BAB 4
IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi

Implementasi dilakukan setelah perancangan selesai dilakukan dan selanjutnya akan diimplementasikan pada bahasa pemrograman yang akan digunakan. Tujuan Implementasi adalah untuk mengkonfirmasikan modul program perancangan pada para pelaku sistem sehingga *user* dapat memberi masukan kepada pembangun sistem.

4.2.1 Perangkat Lunak Pendukung

Perangkat lunak yang digunakan untuk membangun sistem informasi geografis ini dapat dilihat pada Tabel 4.1

Tabel 4.1 Perangkat Lunak Pendukung

Spesifikasi	Perangkat Lunak Yang dibutuhkan
Sistem Operasi	Windows 8
Database	MySql
Browser	Firefox modzila 21.0 ,Google chrome
Sourcemap	MapServer, pmapper, Macromedia Dreamweaver 8 , Notepad ++ ,quantum gis

4.2.2 Perangkat Keras Pendukung

Kebutuhan minimum perangkat keras yang diperlukan untuk mengimplementasikan aplikasi sistem informasi geografis dapat dilihat pada Tabel 4.2

Tabel 4.2 Perangkat Keras Pendukung

Perangkat	Spesifikasi Kebutuhan Minimum
Prosesor	*Intel® Core™2*
Memory	1 GB
VGA Card	512 MB
Harddisk	80 GB
Monitor	Monitor 16”, Resolusi 800 x 600
Keyboard	Standar

Perangkat	Spesifikasi Kebutuhan Minimum
Mouse	Standar
Printer	Epson L800

Peralatan jaringan computer yang digunakan adalah :

1. Switch
2. Kabel *LAN*

4.2.3 Impelematasi Basis Data

Implementasi basis data diambil berdasarkan perancangan basis data yang dibuat sebelumnya, Tabel-tabel berikut menggambarkan struktur tabel yang diimplementasikan pada basis data dapat dilihat pada Tabel 4.3

Tabel 4.3 Perangkat Keras Pendukung

No	Tabel	SQL
1	Desa	CREATE TABLE IF NOT EXISTS `desa` (`id_desa` int(11) NOT NULL AUTO_INCREMENT, `nama_desa` varchar(50) NOT NULL, `id_kecamatan` int(11) NOT NULL, PRIMARY KEY (`id_desa`), KEY `id_kecamatan` (`id_kecamatan`)) ENGINE=InnoDB -- -- Constraints for table `desa` -- ALTER TABLE `desa` ADD CONSTRAINT `desa_ibfk_2` FOREIGN KEY (`id_kecamatan`) REFERENCES `kecamatan` (`id_kecamatan`) ON DELETE CASCADE ON UPDATE CASCADE;
2	Detail_ternak	CREATE TABLE IF NOT EXISTS `detail_ternak` (`id_detail` int(11) NOT NULL AUTO_INCREMENT, `bulan` varchar(10) NOT NULL, `tahun` int(11) NOT NULL, `jumlah` int(11) NOT NULL, `id_hewan` int(11) NOT NULL, PRIMARY KEY (`id_detail`), KEY `id_hewan` (`id_hewan`)) ENGINE=InnoDB -- -- Constraints for table `detail_ternak` -- ALTER TABLE `detail_ternak` ADD CONSTRAINT `detail_ternak_ibfk_1` FOREIGN KEY (`id_hewan`) REFERENCES `hewan_ternak` (`id_hewan`) ON DELETE CASCADE ON UPDATE CASCADE, ADD CONSTRAINT `detail_ternak_ibfk_3` FOREIGN

No	Tabel	SQL
		KEY (`id_kecamatan`) REFERENCES `kecamatan` (`id_kecamatan`) ON DELETE CASCADE ON UPDATE CASCADE;
3	Hewan_ternak	CREATE TABLE IF NOT EXISTS `hewan_ternak` (`id_hewan` int(11) NOT NULL AUTO_INCREMENT, `nama_hewan` varchar(50) NOT NULL, `jk` varchar(10) NOT NULL, `jumlah` varchar(50) NOT NULL, PRIMARY KEY (`id_hewan`)) ENGINE=InnoDB
4	Kecamatan	CREATE TABLE IF NOT EXISTS `kecamatan` (`id_kecamatan` int(11) NOT NULL AUTO_INCREMENT, `nama_kecamatan` varchar(50) NOT NULL, PRIMARY KEY (`id_kecamatan`)) ENGINE=InnoDB ALTER TABLE `kecamatan` ADD CONSTRAINT `kecamatan_ibfk_1` FOREIGN KEY (`FeatureID`) REFERENCES `features0` (`FeatureID`) ON UPDATE CASCADE;
5	Kelompok_ternak	CREATE TABLE IF NOT EXISTS `kelompok_ternak` (`id_kelompokternak` int(11) NOT NULL AUTO_INCREMENT, `nama_kelompok` varchar(50) NOT NULL, `ketua_kelompok` varchar(50) NOT NULL, `alamat` varchar(100) NOT NULL, `id_desa` int(11) NOT NULL, `id_kecamatan` int(11) NOT NULL, `jumlah_ternak` varchar(50) NOT NULL, PRIMARY KEY (`id_kelompokternak`)) ENGINE=InnoDB
6	Layers	CREATE TABLE IF NOT EXISTS `layers` (`LayerID` int(11) NOT NULL, `Shape` char(10) DEFAULT NULL, `Name` char(50) DEFAULT NULL, `FillColor` char(17) DEFAULT NULL, `StrokeColor` char(17) DEFAULT NULL, `StrokeWidth` double DEFAULT NULL, `MinScale` int(11) DEFAULT NULL, `MaxScale` int(11) DEFAULT NULL, `HasAttributes` tinyint(4) DEFAULT NULL, `NameField` char(50) DEFAULT NULL, `LabelColor` char(15) DEFAULT NULL, `LabelSize` int(11) DEFAULT NULL, `IDType` char(10) DEFAULT NULL, `IDField` char(50) DEFAULT NULL, `URL` char(70) DEFAULT NULL, `UseLabels` tinyint(4) DEFAULT NULL, `IsExternal` tinyint(4) DEFAULT NULL, `LegendType` char(50) DEFAULT NULL, `ClassificationField` char(50) DEFAULT NULL, PRIMARY KEY (`LayerID`)) ENGINE=InnoDB
7	Peramalan	CREATE TABLE IF NOT EXISTS `peramalan` (

No	Tabel	SQL
		`id_peramalan` int(11) NOT NULL AUTO_INCREMENT, `hasil` varchar(50) NOT NULL, `target` float NOT NULL, `target_sebelum` float NOT NULL, `id_periode` int(11) NOT NULL, `id_hewan` int(11) NOT NULL, PRIMARY KEY (`id_peramalan`), KEY `id_periode` (`id_periode`), KEY `id_hewan` (`id_hewan`)) ENGINE=InnoDB -- -- Constraints for table `produksi` -- ALTER TABLE `peramalan` ADD CONSTRAINT `peramalan_ibfk_2` FOREIGN KEY (`id_periode`) REFERENCES `periode` (`id_periode`) ON DELETE CASCADE ON UPDATE CASCADE;
8	Periode	CREATE TABLE IF NOT EXISTS `periode` (`id_periode` int(11) NOT NULL AUTO_INCREMENT, `periode` int(11) NOT NULL, PRIMARY KEY (`id_periode`)) ENGINE=InnoDB
9	*User*	CREATE TABLE IF NOT EXISTS `*user*` (`id_*user*` int(11) NOT NULL AUTO_INCREMENT, `*user*name` varchar(50) NOT NULL, `*Password*` varchar(50) NOT NULL, `nama_lengkap` varchar(50) NOT NULL, `email` varchar(50) NOT NULL, PRIMARY KEY (`id_*user*`)) ENGINE=InnoDB
10	produksi	CREATE TABLE IF NOT EXISTS `produksi` (`id_produksi` int(11) NOT NULL AUTO_INCREMENT, `jenis_produksi` varchar(50) NOT NULL, `jumlah` varchar(50) NOT NULL, PRIMARY KEY (`id_produksi`)) ENGINE=InnoDB -- -- Constraints for table `produksi` -- ALTER TABLE `produksi` ADD CONSTRAINT `produksi_ibfk_1` FOREIGN KEY (`id_periode`) REFERENCES `periode` (`id_periode`) ON DELETE CASCADE ON UPDATE CASCADE;

4.2.4 Implementasi Antarmuka

Berikut ini adalah implementasi antarmuka pada sistem informasi geografis yang telah dibangun.

4.2.1.1 Antarmuka Admin

Berikut adalah implementasi antarmuka untuk admin dapat dilihat pada Tabel 4.4

Tabel 4.4 Antarmukan Admin

Menu	Deskripsi
Login	Untuk menangani proses login admin
Beranda	Tampilan awal antarmuka admin
Manajemen *User*	Tampilan antarmuka untuk melihat,menambah,dan mengedit *user*.
Manejemen peta	Tampilan antarmuka untuk melihat data peta
Manajemen kelompok ternak	Tampilan antarmuka untuk mengelola data kelompok ternak, seperti menambah, menghapus , meng*Edit* dan memperbaharui data.
Manajemen hewan ternak	Tampilan antarmuka untuk mengelola data hewan ternak, seperti menambah, menghapus , meng*Edit* dan memperbaharui data.
Manajemen detail ternak	Tampilan antarmuka untuk mengelola data detail ternak, seperti menambah, menghapus , meng*Edit* dan memperbaharui data.
Manajemen kecamatan dan desa	Tampilan antarmuka untuk mengelola data kecamatan dan desa, seperti menambah, menghapus , meng*Edit* dan memperbaharui data.
Manajemen Produksi	Tampilan antarmuka untuk mengelola data produksi, seperti menambah, menghapus , meng*Edit* dan memperbaharui data.
Manejemen peramalan	Tampilan antarmuka untuk mengelola data peramalan, seperti proses dalam peramalan.
Keluar	Untuk proses keluar dari halaman

4.2.1.2 Antarmuka Pimpinan

Berikut adalah implementasi antarmuka pimpinan dapat dilihat pada Tabel 4.5

Tabel 4.5 Antarmukan Pimpinan

Menu	Deskripsi
Login	Untuk menangani proses login admin
Beranda	Tampilan awal antarmuka admin
View detail ternak	Tampilan antarmuka untuk melihat data detail ternak
View peramalan	Tampilan antarmuka untuk melihat data peramalan
View grafik populasi ternak	Tampilan antarmuka untuk melihat grafik populasi hewan ternak

Menu	Deskripsi
View grafik peramalan	Tampilan antarmuka untuk melihat grafik peramalan
Keluar	Untuk proses keluar dari halaman

4.2.1.3 Antarmuka Staff

Berikut adalah implementasi antarmuka staff dapat dilihat pada Tabel 4.6

Tabel 4.6 Antarmukan Staff

Menu	Deskripsi
Menu utama	Tampilan awal antarmuka staff untuk menampilkan data spasial dalam bentuk Sig
Pencarian data	Tampilan antarmuka untuk mencari data popolasi hewan ternak
Cetak	Tampilan antarmuka untuk mencetak data

4.2 Pengujian Sistem

Proses pengujian adalah mencoba program dengan memasukkan data kedalam *form-form* masukan yang telah disediakan, pada tahap ini merupakan kelanjutan dari tahap implementasi yaitu melakukan pengujian-pengujian terhadap sistem yang dibangun.

4.2.1 Pengujian Alpha

Pengujian alpha ialah pengujian sistem yang dibangun yang merepresentasikan kajian pokok dari spesifikasi, perancangan dan pengkodean perangkat lunak. Pengujian ini menggunakan metode *black-box*. Pengujian *black-box* berfokus pada persyaratan fungsional perangkat lunak. Pengujian ini dibagi menjadi tiga skenario yaitu pengujian aplikasi pada Administrator, skenario pengujian aplikasi pada pimpinan dan skenario pengujian aplikasi pada staff/pegawai.

Berikut penjelasan tentang skenario pengujian aplikasi administrator. Pengujian dapat dilihat pada tabel Tabel 4.7

Tabel 4.7 Skenario Pengujian Aplikasi Administrator

Kelas Uji	Butir Uji	Jenis Pengujian
Login Administrator	Isi data Login administrator	*Black Box*
	Validasi *username and Password*	*Black Box*
Data User	Ubah data *user*	*Black Box*
	Tambah data *user*	*Black Box*
Data Kecamatan	Ubah data kecamatan	*Black Box*
	Hapus data kecamatan	*Black Box*
	Tambah data kecamatan	*Black Box*
Data Desa	Ubah data desa	*Blackbox*
	Hapus data desa	*Black Box*
	Tambah data desa	*Black Box*
Data kelompok ternak	Ubah data kelompok ternak	*Blackbox*
	Hapus data kelompok ternak	*Black Box*
	Tambah data kelompok ternak	*Black Box*
Data hewan ternak	Ubah data hewan ternak	*Blackbox*
	Hapus data hewan ternak	*Black Box*
	Tambah data hewan ternak	*Black Box*
Data detail ternak	Ubah data detail ternak	*Blackbox*
	Hapus data detail ternak	*Black Box*
	Tambah data detail ternak	*Black Box*
Data peramalan	Simpan data peramalan	*Black Box*
	Tambah peralaman	*Black Box*
	Proses peramalan	*Black Box*
Keluar	Menghapus *session* administrator	*Black Box*

Berikut penjelasan tentang skenario pengujian aplikasi pimpinan. Pengujian dapat dilihat pada tabel Tabel 4.8

Tabel 4.8 Skenario Pengujian Aplikasi Pimpinan

Kelas Uji	Butir Uji	Jenis Pengujian
Login Pimpinan	Isi data Login administrator	*Black Box*
	Validasi *username and Password*	*Black Box*
View data ternak	Lihat data ternak	*Black Box*
View peramalan	Lihat peramalan	*Black Box*
View grafik populasi	Lihat grafik populasi	*Blackbox*
Keluar	Menghapus *session* administrator	*Black Box*

Berikut penjelasan tentang skenario pengujian aplikasi staff. Pengujian dapat dilihat pada tabel Tabel 4.9

Tabel 4.9 Skenario Pengujian Aplikasi Staff

Kelas Uji	Butir Uji	Jenis Pengujian
Zoom in/out	Lihat data ternak	*Black Box*
Identify	Identifikasi peta	*Black Box*
Select	Pilih peta	*Black Box*
Search	Pencarian kecamatan	*Black Box*

4.2.1.1 Kasus dan Hasil Pengujian Alpha

Berdasarkan rencana pengujian, kasus dan hasil pengujian berisi pemaparan dari rencana pengujian yang telah disusun pada rencana pengujian.

Berikut pengujian pada aplikasi administrator .

1. Pengujian Login

Login digunakan untuk penentuan hak akses pengguna yang dapat dilihat pada Tabel 4.10

Tabel 4.10 Pengujian Login

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Pengisian *username* dan *Password* dengan data yang valid.	Menuju beranda admin	Menuju beranda admin	[√] Diterima [ ] Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
***User*name salah**			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Pengisian *username* dan *Password* dengan data yang tidak valid. Contoh: *User*name : Adm1n *Password* : admin	Menampilkan pesan kesalahan "*User*name dan *Password* Anda salah."	Menampilkan pesan kesalahan *User*name dan *Password* Anda salah."	[√] Diterima [ ] Ditolak
***Password* salah**			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Pengisian *username* dan *Password* dengan data yang tidak valid. Contoh: *User*name : admin *Password* : 4dm1n	Menampilkan pesan kesalahan "*User*name dan *Password* Anda salah."	Menampilkan pesan kesalahan "*User*name dan *Password* Anda salah."	[√] Diterima [ ] Ditolak

2. Pengujian tambah *User*

Pengujian untuk tambah *user* dengan data benar dan data salah dapat terlihat pada Tabel 4.11

Tabel 4.11 Pengujian tambah *User*

Kasus dan Data Hasil Uji (Data Benar)

Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Penambahan *data User* dengan data yang valid.	Data *User* berhasil tambah, menampilkan pesan ” data berhasil ditambah”	Data *User* berhasil tambah,” data berhasil ditambah”	[√] Diterima [ ] Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Nama lengkap Kosong			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan *field* nama lengkap. Contoh nama lengakap:[tidak diisi]	Menampilkan pesan kesalahan “nama tidak boleh kosong”. Penambahan data *User* gagal diinputkan	Menampilkan pesan kesalahan “nama tidak boleh kosong”. Penambahan data *User* gagal diinputkan	[√] Diterima [ ] Ditolak
***Username* kosong**			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan *field Username*. Contoh: Nama lengkap : *Username*:[kosong]	Menampilkan pesan kesalahan “*Username* tidak boleh kosong.” Penambahan data *User* gagal diinputkan	Menampilkan pesan kesalahan “*Username* tidak boleh kosong” . Penambahan data *User* gagal diinputkan	[√] Diterima [ ] Ditolak
***Password* kosong**			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan *field Password*. Contoh: *Password* : [kosong]	Menampilkan pesan kesalahan “*Password* tidak boleh kosong”. Penambahan data *User* gagal diinputkan	Menampilkan pesan kesalahan “*Password* tidak boleh kosong”. Penambahan data *User* gagal diinputkan	[√] Diterima [ ] Ditolak
Re-type Password* tidak sama dengan *Password			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Field Re-type *Password* tidak sama dengan *Password* Contoh: *Password* : 12345678 *Re-type Password :* 1234	Menampilkan pesan kesalahan “Re-Type *Password* tidak sama dengan *Password*” . Penambahan data *User* gagal diinputkan	Menampilkan pesan kesalahan “Re-Type *Password* tidak sama dengan *Password*” . Penambahan data *User* gagal diinputkan	[√] Diterima [ ] Ditolak

3. Pengujian *Edit User*

Pengujian untuk *edit user* dengan data benar dan data salah dapat terlihat pada Tabel 4.12

Tabel 4.12 Pengujian *Edit User*

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
perubahan *data User* dengan data yang valid.	Data *User* berhasil diubah	Data *User* berhasil diubah	[√] Diterima [ ] Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Nama lengkap Kosong			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan field nama lengkap. Contoh : Nama lengkap : [kosong]	Menampilkan pesan kesalahan "nama tidak boleh kosong ". Data *User* gagal diubah	Menampilkan pesan kesalahan "nama tidak boleh kosong." Data *User* gagal diubah	[√] Diterima [ ] Ditolak
***Username* kosong**			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan *field Username*. Contoh: *Username*:[kosong]	Menampilkan pesan kesalahan "*Username* tidak boleh kosong." Data *User* gagal diubah	Menampilkan pesan kesalahan "*Username* tidak boleh kosong." Data *User* gagal diubah	[√] Diterima [ ] Ditolak

4. Pengujian Tambah kecamatan

Pengujian untuk tambah kecamatan dengan data benar dan data salah dapat terlihat pada Tabel 4.13

Tabel 4.13 Pengujian Tambah Kecamatan

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Penambahan data kecamatan dengan data yang valid.	Data kecamatan berhasil ditambahkan dan menampilkan "data berhasil ditambah".	Data kecamatan berhasil ditambahkan dan menampilkan "data berhasil ditambah".	[√] Diterima [ ] Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Nama kecamatan kosong			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan *field* kecamatan Contoh: : [kosong]	Menampilkan pesan kesalahan "nama Kecamatan tidak boleh	Menampilkan pesan kesalahan "nama Kecamatan tidak boleh	[√] Diterima [ ] Ditolak

	kosong dan harus diisi huruf!" Data kecamatan tidak dapat ditambahkan kedalam sistem	kosong dan harus diisi huruf!" Data kecamatan tidak dapat ditambahkan kedalam sistem	

5. Pengujian *Edit* Kecamatan

Pengujian untuk *edit* kecamatan dengan data benar dan data salah dapat terlihat pada Tabel 4.14

Tabel 4.14 Pengujian *Edit* Kecamatan

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
pengubahan data kecamatan dengan data yang valid.	Data kecamatan berhasil diubahdan menampilkan "pesan bahwa data kecamatan berhasil diubah".	Data kecamatan berhasil diubahdan menampilkan "pesan bahwa data kecamatan berhasil diubah".	[√] Diterima [ ] Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Nama kecamatan kosong			
Mengosongkan *field* kecamatan Contoh: Nama kecamatan: [kosong]	Menampilkan pesan kesalahan "nama Kecamatan tidak boleh kosong!" Menampilkan pesan kesalahan Data kecamatan tidak dapat diubah	Menampilkan pesan kesalahan "nama Kecamatan tidak boleh kosong dan harus diisi huruf!"	[√] Diterima [ ] Ditolak

6. Pengujian Hapus Kecamatan

Pengujian untuk hapus kecamatan dengan data benar dapat terlihat pada Tabel 4.15

Tabel 4.15 Pengujian Hapus Kecamatan

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Menekan *icon* delete untuk menghapus kecamatan	Proses *delete* atau hapus kecamatan berhasil dengan menampilkan *:*pesan bahwa data	proses *delete* atau hapus kecamatan berhasil dengan menampilkan *alert* pesan bahwa data	[√] Diterima [ ] Ditolak

	kecamatan telah dihapus”	kecamatan telah terhapus	

7. Pengujian Tambah Desa

Pengujian untuk tambah desa dengan data benar dan data salah dapat terlihat pada Tabel 4.16

Tabel 4.16 Pengujian Tambah Desa

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Penambahan data Desa dengan data yang valid.	Data Desa berhasil ditambahkan dan menampilkan pesan bahwa “data berhasil ditambahka”n.	Data Desa berhasil ditambahkan dan menampilkan *alert* pesan bahwa “data berhasil ditambahkan”.	[√] Diterima [ ] Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Nama desa kosong			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan *field* desa Contoh: : [kosong]	Menampilkan pesan kesalahan “nama Desa tidak boleh kosong dan harus diisi huruf!” Data desa tidak dapat ditambahkan kedalam sistem	Menampilkan pesan kesalahan “nama Desa tidak boleh kosong dan harus diisi huruf!” Data desa tidak dapat ditambahkan kedalam sistem	[√] Diterima [ ] Ditolak

8. Pengujian *Edit* Desa

Pengujian untuk *edit* desa dengan data benar dan data salah dapat terlihat pada Tabel 4.17

Tabel 4.17 Pengujian *Edit* Desa

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
pengubahan data desa dengan data yang valid.	Data desa berhasil diubah dan menampilkan pesan bahwa “data berhasil diubah”.	Data desa berhasil diubahdan menampilkan pesan bahwa “data berhasil diubah”.	[√] Diterima [ ] Ditolak

Kasus dan Data Hasil Uji (Data Salah)			
Nama desa kosong			
Mengosongkan *field* desa Contoh: Nama desa: [kosong]	Menampilkan pesan kesalahan "nama Desa tidak boleh kosong!" Menampilkan pesan kesalahan Data desa tidak dapat diubah	Menampilkan pesan kesalahan "nama Desa tidak boleh kosong dan harus diisi huruf!"	[√] Diterima [ ] Ditolak

9. Pengujian Hapus Desa

Pengujian untuk hapus desa dengan data benar dapat terlihat pada Tabel 4.18

Tabel 4.18 Pengujian Hapus Desa

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Menekan *icon* delete untuk menghapus desa	Proses *delete* atau hapus kelurahan berhasil dengan menampilkan pesan "data telah terhapus "	proses *delete* atau hapus desa berhasil dengan menampilkan pesan data desa telah terhapus"	[√] Diterima [ ] Ditolak

10. Pengujian Tambah Kelompok Ternak

Pengujian untuk kelompok ternak dengan data benar dan data salah dapat terlihat pada Tabel 4.19

Tabel 4.19 Pengujian Tambah Kelompok Ternak

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Penambahan data Kelompok ternak dengan data yang valid.	Data Kelompok ternak berhasil ditambahkan dan menampilkan pesan "data berhasil ditambahkan".	Data Kelompok ternak berhasil ditambahkan dan menampilkan pesan "data berhasil ditambahkan".	[√] Diterima [ ] Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Nama kelompok ternak kosong			
Aktifitas yang	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**

dilakukan			
Mengosongkan *field* kelompok ternak Contoh: : [kosong]	Menampilkan pesan kesalahan "nama Kelompok ternak tidak boleh kosong dan harus diisi huruf!" Data kelompok ternak tidak dapat ditambahkan kedalam sistem	Menampilkan pesan kesalahan "nama Kelompok ternak tidak boleh kosong dan harus diisi huruf!" Data kelompok ternak tidak dapat ditambahkan kedalam sistem	[√] Diterima [ ] Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Nama ketua kelompok kosong			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan *field* nama ketua kelompok Contoh: : [kosong]	Menampilkan pesan kesalahan "nama ketua kelompok tidak boleh kosong dan harus diisi huruf!" Data nama ketua kelompok tidak dapat ditambahkan kedalam sistem	Menampilkan pesan kesalahan "nama ketua kelompok tidak boleh kosong dan harus diisi huruf!" Data nama ketua kelompok tidak dapat ditambahkan kedalam sistem	[√] Diterima [ ] Ditolak

11. Pengujian *Edit* Kelompok ternak

Pengujian untuk *edit* kelompok ternak dengan data benar dan data salah dapat terlihat pada Tabel 4.20

Tabel 4.20 Pengujian *Edit* Kelompok ternak

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
pengubahan data kelompok ternak dengan data yang valid.	Data kelompok ternak berhasil diubah dan menampilkan pesan "data berhasil diubah".	Data kelompok ternak berhasil diubah dan menampilkan pesan "data berhasil diubah".	[√] Diterima [ ] Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Nama kelompok ternak kosong			
Mengosongkan *field* kelompok ternak Contoh: Nama kelompok ternak: [kosong]	Menampilkan pesan kesalahan "nama Kelompok ternak tidak boleh kosong!" Menampilkan pesan kesalahan "Kabupaten/Kota Belum dipilih" Data kelompok ternak tidak dapat diubah	Menampilkan pesan kesalahan "nama Kelompok ternak tidak boleh kosong dan harus diisi huruf!"	[√] Diterima [ ] Ditolak

12. Pengujian Hapus Kelompok Ternak

Pengujian untuk hapus kelompok ternak dengan data benar dapat terlihat pada Tabel 4.21

Tabel 4.21 Pengujian Hapus Kelompok ternak

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Menekan *icon* delete untuk menghapus kelompok ternak	Proses *delete* atau hapus kelurahan berhasil dengan menampilkan pesan "data telah terhapus	Proses *delete* atau hapus kelurahan berhasil dengan menampilkan pesan "data telah terhapus	[√] Diterima [ ] Ditolak

13. Pengujian Tambah Hewan Ternak

Pengujian untuk tambah hewan ternak dengan data benar dapat terlihat pada Tabel 4.22

Tabel 4.22 Pengujian Tambah Hewan Ternak

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Penambahan data Hewan ternak dengan data yang valid.	Data Hewan ternak berhasil ditambahkan dan menampilkan pesan "data ternak ditambahkan."	Data Hewan ternak berhasil ditambahkan dan menampilkan pesan "data ternak ditambahkan."	[√] Diterima [ ] Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Nama hewan ternak kosong			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan *field* hewan ternak Contoh: : [kosong]	Menampilkan pesan kesalahan "nama Hewan ternak tidak boleh kosong dan harus diisi huruf!" Data hewan ternak tidak dapat ditambahkan kedalam sistem	Menampilkan pesan kesalahan "nama Hewan ternak tidak boleh kosong dan harus diisi huruf!" Data hewan ternak tidak dapat ditambahkan kedalam sistem	[√] Diterima [ ] Ditolak

14. Pengujian *Edit* Hewan Ternak

Pengujian untuk *edit* hewan tenak ternak dengan data benar dapat terlihat pada Tabel 4.23

Tabel 4.23 Pengujian *Edit* Hewan Ternak

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
pengubahan data hewan ternak dengan data yang valid.	Data hewan ternak berhasil diubah dan menampilkan pesan "data berhasil diubah"	Data hewan ternak berhasil diubah dan menampilkan pesan "data berhasil diubah"	[√] Diterima [ ] Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Nama hewan ternak kosong			
Mengosongkan *field* hewan ternak Contoh: Nama hewan ternak: [kosong]	Menampilkan pesan kesalahan "nama Hewan ternak tidak boleh kosong!" Menampilkan pesan kesalahan Data hewan ternak tidak dapat diubah	Menampilkan pesan kesalahan "nama Hewan ternak tidak boleh kosong dan harus diisi huruf!"	[√] Diterima [ ] Ditolak

15. Pengujian Hapus Hewan Ternak

Pengujian untuk hapus hewan ternak dengan data benar dapat terlihat pada Tabel 4.24

Tabel 4.24 Pengujian Hapus Hewan Ternak

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Menekan *icon* delete untuk menghapus hewan ternak	Proses *delete* atau hapus kelurahan berhasil dengan menampilkan *alert* pesan bahwa data hewan ternak telah terhapus	proses *delete* atau hapus hewan ternak berhasil dengan menampilkan *alert* pesan bahwa data hewan ternak telah terhapus	[√] Diterima [ ] Ditolak

16. Pengujian Tambah Detail Ternak

Pengujian untuk tambah detail ternak dengan data benar dan data salah dapat terlihat pada Tabel 4.25

Tabel 4.25 Pengujian Tambah Detail Ternak

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Penambahan data Detail ternak dengan data yang valid.	Data Detail ternak berhasil ditambahkan dan menampilkan pesan "data ternak ditambahkan".	Data Detail ternak berhasil ditambahkan dan menampilkan pesan "data ternak ditambahkan".	[√] Diterima [ ] Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Nama detail ternak kosong			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan *field* detail ternak Contoh: : [kosong]	Menampilkan pesan kesalahan "nama Detail ternak tidak boleh kosong dan harus diisi huruf!" Data detail ternak tidak dapat ditambahkan kedalam sistem	Menampilkan pesan kesalahan "nama Detail ternak tidak boleh kosong dan harus diisi huruf!" Data detail ternak tidak dapat ditambahkan kedalam sistem	[√] Diterima [ ] Ditolak

17. Pengujian *Edit* Detail Ternak

Pengujian untuk *edit* detail ternak dengan data benar dan data salah dapat terlihat pada Tabel 4.26

Tabel 4.26 Pengujian *Edit* Detail Ternak

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
pengubahan data detail ternak dengan data yang valid.	Data detail ternak berhasil diubah dan menampilkan pesan "data berhasil diubah."	Data detail ternak berhasil diubah dan menampilkan pesan "data berhasil diubah."	[√] Diterima [ ] Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Nama detail ternak kosong			
Mengosongkan *field*	Menampilkan pesan	Menampilkan pesan	[√] Diterima

detail ternak Contoh: Nama detail ternak: [kosong]	kesalahan "nama Detail ternak tidak boleh kosong!" Menampilkan pesan kesalahan Data detail ternak tidak dapat diubah	kesalahan "nama Detail ternak tidak boleh kosong dan harus diisi huruf!"	[ ] Ditolak

18. Pengujian Hapus Detail Ternak

Pengujian untuk hapus detail ternak dengan data benar dapat terlihat pada Tabel 4.27

Tabel 4.27 Pengujian Hapus Detail Ternak

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Menekan *icon* delete untuk menghapus detail ternak	Proses *delete* atau hapus kelurahan berhasil dengan menampilkan pesan "data telah dihapus"	Proses *delete* atau hapus kelurahan berhasil dengan menampilkan pesan "data telah dihapus"	[√] Diterima [ ] Ditolak

19. Pengujian Tambah Layer

Pengujian untuk tambah layer dengan data benar dan data salah dapat terlihat pada Tabel 4.28

Tabel 4.28 Pengujian Tambah Layer

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Penambahan data Layer dengan data yang valid.	Data Layer berhasil ditambahkan dan menampilkan *alert* pesan bahwa data Layer berhasil ditambahkan.	Data Layer berhasil ditambahkan dan menampilkan *alert* pesan bahwa data Layer berhasil ditambahkan.	[√] Diterima [ ] Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Nama layer kosong			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan *field* layer Contoh:	Menampilkan pesan kesalahan "nama Layer	Menampilkan pesan kesalahan "nama Layer	[√] Diterima [ ] Ditolak

: [kosong]	tidak boleh kosong dan harus diisi huruf!" Data layer tidak dapat ditambahkan kedalam sistem	tidak boleh kosong dan harus diisi huruf!" Data layer tidak dapat ditambahkan kedalam sistem	

20. Pengujian *Edit* Layer

Pengujian untuk *edit* layer dengan data benar dan data salah dapat terlihat pada Tabel 4.29

Tabel 4.29 Pengujian *Edit* Layer

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
pengubahan data layer dengan data yang valid.	Data layer berhasil diubah dan menampilkan *alert* pesan bahwa data layer berhasil diubah.	Data layer berhasil diubahdan menampilkan *alert* pesan bahwa data layer berhasil diubah.	[√] Diterima [ ] Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Nama layer kosong			
Mengosongkan *field* layer Contoh: Nama layer: [kosong]	Menampilkan pesan kesalahan "nama Layer tidak boleh kosong!" Menampilkan pesan kesalahan Data layer tidak dapat diubah	Menampilkan pesan kesalahan "nama Layer tidak boleh kosong dan harus diisi huruf!"	[√] Diterima [ ] Ditolak

21. Pengujian Hapus Layer

Pengujian untuk hapus layer dengan data benar dapat terlihat pada Tabel 4.30

Tabel 4.30 Pengujian Hapus Layer

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Menekan *icon* delete untuk menghapus layer	Proses *delete* atau hapus kelurahan berhasil dengan menampilkan *alert* pesan bahwa data layer telah terhapus	proses *delete* atau hapus layer berhasil dengan menampilkan *alert* pesan bahwa data layer telah terhapus	[√] Diterima [ ] Ditolak

22. Pengujian Menu Peramalan

Pengujian untuk menu peramalan dengan data benar dapat terlihat pada Tabel 4.31

Tabel 4.31 Pengujian menu Peramalan

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memilih menu peramalan	Menampilkan data peramalan	Menampilkan data peramalan	[√] Diterima [ ] Ditolak

23. Pengujian Proses Peramalan

Pengujian untuk proses peramalan dengan data benar dan data salah dapat terlihat pada Tabel 4.32

Tabel 4.32 Pengujian Proses Peramalan

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Penambahan data Peramalan dengan data yang valid.	Menampilkan hasil peramalan berdasarkan produksi	Memilih jenis produksi, memilih akan diramalkan, lalu klik proses.	[√] Diterima [ ] Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Nama peramalan kosong			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan *field* data peramalan Contoh: : [kosong]	Tidak dapat melakukan proses peramalan.	Muncul Pesan data tidak sesuai dengan periode dan tidak dapat melanjutkan proses peramalan	[√] Diterima [ ] Ditolak
Kasus dan Data Hasil Uji (Data Normal)			
Simpan peramalan			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Data hasil proses peramalan	Menyimpan hasil peramalan di *database* dan akan menampilkan pre*view* laporan hasil peramalan yang akan dicetak..	Klik tombol Simpan, akan muncul pesan bahwa data telah tersimpan dan akan menanyakan cetak hasil peramalan jika memilih tombol Yes, maka data akan dicetak	[√] Diterima [ ] Ditolak

24. Pengujian keluar

Pengujian untuk keluar dengan data benar dapat terlihat pada Tabel 4.34

Tabel 4.33 Pengujian Keluar

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Klik menu keluar	Keluar dari sistem dan menghapus *session* login.	Keluar dari sistem dan menghapus *session* login.	[√] Diterima [ ] Ditolak

Berikut pengujian pada aplikasi pimpinan.

1. Pengujian Login

Pengujian untuk Login dengan data benar dapat terlihat pada Tabel 4.34

Tabel 4.34 Pengujian Login

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Pengisian *username* dan *Password* dengan data yang valid.	Menuju beranda admin	Menuju beranda admin	[√] Diterima [ ] Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
***User*name salah**			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Pengisian *username* dan *Password* dengan data yang tidak valid. Contoh: *User*name : Adm1n *Password* : admin	Menampilkan pesan kesalahan "*User*name dan *Password* Anda salah."	Menampilkan pesan kesalahan *User*name dan *Password* Anda salah."	[√] Diterima [ ] Ditolak
***Password* salah**			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Pengisian *username* dan *Password* dengan data yang tidak valid. Contoh: *User*name : admin *Password* : 4dm1n	Menampilkan pesan kesalahan "*User*name dan *Password* Anda salah."	Menampilkan pesan kesalahan "*User*name dan *Password* Anda salah."	[√] Diterima [ ] Ditolak

2. Pengujian *View* Data Ternak

Pengujian untuk *view* data ternak dengan data benar dapat terlihat pada Tabel 4.35

Tabel 4.35 Pengujian *View* Data Ternak

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Klik menu *View* Data Ternak	Menampilkan data ternak.	Menampilkan data ternak.	[√] Diterima [ ] Ditolak

3. Pengujian *View* Data Peramalan

Pengujian untuk *view* data peramalan dengan data benar dapat terlihat pada Tabel 4.36

Tabel 4.36 Pengujian *View* Data Peramalan

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Klik menu *View* Data peramalan	Menampilkan data peramalan.	Menampilkan data peramalan.	[√] Diterima [ ] Ditolak

4. Pengujian *View* Grafik Populasi

Pengujian untuk *view* grafik populasi dengan data benar dapat terlihat pada Tabel 4.37

Tabel 4.37 Pengujian *View* Grafik Populasi

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Klik menu *View* Grafik Populasi	Menampilkan grafik populasi	Menampilkan grafik populasi	[√] Diterima [ ] Ditolak

5. Pengujian keluar

Pengujian untuk keluar dengan data benar dapat terlihat pada Tabel 4.33

Tabel 4.38 Pengujian Keluar

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Klik menu keluar	Keluar dari sistem dan menghapus *session* login.	Keluar dari sistem dan menghapus *session* login.	[√] Diterima [ ] Ditolak

Berikut pengujian pada aplikasi staff.pegawai.

1. **Pengujian *Zoom In/Out***

   Pengujian untuk *zoom in/out* dengan data benar dapat terlihat pada Tabel 4.39

Tabel 4.39 Pengujian *Zoom In/Out*

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Klik menu *zoom in/out*	*Zoom in* untuk menampilkan peta lebih dalam atau dekat dan *zoom out* untuk menampilkan peta lebih jauh.	*Zoom in* dapat menampilkan peta lebih dalam atau dekat dan *zoom out* dapat menampilkan peta lebih jauh .	[√] Diterima [ ] Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
***Zoom in/out* salah satu wilayah**			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mngklik tombol *zom* Contoh: : [zoom salah satu wilayah]	Tidak menampilkan peta yang akan di *zoom*	Tidak menampilkan pea yang akan di *zoom*	[√] Diterima [ ] Ditolak

2. **Pengujian *Identify***

   Pengujian untuk *identify* dengan data benar dapat terlihat pada Tabel 4.40

Tabel 4.40 Pengujian *Identify*

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**

Klik menu *identify*	Menapilkan hasil identifikasi pada peta yang telah di klik.	Menapilkan hasil identifikasi pada peta yang telah di klik. .	[√] Diterima [ ] Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
***Identify* Salah satu peta**			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengklik tombol *identify* Contoh: : [*identify* salah satu peta]	Tidak menampilkan hasil identifikasi	Tidak menampilkan hasil identifikasi	[√] Diterima [ ] Ditolak

3. Pengujian *Select*

Pengujian untuk select dengan data benar dapat terlihat pada Tabel 4.41

Tabel 4.41 Pengujian *Select*

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Klik menu *select*	Menampilkan hasil seleksi peta yang akan ditampilkan.	Menampilkan hasil seleksi peta yang akan ditampilkan.	[√] Diterima [ ] Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
***Select* salah satu peta**			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengklik tombol *select* Contoh: : [*select* salah satu peta]	Tidak menampilkan data dari hasil seleksi.	Tidak menampilkan data dari hasil seleksi.	[√] Diterima [ ] Ditolak

4. Pengujian *Seacrh*

Pengujian untuk *search* dengan data benar dapat terlihat pada Tabel 4.42

Tabel 4.42 Pengujian *Search*

Kasus dan Data Hasil Uji (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Klik menu *Search*	Menampilkan hasil wilayah yang di cari.	Menampilkan hasil wilayah yang di cari.	[√] Diterima [ ] Ditolak

Kasus dan Data Hasil Uji (Data Salah)			
***Search* salah satu wilayah**			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengklik tombol *search* Contoh: : [*search* salah satu wilayah]	Tidak menampilkan data dari pencarian.	Tidak menampilkan data dari pencarian	[√] Diterima [ ] Ditolak

4.2.1.2 Kesimpulan Pengujian Alpha

Berdasarkan hasil pengujian dengan kasus uji sample diatas dapat ditarik kesimpulan bahwa perangkat lunak telah bebas dari kesalahan sintaks dan secara fungsional menghasilkan hasil yang sesuai dengan yang diharapkan.

4.2.2 Pengujian Beta

Pengujian *beta* dilakukan dengan tujuan untuk mengetahui sejauh mana kualitas dari perangkat lunak yang dibangun, apakah sudah sesuai dengan harapan atau belum.

4.2.2.1 Kasus dan Hasil 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*.

Wawancara dilakukan hanya pada kepala sub bagian perencanaan dan program dan Staff. Adapun pertanyaan dan hasil wawancar yang akan diajukan pada saat wawancara adalah sebagai berikut :

a. Pertanyaan dan hasil wawancara wawancara bagian Administrator dapat dilihat pada Tabel 4.43

Tabel 4.43 Pertanyaan dan Hasil Wawancara Bagian Administrator

1. Apakah instruksi atau cara penggunaan yang diberikan dalam aplikasi ini mudah dimengerti?	
Jawaban narasumber	ya, cukup mudah digunakan dan tidak susah dalam mengaplikasikannya.
2. Apakah aplikasi SIG ini memberikan manfaat kepada pihak dinas?	
Jawaban narasumber	iya, aplikasi ini sangat bermanfaat dan membantu dinas terutama dalam melihat penyebaran populasi hewan ternak

3. Apakah fasilitas *forecasting* telah memberikan kemudahan dalam memprediksi kebutuhan produksi ternak?	
Jawaban narasumber	iya , fasilitas *forecasting* sangat membantu dalam memprediksi kebutuhan produksi ternak tetapi tampilannya masih harus diperbaiki
4. Apakah aplikasi tersbut mudah dalam pengelolaan data ternak?	
Jawaban narasumber	iya, pengolahan data lebih praktis data dikelola sistem dengan baik

b. Pertanyaan dan hasil wawancara bagian staff/pegawai dapat dilihat pada Tabel 4.44

Tabel 4.44 Pertanyaan dan Hasil Wawancara Bagian Staff/Pegawai

1. Apakah secara umum program aplikasi SIG ini mudah digunakan?	
Jawaban narasumber	ya, cukup mudah digunakan.
2. Apakah dengan adanya aplikasi ini mempermudah Dinas dalam melihat potensi ternak?	
Jawaban narasumber	ya membantu, pemilihan potensi menjadi lebih terbantu karena kriteria pemilihan potensi dikolola oleh sistem.
3. Apakah aplikasi tersebut mudah dalam pencarian populasi ternak sesuai keinginan?	
Jawaban narasumber	ya , mudah karena pencarian tersebut di fokuskan pada tempat yang dicari sehingga pencarian menjadi lebih optimal.
4. Apakah tampilan aplikasi tersebut menarik?	
Jawaban narasumber	cukup, untuk tampilan sebagai pegawai bagus bila perlu tampilannya dibuat lebih interaktif kembali.

4.2.2.2 Kesimpulan Pengujian Beta

Dari hasil pengujian beta yang telah dilakukan yaitu dengan wawancara kepada *User* yang bersifat pertanyaan di lapangan, diperoleh kesimpulan bahwa sistem yang telah dibangun ini dinilai sudah optimal dalam hal pengelolaan sistem, mudah digunakan dan dapat membantu dinas peternakan dalam mengidentifikasi potensi daerah yang menghasilkan ternak yang lebih baik ,serta ,memudahkan dalam pemantauan perkembangan populasi hewan informasi sedangakan hasil peramalan walaupun belum memenuhi target tetapi data tersebut bisa mamberikan gambaran untuk kebutuhan produksi ternak di tahun yang akan datang.