BAB 4
IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi Sistem

Implementasi sistem merupakan tahap yang dilakukan untuk mengetahui sistem yang telah dibangun dapat beroperasi dengan baik. Tahap ini merupakan tahap dimana sistem siap untuk digunakan, yang terdiri dari penjelasan mengenai lingkungan implementasi, dan implementasi program. Setelah implementasi maka dilakukan pengujian sistem yang baru dimana akan dilihat kekurangan-kekurangan pada aplikasi yang baru untuk pengembangan selanjutnya.

4.1.1 Implementasi Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan dalam pembangunan Sistem Informasi Manajemen Pengelolaan Mitra Petani di CV. Jatiluhur Mas tercantum pada Tabel 4-1.

Tabel 4-1 Perangkat Lunak Yang Digunakan

No	Perangkat Lunak	Keterangan
1	Sistem Operasi	Microsoft Windows 8.1
2	DBMS	C# (C Sharp)
3	Bahasa Pemrograman	PhpMyAdmin
4	Code Editor	Visual Studio 2013
5	UML Modeler	Visual Paradigm for UML

4.1.2 Implementasi Perangkat Keras

Spesifikasi perangkat keras yang digunakan dalam pembangunan Sistem Informasi Manajemen Pengelolaan Mitra Petani di CV. Jatiluhur Mas dapat dilihat pada Tabel 4-2.

Tabel 4-2 Perangkat keras yang digunakan

No	Perangkat Keras	Spesifikasi
1	*Processor*	*Speed 2,26* Ghz
2	Memori	2 GB
3	*Harddisk*	320 GB
5	VGA	1 GB
6	*Lan Card*	10/100Mbps
7	Monitor	16 inch
8	*Keyboard*	Standar
9	*Mouse*	Standar

4.1.3 Implementasi Kelas

Ini deskripsi implementasi class

No	Nama Kelas Perancangan	Nama File Terkait Kelas Perancangan
1	ClassKoneksi	ClassKoneksi.cs
2	Kolam	FormAreaKolam.cs FormKolamSiapPanen.cs FormLihatDaftarKolam.cs Kolam.cs
3	Mitra	FormKelolaPetani.cs FormDetailPiutangMitra.cs Mitra.cs
4	Pakan	FormStockPakan.cs FormTambahPakan.cs Pakan.cs
5	Pengiriman	FormPengiriman.cs FormRutePengiriman.cs FormVerifikasi.cs Pengiriman.cs
6	Petugas	FormPetugasPengiriman.cs FormPengurusKolam.cs Petugas.cs
7	User	FormLogin.cs User.cs
8	ClassCRUD	ClassCRUD.cs

4.1.4 Implementasi Basis Data

Pembuatan basis data dilakukan dengan menggunakan aplikasi DBMS phpMyAdmin. Implementasi basis data dalam bahasa SQL adalah sebagai berikut:

1. Pembuatan *database*

```
1  CREATE DATABASE ‘db_cvjm’;
```

2. Tabel mitra

```
1  CREATE TABLE ‘t_mitra’ (
2  ‘id_mitra’ int (6) NOT NULL AUTO INCREMENT,
3  ‘nama’ varchar (40) NOT NULL,
4  ‘alamat’ varchar (50) NOT NULL,
5  ‘telepon’ varchar (13) NOT NULL,
6  ‘tgl_gabung’ varchar (15) NOT NULL,
7  ‘setoran_awal’ int (10),
8  ‘jenis_kerja_sama’ enum(‘mitra_kecil’,’mitra_sedang’,’mitra_besar),
9  ‘kelompok_mitra’ enum(‘mitra_kecil’,’mitra_sedang’,’mitra_besar),
10 PRIMARY KEY(‘id_mitra’)
11 ) ENGINE = InnoDB;
```

3. Tabel kolam

```
1  CREATE TABLE ‘t_kolam’ (
2  ‘id_kolam’ int (6) NOT NULL AUTO INCREMENT,
3  ‘id_pengurus_kolam’ int (6) NOT NULL,
4  ‘id_mitra’ int (6) NOT NULL,
5  ‘id_area’ int (6) NOT NULL,
6  ‘nama_kolam’ varchar (20) NOT NULL,
7  ‘tanggal_sebar_bibit’ varchar (15) NOT NULL,
8  ‘total_pakan’ int (15) NOT NULL,
9  ‘jumlah_petak’ int (5) NOT NULL,
10 ‘perkiraan_bobot’ double NOT NULL,
11 PRIMARY KEY (‘id_kolam’),
12 FOREIGN KEY fk_pengurus (‘id_pengurus_kolam’)
13 REFERENCES t_pengurus_kolam(‘id_pengurus_kolam’),
14 FOREIGN KEY fk_mitra(‘id_mitra’)
15 REFERENCES t_mitra(‘id_mitra’),
16 FOREIGN KEY fk_area(‘id_area’)
17 REFERENCES t_area_kolam(‘id_area’)
18 ) ENGINE = InnoDB;
```

4. Tabel area kolam

```
1  CREATE TABLE ‘t_area_kolam’ (
2  ‘id_area’ int (6) NOT NULL AUTO INCREMENT,
3  ‘nama_area’ varchar (20) NOT NULL,
4  ‘longtitude’ double NOT NULL,
5  ‘latitude’ double NOT NULL,
6  ‘jarak_area’ double,
7  PRIMARY KEY (‘id_area’)
8  ) ENGINE = InnoDB;
```

5. Tabel pengurus kolam

```
1  CREATE TABLE ‘t_pengurus_kolam’ (
2  ‘id_pengurus_kolam’ int (6) NOT NULL AUTO INCREMENT,
3  ‘nama’ varchar (40) NOT NULL,
4  ‘alamat’ varchar (50) NOT NULL,
5  ‘telepon’ varchar (13) NOT NULL,
6  PRIMARY KEY (‘id_pengurus_kolam’)
7  ) ENGINE = InnoDB;
```

6. Tabel pemesanan

```
1  CREATE TABLE ‘t_pemesanan’ (
2  ‘id_pesanan’ int (6) NOT NULL AUTO INCREMENT,
3  ‘id_kolam’ int (6) NOT NULL,
4  ‘id_pakan’ int (6) NOT NULL,
5  ‘id_petugas_pengiriman’ int (6) NOT NULL,
6  ‘id_pengurus_kolam’ int (6) NOT NULL,
7  ‘banyak_pakan’ int (9) NOT NULL,
8  ‘tanggal_pesan’ varchar (15) NOT NULL,
9  ‘keterangan_pengiriman’ enum (‘belum_dikirim’, ‘sudah_dikirim’),
10 ‘urutan_pengiriman’ int (4),
```

```
11  'jam_sampai' varchar (15),
12  PRIMARY KEY ('id_pesanan'),
13  FOREIGN KEY fk_kolam ('id_kolam')
14  REFERENCES t_kolam('id_kolam'),
15  FOREIGN KEY fk_pakan('id_pakan')
16  REFERENCES t_pakan('id_pakan'),
17  FOREIGN KEY fk_pengirim ('id_petugas_pengiriman')
18  REFERENCES t_petugas_pengiriman ('id_petugas_pengiriman'),
19  FOREIGN KEY fk_pengurus2 ('id_pengurus_kolam')
20  REFERENCES t_pengurus_kolam ('id_ pengurus_kolam')
21  ) ENGINE = InnoDB;
```

7. Tabel pakan

```
1   CREATE TABLE 't_pakan' (
2   'id_pakan' int (6) NOT NULL AUTO INCREMENT,
3   'stok_pakan' int (10) NOT NULL,
4   'jenis_pakan' varchar (20) NOT NULL,
5   'harga_pakan' int (6) NOT NULL,
6   'jenis_mitra' enum ('mitra_kecil','mitra_sedang','mitra_besar'),
7   'harga_beli_pakan' int (6) NOT NULL,
8   PRIMARY KEY ('id_pakan')
9   ) ENGINE = InnoDB;
```

8. Tabel petugas pengiriman

```
1   CREATE TABLE 't_petugas_pengiriman' (
2   'id_petugas_pengiriman' int (6) NOT NULL AUTO INCREMENT,
3   'nama' varchar (40) NOT NULL,
4   'alamat' varchar (50) NOT NULL,
5   'telepon' varchar (13) NOT NULL,
6   'poin' int (6),
7   PRIMARY KEY ('id_ petugas_pengiriman')
8   ) ENGINE = InnoDB;
```

9. Tabel pemasok

```
1   CREATE TABLE 't_pemasok' (
2   'id_pabrik' int (6) NOT NULL AUTO INCREMENT,
3   'id_pakan' int (6) NOT NULL,
4   'tanggal_pemesanan' varchar (15) NOT NULL,
5   'tenggat_waktu' varchar (15) NOT NULL,
6   'jumlah_pakan' int (9) NOT NULL,
7   PRIMARY KEY ('id_pabrik'),
8   FOREIGN KEY fk_pakan2 ('id_pakan')
9   REFERENCES t_pakan ('id_pakan'),
10  ) ENGINE = InnoDB;
```

10. Tabel pembayaran

```
1   CREATE TABLE 't_pembayaran' (
```

```
2   'id_pembayaran' int (6) NOT NULL AUTO INCREMENT,
3   'id_mitra' int (6) NOT NULL,
4   'tanggal_bayar' varchar (15) NOT NULL,
5   'jumlah_bayar' int (10) NOT NULL,
6   'jenis_bayar' enum ('transfer','tunai'),
7   'bukti_bayar' varchar (50) NOT NULL,
8   'sisa_hutang' int (10) NOT NULL,
9   PRIMARY KEY ('id_pembayaran'),
10  FOREIGN KEY fk_mitra3 ('id_mitra')
11  REFRENCES t_mitra ('id_mitra')
12  ) ENGINE = InnoDB;
```

11. Tabel *user*

```
1   CREATE TABLE 't_user' (
2   'id_user' int (6) NOT NULL AUTO INCREMENT,
3   'nama' varchar (40) NOT NUL,
4   'username' varchar (20) NOT NULL,
5   'password' varchar (8) NOT NULL,
6   'jabatan' enum ('petugas','manajer')
7   PRIMARY KEY ('id_user')
8   ) ENGINE = InnoDB;
```

4.1.5 Implementasi Antarmuka

Implementasi antarmuka dilakukan dengan setiap tampilan program yang dibangun. Adapun untuk tampilan implementasi pembangunan Sistem Informasi Manajemen Pengelolaan Mitra Petani di CV. Jatiluhur Mas dilampirkan pada halaman lampiran B. Berikut ini adalah implementasi antarmuka yang terbagi atas antarmuka Pemimpin perusahaan dan Kepala bagian gudang.

Implementasi antarmuka untuk pemimpin perusahaan dapat dilihat pada Tabel 4-3.

Tabel 4-3 Implementasi Antarmuka Pemimpin Perusahaan

No	Menu	Deskripsi	Nama *File*
1	Menentukan rute pengiriman	Tampilan untuk menentukan rute pengiriman berdasarkan pemesanan yang dilakukan oleh mitra petani.	FormRutePengirimanPakan. Designer.cs
2	Memperkirakan kolam siap panen	Tampilan untuk memilih kolam siap panen sesuai permintaan pasar.	FormKolamSiapPanen. Designer.cs
3	Melakukan pemesanan stok pakan	Tampilan untuk mengelola penyediaan stok pakan dari pabrik.	FormStockPakan. Designer.cs
4	Mengelola pembayaran piutang mitra	Tampilan untuk mengelola pembayaran mitra petani yang melakukan pembayaran melalu pengiriman ke rekening CV. Jatiluhur Mas.	FormDetailPiutangMitra. Designer.cs

Implementasi antarmuka untuk Kepala bagian gudang dapat diliha pada Tabel 4-4.

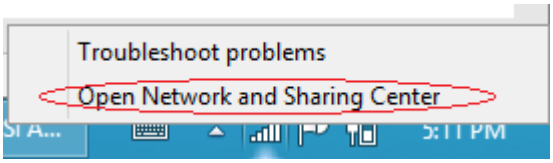
Tabel 4-4 Implementasi Antarmuka Kepala Bagian Gudang

No	Menu	Deskripsi	Nama *File*
1	Mengelola data mitra	Tampilan untuk mengelola data mitra petani.	FormKelolaPetani. Designer.cs
2	Mengelola data kolam	Tampilan untuk mengelola kolam.	FormLihatDaftarKolam. Designer.cs
3	Mengelola data area kolam	Tampilan untuk megelola area kolam.	FormAreaKolam. Designer.cs
4	Mengelola data petugas pengiriman	Tampilan untuk mengelola data petugas pengiriman.	FormPetugasPengiriman. Designer.cs
5	Mengelola data pengurus kolam	Tampilan untuk mengelola data pengurus kolam.	FormPengurusKolam. Designer.cs
6	Mengelola pembayaran piutang mitra	Tampilan untuk mengelola pembayaran mitra petani yang melakukan pembayaran manual.	FormDetailPiutangMitra. Designer.cs

4.1.6 Implementasi Jaringan *Client-Server*

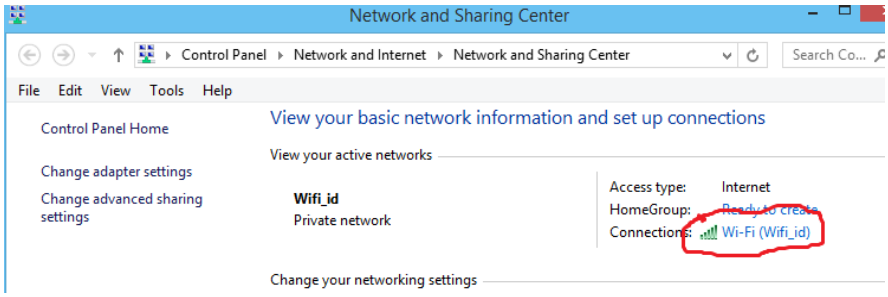
Siapkan alat-alat yang dibutuhkan seperti yang sudah dijelaskan pada tahap perancangan jaringan. Lalu lakukan langkah berikut ini:

1. Nyalakan *hotspot (wifi)*, lalu hubungkan komputer *server* dan komputer *client* ke jaringan *hotspot (wifi)* tersebut.
2. Kedua komputer secara otomatis akan mendapatkan *IP Address*, setelah terhubung dengan jaringan tersebut.
3. Cek kedua *IP Address* yang didapat komputer *server*, dengan cara sebagai berikut:
    a. Klik kanan pada ikon *wifi* yang ada pada *taskbar*, lalu klik ”*open Network and Sharing center*”. Keterangan dapat dilihat pada Gambar 4-1.



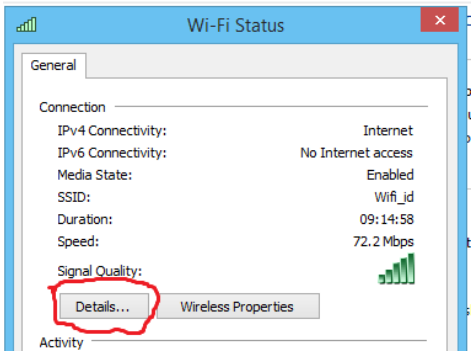
Gambar 4-1 Mengecek Alamat *IP Address Server* (1)

b. Setelah muncul *window Network and Sharing center*, lalu klik pada keterangan status nama *wifi* yang terhubung. Keterangan dapat dilihat pada Gambar 4-2.



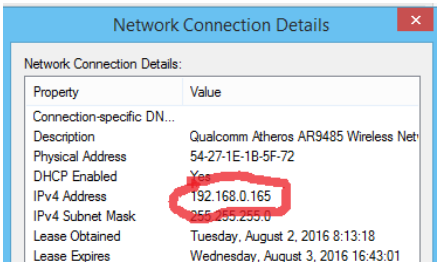
Gambar 4-2 Mengecek Alamat *IP Address Server* (2)

c. Setelah muncul window *wifi* status, lalu klik tombol "*Details*". Keterangan dapay dilihat pada Gambar 4-3.



Gambar 4-3 Mengecek Alamat *IP Address Server* (3)

d. Setelah muncul *window network connection details*, maka akan ada keterangan *IP Address* komputer. Keterangan *IP Address* dapat dilihat pada Gambar 4-4.



Gambar 4-4 Mengecek Alamat *IP Address Server* (4)

4. Setelah kedua *IP Address* diketahui, komputer *server* harus memberikan hak akes *databse* pada *IP Address* milik komputer *client* dengan cara menambahkan kodingan "Allow from (diisi dengan *IP Address* milik *client*)"

pada file httpd-xampp.conf yang biasanya terletak di C:\xampp\apache\conf\extra atau disesuaikan dengan direktori milik xampp pada komputer *server*.

5. Setelah *IP Address* milik komputer *client* sudah diberikan hak akses ke *databse*, maka saat *client* membuka aplikasi akan muncul form untuk mengisikan *IP Address* milik *server*. Setelah mengisikan *IP Address* milik *server* aplikasi sudah dapat dijalankan sesuai kebutuhannya.

4.2 Pengujian Sistem

Pengujian sistem ini merupakan hal terpenting yang bertujuan untuk menemukan kesalahan-kesalahan dan kekurangan-kekurangan pada pernagkat lunak yang diuji. Pengujian bermaksud untuk mengetahui perangkat lunak yang dibuat sudah memenuhi kriteria yang sesuai dengan tujuan perancangan. Berdasarkan hasil pengujian sistem yang telah dilakukan secara keseluruhan dapat ditarik kesimpulan mengenai perangkat lunak yang sedang dibangun. Pengujian perangkat lunak ini menggunakan pengujian *black box.* Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak.

4.2.1 Skenario Pengujian Black Box

Berdasarkan pengujian yang telah dilakukan, dapat diambil kesimpulan bahwa aplikasi Sistem Informasi Manajemen Pengelolaan Mitra Petani di CV. Jatiluhur Mas telah melalui tahap perbaikan dan sudah dimaksimalkan terhadap proses-proses tersebut dan secara fungsional sistem sudah dapat digunakan dan menghasilkan keluaran yang diharapkan. Skenario pegujian *black box* dapat dilihat pada Tabel 4-5.

Tabel 4-5 Skenario Pengujian

No	Kelas Uji	Butir Uji	Jenis Pengujian
1	*Login*	*1.Scenario Main Flow* *2.Scenario Extension*	*Black box* *Black box*
2	Menentukan inisialisasi kolam	*1.Scenario Main Flow* *2.Scenario Extension*	*Black box* *Black box*
3	Memilih kolam siap panen	*1.Scenario Main Flow* *2.Scenario Extension*	*Black box* *Black box*
4	Mencari rute pengiriman	*1.Scenario Main Flow* *2.Scenario Extension*	*Black box* *Black box*
5	Memesanan pakan ke pemasok	*1.Scenario Main Flow*	*Black box*

No	Kelas Uji	Butir Uji	Jenis Pengujian
		2.Scenario Extension	*Black box*
6	Melakukan pembayaran pakan ke pemasok	*1.Scenario Main Flow* *2.Scenario Extension*	*Black box* *Black box*
7	Memproses pengiriman pakan	*1.Scenario Main Flow* *2.Scenario Extension*	*Black box* *Black box*
8	Memverifikasi pengiriman pakan	*1.Scenario Main Flow* *2.Scenario Extension*	*Black box* *Black box*
9	Tambah data mitra	*1.Scenario Main Flow* *2.Scenario Extension*	*Black box* *Black box*
10	Tambah data area kolam	*1.Scenario Main Flow* *2.Scenario Extension*	*Black box* *Black box*
11	Tambah data kolam	*1.Scenario Main Flow* *2.Scenario Extension*	*Black box* *Black box*
12	Tambah Data Petugas	*1.Scenario Main Flow* *2.Scenario Extension*	*Black box* *Black box*
13	Ubah data mitra	*1.Scenario Main Flow* *2.Scenario Extension*	*Black box* *Black box*
14	Ubah Data Area Kolam	*1.Scenario Main Flow* *2.Scenario Extension*	*Black box* *Black box*
15	Ubah data kolam	*1.Scenario Main Flow* *2.Scenario Extension*	*Black box* *Black box*
16	Ubah data petugas	*1.Scenario Main Flow* *2.Scenario Extension*	*Black box* *Black box*
17	Hapus data mitra	*1.Scenario Main Flow* *2.Scenario Extension*	*Black box* *Black box*
18	Hapus data area kolam	*1.Scenario Main Flow* *2.Scenario Extension*	*Black box* *Black box*
19	Hapus data kolam	*1.Scenario Main Flow* *2.Scenario Extension*	*Black box* *Black box*
20	Hapus data petugas	*1.Scenario Main Flow* *2.Scenario Extension*	*Black box* *Black box*
21	Mengelola pembayaran pakan	*1.Scenario Main Flow* *2.Scenario Extension*	*Black box* *Black box*

4.2.2 Kasus dan Hasil Pengujian

Berdasarkan rencana pengujian yang disusun, maka dilakukan pengujian sesuai dengan kelas uji yang telah dijelaskan pada Tabel 4-5.

4.2.2.1 Pengujian *Login*

Login dilakukan untuk penentuan hak akses pengguna kepala bagian gudang dan pemimpin perusahaan. Pengujian login untuk data yang dimasukkan benar dapat dilihat pada Tabel 4-6.

Tabel 4-6 Pengujian *Login* Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukkan *username* pengguna	*Field username* sudah terisi	*Field username* sudah terisi	[√] Diterima [ ] Ditolak
Memasukkan *password* pengguna	*Field password* sudah terisi	*Field password* sudah terisi	[√] Diterima [ ] Ditolak
Memasukkan *username* dan *password* dengan benar dan menekan tombol *login*	Menampilkan halaman utama sesuai jabatan yang dimiliki oleh akun pengguna	Menampilkan halaman utama sesuai jabatan yang dimiliki oleh akun pengguna	[√] Diterima [ ] Ditolak

Pengujian untuk login dengan data yang dimasukkan oleh pengguna salah dapat dilihat pada Tabel 4-7.

Tabel 4-7 Pengujian Login Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Field username tidak diisi dan menekan tombol *login*	Menampilkan pesan "Data *username* belum diisi!"	Menampilkan pesan "Data *username* belum diisi!"	[√] Diterima [ ] Ditolak
Field password tidak diisi dan menekan tombol *login*	Menampilkan pesan "Data *password* belum diisi!"	Menampilkan pesan "Data *password* belum diisi!"	[√] Diterima [ ] Ditolak
Field username dan *password* diisi dengan data yang salah lalu menekan tombol *login*	Menampilkan pesan "*Username* atau *password* yang anda masukkan salah!"	Menampilkan pesan "*Username* atau *password* yang anda masukkan salah!"	[√] Diterima [ ] Ditolak

4.2.2.2 Pengujian Menentukan Inisialisasi Kolam

Menentukan inisialisasi kolam digunakan untuk membantu proses perhitungan perkiraan bobot ikan per kolam. Pengujian menentukan inisialisasi kolam dengan data benar dapat dilihat pada Tabel 4-8.

Tabel 4-8 Pengujian Menentukan Inisialisasi Kolam Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memilih pemilik kolam pada *comboBox* yang tersedia	*comboBox* pemilik kolam sudah dipilih	*comboBox* pemilik kolam sudah dipilih	[√] Diterima [ ] Ditolak
Memilih nama kolam yang akan di inisialisasi pada	*comboBox* nama kolam sudah diilih	*comboBox* nama kolam sudah diilih	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
comboBox yang tersedia			
Memilih tanggal sebar bibit ikan pada *dateTimePicker* yang tersedia	*dateTimePicker* tanggal sebar bibit ikan sudah dipilih	*dateTimePicker* tanggal sebar bibit ikan sudah dipilih	[√] Diterima [ ] Ditolak
Mengisi *field* perkiraan bobot ikan dengan angka	*Field* perkiraan bobot ikan sudah diisi dengan angka	*Field* perkiraan bobot ikan sudah diisi dengan angka	[√] Diterima [ ] Ditolak
Mengisi *field* jumlah petak kolam yang diisi bibit ikan	*Field* jumlah petak yang diisi bibit ikan sudah terisi	*Field* jumlah petak yang diisi bibit ikan sudah terisi	[√] Diterima [ ] Ditolak
Semua *field* pada *groupBox* inisialisasi kolam diisi dengan data yang benar lalu menekan tombol simpan	Menampilkan pesan 'Data berhasil disimpan!'	Menampilkan pesan 'Data berhasil disimpan!'	[√] Diterima [ ] Ditolak

Pengujian untuk menentukan inisialisasi kolam dengan data yang dimasukkan oleh pengguna salah dapat dilihat pada Tabel 4-9.

Tabel 4-9 Pengujian Menentukan Inisialisasi Kolam Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak memilih pemilik kolam pada *comboBox* yang tersedia	Muncul pesan "Pemilik kolam belum dipilih"	Muncul pesan "Pemilik kolam belum dipilih"	[√] Diterima [ ] Ditolak
Tidak memilih nama kolam yang akan di inisialisasi pada *comboBox* yang tersedia	Muncul pesan "Nama kolam belum dipilih"	Muncul pesan "Nama kolam belum dipilih"	[√] Diterima [ ] Ditolak
Tidak memilih tanggal sebar bibit ikan pada *dateTimePicker* yang tersedia	Muncul pesan "Tanggal sebar bibit kolam belum dipilih"	Muncul pesan "Tanggal sebar bibit kolam belum dipilih"	[√] Diterima [ ] Ditolak
Tidak mengisi *field* perkiraan bobot ikan dengan angka	Muncul pesan "Perkiraan bobot ikan per kolam harus berupa angka!"	Muncul pesan "Perkiraan bobot ikan per kolam harus berupa angka!"	[√] Diterima [ ] Ditolak
Tidak mengisi *fiels* perkiraan bobot ikan	Muncul pesan "Perkiraan bobot ikan per kolam harus diisi!"	Muncul pesan "Perkiraan bobot ikan per kolam harus diisi!"	[√] Diterima [ ] Ditolak
Tidak mengisi *field* jumlah petak kolam yang diisi bibit ikan	Muncul pesan "Jumlah petak harus diisi!!"	Muncul pesan "Jumlah petak harus diisi!!"	[√] Diterima [ ] Ditolak
Semua *field* pada *groupBox* inisialisasi kolam diisi dengan	Muncul pesan "Data gagal disimpan!"	Muncul pesan "Data gagal disimpan!"	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
data yang salah lalu menekan tombol simpan			

4.2.2.3 Pengujian Memilih Kolam Siap Panen

Memilih kolam siap panen digunakan pemimpin perusahaan untuk bahan pertimbangan menentukan kolam yang akan dipanen agar permintaan pasar dapat terpenuhi. Pengujian memilih kolam siap panen dengan data benar dapat dilihat pada Tabel 4-10.

Tabel 4-10 Pengujian Memilih Kolam Siap Panen Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memilih ukuran ikan yang diinginkan pada comboBox yang tersedia	Ukuran ikan yang diinginkan sudah dipilih	Ukuran ikan yang diinginkan sudah dipilih	[√] Diterima [ ] Ditolak
Memilih ukuran ikan dan menekan tombol cari kolam	Menampilkan daftar kolam yang sesuai dengan ukuran ikan yang dipilih pada *dataGridView* yang tersedia	Menampilkan daftar kolam yang sesuai dengan ukuran ikan yang dipilih pada *dataGridView* yang tersedia	[√] Diterima [ ] Ditolak

Pengujian untuk memilih kolam siap panen dengan data yang dimasukkan oleh pengguna salah dapat dilihat pada Tabel 4-11.

Tabel 4-11 P engujian Memilih Kolam Siap Panen Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak memilih ukuran ikan yang diinginkan pada comboBox yang tersedia	Muncul pesan "Ukuran ikan harus dipilih!"	Muncul pesan "Ukuran ikan harus dipilih!"	[√] Diterima [ ] Ditolak
Tidak memilih ukuran ikan dan menekan tombol cari kolam	Muncul pesan "Data gagal ditemukan!"	Muncul pesan "Data gagal ditemukan!"	[√] Diterima [ ] Ditolak

4.2.2.4 Pengujian Mencari Rute Pengiriman

Mencari rute pengiriman dilakukan untuk mencegah pengacakan pengiriman pakan oleh petugas pengiriman. Pengujian mencari rute terdekat dengan data benar dapat dilihat pada Tabel 4-12.

Tabel 4-12 Pengujian Mencari Rute Pengiriman Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memilih tanggal pemesanan pakan pada *dateTimePicker* yang tersedia	Tanggal pemesanan sudah dipilih	Tanggal pemesanan sudah dipilih	[√] Diterima [ ] Ditolak
Memilih tanggal pemesanan pakan dan menekan tombol tampilkan tujuan	Menampilkan daftar pemesanan pakan sesuai tanggal yang dipilih pada *dataGridView*	Menampilkan daftar pemesanan pakan sesuai tanggal yang dipilih pada *dataGridView*	[√] Diterima [ ] Ditolak
Terdapat daftar tujuan kolam pada *dataGridView* yang tersedia lalu menekan tombol urutkan rute tujuan	Menampilkan tujuan kolam yang sudah terurut berdasarkan jarak dan jumlah pakan yang dipesan pada *dataGridView*	Menampilkan tujuan kolam yang sudah terurut berdasarkan jarak dan jumlah pakan yang dipesan pada *dataGridView*	[√] Diterima [ ] Ditolak

Pengujian untuk mencari rute pengiriman dengan data yang dimasukkan oleh pengguna salah dapat dilihat pada Tabel 4-13.

Tabel 4-13 P engujian Mencari Rute Pengiriman Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak memilih tanggal pemesanan pakan pada *dateTimePicker* yang tersedia	Muncul pesan ‘Tanggal pemesanan belum dipilih!’	Muncul pesan ‘Tanggal pemesanan belum dipilih!’	[√] Diterima [ ] Ditolak
Tidak memilih tanggal pemesanan pakan dan menekan tombol tampilkan tujuan	Muncul pesan ‘Tanggal pemesanan belum dipilih!’	Muncul pesan ‘Tanggal pemesanan belum dipilih!’	[√] Diterima [ ] Ditolak
Belum ada daftar tujuan kolam pada *dataGridView* yang tersedia lalu menekan tombol urutkan rute tujuan	Muncul pesan ‘Belum ada daftar pesanan yang akan diurutkan!’	Muncul pesan ‘Belum ada daftar pesanan yang akan diurutkan!’	[√] Diterima [ ] Ditolak

4.2.2.5 Pengujian Memesan Pakan Ke Pemasok

Memesan pakan ke pemasok dilakukan untuk menjaga stok pakan yang ada digudang CV. Jatiluhur Mas. Pengujian memesan pakan ke pemasok dengan data benar dapat dilihat pada Tabel 4-14.

Tabel 4-14 Pengujian Memesan Pakan Ke Pemasok Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi *field* jumlah pakan dengan angka	*Field* jumlah pakan diisi dengan angka	*Field* jumlah pakan diisi dengan angka	[√] Diterima [ ] Ditolak
Memilih jenis pakan yang tersedia pada *comboBox*	Jenis pakan sudah dipilih dari *comboBox* yang tersedia	Jenis pakan sudah dipilih dari *comboBox* yang tersedia	[√] Diterima [ ] Ditolak
Pilih tanggal masuk pakan ke gudang pada *dateTimePicker* yang tersedia	Tanggal masuk pakan ke gudang sudah dipilih	Tanggal masuk pakan ke gudang sudah dipilih	[√] Diterima [ ] Ditolak
Pilih tanggal tenggat pembayaran pakan ke pemasok pada *dateTimePicker* yang tersedia	Tanggal tenggat pembayaran ke pemasok sudah dipilih	Tanggal tenggat pembayaran ke pemasok sudah dipilih	[√] Diterima [ ] Ditolak
Mengisi semua *field* yang ada pada *groupBox* pemesanan pakan ke pemasok dengan data yang benar lalu menekan tombol simpan	Menampilkan pesan 'Data berhasil disimpan!'	Menampilkan pesan 'Data berhasil disimpan!'	[√] Diterima [ ] Ditolak

Pengujian untuk memesan pakan ke pemasok dengan data yang dimasukkan oleh pengguna salah dapat dilihat pada Tabel 4-15.

Tabel 4-15 P engujian Memesan Pakan Ke Pemasok Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak mengisi *field* jumlah pakan dengan angka	Muncul pesan 'Jumlah pakan belum diisi!'	Muncul pesan 'Jumlah pakan belum diisi!'	[√] Diterima [ ] Ditolak
Tidak memilih jenis pakan yang tersedia pada *comboBox*	Muncul pesan 'Jenis pakan belum dipilih!'	Muncul pesan 'Jenis pakan belum dipilih!'	[√] Diterima [ ] Ditolak
Tidak memilih tanggal masuk pakan ke gudang pada *dateTimePicker* yang tersedia	Muncul pesan 'Taggal pakan masuk gudang belum dipilih!'	Muncul pesan 'Taggal pakan masuk gudang belum dipilih!'	[√] Diterima [ ] Ditolak
Tidak memilih tanggal tenggat pembayaran pakan ke pemasok pada *dateTimePicker* yang tersedia	Muncul pesan 'Tanggal tenggat pembayaran ke pemasok belum dipilih!'	Muncul pesan 'Tanggal tenggat pembayaran ke pemasok belum dipilih!'	[√] Diterima [ ] Ditolak
Mengisi semua *field* yang ada pada *groupBox* pemesanan pakan ke pemasok	Muncul pesan 'Data gagal disimpan!'	Muncul pesan 'Data gagal disimpan!'	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
dengan data yang salah lalu menekan tombol simpan			

4.2.2.6 Pengujian Melakukan Pembayaran Ke Pemasok

Melakukan pembayaran ke pemasok dilakukan untuk mengolah data pembayaran pakan CV. Jatiluhur Mas kepada pemasok. Pengujian melakukan pembayaran ke pemasok dengan data benar dapat dilihat pada Tabel 4-16.

Tabel 4-16 Pengujian Melakukan Pembayaran Ke Pemasok Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi *field* jumlah pembayaran	*Field* jumlah pembayaran sudah diisi	*Field* jumlah pembayaran sudah diisi	[√] Diterima [ ] Ditolak
Memilih jenis pembayaran pada *comboBox* yang tersedia	Jenis pembayaran sudah dipilih	Jenis pembayaran sudah dipilih	[√] Diterima [ ] Ditolak
Memilih bukti pembayaran berupa foto berformat .jpg atau .png	Bukti pembayaran sudah dipilih dan berformat .jpg ataupun .png	Bukti pembayaran sudah dipilih dan berformat .jpg ataupun .png	[√] Diterima [ ] Ditolak
Memilih tanggal pembayaran pada *dateTimePicker* yang tersedia	Sudah memiih tanggal pembayaran	Sudah memiih tanggal pembayaran	[√] Diterima [ ] Ditolak
Mengisi semua *field* yang tersedia pada *groupBox* pembayaran pakan ke pemasok dengan data yang benar lalu menekan tombol simpan	Muncul pesan 'Data berhasil disimpan!'	Muncul pesan 'Data berhasil disimpan!'	[√] Diterima [ ] Ditolak

Pengujian untuk melakukan pembayaran ke pemasok dengan data yang dimasukkan oleh pengguna salah dapat dilihat pada Tabel 4-17.

Tabel 4-17 P engujian Melakukan Pembayaran Ke Pemasok Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak mengisi *field* jumlah pembayaran	Muncul pesan 'Jumlah pembayaran belum diisi!'	Muncul pesan 'Jumlah pembayaran belum diisi!'	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak memilih jenis pembayaran pada *comboBox* yang tersedia	Muncul pesan 'Jenis pembayaran belum dipilih!'	Muncul pesan 'Jenis pembayaran belum dipilih!'	[√] Diterima [ ] Ditolak
Tidak memilih bukti pembayaran berupa foto berformat .jpg atau .png	Muncul pesan 'Bukti pembayaran belu dipilih!'	Muncul pesan 'Bukti pembayaran belu dipilih!'	[√] Diterima [ ] Ditolak
Tidak memilih tanggal pembayaran pada *dateTimePicker* yang tersedia	Muncul pesan 'Tanggal pembyaran belum dipilih!'	Muncul pesan 'Tanggal pembyaran belum dipilih!'	[√] Diterima [ ] Ditolak
Mengisi semua *field* yang tersedia pada *groupBox* pembayaran pakan ke pemasok dengan data yang yang salah lalu menekan tombol simpan	Muncul pesan 'Data gagal disimpan!'	Muncul pesan 'Data gagal disimpan!'	[√] Diterima [ ] Ditolak

4.2.2.7 Pengujian Memproses Pengiriman Pakan

Memproses pengiriman pakan dilakukan untuk mengolah data pakan yang akan dikirim berdasarkan pemesanan yang dilakukan oleh mitra petani. Pengujian memproses pengiriman pakan dengan data benar dapat dilihat pada Tabel 4-18.

Tabel 4-18 Pengujian Memproses Pengiriman Pakan Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi *field* jumlah pakan yang dengan angka	*Field* jumlah pakan sudah diisi	*Field* jumlah pakan sudah diisi	[√] Diterima [ ] Ditolak
Memilih pemilik kolam pada *comboBox* yang tersedia	Pemilik kolam udah dipilih	Pemilik kolam udah dipilih	[√] Diterima [ ] Ditolak
Memilih nama kolam pada *comboBox* yang tersedia	Nama kolam sudah dipilih	Nama kolam sudah dipilih	[√] Diterima [ ] Ditolak
Memilih jenis pakan pada *comboBox* yang tersedia	Jenis pakan sudah dipilih	Jenis pakan sudah dipilih	[√] Diterima [ ] Ditolak
Memilih tanggal pengiriman pada *dateTimePicker* yang tersedia	Tanggal pengiriman sudah dipilih	Tanggal pengiriman sudah dipilih	[√] Diterima [ ] Ditolak
Mengisi semua *field* yang ada pada *groupBox* pengiriman	Muncul pesan 'Data berhasil disimpan!'	Muncul pesan 'Data berhasil disimpan!'	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
pakan dengan data yang benar dan menekan tombol simpan			

Pengujian untuk memproses pengiriman pakan dengan data yang dimasukkan oleh pengguna salah dapat dilihat pada Tabel 4-19.

Tabel 4-19 P engujian Memproses Pengiriman Pakan Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak mengisi *field* jumlah pakan yang dengan angka	Muncul pesan 'Jumlah pakan belum diisi!'	Muncul pesan 'Jumlah pakan belum diisi!'	[√] Diterima [ ] Ditolak
Tidak memilih pemilik kolam pada *comboBox* yang tersedia	Muncul pesan 'Pemilik kolam belum dipilih!'	Muncul pesan 'Pemilik kolam belum dipilih!'	[√] Diterima [ ] Ditolak
Tidak memilih nama kolam pada *comboBox* yang tersedia	Muncul pesan 'Nama kolam belum dipilih!'	Muncul pesan 'Nama kolam belum dipilih!'	[√] Diterima [ ] Ditolak
Tidak memilih jenis pakan pada *comboBox* yang tersedia	Muncul pesan 'Jenis pakan belum dipilih!'	Muncul pesan 'Jenis pakan belum dipilih!'	[√] Diterima [ ] Ditolak
Tidak memilih tanggal pengiriman pada *dateTimePicker* yang tersedia	Muncul pesan 'Tanggal pengiriman belum dipilih!'	Muncul pesan 'Tanggal pengiriman belum dipilih!'	[√] Diterima [ ] Ditolak
Mengisi semua *field* yang ada pada *groupBox* pengiriman pakan dengan data yang salah dan menekan tombol simpan	Muncul pesan 'Data gaga disimpan!'	Muncul pesan 'Data gaga disimpan!'	[√] Diterima [ ] Ditolak

4.2.2.8 Pengujian Memverifikasi Pengiriman Pakan

Memverifikasi pengiriman pakan dilakukan untuk mengontrol apakah petugas sudah melakukan tugasnya sesuai dengan yang diharapkan. Pengujian memverifikasi pengiriman pakan dengan data benar dapat dilihat pada Tabel 4-20.

Tabel 4-20 Pengujian Memverifikasi Pengiriman Pakan Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memilih tanggal pengiriman pada *comboBox* yang tersedia	Tanggal pengiriman sudah dipilih	Tanggal pengiriman sudah dipilih	[√] Diterima [ ] Ditolak
Memilih tujuan kolam pada *comboBox* yang tersedia	Tujuan kolam sudah dipilih	Tujuan kolam sudah dipilih	[√] Diterima [ ] Ditolak
Memilih keterangan pengiriman pada *comboBox* yang tersedia	Keterangan pengiriman sudah dipilih	Keterangan pengiriman sudah dipilih	[√] Diterima [ ] Ditolak
Mengisi urutan sampai pada *field* yang tersedia	Urutan pengantaran pakan sudah dipilih	Urutan pengantaran pakan sudah dipilih	[√] Diterima [ ] Ditolak
Mengisi waktu sampai pada *field* yang tersedia	Waktu pakan sampai sudah diisi	Waktu pakan sampai sudah diisi	[√] Diterima [ ] Ditolak
Memilih petugas pengiriman pada *comboBox* yang tersedia	Petugas pengiriman sudah dipilih	Petugas pengiriman sudah dipilih	[√] Diterima [ ] Ditolak
Memilih petugas penerima pada *comboBox* yang tersedia	Petugas penerima pakan sudah dipilih	Petugas penerima pakan sudah dipilih	[√] Diterima [ ] Ditolak
Mengisi semua *field* yang ada pada form verifikasi dengan data yang benar dan menekan tombol simpan	Muncul pesan ‘Data berhasil disimpan!’	Muncul pesan ‘Data berhasil disimpan!’	[√] Diterima [ ] Ditolak

Pengujian untuk memverifikasi pengiriman pakan dengan data yang dimasukkan oleh pengguna salah dapat dilihat pada Tabel 4-21.

Tabel 4-21 Pengujian Memverifikasi Pengiriman Pakan Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak memilih tanggal pengiriman pada *comboBox* yang tersedia	Muncul pesan ‘Tanggal pengiriman belum dipilih!’	Muncul pesan ‘Tanggal pengiriman belum dipilih!’	[√] Diterima [ ] Ditolak
Tidak memilih tujuan kolam pada *comboBox* yang tersedia	Muncul pesan ‘Kolam tujuan belum dipilih!’	Muncul pesan ‘Kolam tujuan belum dipilih!’	[√] Diterima [ ] Ditolak
Tidak memilih keterangan pengiriman pada	Muncul pesan ‘Keterangan	Muncul pesan ‘Keterangan	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
comboBox yang tersedia	pengiriman belum dipilih!'	pengiriman belum dipilih!'	
Tidak mengisi urutan sampai pada *field* yang tersedia	Muncul pesan 'Urutan pengiriman belum diisi!'	Muncul pesan 'Urutan pengiriman belum diisi!'	[√] Diterima [ ] Ditolak
Tidak mengisi waktu sampai pada *field* yang tersedia	Muncul pesan 'Waktu pakan sampai belum diisi!'	Muncul pesan 'Waktu pakan sampai belum diisi!'	[√] Diterima [ ] Ditolak
Tidak memilih petugas pengiriman pada *comboBox* yang tersedia	Muncul pesan 'Petugas pengiriman belum dipilih!'	Muncul pesan 'Petugas pengiriman belum dipilih!'	[√] Diterima [ ] Ditolak
Tidak memilih petugas penerima pada *comboBox* yang tersedia	Muncul pesan 'Petugas penerima pakan belum dipilih!'	Muncul pesan 'Petugas penerima pakan belum dipilih!'	[√] Diterima [ ] Ditolak
Mengisi semua *field* yang ada pada form verifikasi dengan data yang salah dan menekan tombol simpan	Muncul pesan 'Data gagal disimpan!'	Muncul pesan 'Data gagal disimpan!'	[√] Diterima [ ] Ditolak

4.2.2.9 Pengujian Menambahkan Data Mitra

Menambahkan data mitra dilakukan untuk menambahkan data mitra petani baru pada *database.* Pengujian menambahkan data mitra dengan data benar dapat dilihat pada Tabel 4-22.

Tabel 4-22 Pengujian Menambahkan Data Mitra Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi *field* nama mitra	*Field* nama mitra sudah diisi	*Field* nama mitra sudah diisi	[√] Diterima [ ] Ditolak
Mengisi *field* alamat mitra	*Field* alamat mitra sudah diisi	*Field* alamat mitra sudah diisi	[√] Diterima [ ] Ditolak
Mengisi *field* nomor telepon mitra	*Field* nomor telepon mitra sudah diisi	*Field* nomor telepon mitra sudah diisi	[√] Diterima [ ] Ditolak
Memilih jenis kerja sama mitra pada *comboBox yang tersedia*	Jenis kerja sama sudah dipilih	Jenis kerja sama sudah dipilih	[√] Diterima [ ] Ditolak
Memilih jenis mitra pada *comboBox* yang tersedia	Jenis mitra sudah dipilih	Jenis mitra sudah dipilih	[√] Diterima [ ] Ditolak
Mengisi *field* setoran awal mitra	*Field* setoran awal mitra sudah diisi	*Field* setoran awal mitra sudah diisi	[√] Diterima [ ] Ditolak
Memilih tanggal bergabung mitra	Tanggal gabung mitra sudah dipilih	Tanggal gabung mitra sudah dipilih	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi *field* banyak kolam yang dimiliki mitra	*Field* banyak kolam mitra sudah diisi	*Field* banyak kolam mitra sudah diisi	[√] Diterima [ ] Ditolak
Mengisi semua *field* yang ada pada *groupBox* tambah mitra petani dengan data yang benar dan menekan tombol simpan	Muncul pesan 'Data berhasil disimpan!'	Muncul pesan 'Data berhasil disimpan!'	[√] Diterima [ ] Ditolak

Pengujian untuk menambahkan data mitra dengan data yang dimasukkan oleh pengguna salah dapat dilihat pada Tabel 4-23.

Tabel 4-23 P engujian Menambahkan Data Mitra Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak mengisi *field* nama mitra	Muncul pesan 'Nama mitra belum diisi!'	Muncul pesan 'Nama mitra belum diisi!'	[√] Diterima [ ] Ditolak
Tidak mengisi *field* alamat mitra	Muncul pesan 'Alamat mitra belum diisi!'	Muncul pesan 'Alamat mitra belum diisi!'	[√] Diterima [ ] Ditolak
Tidak mengisi *field* nomor telepon mitra	Muncul pesan 'Nomor telepon belum diisi!'	Muncul pesan 'Nomor telepon belum diisi!'	[√] Diterima [ ] Ditolak
Tidak memilih jenis kerja sama mitra pada *comboBox yang tersedia*	Muncul pesan 'Jenisa kerja sama belum dipilih!'	Muncul pesan 'Jenisa kerja sama belum dipilih!'	[√] Diterima [ ] Ditolak
Tidak memilih jenis mitra pada *comboBox* yang tersedia	Muncul pesan 'Jenisa mitra belum dipilih!'	Muncul pesan 'Jenisa mitra belum dipilih!'	[√] Diterima [ ] Ditolak
Tidak mengisi *field* setoran awal mitra	Muncul pesan 'Setoran awal belum diisi!'	Muncul pesan 'Setoran awal belum diisi!'	[√] Diterima [ ] Ditolak
Tidak memilih tanggal bergabung mitra	Muncul pesan 'Tanggal bergabung belum dipilih!'	Muncul pesan 'Tanggal bergabung belum dipilih!'	[√] Diterima [ ] Ditolak
Tidak mengisi *field* banyak kolam yang dimiliki mitra	Muncul pesan 'Banyak kolam belum diisi!'	Muncul pesan 'Banyak kolam belum diisi!'	[√] Diterima [ ] Ditolak
Mengisi semua *field* yang ada pada *groupBox* tambah mitra petani dengan data yang salah dan menekan tombol simpan	Muncul pesan 'Data gagal disimpan!'	Muncul pesan 'Data gagal disimpan!'	[√] Diterima [ ] Ditolak

4.2.2.10 Pengujian Menambahkan Data Area Kolam

Menambahkan data area kolam dilakukan jika ada penambahan area baru mulik mitra petani yang sebelumnya tidak ada. Pengujian menambahkan data area kolam dengan data benar dapat dilihat pada Tabel 4-24.

Tabel 4-24 Pengujian Menambahkan Data Area Kolam Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi *field* nama area	*Field* nama area sudah diisi	*Field* nama area sudah diisi	[√] Diterima [ ] Ditolak
Mengisi *field* koordinat longitude area	*Field* koordinat longitude area sudah diisi	*Field* koordinat longitude area sudah diisi	[√] Diterima [ ] Ditolak
Mengisi *field* koordinat latitude area	*Field* koordinat latitude area sudah diisi	*Field* koordinat latitude area sudah diisi	[√] Diterima [ ] Ditolak
Mengisi semua *field* pada *groupBox* tambah area dengan data yang benar lalu menekan tombol simpan	Muncul pesan 'Data berhasil disimpan!'	Muncul pesan 'Data berhasil disimpan!'	[√] Diterima [ ] Ditolak

Pengujian untuk menambahkan data area kolam dengan data yang dimasukkan oleh pengguna salah dapat dilihat pada Tabel 4-25.

Tabel 4-25 P engujian Menambahkan Data Area Kolam Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak mengisi *field* nama area	Muncul pesan 'Nama area belum diisi!'	Muncul pesan 'Nama area belum diisi!'	[√] Diterima [ ] Ditolak
Tidak mengisi *field* koordinat longitude area	Muncul pesan 'Koordinat longitude area belum diisi!'	Muncul pesan 'Koordinat longitude area belum diisi!'	[√] Diterima [ ] Ditolak
Tidak mengisi *field* koordinat latitude area	Muncul pesan 'Koordinat latitude area belum diisi!'	Muncul pesan 'Koordinat latitude area belum diisi!'	[√] Diterima [ ] Ditolak
Mengisi semua *field* pada *groupBox* tambah area dengan data yang salah lalu menekan tombol simpan	Muncul pesan 'Data gagal disimpan!'	Muncul pesan 'Data gagal disimpan!'	[√] Diterima [ ] Ditolak

4.2.2.11 Pengujian Menambahkan Data Kolam

Menambahkan data kolam dilakukan untuk menambahkan data kolam baru yang dimiliki mitra petani. Pengujian menambahkan data kolam menggunakan data yang benar dapat dilihat pada Tabel 4-26.

Tabel 4-26 Pengujian Menambahkan Data Kolam Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi *field* nama kolam	*Field* nama kolam sudah diisi	*Field* nama kolam sudah diisi	[√] Diterima [ ] Ditolak
Memilih pemilik kolam pada *comboBox* yang tersedia	Pemilik kolam sudah dipilih	Pemilik kolam sudah dipilih	[√] Diterima [ ] Ditolak
Memilih letak area kolam pada *comboBox* yang tersedia	Letak area kolam sudah dipilih	Letak area kolam sudah dipilih	[√] Diterima [ ] Ditolak
Memilih tanggal sebar bibit ikan pada *dateTimePicker* yang tersedia	Tanggal sebar bibit ikan di kolam sudah dipilih	Tanggal sebar bibit ikan di kolam sudah dipilih	[√] Diterima [ ] Ditolak
Memilih petugas pengurus kolam pada *comboBox* yang tersedia	Petugas pengurus kolam sudah dipilih	Petugas pengurus kolam sudah dipilih	[√] Diterima [ ] Ditolak
Mengisi *field* banyak petak dengan angka	Banyak petak pada kolam sudah diisi	Banyak petak pada kolam sudah diisi	[√] Diterima [ ] Ditolak
Mengisi semua *field* yang ada pada *groupBox* tambah data kolam dengan data yang benar dan menekan tombol simpan	Muncul pesan ‘Data berhasil disimpan!’	Muncul pesan ‘Data berhasil disimpan!’	[√] Diterima [ ] Ditolak

Pengujian untuk menambahkan data kolam dengan data yang dimasukkan oleh pengguna salah dapat dilihat pada Tabel 4-27.

Tabel 4-27 P engujian Menambahkan Data Kolam Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak mengisi *field* nama kolam	Muncul pesan ‘Nama kolam belum diisi!’	Muncul pesan ‘Nama kolam belum diisi!’	[√] Diterima [ ] Ditolak
Tidak memilih pemilik kolam pada *comboBox* yang tersedia	Muncul pesan ‘Pemelik kolam belum dipilih’	Muncul pesan ‘Pemelik kolam belum dipilih’	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak memilih letak area kolam pada *comboBox* yang tersedia	Muncul pesan 'Letak area kolam belum dipilih!'	Muncul pesan 'Letak area kolam belum dipilih!'	[√] Diterima [ ] Ditolak
Tidak memilih tanggal sebar bibit ikan pada *dateTimePicker* yang tersedia	Muncul pesan 'Tanggal sebar bibit ikan di kolam belum diisi!'	Muncul pesan 'Tanggal sebar bibit ikan di kolam belum diisi!'	[√] Diterima [ ] Ditolak
Tidak memilih petugas pengurus kolam pada *comboBox* yang tersedia	Muncul pesan 'Petugas pengurus kolam belum dipilih!'	Muncul pesan 'Petugas pengurus kolam belum dipilih!'	[√] Diterima [ ] Ditolak
Tidak mengisi *field* banyak petak dengan angka	Muncul pesan 'Banyak petak di kolam belum diisi!'	Muncul pesan 'Banyak petak di kolam belum diisi!'	[√] Diterima [ ] Ditolak
Mengisi semua *field* yang ada pada *groupBox* tambah data kolam dengan data yang salah dan menekan tombol simpan	Muncul pesan 'Data gagal disimpan!'	Muncul pesan 'Data gagal disimpan!'	[√] Diterima [ ] Ditolak

4.2.2.12 Pengujian Menambahkan Data Petugas

Menambahkan data petugas dilakukan untuk menambahkan data petugas baru yang bertugas mengirimi pakan maupun petugas yang mengurus kolam. Pengujian mmenambahkan data petugas menggunakan data yang benar dapat dilihat pada Tabel 4-28.

Tabel 4-28 P engujian Menambahkan Data Petugas Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi *field* nama petugas	*Field* nama petugas sudah diisi	*Field* nama petugas sudah diisi	[√] Diterima [ ] Ditolak
Mengisi *field* alamat petugas	*Field* alamat petugas sudah diisi	*Field* alamat petugas sudah diisi	[√] Diterima [ ] Ditolak
Mengisi *field* nomor telepon petugas	*Field* nomor telelpon petugas sudah diisi	*Field* nomor telelpon petugas sudah diisi	[√] Diterima [ ] Ditolak
Mengisi semua *field* pada *groupBox* tambah data petugas dengan benar dan menekan tombol simpan	Muncul pesan 'Data berhasil disimpan!'	Muncul pesan 'Data berhasil disimpan!'	[√] Diterima [ ] Ditolak

Pengujian untuk menambahkan data petugas dengan data yang dimasukkan oleh pengguna salah dapat dilihat pada Tabel 4-29.

Tabel 4-29 P engujian Menambahkan Data Petugas Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak mengisi *field* nama petugas	Muncul pesan ’Nama petugas belum diisi!’	Muncul pesan ’Nama petugas belum diisi!’	[√] Diterima [ ] Ditolak
Tidak mengisi *field* alamat petugas	Muncul pesan ‘Alamat petugas belum diisi!’	Muncul pesan ‘Alamat petugas belum diisi!’	[√] Diterima [ ] Ditolak
Tidak mengisi *field* nomor telepon petugas	Muncul pesan ‘Nomor telepon petugas belum diisi!’	Muncul pesan ‘Nomor telepon petugas belum diisi!’	[√] Diterima [ ] Ditolak
Mengisi semua *field* pada *groupBox* tambah data petugas dengan salah dan menekan tombol simpan	Muncul pesan ‘Data gagal disimpan!’	Muncul pesan ‘Data gagal disimpan!’	[√] Diterima [ ] Ditolak

4.2.2.13 Pengujian Mengubah Data Mitra

Mengubah data mitra dilakukan untuk memperbarui data mitra yang sudha ada sebelumnya. Pengujian mengubah data itra menggunakan data yang benar dapat dilihat pada Tabel 4-30.

Tabel 4-30 Pengujian Mengubah Data Mitra Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi *field* nama mitra	*Field* nama mitra sudah diisi	*Field* nama mitra sudah diisi	[√] Diterima [ ] Ditolak
Mengisi *field* alamat mitra	*Field* alamat mitra sudah diisi	*Field* alamat mitra sudah diisi	[√] Diterima [ ] Ditolak
Mengisi *field* nomor telepon mitra	*Field* nomor telepon mitra sudah diisi	*Field* nomor telepon mitra sudah diisi	[√] Diterima [ ] Ditolak
Memilih jenis kerja sama mitra pada *comboBox yang tersedia*	Jenis kerja sama sudah dipilih	Jenis kerja sama sudah dipilih	[√] Diterima [ ] Ditolak
Memilih jenis mitra pada *comboBox* yang tersedia	Jenis mitra sudah dipilih	Jenis mitra sudah dipilih	[√] Diterima [ ] Ditolak
Mengisi *field* setoran awal mitra	*Field* setoran awal mitra sudah diisi	*Field* setoran awal mitra sudah diisi	[√] Diterima [ ] Ditolak
Memilih tanggal bergabung mitra	Tanggal gabung mitra sudah dipilih	Tanggal gabung mitra sudah dipilih	[√] Diterima [ ] Ditolak
Mengisi *field* banyak kolam yang dimiliki mitra	*Field* banyak kolam mitra sudah diisi	*Field* banyak kolam mitra sudah diisi	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi semua *field* yang ada pada *groupBox* tambah mitra petani dengan data yang benar dan menekan tombol simpan	Muncul pesan 'Data berhasil disimpan!'	Muncul pesan 'Data berhasil disimpan!'	[√] Diterima [ ] Ditolak

Pengujian untuk mengubah data mitra dengan data yang dimasukkan oleh pengguna salah dapat dilihat pada Tabel 4-31.

Tabel 4-31 P engujian Mengubah Data Mitra Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak mengisi *field* nama mitra	Muncul pesan 'Nama mitra belum diisi!'	Muncul pesan 'Nama mitra belum diisi!'	[√] Diterima [ ] Ditolak
Tidak mengisi *field* alamat mitra	Muncul pesan 'Alamat mitra belum diisi!'	Muncul pesan 'Alamat mitra belum diisi!'	[√] Diterima [ ] Ditolak
Tidak mengisi *field* nomor telepon mitra	Muncul pesan 'Nomor telepon belum diisi!'	Muncul pesan 'Nomor telepon belum diisi!'	[√] Diterima [ ] Ditolak
Tidak memilih jenis kerja sama mitra pada *comboBox yang tersedia*	Muncul pesan 'Jenisa kerja sama belum dipilih!'	Muncul pesan 'Jenisa kerja sama belum dipilih!'	[√] Diterima [ ] Ditolak
Tidak memilih jenis mitra pada *comboBox* yang tersedia	Muncul pesan 'Jenisa mitra belum dipilih!'	Muncul pesan 'Jenisa mitra belum dipilih!'	[√] Diterima [ ] Ditolak
Tidak mengisi *field* setoran awal mitra	Muncul pesan 'Setoran awal belum diisi!'	Muncul pesan 'Setoran awal belum diisi!'	[√] Diterima [ ] Ditolak
Tidak memilih tanggal bergabung mitra	Muncul pesan 'Tanggal bergabung belum dipilih!'	Muncul pesan 'Tanggal bergabung belum dipilih!'	[√] Diterima [ ] Ditolak
Tidak mengisi *field* banyak kolam yang dimiliki mitra	Muncul pesan 'Banyak kolam belum diisi!'	Muncul pesan 'Banyak kolam belum diisi!'	[√] Diterima [ ] Ditolak
Mengisi semua *field* yang ada pada *groupBox* tambah mitra petani dengan data yang salah dan menekan tombol simpan	Muncul pesan 'Data gagal disimpan!'	Muncul pesan 'Data gagal disimpan!'	[√] Diterima [ ] Ditolak

4.2.2.14 Pengujian Mengubah Data Area Kolam

Mengubah data area kolam dilakukan untuk memperbarui data area kolam yang sudah ada sebelumnya. Pengujian mengubah data area kolam menggunakan data yang benar dapat dilihat pada Tabel 4-32.

Tabel 4-32 Pengujian Mengubah Data Area Kolam Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi *field* nama area	*Field* nama area sudah diisi	*Field* nama area sudah diisi	[√] Diterima [ ] Ditolak
Mengisi *field* koordinat longitude area	*Field* koordinat longitude area sudah diisi	*Field* koordinat longitude area sudah diisi	[√] Diterima [ ] Ditolak
Mengisi *field* koordinat latitude area	*Field* koordinat latitude area sudah diisi	*Field* koordinat latitude area sudah diisi	[√] Diterima [ ] Ditolak
Mengisi semua *field* pada *groupBox* tambah area dengan data yang benar lalu menekan tombol simpan	Muncul pesan 'Data berhasil disimpan!'	Muncul pesan 'Data berhasil disimpan!'	[√] Diterima [ ] Ditolak

Pengujian untuk mengubah data area kolam dengan data yang dimasukkan oleh pengguna salah dapat dilihat pada Tabel 4-33.

Tabel 4-33 P engujian Mengubah Data Area Kolam Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak mengisi *field* nama area	Muncul pesan 'Nama area belum diisi!'	Muncul pesan 'Nama area belum diisi!'	[√] Diterima [ ] Ditolak
Tidak mengisi *field* koordinat longitude area	Muncul pesan 'Koordinat longitude area belum diisi!'	Muncul pesan 'Koordinat longitude area belum diisi!'	[√] Diterima [ ] Ditolak
Tidak mengisi *field* koordinat latitude area	Muncul pesan 'Koordinat latitude area belum diisi!'	Muncul pesan 'Koordinat latitude area belum diisi!'	[√] Diterima [ ] Ditolak
Mengisi semua *field* pada *groupBox* tambah area dengan data yang salah lalu menekan tombol simpan	Muncul pesan 'Data gagal disimpan!'	Muncul pesan 'Data gagal disimpan!'	[√] Diterima [ ] Ditolak

4.2.2.15 Pengujian Mengubah Data Kolam

Mengubah data kolam dilakukan untuk memperbarui data kolam yang sudah ada sebelumnya. Pengujian mengubah data kolm menggunakan data yang benar dapat dilihat pada Tabel 4-34.

Tabel 4-34 Pengujian Mengubah Data Kolam Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi *field* nama kolam	*Field* nama kolam sudah diisi	*Field* nama kolam sudah diisi	[√] Diterima [ ] Ditolak
Memilih pemilik kolam pada *comboBox* yang tersedia	Pemilik kolam sudah dipilih	Pemilik kolam sudah dipilih	[√] Diterima [ ] Ditolak
Memilih letak area kolam pada *comboBox* yang tersedia	Letak area kolam sudah dipilih	Letak area kolam sudah dipilih	[√] Diterima [ ] Ditolak
Memilih tanggal sebar bibit ikan pada *dateTimePicker* yang tersedia	Tanggal sebar bibit ikan di kolam sudah dipilih	Tanggal sebar bibit ikan di kolam sudah dipilih	[√] Diterima [ ] Ditolak
Memilih petugas pengurus kolam pada *comboBox* yang tersedia	Petugas pengurus kolam sudah dipilih	Petugas pengurus kolam sudah dipilih	[√] Diterima [ ] Ditolak
Mengisi *field* banyak petak dengan angka	Banyak petak pada kolam sudah diisi	Banyak petak pada kolam sudah diisi	[√] Diterima [ ] Ditolak
Mengisi semua *field* yang ada pada *groupBox* tambah data kolam dengan data yang benar dan menekan tombol simpan	Muncul pesan ‘Data berhasil disimpan!’	Muncul pesan ‘Data berhasil disimpan!’	[√] Diterima [ ] Ditolak

Pengujian untuk mengubah data kolam dengan data yang dimasukkan oleh pengguna salah dapat dilihat pada Tabel 4-35.

Tabel 4-35 P engujian Mengubah Data Kolam Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak mengisi *field* nama kolam	Muncul pesan ‘Nama kolam belum diisi!’	Muncul pesan ‘Nama kolam belum diisi!’	[√] Diterima [ ] Ditolak
Tidak memilih pemilik kolam pada *comboBox* yang tersedia	Muncul pesan ‘Pemelik kolam belum dipilih’	Muncul pesan ‘Pemelik kolam belum dipilih’	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak memilih letak area kolam pada *comboBox* yang tersedia	Muncul pesan 'Letak area kolam belum dipilih!'	Muncul pesan 'Letak area kolam belum dipilih!'	[√] Diterima [ ] Ditolak
Tidak memilih tanggal sebar bibit ikan pada *dateTimePicker* yang tersedia	Muncul pesan 'Tanggal sebar bibit ikan di kolam belum diisi!'	Muncul pesan 'Tanggal sebar bibit ikan di kolam belum diisi!'	[√] Diterima [ ] Ditolak
Tidak memilih petugas pengurus kolam pada *comboBox* yang tersedia	Muncul pesan 'Petugas pengurus kolam belum dipilih!'	Muncul pesan 'Petugas pengurus kolam belum dipilih!'	[√] Diterima [ ] Ditolak
Tidak mengisi *field* banyak petak dengan angka	Muncul pesan 'Banyak petak di kolam belum diisi!'	Muncul pesan 'Banyak petak di kolam belum diisi!'	[√] Diterima [ ] Ditolak
Mengisi semua *field* yang ada pada *groupBox* tambah data kolam dengan data yang salah dan menekan tombol simpan	Muncul pesan 'Data gagal disimpan!'	Muncul pesan 'Data gagal disimpan!'	[√] Diterima [ ] Ditolak

4.2.2.16 Pengujian Mengubah Data Petugas

Mengubah data petugas dilakukan untuk memperbarui data petugas yang sudah ada sebelumnya. Pengujian mengubah data petugas menggunakan data yang benar dapat dilihat pada Tabel 4-36.

Tabel 4-36 Pengujian Mengubah Data Petugas Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi *field* nama petugas	*Field* nama petugas sudah diisi	*Field* nama petugas sudah diisi	[√] Diterima [ ] Ditolak
Mengisi *field* alamat petugas	*Field* alamat petugas sudah diisi	*Field* alamat petugas sudah diisi	[√] Diterima [ ] Ditolak
Mengisi *field* nomor telepon petugas	*Field* nomor telelpon petugas sudah diisi	*Field* nomor telelpon petugas sudah diisi	[√] Diterima [ ] Ditolak
Mengisi semua *field* pada *groupBox* tambah data petugas dengan benar dan menekan tombol simpan	Muncul pesan 'Data berhasil disimpan!'	Muncul pesan 'Data berhasil disimpan!'	[√] Diterima [ ] Ditolak

Pengujian untuk mengubah data petugas dengan data yang dimasukkan oleh pengguna salah dapat dilihat pada Tabel 4-37.

Tabel 4-37 P engujian Mengubah Data Petugas Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak mengisi *field* nama petugas	Muncul pesan ’Nama petugas belum diisi!’	Muncul pesan ’Nama petugas belum diisi!’	[√] Diterima [ ] Ditolak
Tidak mengisi *field* alamat petugas	Muncul pesan ‘Alamat petugas belum diisi!’	Muncul pesan ‘Alamat petugas belum diisi!’	[√] Diterima [ ] Ditolak
Tidak mengisi *field* nomor telepon petugas	Muncul pesan ‘Nomor telepon petugas belum diisi!’	Muncul pesan ‘Nomor telepon petugas belum diisi!’	[√] Diterima [ ] Ditolak
Mengisi semua *field* pada *groupBox* tambah data petugas dengan salah dan menekan tombol simpan	Muncul pesan ‘Data gagal disimpan!’	Muncul pesan ‘Data gagal disimpan!’	[√] Diterima [ ] Ditolak

4.2.2.17 Pengujian Menghapus Data Mitra

Mengahapus data mitra dilakukan untuk menghapus data mitra yang sudah ada sebelumya. Pengujian menghapus data mitra dengan data yang benar dapat dilihat pada Tabel 4-38.

Tabel 4-38 Pengujian Menghapus Data Mitra Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Terdapat data mitra pada *dataGridView* yang tersedia	Data mitra sudah ada pada *dataGridView* yang tersedia	Data mitra sudah ada pada *dataGridView* yang tersedia	[√] Diterima [ ] Ditolak
Memilih data yang akan dihapus pada *dataGridView* yang tersedia dan menekan tombol hapus	Muncul pesan ‘Data berhasil dihapus’	Muncul pesan ‘Data berhasil dihapus’	[√] Diterima [ ] Ditolak

Pengujian untuk login dengan data yang dimasukkan oleh pengguna salah dapat dilihat pada Tabel 4-39.

Tabel 4-39 P engujian Menghapus Data Mitra Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak terdapat data mitra pada *dataGridView* yang tersedia	Muncul pesan ‘Data mitra gagal ditampilkan!’	Muncul pesan ‘Data mitra gagal ditampilkan!’	[√] Diterima [ ] Ditolak
Tidak memilih data yang akan dihapus pada *dataGridView*	Muncul pesan ‘Data mitra gagal diubah!’	Muncul pesan ‘Data mitra gagal diubah!’	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
yang tersedia dan menekan tombol hapus			

4.2.2.18 Pengujian Menghapus Data Area Kolam

Menghapus data area kolam dilakukan untuk menghapus data area kolam yang sudah terdaftar sebelumnya. Pengujian menghapus data area kolam dengan data yang benar dapat dilihat pada Tabel 4-40.

Tabel 4-40 P engujian Menghapus Data Area Kolam Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Terdapat data area kolam pada *dataGridView* yang tersedia	Data area kolam sudah ada pada *dataGridView* yang tersedia	Data area kolam sudah ada pada *dataGridView* yang tersedia	[√] Diterima [ ] Ditolak
Memilih data yang akan dihapus pada *dataGridView* yang tersedia dan menekan tombol hapus	Muncul pesan 'Data berhasil dihapus'	Muncul pesan 'Data berhasil dihapus'	[√] Diterima [ ] Ditolak

Pengujian untuk login dengan data yang dimasukkan oleh pengguna salah dapat dilihat pada Tabel 4-41.

Tabel 4-41 P engujian Menghapus Data Area Kolam Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak terdapat data area kolam pada *dataGridView* yang tersedia	Muncul pesan 'Data area kolam gagal ditampilkan!'	Muncul pesan 'Data area kolam gagal ditampilkan!'	[√] Diterima [ ] Ditolak
Tidak memilih data yang akan dihapus pada *dataGridView* yang tersedia dan menekan tombol hapus	Muncul pesan 'Data area kolam gagal diubah!'	Muncul pesan 'Data area kolam gagal diubah!'	[√] Diterima [ ] Ditolak

4.2.2.19 Pengujian Menghapus Data Kolam

Menghapus data kolam dilakukan untuk menghapus data kolam yang sudah terdaftar sebelumnya. Pengujian menghapus data kolam dengan data yang benar dapat dilihat pada Tabel 4-42.

Tabel 4-42 P engujian Menghapus Data Kolam Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Terdapat data kolam pada *dataGridView* yang tersedia	Data kolam sudah ada pada *dataGridView* yang tersedia	Data kolam sudah ada pada *dataGridView* yang tersedia	[√] Diterima [ ] Ditolak
Memilih data yang akan dihapus pada *dataGridView* yang tersedia dan menekan tombol hapus	Muncul pesan 'Data berhasil dihapus'	Muncul pesan 'Data berhasil dihapus'	[√] Diterima [ ] Ditolak

Pengujian untuk login dengan data yang dimasukkan oleh pengguna salah dapat dilihat pada Tabel 4-43.

Tabel 4-43 Pengujian Menghapus Data Kolam Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak terdapat data kolam pada *dataGridView* yang tersedia	Muncul pesan 'Data kolam gagal ditampilkan!'	Muncul pesan 'Data kolam gagal ditampilkan!'	[√] Diterima [ ] Ditolak
Tidak memilih data yang akan dihapus pada *dataGridView* yang tersedia dan menekan tombol hapus	Muncul pesan 'Data kolam gagal diubah!'	Muncul pesan 'Data kolam gagal diubah!'	[√] Diterima [ ] Ditolak

4.2.2.20 Pengujian Menghapus Data Petugas

Menghapus data petugas dilakukan untuk menghapus data petugas yang sudah terdaftar sebelumnya. Pengujian menghapus data petugas dengan data yang benar dapat dilihat pada Tabel 4-44.

Tabel 4-44 Pengujian Menghapus Data Petugas Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Terdapat data petugas pada *dataGridView* yang tersedia	Data petugas sudah ada pada *dataGridView* yang tersedia	Data petugas sudah ada pada *dataGridView* yang tersedia	[√] Diterima [ ] Ditolak
Memilih data yang akan dihapus pada	Muncul pesan 'Data berhasil dihapus'	Muncul pesan 'Data berhasil dihapus'	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
dataGridView yang tersedia dan menekan tombol hapus			

Pengujian untuk login dengan data yang dimasukkan oleh pengguna salah dapat dilihat pada Tabel 4-45.

Tabel 4-45 Pengujian Menghapus Data Petugas Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak terdapat data petugas pada *dataGridView* yang tersedia	Muncul pesan 'Data petugas gagal ditampilkan!'	Muncul pesan 'Data petugas gagal ditampilkan!'	[√] Diterima [ ] Ditolak
Tidak memilih data yang akan dihapus pada *dataGridView* yang tersedia dan menekan tombol hapus	Muncul pesan 'Data petugas gagal diubah!'	Muncul pesan 'Data petugas gagal diubah!'	[√] Diterima [ ] Ditolak

4.2.2.21 Pengujian Mengelola Pembayaran Pakan

Mengelola pembayaran pakan dilakukan untuk mengolah data pembayaran pakan ikan yang sudah dipesan oleh para mitra petani. Pengujian mengolah data pembayaran pakan dengan data benar dapat dilihat pada Tabel 4-46.

Tabel 4-46 Pengujian Mengelola Pembayaran Pakan Data Benar

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memilih nama mitra yang tersedia pada *comboBox*	Nama mitra sudah dipilih	Nama mitra sudah dipilih	[√] Diterima [ ] Ditolak
Mengisi *field* jumlah pembayaran	Jumlah pembayaran sudah diisi	Jumlah pembayaran sudah diisi	[√] Diterima [ ] Ditolak
Memilih cara pembayaran yang tersedia pada *comboBox*	Cara pembayaran sudah dipilih	Cara pembayaran sudah dipilih	[√] Diterima [ ] Ditolak
Memilih bukti pembayaran berupa foto yang berformat .jpg atau .png	Bukti pembayaran sudah dipilih dan berupa foto yang berformat .jpg atau .png	Bukti pembayaran sudah dipilih dan berupa foto yang berformat .jpg atau .png	[√] Diterima [ ] Ditolak
Memilih tanggal pembayaran pada *dateTimePicker* yang tersedia	Tanggal pembayaran sudah dipilih	Tanggal pembayaran sudah dipilih	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi semua *field* yang ada pada *groupBox* pembayaran mitra dengan benar dan menekan tombol simpan	Muncul pesan 'Data berhasil disimpan!'	Muncul pesan 'Data berhasil disimpan!'	[√] Diterima [ ] Ditolak

Pengujian untuk login dengan data yang dimasukkan oleh pengguna salah dapat dilihat pada Tabel 4-47.

Tabel 4-47 Pengujian Mengelola Pembayaran Pakan Data Salah

Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak memilih nama mitra yang tersedia pada *comboBox*	Muncul pesan 'Nama mitra belum dipilih!'	Muncul pesan 'Nama mitra belum dipilih!'	[√] Diterima [ ] Ditolak
Tidak mengisi *field* jumlah pembayaran	Muncul pesan 'Jumlah pembayaran belum diisi!'	Muncul pesan 'Jumlah pembayaran belum diisi!'	[√] Diterima [ ] Ditolak
Tidak memilih cara pembayaran yang tersedia pada *comboBox*	Muncul pesan 'Cara pembayaran belum dipilih!'	Muncul pesan 'Cara pembayaran belum dipilih!'	[√] Diterima [ ] Ditolak
Tidak memilih bukti pembayaran berupa foto yang berformat .jpg atau .png	Muncul pesan 'Bukti pembayaran belum dipilih atau tidak format gambar tidak sesuai!'	Muncul pesan 'Bukti pembayaran belum dipilih atau tidak format gambar tidak sesuai!'	[√] Diterima [ ] Ditolak
Tidak memilih tanggal pembayaran pada *dateTimePicker* yang tersedia	Muncul pesan 'Tanggal pembayaran belum dipilih!'	Muncul pesan 'Tanggal pembayaran belum dipilih!'	[√] Diterima [ ] Ditolak
Mengisi semua *field* yang ada pada *groupBox* pembayaran mitra dengan data yang salah dan menekan tombol simpan	Muncul pesan 'Data gagal disimpan!'	Muncul pesan 'Data gagal disimpan!'	[√] Diterima [ ] Ditolak

4.2.3 Kesimpulan Pengujian Blacbox

Berdasarkan hasil pengujian *black box* yang dilakukan dengan kasus uji pada Tabel 4-5, dapat ditarik kesimpulan sebagai berikut:

1. Jika ada data masukkan benar, maka sistem akan mengeluarkan keluaran sesuai harapan.

2. Jika data yang dimasukkan salah, maka sistem akan menunjukkan pesan kesalahan sesuai dengan kesalahannya dan mengarahkan pengguna agar memberi masukan yang benar agar fungsional sistem dapat mengeluarkan hasil yang diharapkan.

4.2.4 Skenario Pengujian Beta

Pengujian beta merupakan pengujian yang dilakukan secara objektif dimana diuji secara langsung ke lapangan yaitu instansi yang bersangkutan mengenai kepuasan pengguna dengan kandungan poin yaitu pemenuhan kebutuhan dari tujuan awal pembangunan Sistem Informasi Manajemen Pengelolaan Mitra Petani di CV. Jatiluhur Mas dan tampilan antarmuka dari sistem yang dibangun. Pengujian beta dilakukan melalui sebuah teknik pengambilan data, yaitu melalui wawancara. Wawancara dilakukan terhadap pengujian beta untuk halaman pimpinan perusahaan dan Kepala bagian gudang.

4.2.5 Wawancara Pengguna

Wawancara dilakukan dengan menggunakan teknik kulitatif melalui pertanyaan terbuka. Dari hasil wawancara tersebut akan dilakukan pengambilan kesimpulan terhadap penilaian penerapan sistem yang baru. Wawancara terdiri dari 5 pertanyaan yang akan ditanyakan kepada narasumber.

4.2.5.1 Wawancara Pengujian Beta Untuk Pemimpin Perusahaan

Wawancara dilakukan untuk pengujian beta terhadap Sistem Informasi Manajemen Pengelolaan Peternakan Ikan di CV. Jatiluhur Mas kepada pemimpin perusahaan, yaitu Bapak Ayudya Mas Bayunianto. Kegiatan wawancara ini dilakukan pada tanggal 28 Juli 2016 yang bertempat di PIT STOP (tempat pencucian mobil milik Bapak Ayudya). Daftar Pertanyaan dan hasil wawancara kepada pemimpin perusahaan dapat dilihat pada Tabel 4-48.

Tabel 4-48 Wawancara Pemimpin Perusahaan

1. Apakah manfaat yang anda dapatkan dari sistem yang dibangun?	
Jawaban Narasumber	Manfaatnya program jadi lebih terstruktur, lebih terperinci, lebih terarah. Tujuannya yang asalnya acak-acakan setelah ada sistem kan sistemnya berurutan nanti kalau sudah berurutan kita dapat mengambil keputusan juga lebih terarah dan lebih spesifik.

2. Apakah dengan sistem yang dibangun akan lebih memudahkan pekerjaan anda?	
Jawaban Narasumber	Ya jelas, dengan itu kan jadi yang awalnya yang lebih ribet, sering ada miss dari tim panen dengan kita yang input data kan jadi segala kemungkinan salah input atau salah lapor itu kan bisa ditanggulangi.
3. Apakah sistem yang sedang dibangun dapat mengurangi masalah-masalah yang terjadi sebelumnya?	
Jawaban Narasumber	Bisa, ga nutup kemungkinan dengan sistem yang dibuat jadi lebih baik, tapi dari kita dengan SDM nya sendiri hasrus lebih ditingkatkan. Perannya juga bukan hanya sistem ini kan, tetapi juga SDM nya.
4. Apakah sistem yang dibangun dapat membantu pengelolaan peternakan ikan yang sudah ada?	
Jawaban Narasumber	Bisa.
5. Bagaimana pendapat anda untuk sistem yang dibangun, apakah cukup mudah dipelajari?	
Jawaban Narasumber	Belum sih, kan baru di demoin.
6. Menurut anda apakah menu yang terdapat pada sistem yang dibangun sudah sesuai dengan kebutuhan perusahaan?	
Jawaban Narasumber	Ya, sesuai.
7. Menurut anda apakah navigasi pada sistem yang dibangun sudah cukup baik?	
Jawaban Narasumber	Sudah, nanti tinggal kita yang pelajari.
8. Apakah tampilan sistem yang dibangun sudah baik?	
Jawaban Narasumber	Ya, sudah.
9. Apakah anda sudah cukup puas dengan sistem yang dibangun?	
Jawaban Narasumber	Ya, sudah.
10. Apakah anda memiliki kritik dan saran terhadap sistem yang dibangun agar kedepannya sistem ini apat lebih baik lagi?	
Jawaban Narasumber	Enggak sih, liat aja nanti.

4.2.5.2 Wawancara Pengujian Beta Untuk Kepala Bagian Gudang

Wawancara dilakukan untuk pengujian beta terhadap Sistem Informasi Manajemen Pengelolaan Peternakan Ikan di CV. Jatiluhur Mas kepada pemimpin perusahaan, yaitu Bapak Sutanto. Kegiatan wawancara ini dilakukan pada tanggal 25 Juli 2016 yang bertempat digudang pakan milik CV. Jatiluhur Mas. Daftar Pertanyaan dan hasil wawancara kepada pemimpin perusahaan dapat dilihat pada Tabel 4-48.

Tabel 4-49 Wawancara Kabag Gudang

1. Apakah manfaat yang anda dapatkan dari sistem yang dibangun?	
Jawaban Narasumber	Setelah adanya sistem semua data lebih teratur
2. Apakah dengan sistem yang dibangun akan lebih memudahkan pekerjaan anda?	

Jawaban Narasumber	Iya, karena yang biasanya data oakan sering salah sekarang jadi lebih kelihatah alurnya, tidak usah bulak-balik lagi.
3. Apakah sistem yang sedang dibangun dapat mengurangi masalah-masalah yang terjadi sebelumnya?	
Jawaban Narasumber	Iya memudahkan, jadi pekerjaan saya bisa jadi lebih cepat.
4. Apakah sistem yang dibangun dapat membantu pengelolaan peternakan ikan yang sudah ada?	
Jawaban Narasumber	Iya,
5. Bagaimana pendapat anda untuk sistem yang dibangun, apakah cukup mudah dipelajari?	
Jawaban Narasumber	Lumayan mudah, tapi perlu belajar lagi karena kan belum terbiasa dengan komputer.
6. Menurut anda apakah menu yang terdapat pada sistem yang dibangun sudah sesuai dengan kebutuhan perusahaan?	
Jawaban Narasumber	Sudah.
7. Menurut anda apakah navigasi pada sistem yang dibangun sudah cukup baik?	
Jawaban Narasumber	Sudah baik, jadi jelas menunya sesuai fungsinya kan,
8. Apakah tampilan sistem yang dibangun sudah baik?	
Jawaban Narasumber	Iya, sudah.
9. Apakah anda sudah cukup puas dengan sistem yang dibangun?	
Jawaban Narasumber	Ya, sudah puas.
10. Apakah anda memiliki kritik dan saran terhadap sistem yang dibangun agar kedepannya sistem ini apat lebih baik lagi?	
Jawaban Narasumber	Tidak.

4.2.6 Kesimpulan Pengujian Beta

Berdasarkan hasil jawaban dari pengguna pemimpin perusahaan dan kepala bagian gudang terhadap pertanyaan yang diajukan pada pengujian beta, maka dapat ditarik kesimpulan bahwa sistem yang dibangun sudah memenuhi kebutuhan perusahaan dan cukup mudah dipahami. Sistem yang lebih terstruktur dapat membantu pemimpin perusahaan mengambil keputusan yang lebih terarah dan spesifik.