelaskan pendefinisian kasus uji kondisi input *(Cause)* dan kondisi output *(effect)*yang terjadi pada proses tambah kota . Berikut *Cause effect testing* untuk tambah kota :

Input states :

1. Nama Provinsi memenuhi yang telah ditentukan
2. Nama Kota memenuhi yang telah ditentukan

Output states :

A. Data disimpan ke dalam database

B. Menampilkan pesan kesalahan dari Nama Provinsi yang diinputkan

C. Menampilkan pesan kesalahan dari Nama Kota yang diinputkan

Dari input states dan output states yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel cause effect testing untuk mempermudah membaca hasil dari uji atau testing dari tambah kota. Tabel cause effect testing untuk tambah kota dapat dilihat pada tabel 4.68

Tabel 4.68 *Cause effect testing* untuk Tambah Kota

:st cause / *Cause effect*	**1**	**2**	**3**
1	1	0	1
2	1	1	0
A	1	0	0
B	0	1	0
C	0	0	1

4.4.4.9 Cause Effect Testing Tambah Jendral

Cause effect testing tambah jendral menjelaskan pendefinisian kasus uji kondisi input *(Cause)* dan kondisi output *(effect)*yang terjadi pada proses tambah jendral . Berikut *Cause effect testing* untuk tambah jendral :

Input states :

1. Username memenuhi yang telah ditentukan
2. Email memenuhi yang telah ditentukan
3. Nama Jendral memenuhi yang telah ditentukan
4. Alamat memenuhi yang telah ditentukan
5. Provinsi memenuhi yang telah ditentukan

6. Kota memenuhi yang telah ditentukan
7. No. Telepon memenuhi yang telah ditentukan
8. Pin BB memenuhi yang telah ditentukan
9. Nama Kendaraaan memenuhi yang telah ditentukan
10. Plat Nomor memenuhi yang telah ditentukan
11. ID Twitter memenuhi yang telah ditentukan

Output states :

A. Data disimpan ke dalam database
B. Menampilkan pesan kesalahan dari Username yang diinputkan
C. Menampilkan pesan kesalahan dari Email yang diinputkan
D. Menampilkan pesan kesalahan dari Nama Jendral yang diinputkan
E. Menampilkan pesan kesalahan dari Alamat yang diinputkan
F. Menampilkan pesan kesalahan dari Provinsi yang diinputkan
G. Menampilkan pesan kesalahan dari Kota yang diinputkan
H. Menampilkan pesan kesalahan dari No. Telepon yang diinputkan
I. Menampilkan pesan kesalahan dari Nama Kendaraan yang diinputkan
J. Menampilkan pesan kesalahan dari Plat Nomor yang diinputkan
K. Menampilkan pesan kesalahan dari ID Twitter yang diinputkan

Dari input states dan output states yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel cause effect testing untuk mempermudah membaca hasil dari uji atau testing dari tambah Jendral. Tabel cause effect testing untuk tambah Jendral dapat dilihat pada tabel 4.66

Tabel 4.69 *Cause effect testing* Tambah Jendral

Test cause / *Cause effect*	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
1	1	0	1	1	1	1	1	1	1	1	1	1
2	1	1	0	1	1	1	1	1	1	1	1	1
3	1	1	1	0	1	1	1	1	1	1	1	1
4	1	1	1	1	0	1	1	1	1	1	1	1
5	1	1	1	1	1	0	1	1	1	1	1	1
6	1	1	1	1	1	1	0	1	1	1	1	1
7	1	1	1	1	1	1	1	0	1	1	1	1
8	1	1	1	1	1	1	1	1	0	1	1	1
9	1	1	1	1	1	1	1	1	1	0	1	1
10	1	1	1	1	1	1	1	1	1	1	0	1
11	1	1	1	1	1	1	1	1	1	1	1	0
A	1	0	0	0	0	0	0	0	1	0	0	0
B	0	1	0	0	0	0	0	0	0	0	0	0
C	0	0	1	0	0	0	0	0	0	0	0	0
D	0	0	0	1	0	0	0	0	0	0	0	0
E	0	0	0	0	1	0	0	0	0	0	0	0
F	0	0	0	0	0	1	0	0	0	0	0	0
G	0	0	0	0	0	0	1	0	0	0	0	0
H	0	0	0	0	0	0	0	1	0	0	0	0
I	0	0	0	0	0	0	0	0	0	1	0	0
J	0	0	0	0	0	0	0	0	0	0	1	0
K	0	0	0	0	0	0	0	0	0	0	0	1

4.4.4.10 *Cause Effect Testing* Tambah Data Penjualan

Cause effect testing tambah data penjualan menjelaskan pendefinisian kasus uji kondisi input *(Cause)* dan kondisi output *(effect)*yang terjadi pada proses tambah data penjualan. Berikut *Cause effect testing* untuk tambah data penjualan:

Input states :

1. Nama Jendral memenuhi yang telah ditentukan
2. Nama Jenis memenuhi yang telah ditentukan
3. Level memenuhi yang telah ditentukan
4. Jumlah produk yang telah ditentukan

Output states :

A. Data disimpan ke dalam database
B. Menampilkan pesan kesalahan dari Nama Jendral yang diinputkan
C. Menampilkan pesan kesalahan dari Nama Jenis yang diinputkan
D. Menampilkan pesan kesalahan dari Level yang diinputkan
E. Menampilkan pesan kesalahan dari Jumlah Produk yang diinputkan

Dari input states dan output states yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel cause effect testing untuk mempermudah membaca hasil dari uji atau testing dari tambah data penjualan. Tabel cause effect testing untuk tambah data penjualan dapat dilihat pada tabel 4.67

Tabel 4.70 *Cause effect testing* untuk Tambah Data Penjualan

st cause / *Cause effect*	**1**	**2**	**3**	**4**	**5**
1	1	0	1	1	1
2	1	1	0	1	1
3	1	1	1	0	1
4	1	1	1	1	0
A	1	0	0	0	0
B	0	1	0	0	0
C	0	0	1	0	0
D	0	0	0	1	0
E	0	0	0	0	1

4.4.5 Kesimpulan Pengujian Alpa

Berdasarkan hasil pengujian sistem yang telah dilakukan secara keseluruhan pelangganikan kesimpulan bahwa pada proses pada aplikasi ini telah melalui tahap perbaikan dan sudah di maksimalkan terhadap proses-proses tersebut dan secara fungsional sistem sudah dapat digunakan dan menghasilkan output yang diharapkan.

4.5 Pengujian Beta

Pengujian beta merupakan pengujian yang dilakukan secara objektif dimana diuji secara langsung ke lapangan yaitu instansi yang bersangkutan mengenai kepuasan user dengan kandungan poin yaitu pemenuhan kebutuhan dari tujuan awal pembangunan aplikasi tracking berbasis *mobile/web* dan tampilan antarmuka dari aplikasi *tracking* tersebut. Pengujian beta ini dilakukan dengan

cara melakukan pengujian langsung di tempat penelitian. Penelitian ini dilakukan di dua lingkungan :

1. Lingkungan penjualan di PT Maicih Inti Synergi

Penelitian di lingkungan penjual di PT Maicih Inti Synergi dilakukan kepada administrator dengan metode penelitian yang dilakukan yaitu wawancara. Penelitian ini dilakukan supaya dapat mengetahui sejauh mana sistem yang dibangun dapat menjadi alternatif penyelesaian permasalahan yang telah dijelaskan di awal.

2. Lingkungan masyarakat umum

Penelitian di lingkungan masyarakat umum dilakukan kepada Jendral dan pelanggan khususnya yang sudah terbiasa dengan teknologi *internet*, penelitian ini dilakukan dengan teknik kuesioner, tujuan penelitian ini untuk mengetahui sejauh mana sistem yang dibangun dapat memberikan kemudahan masyarakat umum untuk melakukan *tracking* kepada Jendral dan pelanggan.

4.5.1 Kuesioner Pengguna

Kuesioner pengguna ini disebarkan menggunakan teknik sampling yaitu *Simple Random Sampling* yang disebarkan kepada 10 jendral dan 10 pelanggan. Dari hasil kuesioner tersebut akan dilakukan perhitungan agar dapat diambil kesimpulan terhadap penilaian penerapan sistem yang baru.

Perhitungan tersebut menggunakan rumus :

$Y = P/Q * 100\%$

Keterangan : P = Banyaknya jawaban responden tiap soal.

Q = Jumlah responden

Y = Nilai persentase

Kuisioner ini terdiri dari 10 pertanyaan (contoh kuisioner dapat dilihat pada lampiran) dengan menggunakan skala *likert* dengan skala 1 sampai 10.

4.5.1.1 Kuesioner Pengujian Beta Untuk Pelanggan

1. Apakah Anda setuju bahwa tampilan *aplikasi tracking* cukup menarik dan mudah dipahami / *user friendly*?

   A. Sangat Setuju
   B. Setuju
   C. Cukup Setuju
   D. Kurang Setuju
   E. Tidak Setuju

2. Apakah Anda setuju bahwa aplikasi tracking berbasis mobile ini dapat membantu Anda untuk mendapatkan informasi terbaru tentang lokasi penjualan Maicih?

   A. Sangat Setuju
   B. Setuju
   C. Cukup Setuju
   D. Kurang Setuju
   E. Tidak Setuju

3. Apakah Anda setuju bahwa *aplikasi tracking* ini akan lebih mempermudah Anda dalam melakukan pencarian untuk membeli produk Maicih di bandingkan mengandalkan akun twitter?

   A. Sangat Setuju
   B. Setuju
   C. Cukup Setuju
   D. Kurang Setuju
   E. Tidak Setuju

4. Apakah Anda setuju dengan adanya *aplikasi tracking mobile* ini pemilik dapat meningkatkan penjualan produk Maicih?

   A. Sangat Setuju
   B. Setuju
   C. Cukup Setuju
   D. Kurang Setuju
   E. Tidak Setuju

5. Apakah anda setuju bahwa sistem tracking yang digunakan akan lebih memudahkankan anda?

   A. Sangat Setuju
   B. Setuju
   C. Cukup Setuju
   D. Kurang Setuju
   E. Tidak Setuju

6. Apakah anda setuju, GPS pada *mobile* anda dapat berjalan dengan baik ?

   A. Sangat Setuju
   B. Setuju
   C. Cukup Setuju
   D. Kurang Setuju
   E. Tidak Setuju

7. Apakah anda setuju aplikasi mobile android sangat diminatin kalangan masyarakat?

   A. Sangat Setuju
   B. Setuju
   C. Cukup Setuju
   D. Kurang Setuju
   E. Tidak Setuju

8. Apakah anda setuju dengan adanya aplikasi ini sebagai media promosi?

   A. Sangat Setuju
   B. Setuju
   C. Cukup Setuju
   D. Kurang Setuju
   E. Tidak Setuju

Berdasarkan data hasil kusioner tersebut, dapat dicari prosentase masing-masing jawaban dengan menggunakan rumus : Y = P/Q * 100%

Keterangan :

P = Banyaknya jawaban responden tiap soal.

Q = Jumlah responden

Y = Nilai persentase

Berikut ini adalah hasil persentase masing-masing jawaban yang sudah dihitung nilainya dengan menggunakan rumus diatas. Kuisioner ini diujikan kepada 10 orang.

1. Apakah Anda setuju bahwa tampilan aplikasi *tracking* cukup menarik dan mudah dipahami / *user friendly*?

Tabel 4.71 Hasil pengujian kuesioner soal nomor 1

Pernyataan	No.	Keterangan	Responden	Prosentase (%)
1	1	Sangat Setuju	3	30
	2	Setuju	7	70
	3	Setuju	0	0
	4	Kurang Setuju	0	0
	5	Tidak Setuju	0	0
Jumlah			10	100

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 3 atau 30% menyatakan sangat setuju, dan 7 atau 70% menyatakan setuju bahwa aplikasi *tracking mobile* ini cukup menarik dan mudah untuk dipahami/ user friendly.

2. Apakah Anda setuju bahwa aplikasi tracking berbasis mobile ini dapat membantu Anda untuk mendapatkan informasi terbaru tentang lokasi penjualan Maicih?

Tabel 4.72 Hasil pengujian kuesioner soal nomor 2

Pernyataan	No.	Keterangan	Responden	Prosentase (%)
2	1	Sangat Setuju	3	30
	2	Setuju	6	60
	3	Setuju	1	10
	4	Kurang Setuju	0	0
	5	Tidak Setuju	0	0
Jumlah			10	100

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 3 orang atau 30% menyatakan sangat setuju, 6 orang atau 60% menyatakan setuju dan 1 orang atau 10% menyatakan cukup setuju bahwa informasi lokasi terbaru untuk mendapatkan produk Maicih.

3. Apakah Anda setuju bahwa *aplikasi tracking* ini akan lebih mempermudah Anda dalam melakukan pencarian untuk membeli produk Maicih di bandingkan mengandalkan akun twitter?

Tabel 4.73 Hasil pengujian kuesioner soal nomor 3

Pernyataan	No.	Keterangan	Responden	Prosentase (%)
3	1	Sangat Setuju	2	20
	2	Setuju	6	60
	3	Setuju	2	20
	4	Kurang Setuju	0	0
	5	Tidak Setuju	0	0
Jumlah			10	100

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 2 atau 20% menyatakan sangat setuju, 6 orang atau 60 % menyatakan setuju dan, 2 atau 20 % menyatakan cukup setuju bahwa website ini akan lebih memudahkan produk Maicih dibanding mengandalkan twitter.

4. Apakah Anda setuju dengan adanya *aplikasi tracking mobile* ini pemilik dapat meningkatkan penjualan produk Maicih?

Tabel 4.74 Hasil pengujian kuesioner soal nomor 4

Pernyataan	No.	Keterangan	Responden	Prosentase (%)
4	1	Sangat Bagus	6	60
	2	Bagus	3	30
	3	Cukup	1	10
	4	Kurang Bagus	0	0
	5	Tidak Bagus	0	0
Jumlah			10	100

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 6 orang atau 60% menyatakan sangat setuju,3 orang atau 30% menyatakan setuju dan 1 orang atau 10 menyatakan cukup setuju bahwa aplikasi tracking mobile ini akan meningkatkan penjualan produk Maicih

5. Apakah anda setuju bahwa sistem tracking yang digunakan akan lebih memudahkan anda.

Tabel 4.75 Hasil pengujian kuesioner soal nomor 5

Pernyataan	No.	Keterangan	Responden	Prosentase (%)
5	1	Sangat Bagus	2	20
	2	Bagus	4	40
	3	Cukup	2	20
	4	Kurang Bagus	0	0
	5	Tidak Bagus	0	0
Jumlah			10	100

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 2 orang atau 20% menyatakan sangat setuju, 4 orang atau 40% menyatakan setuju dan 2 orang atau 20% menyatakan cukup setuju bahwa sistem *tracking* yang digunakan akan lebih memudahkan.

6. Apakah anda setuju, GPS pada *mobile* anda dapat berjalan dengan baik ?

Tabel 4.76 Hasil pengujian kuesioner soal nomor 6

Pernyataan	No.	Keterangan	Responden	Prosentase (%)
6	1	Sangat Setuju	0	0
	2	Setuju	4	40
	3	Setuju	6	60
	4	Kurang Setuju	0	0
	5	Tidak Setuju	0	0
Jumlah			10	100

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 4 orang atau 40% menyatakan setuju dan 2 orang atau 20% menyatakan cukup setuju bahwa koneksi internetnya dapat mengakses GPS secara lancar.

7. Apakah anda setuju aplikasi mobile android sangat diminatin kalangan masyarakat?

Tabel 4.77 Hasil pengujian kuesioner soal nomor 7

Pernyataan	No.	Keterangan	Responden	Prosentase (%)
7	1	Sangat Setuju	5	50
	2	Setuju	5	50
	3	Setuju	0	00
	4	Kurang Setuju	0	0
	5	Tidak Setuju	0	0
Jumlah			10	100

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 5 orang atau 50% menyatakan sangat setuju dan 5 orang atau 50% menyatakan setuju bahwa sistem *tracking* yang digunakan akan lebih memudahkan.

8. Apakah anda setuju dengan adanya aplikasi ini sebagai media promosi?

Tabel 4.78 Hasil pengujian kuesioner soal nomor 8

Pernyataan	No.	Keterangan	Responden	Prosentase (%)
8	1	Sangat Setuju	2	20
	2	Setuju	7	70
	3	Setuju	1	10
	4	Kurang Setuju	0	0
	5	Tidak Setuju	0	0
Jumlah			10	100

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 2 orang atau 20% menyatakan sangat setuju, 7 orang atau 70% menyatakan setuju dan 1 orang atau 10% menyatakan cukup setuju bahwa aplikasi ini dapat digunakan sebagai media promosi.

4.5.1.2 Kuesioner Pengujian Beta Untuk Jendral

1. Apakah Anda setuju bahwa tampilan *aplikasi tracking* cukup menarik dan mudah dipahami / *user friendly*?

   A. Sangat Setuju
   B. Setuju
   C. Cukup Setuju
   D. Kurang Setuju
   E. Tidak Setuju

2. Apakah Anda setuju bahwa aplikasi tracking berbasis mobile ini dapat membantu Anda untuk mendapatkan informasi terbaru tentang lokasi penjualan Maicih?

   A. Sangat Setuju
   B. Setuju
   C. Cukup Setuju
   D. Kurang Setuju
   E. Tidak Setuju

3. Apakah Anda setuju bahwa *aplikasi tracking* ini akan lebih mempermudah Anda dalam melakukan pencarian untuk membeli produk Maicih di bandingkan mengandalkan akun twitter?

   A. Sangat Setuju
   B. Setuju
   C. Cukup Setuju
   D. Kurang Setuju
   E. Tidak Setuju

4. Apakah Anda setuju dengan adanya *aplikasi tracking mobile* ini pemilik dapat meningkatkan penjualan produk Maicih?

   A. Sangat Setuju
   B. Setuju
   C. Cukup Setuju
   D. Kurang Setuju
   E. Tidak Setuju

5. Apakah anda setuju bahwa sistem tracking yang digunakan akan lebih memudahkankan anda?

   A. Sangat Setuju
   B. Setuju
   C. Cukup Setuju
   D. Kurang Setuju
   E. Tidak Setuju

6. Apakah anda setuju, GPS pada *mobile* anda dapat berjalan dengan baik ?

   A. Sangat Setuju
   B. Setuju
   C. Cukup Setuju
   D. Kurang Setuju
   E. Tidak Setuju

7. Apakah anda setuju aplikasi mobile android sangat diminatin kalangan masyarakat?

   A. Sangat Setuju
   B. Setuju
   C. Cukup Setuju
   D. Kurang Setuju
   E. Tidak Setuju

8. Apakah anda setuju dengan adanya aplikasi ini sebagai media promosi?

A. Sangat Setuju
B. Setuju
C. Cukup Setuju
D. Kurang Setuju
E. Tidak Setuju

Berdasarkan data hasil kusioner tersebut, dapat dicari prosentase masing-masing jawaban dengan menggunakan rumus : Y = P/Q * 100%

Keterangan :

P = Banyaknya jawaban responden tiap soal.

Q = Jumlah responden

Y = Nilai persentase

Berikut ini adalah hasil persentase masing-masing jawaban yang sudah dihitung nilainya dengan menggunakan rumus diatas. Kuisioner ini diujikan kepada 10 orang.

1. Apakah Anda setuju bahwa tampilan aplikasi *tracking* cukup menarik dan mudah dipahami / *user friendly*?

Tabel 4.79 Hasil pengujian kuesioner soal nomor 1

Pernyataan	No.	Keterangan	Responden	Prosentase (%)
1	1	Sangat Setuju	4	40
	2	Setuju	6	60
	3	Cukup Setuju	0	0
	4	Kurang Setuju	0	0
	5	Tidak Setuju	0	0
Jumlah			10	100

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 4 atau 40% menyatakan sangat setuju, dan 6 atau 60% menyatakan setuju bahwa aplikasi *tracking mobile* ini cukup menarik dan mudah untuk dipahami para Jendral

2. Apakah Anda setuju bahwa aplikasi tracking berbasis mobile ini dapat membantu Anda untuk mendapatkan informasi terbaru tentang lokasi penjualan Maicih?

Tabel 4.80 Hasil pengujian kuesioner soal nomor 2

Pernyataan	No.	Keterangan	Responden	Prosentase (%)
2	1	Sangat Setuju	3	30
	2	Setuju	7	70
	3	Cukup Setuju	0	0
	4	Kurang Setuju	0	0
	5	Tidak Setuju	0	0
Jumlah			10	100

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 3 orang atau 30% menyatakan sangat setuju, 7 orang atau 70% menyatakan setuju bahwa informasi lokasi terbaru untuk mendapatkan produk Maicih.

3. Apakah Anda setuju bahwa *aplikasi tracking* ini akan lebih mempermudah Anda dalam melakukan pencarian untuk membeli produk Maicih di bandingkan mengandalkan akun twitter?

Tabel 4.81 Hasil pengujian kuesioner soal nomor 3

Pernyataan	No.	Keterangan	Responden	Prosentase (%)
3	1	Sangat Setuju	2	20
	2	Setuju	5	50
	3	Cukup Setuju	3	30
	4	Kurang Setuju	0	0
	5	Tidak Setuju	0	0
Jumlah			10	100

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 2 atau 20% menyatakan sangat setuju, 5 orang atau 50 % menyatakan setuju dan, 3 atau 30 % menyatakan cukup setuju bahwa website ini akan lebih memudahkan produk Maicih dibanding mengandalkan twitter.

4. Apakah Anda setuju dengan adanya *aplikasi tracking mobile* ini pemilik dapat meningkatkan penjualan produk Maicih?

Tabel 4.82 Hasil pengujian kuesioner soal nomor 4

Pernyataan	No.	Keterangan	Responden	Prosentase (%)
4	1	Sangat Bagus	5	50
	2	Bagus	4	40
	3	Cukup	1	10
	4	Kurang Bagus	0	0
	5	Tidak Bagus	0	0
Jumlah			10	100

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 5 orang atau 50% menyatakan sangat setuju,4 orang atau 40% menyatakan setuju dan 1 orang atau 10 menyatakan cukup setuju bahwa aplikasi tracking mobile ini akan meningkatkan penjualan produk Maicih

5. Apakah anda setuju bahwa sistem tracking yang digunakan akan lebih memudahkan anda?

Tabel 4.83 Hasil pengujian kuesioner soal nomor 5

Pernyataan	No.	Keterangan	Responden	Prosentase (%)
5	1	Sangat Bagus	3	30
	2	Bagus	4	40
	3	Cukup	3	30
	4	Kurang Bagus	0	0
	5	Tidak Bagus	0	0
Jumlah			10	100

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 3 orang atau 30% menyatakan sangat setuju, 4 orang atau 40% menyatakan setuju dan 3 orang atau 30% menyatakan cukup setuju bahwa sistem *tracking* yang digunakan akan lebih memudahkan.

6. Apakah anda setuju, GPS pada *mobile* anda dapat berjalan dengan baik ?

Tabel 4.84 Hasil pengujian kuesioner soal nomor 6

Pernyataan	No.	Keterangan	Responden	Prosentase (%)
6	1	Sangat Setuju	2	20
	2	Setuju	4	40
	3	Setuju	4	20
	4	Kurang Setuju	0	0
	5	Tidak Setuju	0	0
Jumlah			10	100

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 2 orang atau 20% menyatakan sangat setuju, 4 orang atau 40% menyatakan setuju dan 4 orang atau 40% menyatakan cukup setuju bahwa koneksi internetnya dapat mengakses GPS secara lancar

7. Apakah anda setuju aplikasi mobile android sangat diminati dikalangan masyarakat?

Tabel 4.85 Hasil pengujian kuesioner soal nomor 7

Pernyataan	No.	Keterangan	Responden	Prosentase (%)
7	1	Sangat Setuju	4	40
	2	Setuju	4	40
	3	Setuju	2	20
	4	Kurang Setuju	0	0
	5	Tidak Setuju	0	0
Jumlah			10	100

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 4 orang atau 40% menyatakan sangat setuju, 4 orang atau 40% menyatakan setuju dan 2 orang atau 20% menyatakan cukup setuju aplikasi mobile android diminati dikalangan masyarakat

8. Apakah anda setuju dengan adanya aplikasi ini sebagai media promosi?

Tabel 4.86 Hasil pengujian kuesioner soal nomor 8

Pernyataan	No.	Keterangan	Responden	Prosentase (%)
8	1	Sangat Setuju	2	20
	2	Setuju	4	40
	3	Setuju	2	20
	4	Kurang Setuju	0	0
	5	Tidak Setuju	0	0
Jumlah			10	100

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 2 orang atau 20% menyatakan sangat setuju, 4 orang atau 40% menyatakan setuju dan 2 orang atau 20% menyatakan cukup setuju bahwa sistem *tracking* yang digunakan akan lebih memudahkan.

4.5.2 Wawancara Administrator

Wawancara ini dilakukan dengan menggunakan teknik kualitatif melalui pertanyaan terbuka. Wawancara mengenai *web* administrator dilakukan terhadap 2 orang yaitu panglima jendral dan menteri di PT Maicih Inti Synergi. Dari hasil wawancara tersebut akan dilakukan pengambilan kesimpulan terhadap penilaian penerapan sistem yang baru. Wawancara terdiri dari 5 pertanyaan terbuka yang akan ditanyakan kepada narasumber.

Berikut daftar pertanyaan yang dilakukan kepada narasumber:

1. Apakah Anda yakin bahwa pembangunan *web* administrator di aplikasi ini memudahkan dalam proses penjualan?
2. Menurut Anda, bagaimana tampilan *interface* pada halaman *web* admin ini?

3. Menurut Anda, bagaimana sistem pengolahan data Jendral yang ada di halaman *web* administrator ini?
4. Menurut Anda, bagaimana sistem pengolahan data pelanggan yang ada di halaman *web* administrator ini?
5. Menurut Anda, apakah perlu tambahan pengolahan data yang lain di halaman *web* administrator ini?

Dari hasil wawancara tersebut, menghasilkan jawaban dari narasumber. Rincian hasil jawaban yang diperoleh dari wawancara kepada narasumber terdapat pada tabel sebagai berikut:

Tabel 4.87 Hasil pertanyaan pertama

1. Apakah Anda yakin bahwa pembangunan *web* administrator di aplikasi ini memudahkan dalam proses penjualan?	
Narasumber 1	Yakin, karena proses penjualan di *web*, pengolahan data nya begitu mudah.
Narasumber 2	Yakin, karena pada proses penjualan, saya bisa melihat histori transaksi begitu mudah.
Kesimpulan	Proses penjualan pada *web* administrator memudahkan untuk digunakan

Tabel 4.88 Hasil pertanyaan kedua

2. Menurut Anda, bagaimana tampilan *interface* pada halaman *web* admin ini?	
Narasumber 1	Cukup menarik
Narasumber 2	Sungguh bagus, tampilan nya sesuai dengan tema Maicih
Kesimpulan	Tampilan *interface pada* halaman *web* admin ini cukup menarik

Tabel 4.89 Hasil pertanyaan ketiga

3. Menurut Anda, bagaimana sistem pengolahan data Jendral yang ada di halaman *web* administrator ini?	
Narasumber 1	Bagus, Pengolahan data Jendral memudahkan kita dalam mengatur Jendral siapa aja yang berapa pada proses penjualan
Narasumber 2	Bagus, dengan pengolahan data jendral memudahkan kita mengetahui siapa saja yang mengambil produk pada proses penjualan
Kesimpulan	Pengolahan data jendral pada *web* administrator ini cukup bagus.

Tabel 4.90 Hasil pertanyaan keempat

4. Menurut Anda, bagaimana sistem pengolahan data pelanggan yang ada di halaman *web* administrator ini?

Narasumber 1	Bagus, Pengolahan data pelanggan memudahkan kita untuk mengetahu siapa saja pelanggan kita selama ini
Narasumber 2	Bagus, dengan pengolahan data pelanggan memudahkan kita berinteraksi langsung dengan pelanggan
Kesimpulan	Pengolahan data pelanggan pada *web* administrator ini cukup berguna untuk dapat berinteraksi langsung dengan pelanggan

Tabel 4.91 Hasil pertanyaan kelima

5. Menurut Anda, apakah perlu tambahan pengolahan data yang lain di halaman *web* administrator ini?	
Narasumber 1	Tidak ada
Narasumber 2	Ada, mungkin bisa ditambahkan histori data posisi penjualan jendral yang bertugas
Kesimpulan	Perlu ditambahkan pengolahan data histori posisi penjualan jendral yang sedang bertugas

Berdasarkan hasil kesimpulan dari setiap jawaban yang diberikan oleh narasumber terhadap pertanyaan pengujian beta untuk *web* administrator, dapat ditarik kesimpulan utama yaitu *website* administrator aplikasi tracking penjualan maicih sudah memenuhi tujuan awal pembangunan website tersebut serta *website* administrator tersebut memiliki fungsionalitas yang sesuai dengan kebutuhan pada aplikasi *tracking* penjualan produk maicih. Untuk tampilan antarmuka dari

halaman administrator *website ini*, pengguna menyukai karena memiliki tampilan yang bertema Maicih dan fungsionalitas cukup mudah digunakan dan untuk pengolahan data lain, sebaiknya ditambahkan pengolahan data histori posisi jendral yang sedang bertugas.

4.5.3 Kesimpulan Pengujian Beta

Berdasarkan hasil persentase diatas yang didapatkan dari pengujian beta pengguna administrator, Jendral, maka sistem penerapan aplikasi *tracking* untuk sistem penjualan di PT Maicih Inti Synergi ini sudah sesuai dengan tujuan yaitu:

1. Memberikan informasi bagi vendor ataupun distributor PT MAICIH INTI SYNERGI dan juga bagi pelanggan dalam mengetahui keberadaan semua Jendral yang sedang bertugas secara akurat.
2. Memberikan kemudahan mencari posisi Jendral yang terdekat.
3. Memberikan kemudahan kepada Jendral untuk mengetahui tempat lokasi berjualan yang sedang banyak peminatnya

Selain kesimpulan dari fungsionalitas dan pemenuhan tujuan awal dari pembangunan aplikasi *tracking* tersebut, kesimpulan lain yang diperoleh dari pengujian beta yaitu bahwa *interface* (antarmuka) dari aplikasi *tracking* berbasis *mobile* dan *web* memiliki tampilan yang sederhana pada halaman administrator dan cukup dimengerti untuk aplikasi *mobile* Jendral dan pelanggan sehingga mudah untuk digunakan dan dipelajari. Begitu pun dengan halaman pelanggan yang sesuai dengan target usernya yaitu pelanggan dengan mayoritas berjenis kelamin perempuan sehingga dibuat menarik dan sebagian besar pengguna menyukai *interface* dari aplikasi *tracking* PT. Maicih Inti Synergi.