BAB 4

IMPLEMENTASI DAN PENGUJIAN

Pada bab ini akan dilakukan implementasi dan pengujian terhadap sistem yang baru. Tahapan ini dilakukan setelah perancangan selesai dilakukan dan selanjutnya akan diimplementasikan pada bahasa pemrograman. Setelah implementasi maka dilakukan pengujian terhadap sistem yang baru dan akan dilihat kekurangan-kekurangan pada aplikasi yang baru untuk pengembangan sistem selanjutnya.

4.1 Implementasi Sistem

Tahap implementasi sistem merupakan tahap menterjemahkan perancangan berdasarkan hasil analisis dalam bahasa yang dapat dimengerti oleh mesin serta penerapan perangkat lunak pada keadaan yang sesungguhnya.

4.1.1 Perangkat Keras yang Digunakan

Perangkat keras yang digunakan untuk mengimplementasikan sistem ini adalah sebagai berikut :

a. *Processor* Pentium IV dengan kecepatan 2.8 GHz.
b. RAM dengan kapasitas 512 MB.
c. Hardisk berkapasitas 40 GB.
d. Monitor LCD berukuran 14”.
e. Keyboard.
f. Mouse.

4.1.2 Perangkat Lunak yang Digunakan

Perangkat lunak yang digunakan untuk mengimplementasikan aplikasi ini adalah sebagai berikut :

1. Spesifikasi perangkat lunak mimimal yang dibutuhkan adalah :
   a. Sistem Operasi Windows.
   b. Bahasa pemrograman PHP dan JavaScript.
   c. Web Server XAMPP.
   d. Database Server MySQL.
   e. Web Browser.
   f. Code editor macromedia dreamweaver 8

4.1.3 Implementasi *Database*

Pembuatan *database* dilakukan dengan menggunakan aplikasi *DBMS MySQL*. Implementasi *database* dalam bahasa *SQL* adalah sebagai berikut:

Tabel 4.1 Implementasi Basis data

No	Nama Tabel	Query
1.	Tabel Pegawai	`CREATE TABLE IF NOT EXISTS `petugas_lapangan` ( `nip` varchar(8) NOT NULL, `nama_Admin` varchar(50) DEFAULT NULL, `status` char(1) NOT NULL DEFAULT 'A', `alamat` varchar(50) DEFAULT NULL, `no_handphone` varchar(12) DEFAULT NULL, PRIMARY KEY (`nip`) ) ENGINE=InnoDB DEFAULT CHARSET=latin1;`
2.	Tabel jenis_aset	`CREATE TABLE IF NOT EXISTS `jenis_aset` ( `id_jenis_aset` int(11) NOT NULL AUTO_INCREMENT, `nama_jenis_aset` varchar(50) NOT NULL, PRIMARY KEY (`id_jenis_aset`) ) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=5 ;`
3.	Tabel aset	`CREATE TABLE IF NOT EXISTS `aset` ( `id_aset` int(11) NOT NULL AUTO_INCREMENT, `nama_aset` varchar(25) NOT NULL DEFAULT '', `id_jenis_aset` int(11) NOT NULL,`

No	Nama Tabel	Query
		PRIMARY KEY (`id_aset`), KEY `id_jenis_aset` (`id_jenis_aset`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=27 ; ALTER TABLE `aset` ADD CONSTRAINT `aset_ibfk_1` FOREIGN KEY (`id_jenis_aset`) REFERENCES `jenis_aset` (`id_jenis_aset`);
4.	Tabel jenis_jasa	CREATE TABLE IF NOT EXISTS `jenis_jasa` (`id_jenis_jasa` int(11) NOT NULL AUTO_INCREMENT, `nama_jenis_jasa` varchar(25) DEFAULT NULL, PRIMARY KEY (`id_jenis_jasa`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;
5	Tabel kecamatan	CREATE TABLE IF NOT EXISTS `kecamatan` (`id_kecamatan` varchar(20) NOT NULL, `nama_kecamatan` varchar(25) DEFAULT NULL, PRIMARY KEY (`id_kecamatan`)) ENGINE=InnoDB DEFAULT CHARSET=latin1;
6.	Tabel lokasi	CREATE TABLE IF NOT EXISTS `lokasi` (`id_lokasi` int(11) NOT NULL AUTO_INCREMENT, `lokasi` varchar(25) DEFAULT NULL, `nama` varchar(25) NOT NULL, `id_kecamatan` varchar(20) NOT NULL, PRIMARY KEY (`id_lokasi`), KEY `id_kecamatan` (`id_kecamatan`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=11 ; ALTER TABLE `lokasi` ADD CONSTRAINT `lokasi_ibfk_1` FOREIGN KEY (`id_kecamatan`) REFERENCES `kecamatan` (`id_kecamatan`);
7.	Tabel lokasi_detail	CREATE TABLE IF NOT EXISTS `lokasi_detail` (`ld_idx` int(11) NOT NULL AUTO_INCREMENT, `id_lokasi` int(11) NOT NULL, `id_aset` int(11) NOT NULL, PRIMARY KEY (`ld_idx`), KEY `id_lokasi` (`id_lokasi`), KEY `id_aset` (`id_aset`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=34 ; ALTER TABLE `lokasi_detail` ADD CONSTRAINT `lokasi_detail_ibfk_1` FOREIGN KEY (`id_lokasi`) REFERENCES `lokasi` (`id_lokasi`), ADD CONSTRAINT `lokasi_detail_ibfk_2` FOREIGN KEY (`id_aset`) REFERENCES `aset` (`id_aset`);
8.	Tabel jarak	CREATE TABLE IF NOT EXISTS `jarak` (`id_jarak` int(11) NOT NULL AUTO_INCREMENT, `id_lokasi` int(11) NOT NULL, `jarak` int(11) NOT NULL DEFAULT '0', `waktu` int(11) NOT NULL DEFAULT '0', PRIMARY KEY (`id_jarak`), KEY `lokasi_awal` (`lokasi_awal`),

No	Nama Tabel	Query
		) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=94 ; ALTER TABLE `jarak` ADD CONSTRAINT `jarak_ibfk_1` FOREIGN KEY (`id_lokasi `) REFERENCES `lokasi` (`id_lokasi`),
9.	Tabel admin	CREATE TABLE IF NOT EXISTS `Admin` (`id_admin` varchar(8) NOT NULL, `nama_admin` varchar(25) NOT NULL DEFAULT '', `jabatan` varchar(50) NOT NULL, `username` varchar(25) NOT NULL DEFAULT '', `password` varchar(32) NOT NULL, PRIMARY KEY (`id_Admin`)) ENGINE=InnoDB DEFAULT CHARSET=latin1;
10.	Tabel pengerjaan	CREATE TABLE IF NOT EXISTS `pengerjaan` (`id_pengerjaan` int(11) NOT NULL AUTO_INCREMENT, `id_aset` int(11) NOT NULL, `id_jenis_jasa` int(11) NOT NULL, `waktu_pengerjaan` int(11) NOT NULL, PRIMARY KEY (`id_pengerjaan`), KEY `id_aset` (`id_aset`), KEY `id_jenis_jasa` (`id_jenis_jasa`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=53 ; ALTER TABLE `pengerjaan` ADD CONSTRAINT `pengerjaan_ibfk_1` FOREIGN KEY (`id_aset`) REFERENCES `aset` (`id_aset`), ADD CONSTRAINT `pengerjaan_ibfk_2` FOREIGN KEY (`id_jenis_jasa`) REFERENCES `jenis_jasa` (`id_jenis_jasa`);
11.	Tabel Proses	CREATE TABLE IF NOT EXISTS `proses` (`id_proses` varchar(20) NOT NULL, `gen` smallint(5) unsigned NOT NULL DEFAULT '0', `populasi` smallint(5) unsigned NOT NULL DEFAULT '0', `prob_crossover` varchar(10) NOT NULL DEFAULT '2.5', `prob_mutasi` varchar(10) NOT NULL DEFAULT '0.2', `generasi` int(10) unsigned NOT NULL DEFAULT '1', `hasil_optimasi` int(11) NOT NULL DEFAULT '1', `lokasi_awal` int(11) NOT NULL, `waktu_pengerjaan` int(11) NOT NULL DEFAULT '0', PRIMARY KEY (`id_proses`), KEY `lokasi_awal` (`lokasi_awal`)) ENGINE=InnoDB DEFAULT CHARSET=latin1;
12.	Tabel Aset_awal_proses	CREATE TABLE IF NOT EXISTS `aset_awal_proses` (`id_aap` int(11) NOT NULL AUTO_INCREMENT, `id_proses` varchar(20) NOT NULL,

No	Nama Tabel	Query
		`id_aset` int(11) NOT NULL, `id_jenis_jasa` int(11) NOT NULL, `waktu_pengerjaan` int(11) NOT NULL, PRIMARY KEY (`id_aap`), KEY `id_proses` (`id_proses`), KEY `id_aset` (`id_aset`), KEY `id_jenis_jasa` (`id_jenis_jasa`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=45 ; ALTER TABLE `aset_awal_proses` ADD CONSTRAINT `aset_awal_proses_ibfk_1` FOREIGN KEY (`id_proses`) REFERENCES `proses` (`id_proses`), ADD CONSTRAINT `aset_awal_proses_ibfk_2` FOREIGN KEY (`id_aset`) REFERENCES `aset` (`id_aset`), ADD CONSTRAINT `aset_awal_proses_ibfk_3` FOREIGN KEY (`id_jenis_jasa`) REFERENCES `jenis_jasa` (`id_jenis_jasa`);
13.	Tabel detail_proses	CREATE TABLE IF NOT EXISTS `detail_proses` (`id_detail_proses` varchar(20) NOT NULL, `id_proses` varchar(20) NOT NULL, `id_lokasi` int(11) NOT NULL DEFAULT '0', `waktu_pengerjaan` int(11) NOT NULL DEFAULT '0', PRIMARY KEY (`id_detail_proses`), KEY `id_proses` (`id_proses`), KEY `id_lokasi` (`id_lokasi`)) ENGINE=InnoDB DEFAULT CHARSET=latin1; ALTER TABLE `detail_proses` ADD CONSTRAINT `detail_proses_ibfk_1` FOREIGN KEY (`id_proses`) REFERENCES `proses` (`id_proses`);
14.	Tabel detail_aset_proses	CREATE TABLE IF NOT EXISTS `detail_aset_proses` (`dap_id` int(11) NOT NULL AUTO_INCREMENT, `id_detail_proses` varchar(20) NOT NULL, `id_aset` int(11) NOT NULL, `id_jenis_jasa` int(11) NOT NULL, `waktu_pengerjaan` int(11) NOT NULL, PRIMARY KEY (`dap_id`), KEY `id_detail_proses` (`id_detail_proses`), KEY `id_aset` (`id_aset`), KEY `id_jenis_jasa` (`id_jenis_jasa`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=50 ; ALTER TABLE `detail_aset_proses` ADD CONSTRAINT `detail_aset_proses_ibfk_1` FOREIGN KEY (`id_detail_proses`) REFERENCES `detail_proses` (`id_detail_proses`), ADD CONSTRAINT `detail_aset_proses_ibfk_2` FOREIGN KEY (`id_aset`) REFERENCES `aset` (`id_aset`), ADD CONSTRAINT `detail_aset_proses_ibfk_3` FOREIGN KEY (`id_jenis_jasa`) REFERENCES `jenis_jasa` (`id_jenis_jasa`);

No	Nama Tabel	Query
15.	Tabel Hasil proses	`CREATE TABLE IF NOT EXISTS `hasil_proses` ( `id_hasil_proses` int(11) NOT NULL auto_increment, `id_proses` varchar(20) NOT NULL, `generasi` int(11) NOT NULL default '1', `hasil` text NOT NULL, `jarak_optimal` int(11) NOT NULL default '0', `waktu_optimal` int(11) NOT NULL default '0', `optimal_value` char(1) NOT NULL default 'F', PRIMARY KEY (`id_hasil_proses`), KEY `id_proses` (`id_proses`) ) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=39 ; ADD CONSTRAINT `hasil_proses_ibfk_1` FOREIGN KEY (`id_proses`) REFERENCES `proses` (`id_proses`);`

4.1.4 Implementasi Antarmuka

Implementasi antarmuka dilakukan dengan setiap tampilan program yang dibuat dan pengkodeannya dalam bentuk *file* program. Berikut ini adalah implementasi antarmuka yang dibuat.

Tabel 4.2 Deskripsi *File* Aplikasi

Fungsi	Deskripsi	Proses	File
Login	Untuk menangani proses *login* Admin	1.0 Login	login.php
Data Pegawai	Untuk mengolah data Pegawai seperti tambah Pegawai,edit Pegawai, cari Pegawai , hapus Pegawai	2.1 Pengolahan Data Pegawai	action.php
Data Jenis Aset	Untuk mengolah data Jenis Aset seperti tambah jenis aset,edit jenis aset, cari jenis aset dan hapus jenis aset	2.2 Pengolahan Data Jenis Aset	action.php
Data Aset	Untuk mengolah data Aset seperti tambah Aset, cari Aset, Edit Aset dan hapus Aset	2.3 Pengolahan Data Aset	action.php
Data Jenis Jasa	Untuk mengolah data jenis jasa seperti tambah jenis jasa, edit jenis jasa, cari jenis jasa dan hapus jenis jasa	2.4 Pengolahan Data Jenis Jasa	action.php
Data Kecamatan	Untuk mengolah data kecamatan seperti tambah kecamatan,edit kecamatan, cari kecamatan, dan hapus kecamatan	2.5 Pengolahan Data Kecamatan	action.php
Data Lokasi	Untuk mengolah data lokasi seperti	2.6 Pengolahan	action.php

Fungsi	Deskripsi	Proses	File
	tambah lokasi,edit lokasi, cari lokasi dan hapus lokasi	Data Lokasi	
Data Jarak	Untuk mengolah data jarak seperti tambah jarak,edit jarak, cari jarak dan hapus jarak	2.7 Pengolahan Jarak	jarak.php
Data Admin	Untuk mengolah data Admin seperti edit Admin, dan ubah password	2.8 Pengolahan Data Admin	user.php
Penjadwalan	Untuk memproses penjadwalan	3.0 Proses Penjadwalan	genetik.php
Pengolahan Laporan Penjadwalan	Untuk mengolah data Laporan Penjadwalan	4.1 Pengolahan Laporan Penjadwalan	reporting.php
Pengolahan Laporan Surat Perintah Kerja	Untuk mengolah data Laporan Surat Perintah Kerja	4.2 Pengolahan Laporan Surat Perintah Kerja	reporting.php

Tabel 4.3 Antarmuka Admin

Menu	Deskripsi
- Login	*Login* sebagai Admin
- Menu Utama	Berupa *form* utama
- Logout	Keluar dari program
Pengolahan data master :	
- Data Pegawai	Berupa *form* untuk pengolahan data Pegawai
- Data Jenis Aset	Berupa *form* untuk pengolahan data jenis aset
- Data Aset	Berupa *form* untuk pengolahan data aset
- Data Jenis Jasa	Berupa *form* untuk pengolahan data jenis jasa
- Data Kecamatan	Berupa *form* untuk pengolahan data kecamatan
- Data Lokasi	Berupa *form* untuk pengolahan data lokasi
- Data Jarak	Berupa *form* untuk pengolahan data jarak
- Data Admin	Berupa *form* untuk pengolahan data Admin
Penjadwalan	
- Penjadwalan	Berupa *form* untuk Penjadwalan
Laporan	
- Laporan Penjadwalan	Berupa laporan Penjadwalan
- Surat perintah kerja	Berupa Surat Perintah Kerja

4.2 Pengujian Alpha

Pengujian yang digunakan untuk menguji sistem yang baru adalah metode pengujian *black box*. Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak.

4.2.1 Rencana Pengujian

Rencana pengujian yang akan dilakukan dengan menguji sistem secara *alpha* dan *beta*. Pengujian Aplikasi Penjadwalan Pemeliharaan dan Perbaikan Jaringan Listrik di PT PLN (Persero) APJ Majalaya berikut menggunakan data uji berdasarkan data dari PT PLN (Persero) APJ Majalaya. Rencana pengujian selengkapnya terlihat pada tabel dibawah ini.

Tabel 4.4 Rencana Pengujian Aplikasi Penjadwalan Pemeliharaan dan Perbaikan Jaringan Listrik di PT PLN (Persero) APJ Majalaya

Item Uji	Detail Pengujian	Jenis Pengujian
Login	Verifikasi Username	*Black Box*
	Verifikasi Password	*Black Box*
Data Pegawai	Tambah Pegawai	*Black Box*
	Edit Pegawai	*Black Box*
	Cari Pegawai	*Black Box*
	Hapus Pegawai	*Black Box*
Data Jenis Aset	Tambah Jenis Aset	*Black Box*
	Edit Jenis Aset	*Black Box*
	Cari Jenis Aset	*Black Box*
	Hapus Jenis Aset	*Black Box*
Data Aset	Tambah Aset	*Black Box*
	Edit Aset	*Black Box*
	Cari Aset	*Black Box*
	Hapus Aset	*Black Box*
Data Jenis Jasa	Tambah Jenis Jasa	*Black Box*
	Cari Jenis Jasa	*Black Box*
	Edit Jenis Jasa	*Black Box*
	Hapus Jenis Jasa	*Black Box*
Data Kecamatan	Tambah Kecamatan	*Black Box*
	Cari Kecamatan	*Black Box*
	Edit Kecamatan	*Black Box*
	Hapus Kecamatan	*Black Box*
Data Lokasi	Tambah Lokasi	*Black Box*
	Cari Lokasi	*Black Box*
	Edit Lokasi	*Black Box*
	Hapus Lokasi	*Black Box*
Data Jarak	Tambah Jarak	*Black Box*
	Cari Jarak	*Black Box*
	Edit Jarak	*Black Box*
	Hapus Jarak	*Black Box*
Data Admin	Edit Admin	*Black Box*
	Ganti Password	*Black Box*

Penjadwalan	Proses	*Black Box*
	Pencetakan hasil penjadwalan	*Black Box*
	Pencetakan Surat Perintah Kerja	*Black Box*
Laporan Penjadwalan	Cari Jadwal	*Black Box*
	Pencetakan Jadwal	*Black Box*
Laporan Surat Perintah Kerja	Cari Surat Perintah Kerja	*Black Box*
	PencetakanSurat Perintah Kerja	*Black Box*
Logout	Logout	*Black Box*

4.2.2 Kasus dan Hasil Pengujian (Pengujian Alpha)

Berdasarakan rencana pengujian yang telah disusun, maka dapat dilakukan pengujian sebagai berikut.

4.2.2.1 Pengujian Login

Tabel 4.5 Pengujian Login Admin

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Username : admin	Tercantum pada *text field username*	Dapat mengisi data *username*. sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Password : admin	Tercantum pada *text field Password*	Dapat mengisi data *password*. sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik tombol Login	Data login dicari di *database* tabel *username*, login valid dan masuk ke menu utama	Tombol login dapat berfungsi. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik tombol reset	Menghapus *username* dan *password* yang tidak dikehendaki	Tombol reset dapat berfungsi. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data kosong)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
username : *Password* : (kosong)	Tidak dapat login dan menampilkan pesan“Username atau Password salah”	User tidak dapat login dan memberikan pesan “Username atau Password salah” Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
username : aaa *Password* : aaa (tidak terdaftar)	Tidak dapat login dan menampilkan pesan“Username atau Password salah”	User tidak dapat login dan memberikan pesan “Username atau Password salah” Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak

4.2.2.2 Pengujian Pengolahan Data Pegawai

Tabel 4.6 Pengujian Pengolahan Pegawai

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik Data Pegawai	Menampilkan data Pegawai	Dapat menampilkan data Pegawai. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik Tambah	Menampilkan *form* tambah Pegawai	Dapat menampilkan *form* tambah Pegawai. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik cari	Menampilkan data Pegawai yang dicari	Dapat menampilkan data Pegawai yang dicari. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik Edit	Menampilkan *Form* edit Pegawai	Dapat menampilkan *form* edit Pegawai. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik hapus	Menampilkan Pesan data Pegawai yang akan dihapus	Dapat menampilkan pesan Pegawai yang akan dihapus. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak

4.2.2.3 Pengujian Tambah Pegawai

Tabel 4.7 Pengujian Tambah Pegawai

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik Tambah	Menampilkan *form* tambah Pegawai	Dapat menampilkan *form* tambah Pegawai. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Mengisi Data Pegawai Contoh: NIP: 6993093L Nama Pegawai: Toto Sumiarto Alamat: Jln Cibaduyut No.122 No. Hanphone: 085624204670	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 Pegawai akan tersimpan didalam *database* dan muncul pesan ”data berhasil di simpan”. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Kosong)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**

NIP tidak diisi (kosong)	Menampilkan pesan kesalahan	Dapat menampilkan pesan " silahkan cek kembali data inputan anda". Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
NIP diisi 10 digit	Menampilkan pesan kesalahan	Dapat menampilkan pesan " silahkan cek kembali data inputan anda". Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak

4.2.2.4 Pengujian Tambah Pegawai

Tabel 4.8 Pengujian Edit Pegawai

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
NIP 6993093L	NIP tidak dapat di edit	NIP tidak dapat di edit. Sesuai yang diharapkan	[√] Diterima [ ] Ditolak
Nama Pegawai: Toto Sumiarto	diisi pada *text field*	*Text field* dapat berfungsi. Sesuai yang diharapkan	[√] Diterima [ ] Ditolak
Alamat: Jln Cibaduyut No.122	diisi pada *text field*	*Text field* dapat berfungsi. Sesuai yang diharapkan	[√] Diterima [ ] Ditolak
No. Handphone: 085624204670	diisi pada *text field*	*Text field* dapat berfungsi. Sesuai yang diharapkan	[√] Diterima [ ] Ditolak
Klik tombil simpan	Edit data Pegawai muncul pesan "edit Pegawai sukses"	Tombol simpan berfungsi. Sesuai yang diharapkan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
NIP diisi 10 digit	Menampilkan pesan kesalahan	Dapat menampilkan pesan " silahkan cek kembali data inputan anda". Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak

4.2.2.5 Pengujian Pengolahan Data Jenis Aset

Tabel 4.9 Pengujian pengolahan data jenis aset

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik Data Jenis Aset	Menampilkan data jenis aset	Dapat menampilkan data jenis aset. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak

Klik Tambah Jenis Aset	Menampilkan *form* tambah jenis aser	Dapat menampilkan *form* tambah jenis aset. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik cari	Menampilkan data jenis aset yang dicari	Dapat menampilkan data jenis aset yang dicari. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik Edit	Menampilkan *Form* edit jenis aset	Dapat menampilkan *form* edit jenis aset. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik hapus	Menampilkan Pesan data jenis aset yang akan dihapus	Dapat menampilkan pesan jenis aset yang akan dihapus. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak

4.2.2.6 Pengujian Tambah Jenis Aset

Tabel 4.10 Pengujian Tambah Jenis Aset

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik Tambah	Menampilkan *form* tambah jenis aset	Dapat menampilkan *form* tambah jenis aset. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Mengisi Data jenis aset Contoh: Jenis Aset : Travo	Jika 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 field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Klik Tombol Simpan	Data pengiriman akan tersimpan didalam *database*	Data jenis aset akan tersimpan didalam *database* dan muncul pesan ”data berhasil di simpan”. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Jenis aset tidak diisi (kosong)	Menampilkan pesan kesalahan	Dapat menampilkan pesan “ silahkan cek kembali data inputan anda”. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak

4.2.2.7 Pengujian Edit Data Jenis Aset

Tabel 4.11 Pengujian Edit Jenis Aset

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengubah Data jenis aset Contoh: Jenis Aset : Travo	Jika 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 field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Klik Tombol Simpan	Data pengiriman akan tersimpan didalam *database*	Data jenis aset akan tersimpan didalam *database* dan muncul pesan ”data berhasil di simpan”. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Jenis aset tidak diisi (kosong)	Menampilkan pesan kesalahan	Dapat menampilkan pesan “ silahkan cek kembali data inputan anda”. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak

4.2.2.8 Pengujian Pengolahan Data Aset

Tabel 4.12 Pengujian Pengolahan Data Aset

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik data aset	Menampilkan data aset	Dapat menampilkan data aset. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik tambah aset	Menampilkan *form* tambah aset	Dapat menampilkan *form* tambah aset. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik cari	Menampilkan data aset yang dicari	Dapat menampilkan data aset yang dicari. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik Edit	Menampilkan *Form* edit aset	Dapat menampilkan *form* edit aset. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik hapus	Menampilkan Pesan data aset yang akan dihapus	Dapat menampilkan pesan aset yang akan dihapus. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak

4.2.2.9 Pengujian Tambah Aset

Tabel 4.13 Pengujian Tambah Aset

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik Tambah	Menampilkan *form* tambah aset	Dapat menampilkan *form* tambah aset. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Mengisi Data aset Contoh: Jenis Aset : Travo 315 KV	Jika 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 field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Klik Tombol Simpan	Data pengiriman akan tersimpan didalam *database*	Data aset akan tersimpan didalam *database* dan muncul pesan ”data berhasil di simpan”. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Aset tidak diisi (kosong)	Menampilkan pesan kesalahan	Dapat menampilkan pesan “ silahkan cek kembali data inputan anda”. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak

4.2.2.10 Pengujian Edit Data Aset

Tabel 4.14 Pengujian Tambah Aset

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Jenis Aset : Travo 315 KV	Jika 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 field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Klik Tombol Simpan	Data pengiriman akan tersimpan didalam *database*	Data aset akan tersimpan didalam *database* dan muncul pesan ”data berhasil di simpan”. Sesuai yang	[√] Diterima [ ] Ditolak

		diharapkan.	
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Aset tidak diisi (kosong)	Menampilkan pesan kesalahan	Dapat menampilkan pesan " silahkan cek kembali data inputan anda". Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak

4.2.2.11 Pengujian Pengolahan Data Jenis Jasa

Tabel 4.15 Pengujian Pengolahan Data Jenis Jasa

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik Data Jenis Jasa	Menampilkan data jenis jasa	Dapat menampilkan data jenis jasa. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik Tambah Jenis Jasa	Menampilkan *form* tambah jenis jasa	Dapat menampilkan *form* tambah jenis aset. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik cari	Menampilkan data jenis jasa yang dicari	Dapat menampilkan data jenis jasa yang dicari. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik Edit	Menampilkan *Form* edit jenis jasa	Dapat menampilkan *form* edit jenis jasa. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik hapus	Menampilkan Pesan data jenis jasa yang akan dihapus	Dapat menampilkan pesan jenis jasa yang akan dihapus. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak

4.2.2.12 Pengujian Tambah Jenis Jasa

Tabel 4.16 Pengujian Tambah Jenis Jasa

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik Tambah	Menampilkan *form* tambah jenis jasa	Dapat menampilkan *form* tambah jenis jasa. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Mengisi Data jenis Jasa Contoh: Jenis Jasa : Pemeliharaan	Jika 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 field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Klik Tombol	Data pengiriman akan	Data jenis jasa akan	[√] Diterima

Simpan	tersimpan didalam *database*	tersimpan didalam *database* dan muncul pesan ”data berhasil di simpan”. Sesuai yang diharapkan.	[ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Jenis Jasa tidak diisi (kosong)	Menampilkan pesan kesalahan	Dapat menampilkan pesan “ silahkan cek kembali data inputan anda”. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak

4.2.2.13 Pengujian Edit Data Jenis Jasa

Tabel 4.17 Pengujian Edit Jenis Jasa

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik Edit	Menampilkan *form* edit jenis jasa	Dapat menampilkan *form* edit jenis jasa. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Mengisi Data jenis Jasa Contoh: Jenis Jasa : Pemeliharaan	Jika 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 field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Klik Tombol Simpan	Data pengiriman akan tersimpan didalam *database*	Data jenis jasa akan tersimpan didalam *database* dan muncul pesan ”data berhasil di simpan”. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Jenis Jasa tidak diisi (kosong)	Menampilkan pesan kesalahan	Dapat menampilkan pesan “ silahkan cek kembali data inputan anda”. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak

4.2.2.14 Pengujian Pengolahan Data Kecamatan

Tabel 4.18 Pengujian Pengolahan Data Kecamatan

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik Data Kecamatan	Menampilkan data kecamatan	Dapat menampilkan data kecamatan. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik Tambah Kecamatan	Menampilkan *form* tambah kecamatan	Dapat menampilkan *form* tambah kecamatan. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik cari	Menampilkan data kecamatan yang dicari	Dapat menampilkan data kecamatan yang dicari. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik Edit	Menampilkan *Form* edit kecamatan	Dapat menampilkan *form* edit kecamatan. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik hapus	Menampilkan Pesan data kecamatan yang akan dihapus	Dapat menampilkan pesan kecamatan yang akan dihapus. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak

4.2.2.15 Pengujian Tambah Kecamatan

Tabel 4.19 Pengujian Tambah Kecamatan

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik Tambah	Menampilkan *form* tambah kecamatan	Dapat menampilkan *form* tambah kecamatan. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Mengisi Data Kecamatan Contoh: Kecamatan : Majalaya	Jika 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 field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Klik Tombol Simpan	Data pengiriman akan tersimpan didalam *database*	Data kecamatan akan tersimpan didalam *database* dan muncul pesan ”data berhasil di simpan”. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Kosong)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Kecamatan tidak diisi (kosong)	Menampilkan pesan kesalahan	Dapat menampilkan pesan “ silahkan cek	[√] Diterima [ ] Ditolak

		kembali data inputan anda". Sesuai yang diharapkan.	
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Kecamatan : Banjaran (data ini sudah terdapat dalam *database*)	Menampilkan pesan kesalahan	Dapat menampilkan pesan " Data Sudah tersedia". Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak

4.2.2.16 Pengujian Edit Data Kecamatan

Tabel 4.20 Pengujian Edit Kecamatan

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik Edit	Menampilkan *form* edit kecamatan	Dapat menampilkan *form* edit kecamatan. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
mengubah Data Kecamatan Contoh: Kecamatan : Majalaya	Jika 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 field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Klik Tombol Simpan	Data pengiriman akan tersimpan didalam *database*	Data kecamatan akan tersimpan didalam *database* dan muncul pesan "data berhasil di simpan". Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Kosong)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Kecamatan tidak diisi (kosong)	Menampilkan pesan kesalahan	Dapat menampilkan pesan " silahkan cek kembali data inputan anda". Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Kecamatan : Banjaran (data ini sudah terdapat dalam *database*)	Menampilkan pesan kesalahan	Dapat menampilkan pesan " Data Sudah tersedia". Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak

4.2.2.17 Pengujian Pengolahan Data Lokasi

Tabel 4.21 Pengujian Pengolahan Data Lokasi

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik Data Lokasi	Menampilkan data lokasi	Dapat menampilkan data lokasi. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik Tambah Lokasi	Menampilkan *form* tambah lokasi	Dapat menampilkan *form* tambah lokasi. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik cari	Menampilkan data lokasi yang dicari	Dapat menampilkan data lokasi yang dicari. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik Edit	Menampilkan *Form* edit lokasi	Dapat menampilkan *form* edit lokasi. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik Hapus	Menampilkan lokasi yang akan dihapus	Dapat menampilkan lokasi yang akan di hapus. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak

4.2.2.18 Pengujian Tambah Lokasi

Tabel 4.22 Pengujian Tambah Lokasi

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik Tambah	Menampilkan *form* tambah lokasi	Dapat menampilkan *form* tambah lokasi. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Mengisi Data Lokasi Contoh: Lokasi : MJLY Nama : BIPE Kecamatan: Majalaya	Jika 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 field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Klik Tombol Simpan	Data pengiriman akan tersimpan didalam *database*	Data lokasi akan tersimpan didalam *database* dan muncul pesan ”data berhasil di simpan”. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
lokasi tidak diisi (kosong)	Menampilkan pesan kesalahan	Dapat menampilkan pesan “ silahkan cek	[√] Diterima [ ] Ditolak

		kembali data inputan anda". Sesuai yang diharapkan.	
lokasi diisi lokasi yang sama	Menampilkan pesan kesalahan	Dapat menampilkan pesan " data yang di tambah sudah tersedia, silahkan cek kembali data inputan anda". Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak

4.2.2.19 Pengujian Edit Data Lokasi

Tabel 4.23 Pengujian Edit Lokasi

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik Edit	Menampilkan *form* edit lokasi	Dapat menampilkan *form* edit lokasi. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Mengisi Data Lokasi Contoh: Lokasi : MJLY Nama : BIPE Kecamatan: Majalaya	Jika 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 field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Klik Tombol Simpan	Data pengiriman akan tersimpan didalam *database*	Data lokasi akan tersimpan didalam *database* dan muncul pesan "data berhasil di simpan". Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
lokasi tidak diisi (kosong)	Menampilkan pesan kesalahan	Dapat menampilkan pesan " silahkan cek kembali data inputan anda". Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
lokasi diisi lokasi yang sama	Menampilkan pesan kesalahan	Dapat menampilkan pesan " data yang di tambah sudah tersedia, silahkan cek kembali data inputan anda". Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak

4.2.2.20 Pengujian Pengolahan Data Jarak

Tabel 4.24 Pengujian Pengolahan Data Jarak

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik Data Jarak	Menampilkan data jarak	Dapat menampilkan data jarak. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik Tambah Jarak	Menampilkan *form* tambah jarak	Dapat menampilkan *form* tambah jarak. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik cari	Menampilkan data jarak yang dicari	Dapat menampilkan data jarak yang dicari. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik Edit	Menampilkan *Form* edit jarak	Dapat menampilkan *form* edit jarak. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik Hapus	Menampilkan hapus jarak	Dapat menampilkan jarak yang akan di hapus. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak

4.2.2.21 Pengujian Tambah Jarak

Tabel 4.25 Pengujian Tambah Jarak

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik Tambah	Menampilkan *form* tambah jarak	Dapat menampilkan *form* tambah jarak. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Mengisi Data Jarak Contoh: Lokasi Awal : BIPE Lokasi Akhir : CWDY Jarak: 500 Waktu: 50 Menit	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 field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Klik Tombol Simpan	Data pengiriman akan tersimpan didalam *database*	Data jarak akan tersimpan didalam *database* dan muncul pesan ”data berhasil di simpan”. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Kosong)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**

Jarak tidak diisi (kosong)	Menampilkan pesan kesalahan	Dapat menampilkan pesan " silahkan cek kembali data inputan anda". Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Lokasi Awal dan Lokasi Akhir diisi data yang sama Lokasi Awal: BIPE Lokasi Akhir : BIPE	Menampilkan pesan kesalahan	Dapat menampilkan pesan " Data yang ditambahkan sudah tersedia, silahkan cek kembali data inputan anda". Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak

4.2.2.22 Pengujian Edit Data Jarak

Tabel 4.26 Pengujian Edit Jarak

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik Edit	Menampilkan *form* edit jarak	Dapat menampilkan *form* edit jarak. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Mengisi Data Jarak Contoh: Lokasi Awal : BIPE Lokasi Akhir : CWDY Jarak: 500 Waktu: 50 Menit	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 field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Klik Tombol Simpan	Data pengiriman akan tersimpan didalam *database*	Data jarak akan tersimpan didalam *database* dan muncul pesan "data berhasil di simpan". Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Kosong)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Jarak tidak diisi (kosong)	Menampilkan pesan kesalahan	Dapat menampilkan pesan " silahkan cek kembali data inputan anda". Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			

Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Data Lokasi Awal dan Lokasi Akhir diisi data yang sama Lokasi Awal: BIPE Lokasi Akhir : BIPE	Menampilkan pesan kesalahan	Dapat menampilkan pesan " Data yang ditambahkan sudah tersedia, silahkan cek kembali data inputan anda". Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak

4.2.2.23 Pengujian Pengolahan Data Admin

Tabel 4.27 Pengujian Pengolahan Data Admin

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik Edit	Menampilkan *Form* Admin	Dapat menampilkan *form* Admin. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik Ubah *Password*	Menampilkan Ubah *Password*	Dapat menampilkan ubah *password*. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak

4.2.2.24 Pengujian Penjadwalan

Tabel 4.28 Pengujian Penjadwalan

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik Penjadwalan	Menampilkan *form* penjadwalan.	Dapat menampilkan *form* penjadwalan. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik tombol Proses Penjadwalan	Menampilkan proses penjadwalan dan hasil dari penjadwalan.	Dapat menampilkan hasil penjadwalan. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak

4.2.2.25 Pengujian Laporan Penjadwalan

Tabel 4.29 Pengujian Laporan Penjadwalan

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Isi data filter yang dipilih (Tanggal)	Data filter terisi.	Data filter terisi. Sesuai dengan yang diharapkan.	[√] Diterima [ ] Ditolak
Klik tombol Cari	Jika data yang dicari	Jika data yang dicari	[√] Diterima

	ditemukan, tampilkan laporan berdasarkan kriteria yang telah difilter. Jika data tidak ditemukan menampilkan pesan "Data tidak ditemukan!".	ditemukan, tampilkan laporan berdasarkan kriteria yang telah difilter. Jika data tidak ditemukan menampilkan pesan "Data tidak ditemukan!". Sesuai yang diharapkan.	[ ] Ditolak
Klik tombol Cetak	Mencetak hasil laporan.	Mencetak hasil laporan. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak

Tabel 4.30 Pengujian Laporan Penjadwalan (Lanjutan)

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Pencarian kata tidak diisi	Menampilkan pesan peringatan "Data filter belum diisi".	Menampilkan pesan peringatan "Data filter belum diisi".Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak

4.2.2.26 Pengujian Laporan Surat Perintah Kerja

Tabel 4.31 Pengujian Laporan Surat Perintah Kerja

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Isi data filter yang dipilih (Tanggal)	Data filter terisi.	Data filter terisi. Sesuai dengan yang diharapkan.	[√] Diterima [ ] Ditolak
Klik tombol Cari	Jika data yang dicari ditemukan, tampilkan laporan berdasarkan kriteria yang telah difilter. Jika data tidak ditemukan menampilkan pesan "Data tidak ditemukan!".	Jika data yang dicari ditemukan, tampilkan laporan berdasarkan kriteria yang telah difilter. Jika data tidak ditemukan menampilkan pesan "Data tidak ditemukan!". Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik tombol Cetak	Mencetak hasil laporan.	Mencetak hasil laporan. Sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Pencarian kata kunci tidak diisi	Menampilkan pesan peringatan "Data	Menampilkan pesan peringatan "Data filter	[√] Diterima [ ] Ditolak

	filter belum diisi".	belum diisi".Sesuai yang diharapkan.	

4.2.2.27 Pengujian Logout

Tabel 4.32 Pengujian Logout

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik logout	Tampilan akan kembali ke login aplikasi	Link logout berfungsi sesuai dengan yang diharapkan.	[√] Diterima [ ] Ditolak

4.2.3 Kesimpulan Hasil Pengujian Alpha

Berdasarkan hasil pengujian dengan kasus *sample* uji yang telah dilakukan memberikan kesimpulan bahwa secara fungsional sistem sudah dapat menghasilkan output yang diharapkan.

4.2.4 Kasus Dan Hasil Pengujian (Pengujian Beta)

Pengujian beta merupakan pengujian yang dilakukan secara objektif dimana diuji secara langsung ke lapangan yaitu instansi yang bersangkutan dengan wawancara langsung kepada *user* yang bersifat pertanyaan mengenai kepuasan *user*. *User* tersebut Admin.

Hasil pengujian terhadap Admin

1. Menurut anda, apakah hasil jadwal dari aplikasi penjadwalan pemeliharaan dan perbaikan jaringan listrik yang telah dibuat sudah optimal dalam hal jarak dan waktu pengerjaan?

   Berdasarkan hasil wawancara di atas maka dapat disimpulkan bahwa aplikasi yang dibuat sudah optimal dalam hal jarak dan waktu.

2. Menurut anda, apakah aplikasi penjadwalan pemeliharaan dan perbaikan jaringan listrik yang telah dibuat menjadikan pemeliharaan dan perbaikan jaringan listrik terkontrol?

   Berdasarkan hasil wawancara di atas maka dapat disimpulkan bahwa aplikasi yang dibuat menjadikan pemeliharaan dan perbaikan jaringan listrik menjadi terkontrol.

3. Menurut anda, apakah aplikasi penjadwalan pemeliharaan dan perbaikan jaringan listrik dapat mempermudah dan mempercepat proses pengolahan data yang berhubungan dengan pemeliharaan dan perbaikan jaringan listrik?

   Berdasarkan hasil wawancara di atas maka dapat disimpulkan bahwa aplikasi yang dibuat dapat mempermudah dan mempercepat proses pengolahan data yang berhubungan dengan pemeliharaan dan perbaikan jaringan listrik.

4. Menurut Anda, apakah informasi penjadwalan yang dihasilkan mudah dimengerti dan sesuai dengan yang diharapkan?

   Berdasarkan hasil wawancara di atas maka dapat disimpulkan bahwa aplikasi yang dibuat informasi penjadwalan yang dihasilkan mudah dimengerti dan sesuai dengan yang diharapkan.

5. Menurut Anda, apakah fungsionalitas aplikasi ini memenuhi kebutuhan?

   Berdasarkan hasil wawancara di atas maka dapat disimpulkan bahwa aplikasi yang dibuat memenuhi kebutuhan.

6. Menurut Anda, Apakah tampilan (antarmuka) aplikasi ini terlihat menarik?

   Berdasarkan hasil wawancara di atas maka dapat disimpulkan bahwa aplikasi yang dibuat terlihat sangat menarik.

7. Menurut Anda, Apakah aplikasi ini mudah digunakan?

   Berdasarkan hasil wawancara di atas maka dapat disimpulkan bahwa aplikasi yang dibuat mudah digunakan.

8. Menurut Anda, apakah aplikasi ini bermanfaat?

   Berdasarkan hasil wawancara di atas maka dapat disimpulkan bahwa aplikasi yang dibuat sangat bermanfaat.

4.2.5 Kesimpulan Hasil Pengujian Beta

Dari hasil pengujian beta yang telah dilakukan yaitu dengan wawancara langsung kepada *user* yang bersifat pertanyaan di lapangan diperoleh kesimpulan bahwa aplikasi penjadwalan ini dinilai sudah optimal dalam hal jarak dan waktu, menjadikan pemeliharaan dan perbaikan jaringan listrik menjadi terkontrol, dapat mempermudah dan mempercepat proses pengolahan data yang berhubungan dengan pemeliharaan dan perbaikan jaringan listrik, informasi penjadwalan yang dihasilkan mudah dimengerti dan sesuai dengan yang diharapkan, memenuhi kebutuhan, terlihat sangat menarik, mudah digunakan dan sangat bermanfaat.