BAB 4
IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi Sistem

Pada bab ini dilakukan implementasi dan pengujian terhadap sistem informasi yang dibangun. Tahapan implementasi merupakan tahap penciptaan perangkat lunak, tahap kelanjutan dari kegiatan perancangan sistem. Tahap ini merupakan tahap di mana sistem siap untuk dioperasikan yang terdiri dari penjelasan mengenai lingkungan implementasi dan implementasi dari program.

4.1.1 Lingkungan Implementasi

Aplikasi yang akan diterapkan pada lingkungan implementasi membutuhkan dukungan, maka dalam hal ini yang harus diperhatikan adalah perangkat keras dan perangkat lunak yang menunjang untuk sistem informasi *inventory control* yang dibangun.

4.1.2 Implementasi Perangkat Keras

Spesifikasi perangkat keras yang digunakan untuk mengimplementasikan sistem infromasi *invenotry control* adalah seperti yang tertera pada Tabel 4.1. berikut.

Tabel 4.1 Perangkat Keras yang digunakan

No	Perangkat Keras	Spesifikasi
1	Prosesor	Intel Pentium 4 2.8 Ghz
2	Monitor	Monitor 15 inch
3	Harddisk	250 GB
4	VGA	NVDIA G-FORCE 9500GT 128 Mb
5	Memory	2 GB
6	Mouse	Standar
7	Keyboard	Standar
8	Jaringan	LAN
9	Printer	Printer warna

4.1.3 Implementasi Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan untuk mengimplementasikan sistem infromasi *invenotry control* tertera terdapat pada tabel 4.2 sebagai berikut.

Tabel 4.2 Perangkat Lunak yang digunakan

No	Perangkat Keras	Spesifikasi
1	Sistem Operasi	Windows XP
2	*Web Browser*	Mozilla Firefox, Google Chrome
3	*Code Editor*	Macromedia Dreamweaver 8 atau NetBeans 7.0.1
4	DBMS	MySQL 5.5
5	*Web Server*	Wamp 2.2
6	Analisis Pemodelan	StarUML

4.1.4 Implementasi Basis Data

Pembuatan basis data dilakukan dengan menggunakan aplikasi pemrograman MySQL. Implementasi basis data yang di antaranya adalah sebagai berikut:

1. Implementasi Tabel kategori

```
CREATE TABLE IF NOT EXISTS `kategori` (
  `id_kategori` int(2) NOT NULL AUTO_INCREMENT,
  `kd_kategori` varchar(15) NOT NULL,
  `nm_kategori` varchar(50) NOT NULL,
  PRIMARY KEY (`id_kategori`)
);
```

2. Implementasi Tabel barang

```
CREATE TABLE IF NOT EXISTS `barang` (
  `id_barang` int(10) NOT NULL AUTO_INCREMENT,
  `kd_barang` varchar(15) NOT NULL,
  `nm_barang` varchar(50) NOT NULL,
  `stok` int(11) NOT NULL,
  `stok_min` int(11) NOT NULL,
  `id_kategori` int(11) DEFAULT NULL,
  `nm_satuan` int(11) DEFAULT NULL,
  PRIMARY KEY (`id_barang`),
  KEY (`id_kategori`)  );
```

```
ALTER TABLE `barang`
   ADD CONSTRAINT `FK_id_kategori` FOREIGN KEY (`id_kategori`)
REFERENCES `kategori` (`id_kategori`) ON UPDATE CASCADE ON DELETE
CASCADE;
```

3. Implementasi Tabel supplier

```
CREATE TABLE IF NOT EXISTS `supplier` (
  `id_supp` int(5) NOT NULL AUTO_INCREMENT,
  `nm_supp` varchar(35) NOT NULL,
  `no_tlp` varchar(15) NOT NULL,
  `alamat_supp` text NOT NULL,
  PRIMARY KEY (`id_supp`) );
```

4. Implementasi Tabel petugas

```
CREATE TABLE IF NOT EXISTS `petugas` (
  `id_petugas` int(5) NOT NULL AUTO_INCREMENT,
  `nip` varchar(10) NOT NULL,
  `nm_petugas` varchar(20) NOT NULL,
  `password` varchar(50) NOT NULL,
  `id_kategori_petugas` int(11) NOT NULL,
  PRIMARY KEY (`id_petugas`),
  KEY (`id_kategori_petugas`) );

ALTER TABLE `petugas`
   ADD    CONSTRAINT    `FK_id_kategori_petugas`    FOREIGN    KEY
(`id_kategori_petugas`)       REFERENCES       `kategori_petugas`
(`id_kategori_petugas`) ON UPDATE CASCADE ON DELETE CASCADE;
```

5. Implementasi Tabel kategori_petugas

```
CREATE TABLE IF NOT EXISTS `kategori_petugas` (
  `id_kategori_petugas` int(5) NOT NULL AUTO_INCREMENT,
  `nama_kategori` varchar(15) NOT NULL,
  `fungsi` text NOT NULL,
  PRIMARY KEY (`id_kategori_petugas`)
);
```

6. Implementasi Tabel pemesanan

```
CREATE TABLE IF NOT EXISTS `pemesanan` (
  `id_pemesanan` int(10) NOT NULL AUTO_INCREMENT,
  `no_pemesanan` varchar(20) NOT NULL,
  `id_petugas` int(5) NOT NULL,
  `id_supp` int(5) DEFAULT NULL,
  `tgl_psn` date NOT NULL,
  PRIMARY KEY (`id_pemesanan`),
  KEY `id_supp` (`id_supp`),
  KEY `id_petugas` (`id_petugas`) );

ALTER TABLE `pemesanan`
   ADD CONSTRAINT `FK_id_supp` FOREIGN KEY (`id_supp`) REFERENCES
`supplier` (`id_supp`) ON UPDATE CASCADE ON DELETE CASCADE,
   ADD CONSTRAINT `FK_id_petugas` FOREIGN KEY (`id_petugas`)
REFERENCES `petugas` (`id_petugas`) ON UPDATE CASCADE ON DELETE
CASCADE;
```

7. Implementasi Tabel detail_pemesanan

```
CREATE TABLE IF NOT EXISTS `detail_pemesanan` (
  `id_detail_psn` int(10) NOT NULL AUTO_INCREMENT,
  `id_barang` int(10) DEFAULT NULL,
  `id_pemesanan` int(10) DEFAULT NULL,
  `jumlah` int(11) NOT NULL,
  `keterangan` varchar(25) DEFAULT NULL,
  `sisa` int(11) NOT NULL,
  PRIMARY KEY (`id_detail_psn`) );

ALTER TABLE `detail_pemesanan`
   ADD CONSTRAINT `FK_id_pemesanan` FOREIGN KEY (`id_ pemesanan`)
REFERENCES ` pemesanan ` (`id_ pemesanan `) ON UPDATE CASCADE ON
DELETE CASCADE,
   ADD CONSTRAINT `FK_id_barang` FOREIGN KEY (`id_barang`)
REFERENCES ` barang ` (`id_barang `) ON UPDATE CASCADE ON DELETE
CASCADE;
```

8. Implementasi Tabel penerimaan

```
CREATE TABLE IF NOT EXISTS `penerimaan` (
  `id_penerimaan` int(10) NOT NULL AUTO_INCREMENT,
  `no_faktur` varchar(20) NOT NULL,
  `id_supp` int(5) NOT NULL,
  `id_petugas` int(5) NOT NULL,
  `tgl_terima` date NOT NULL,
  PRIMARY KEY (`id_penerimaan`) ) ;
```

```
ALTER TABLE `penerimaan`
   ADD CONSTRAINT `FK_id_supp` FOREIGN KEY (`id_supp`) REFERENCES
`supplier` (`id_supp`) ON UPDATE CASCADE ON DELETE CASCADE,
   ADD CONSTRAINT `FK_id_petugas` FOREIGN KEY (`id_petugas`)
REFERENCES `petugas` (`id_petugas`) ON UPDATE CASCADE ON DELETE
CASCADE;
```

9. Implementasi Tabel detail_penerimaan

```
CREATE TABLE IF NOT EXISTS `detail_penerimaan` (
  `id_detail_penerimaan` int(10) NOT NULL AUTO_INCREMENT,
  `id_detail_psn` int(10) DEFAULT NULL,
  `id_pemesanan` int(10) DEFAULT NULL,
  `id_barang` int(10) DEFAULT NULL,
  `jumlah` int(11) DEFAULT NULL,
  `keterangan` varchar(30) DEFAULT NULL,
  `id_penerimaan` int(10) DEFAULT NULL,
  PRIMARY KEY (`id_detail_penerimaan`) );

ALTER TABLE `detail_penerimaan`
   ADD CONSTRAINT `FK_id_detail_psn` FOREIGN KEY (`id_detail_psn`)
REFERENCES `detail_pemesanan` (`id_detail_psn`) ON UPDATE CASCADE
ON DELETE CASCADE,
   ADD CONSTRAINT `FK_id_pemesanan` FOREIGN KEY (`id_pemesanan`)
REFERENCES `pemesanan` (`id_pemesanan`) ON UPDATE CASCADE ON
DELETE CASCADE,
   ADD CONSTRAINT `FK_id_penerimaan` FOREIGN KEY (`id_penerimaan`)
REFERENCES `penerimaan` (`id_penerimaan`) ON UPDATE CASCADE ON
DELETE CASCADE,
   ADD CONSTRAINT `FK_id_barang` FOREIGN KEY (`id_barang`)
REFERENCES `barang` (`id_barang`) ON UPDATE CASCADE ON DELETE
CASCADE,
```

10. Implementasi Tabel pengeluaran

```
CREATE TABLE IF NOT EXISTS `pengeluaran` (
  `id_pengeluaran` int(10) NOT NULL AUTO_INCREMENT,
  `id_petugas` int(10) NOT NULL,
  `no_faktur` varchar(20) NOT NULL,
  `nama_kacab` int(5) NOT NULL,
  `ptgs_kacab` varchar(25) NOT NULL,
  `no_telp` varchar(15) NOT NULL,
  `tgl_keluar` date NOT NULL,
  `no_permintaan` int(30) NOT NULL,
  PRIMARY KEY (`id_pengeluaran`) );
```

```
ALTER TABLE `pengeluaran`
   ADD CONSTRAINT `FK_id_petugas` FOREIGN KEY (`id_petugas`)
REFERENCES `petugas` (`id_petugas`) ON UPDATE CASCADE ON DELETE
CASCADE;
```

11. Implementasi Tabel detail_pengeluaran

```
CREATE TABLE IF NOT EXISTS `detail_pengeluaran` (
  `id_detail_pengeluaran` int(10) NOT NULL AUTO_INCREMENT,
  `id_pengeluaran` int(10) NOT NULL,
  `id_barang` int(10) NOT NULL,
  `jumlah` int(11) NOT NULL,
  PRIMARY KEY (`id_detail_pengeluaran`) );
ALTER TABLE `pengeluaran`
   ADD     CONSTRAINT     `FK_id_pengeluaran`     FOREIGN     KEY
(`id_pengeluaran`) REFERENCES `pengeluaran` (`id_pengeluaran`) ON
UPDATE CASCADE ON DELETE CASCADE,
    ADD  CONSTRAINT  `FK_id_barang`  FOREIGN  KEY  (`id_barang`)
REFERENCES `barang` (`id_barang `) ON UPDATE CASCADE ON DELETE
CASCADE;
```

12. Implementasi Tabel tabel_b

```
CREATE TABLE IF NOT EXISTS `tabel_b` (
  `id` int(5) NOT NULL AUTO_INCREMENT,
  `z` varchar(10) NOT NULL,
  `alfa` varchar(10) NOT NULL,
  `ordinat` varchar(10) NOT NULL,
  `ekspektasi` varchar(10) NOT NULL,
  PRIMARY KEY (`id`) );
```

13. Implementasi Tabel pemesanan_v

```
CREATE OR REPLACE VIEW pemesanan_v AS select
`pemesanan`.`id_pemesanan` AS
`id_pemesanan`,`pemesanan`.`no_pemesanan` AS
`no_pemesanan`,`pemesanan`.`id_petugas` AS
`id_petugas`,`pemesanan`.`id_supp` AS
`id_supp`,`pemesanan`.`tgl_psn` AS
`tgl_psn`,`detail_pemesanan`.`id_detail_psn` AS
`id_detail_psn`,`detail_pemesanan`.`id_barang` AS
`id_barang`,`detail_pemesanan`.`jumlah` AS
`jumlah`,`detail_pemesanan`.`keterangan` AS
`keterangan`,`petugas`.`nm_petugas` AS
`nm_petugas`,`supplier`.`nm_supp` AS
`nm_supp`,`kategori`.`nm_kategori` AS
`nm_kategori`,`barang`.`nm_barang` AS
`nm_barang`,`satuan`.`nm_satuan` AS `nm_satuan` FROM
((((((`pemesanan` left join `detail_pemesanan`
on((`pemesanan`.`id_pemesanan` =
`detail_pemesanan`.`id_pemesanan`))) left join `petugas`
on((`pemesanan`.`id_petugas` = `petugas`.`id_petugas`)))
left join `supplier` on((`pemesanan`.`id_supp` =
`supplier`.`id`))) left join `barang`
on((`detail_pemesanan`.`id_barang` = `barang`.`id`))) );
```

14. Implementasi Tabel penerimaan_v

```
CREATE OR REPLACE VIEW pemesanan_v AS select
`penerimaan`.`id_penerimaan` AS
`id_penerimaan`,`penerimaan`.`no_faktur` AS
`no_faktur`,`penerimaan`.`id_petugas` AS
`id_petugas`,`penerimaan`.`id_supp` AS
`id_supp`,`penerimaan`.`tgl_terima` AS
`tgl_terima`,`detail_penerimaan`.`id_detail_penerimaan` AS
`id_detail_penerimaan`,`detail_penerimaan`.`id_detail_psn`
AS `id_detail_psn`,`detail_penerimaan`.`id_pemesanan` AS
`id_pemesanan`,`detail_penerimaan`.`id_barang` AS
`id_barang`,`detail_penerimaan`.`jumlah` AS
`jumlah`,`detail_penerimaan`.`keterangan` AS
`keterangan`,`petugas`.`nm_petugas` AS
`nm_petugas`,`supplier`.`nm_supp` AS
`nm_supp`,`kategori`.`nm_kategori` AS
`nm_kategori`,`barang`.`nm_barang` AS
`nm_barang`,`satuan`.`nm_satuan` AS `nm_satuan` FROM
((((((`penerimaan` left join `detail_penerimaan`
on((`penerimaan`.`id_penerimaan` =
`detail_penerimaan`.`id_penerimaan`))) left join `petugas`
on((`penerimaan`.`id_petugas` = `petugas`.`id_petugas`)))
left join `supplier` on((`penerimaan`.`id_supp` =
`supplier`.`id`))) left join `barang`
on((`detail_penerimaan`.`id_barang` = `barang`.`id`))) );
```

15. Implementasi Tabel pemesanan_v

```
CREATE OR REPLACE VIEW pengeluaran_v

AS select `pengeluaran`.`id_pengeluaran` AS
`id_pengeluaran`,`pengeluaran`.`no_faktur` AS
`no_faktur`,`pengeluaran`.`id_petugas` AS
`id_petugas`,`pengeluaran`.`nama_kacab` AS `nama_kacab`,
`pengeluaran`.`ptgs_kacab` AS
`ptgs_kacab`,`pengeluaran`.`no_telp` AS ` no_telp `,
`pengeluaran`.`tgl_keluar` AS
`tgl_keluar`,`detail_pengeluaran`.`id_detail_pengeluaran` AS
`id_detail_pengeluaran`,`detail_pengeluaran`.`id_barang` AS
`id_barang`,`detail_pengeluaran`.`jumlah` AS
`jumlah`,`petugas`.`nm_petugas` AS
`nm_petugas`,`kacab`.`nama_kacab` AS
`nama_kacab`,`kategori`.`nm_kategori` AS
`nm_kategori`,`barang`.`nm_barang` AS
`nm_barang`,`satuan`.`nm_satuan` AS `nm_satuan` from
((((((`pengeluaran` left join `detail_pengeluaran`
on((`pengeluaran`.`id_pengeluaran` =
`detail_pengeluaran`.`id_pengeluaran`))) left join `petugas`
on((`pengeluaran`.`id_petugas` = `petugas`.`id_petugas`)))
left join `barang` on((`detail_pengeluaran`.`id_barang` =
`barang`.`id`))) );
```

4.1.5 Implementasi Kelas

Implementasi kelas adalah penerapan *class-class* yang dibangun dan diterapkan pada aplikasi. Implementasi kelas pada sistem informasi *inventory control* dapat dilihat pada Tabel 4.3 dan Tabel 4.4.

Tabel 4.3 Implementasi Kelas

Nama Class	Deskripsi	Diterapkan di File
SiteController	Digunakan sebagai validasi data masukan dari *login* dan meminta *class model* untuk mengakses *database*.	controllers/ SiteController.php
LoginForm	Digunakan sebagai proses lanjutan proses *login* dari SiteController yang berhubungan dengan *database*	models/ LoginForm.php
KategoriController	Digunakan untuk memvalidasi data masukan dari tambah, ubah, cari, dan hapus, meminta *class model* untuk mengakses *database*	controllers/ KategoriController.php
Kategori	Kelas model untuk mengakses *database* yang diperlukan untuk menampilkan informasi, mengolah data kategori.	models/Kategori.php
BarangController	Digunakan untuk memvalidasi data masukan dari tambah, ubah, cari, dan hapus, meminta *class* model untuk mengakses *database*	controllers/ BarangController.php
Barang	Kelas model untuk mengakses *database* yang diperlukan untuk menampilkan informasi, mengolah data barang.	models/Barang.php
SupplierController	Digunakan untuk memvalidasi data masukan dari tambah, ubah, cari, dan hapus, meminta *class* model untuk mengakses *database*.	controllers/ SupplierController.php
Supplier	Kelas model untuk mengakses *database* yang diperlukan untuk menampilkan informasi, mengolah data *supplier*.	models/Supplier.php
PetugasController	Digunakan untuk memvalidasi data masukan dari tambah, ubah, cari, dan hapus, meminta *class* model untuk mengakses *database*.	controllers/ PetugasController.php
Petugas	Kelas model untuk mengakses *database* yang diperlukan untuk menampilkan informasi, mengolah data petugas.	models/Petugas.php

Tabel 4.4 Implementasi Kelas (Lanjutan)

Nama Class	Deskripsi	Diterapkan di File
PemesananController	Digunakan untuk memvalidasi data masukan dari tambah, ubah, cari, dan hapus, meminta *class* model untuk mengakses *database*.	controllers/ PemesananController.php
Pemesanan	Kelas model untuk mengakses *database* yang diperlukan untuk menampilkan informasi, mengolah data pemesanan.	models/Pemesanan.php
DetailPemesanan	Kelas model untuk mengakses *database* yang diperlukan untuk menampilkan informasi, mengolah data detail pemesanan.	models/ DetailPemesanan.php
PenerimaanController	Digunakan untuk memvalidasi data masukan dari tambah, ubah, cari, dan hapus, meminta *class* model untuk mengakses *database*.	controllers/ PenerimaanController.php
Penerimaan	Kelas model untuk mengakses *database* yang diperlukan untuk menampilkan informasi, mengolah data penerimaan.	models/Penerimaan.php
DetailPenerimaan	Kelas model untuk mengakses *database* yang diperlukan untuk menampilkan informasi, mengolah data detail penerimaan.	models/ DetailPenerimaan.php
PengeluaranController	Digunakan untuk memvalidasi data masukan dari tambah, ubah, cari, dan hapus, meminta *class* model untuk mengakses *database*.	controllers/ PenngeluaranController.php
Pengeluaran	Kelas model untuk mengakses *database* yang diperlukan untuk menampilkan informasi, mengolah data pengeluaran.	models/Pengeluaran.php
DetailPengeluaran	Kelas model untuk mengakses *database* yang diperlukan untuk menampilkan informasi, mengolah data detail pengeluaran.	models/ DetailPengeluaran.php
RencanaInventoriController	Digunakan untuk memanggil class lainnya guna menghitung rencana inventori dan validasi	controllers/ RencanaInventoriController.php
LaporanController	Digunakan mengatur hubungan laporanView dengan kelas model dalam mengolah proses cari hingga cetak laporan	controllers/ LaporanController.php

4.1.6 Implementasi Antarmuka

Implementasi antarmuka merupakan tampilan dari aplikasi yang dibangun. Aplikasi pengolahan data proses memilki beberapa bagian tampilan menu dapat dilihat pada Tabel 4.5, Tabel 4.6, Tabel 4.7, Tabel 4.7.

Tabel 4.5 Implementasi Antarmuka Admin

Menu	Deskripsi	Nama File
Login	Digunakan sebagai halaman *login* admin.	site/login.php
Beranda Admin	Digunakan sebagai halaman beranda admin dan terdapat beberapa menu diantaranya menu data master, data pemesanan, penerimaan dan data pengeluaran.	index.php
Pengaturan Kategori	Halaman awal untuk melihat daftar kategori barang.	kategori/admin.php
Tambah Kategori	Halaman yang digunakan untuk menambah data kategori.	kategori/create.php
Ubah Kategori	Halaman yang digunakan untuk mengubah data kategori.	kategori/update.php
Lihat Kategori	Halaman yang digunakan untuk melihat data kategori secara detail.	kategori/view.php
Pengaturan Barang	Halaman awal untuk melihat daftar barang.	barang/admin.php
Tambah Barang	Halaman yang digunakan untuk menambah data barang.	barang/create.php
Ubah Barang	Halaman yang digunakan untuk mengubah data barang.	barang/update.php
Lihat Barang	Halaman yang digunakan untuk melihat data barang secara detail.	barang/view.php
Pengaturan Supplier	Halaman awal untuk melihat daftar *supplier*.	supplier/admin.php
Tambah Supplier	Halaman yang digunakan untuk menambah data *supplier*.	supplier/create.php
Ubah Supplier	Halaman yang digunakan untuk mengubah data *supplier*.	supplier/update.php
Lihat Supplier	Halaman yang digunakan untuk melihat data *supplier* secara detail.	supplier/view.php
Pengaturan Petugas	Halaman awal untuk melihat daftar petugas.	petugas/admin.php
Tambah Petugas	Halaman yang digunakan untuk menambah data petugas.	petugas/create.php
Ubah Petugas	Halaman yang digunakan untuk mengubah data petugas.	petugas/update.php

Tabel 4.6 Implementasi Antarmukan Admin (Lanjutan)

Menu	Deskripsi	Nama File
Lihat Petugas	Halaman yang digunakan untuk melihat data petugas secara detail.	petugas/view.php
Pengaturan Pemesanan	Halaman awal untuk melihat daftar Pemesanan.	pemesanan/admin.php
Ubah Pemesanan	Halaman yang digunakan untuk mengubah data pemesanan.	pemesanan/admin.php
Lihat Pemesanan	Halaman yang digunakan untuk melihat data pemesanan.	pemesanan/view.php
Rencana Inventori	Halaman yang digunakan untuk menghitung rencana inventori.	rencanaInventori/index.php

Tabel 4.7 Implementasi Antarmuka Petugas

Menu	Deskripsi	Nama File
Login	Digunakan sebagai halaman login petugas.	site/login.php
Beranda Petugas	Digunakan sebagai halaman beranda petugas dan terdapat beberapa menu diantaranya menu data pemesanan, penerimaan, data pengeluaran, laporan, dan rencana inventori.	index.php
Pengaturan Barang	Halaman awal untuk melihat daftar barang.	barang/admin.php
Lihat barang	Halaman yang digunakan untuk melihat data barang secara detail.	barang/view.php
Pengaturan Pemesanan	Halaman awal untuk melihat daftar pemesanan.	pemesanan/admin.php
Tambah Pemesanan	Halaman yang digunakan untuk menambah data pemesanan.	pemesanan/ create.php
Lihat Pemesanan	Halaman yang digunakan untuk melihat data pemesanan secara detail.	pemesanan/view.php
Pengaturan Penerimaan	Halaman awal untuk melihat daftar penerimaan.	penerimaan/admin.php
Tambah Penerimaan	Halaman yang digunakan untuk menambah data penerimaan.	penerimaan/create.php
Lihat Penerimaan	Halaman yang digunakan untuk melihat data penerimaan secara detail.	penerimaan/view.php
Pengaturan Pengeluaran	Halaman awal untuk melihat daftar pengeluaran.	pengeluaran/admin.php
Tambah Pengeluaran	Halaman yang digunakan untuk menambah data pengeluaran.	pengeluaran/create.php

Tabel 4.8 Implementasi Antarmuka Petugas (Lanjutan)

Menu	Deskripsi	Nama File
Lihat Pengeluaran	Halaman yang digunakan untuk melihat data pengeluaran secara detail.	pengeluaran/view.php
Laporan Pemesanan	Halaman yang digunakan untuk mengelola laporan pemesanan.	laporan/pemesanan.php
Laporan Penerimaan	Halaman yang digunakan untuk mengelola laporan penerimaan.	laporan/penerimaan.php
Laporan Pengeluaran	Halaman yang digunakan untuk mengelola laporan pengeluaran.	laporan/pengeluaran.php

4.2 Pengujian Perangkat Lunak

Pengujian perangkat lunak merupakan tahapan lanjutan setelah pembangunan aplikasi serta dianggap penting karena bertujuan mencari kelemahan-kelemahan atau kekurangan-kekurangan perangkat lunak yang diuji. Pengujian juga dimaksudkan untuk mengetahui apakah sudah memenuhi kriteria yang sesuai dengan tujuan perancangan perangkat lunak.

Pengujian perangkat lunak ini menggunakan pengujian *black box* yaitu pengujian yang fokus kepada persyaratan fungsional perangkat lunak.

4.2.1 Pengujian *Black Box*

Pengujian *black box* (*black box testing*) merupakan salah satu metode pengujian perangkat lunak yang berfokus pada sisi fungsionalitas, khususnya pada *input* dan *output* aplikasi. Tahap pengujian atau *testing* merupakan salah satu tahap yang harus ada dalam sebuah siklus pengembangan perangkat lunak. Pengujian *blackbox* yang dilakukan pada aplikasi sistem informasi *inventory control* dibagi menjadi tiga bagian yaitu skenario pengujian, kasus dan hasil pengujian, dan kesimpulan pengujian.

4.2.1.1 Skenario Pengujian

Skenario pengujian *black box* di PT. PLN Persero APJ Majalaya terdiri dari dua pelaku sistem yaitu *supervisor* logistik sebagai Admin, *junior* logistik sebagai Petugas.

Tabel 4.9 Skenario Pengujian Admin

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Isi data *login*	*Black Box*
	Validasi *username* dan *password*	*Black Box*
Pengolahan data Kategori	Cari Kategori	*Black Box*
	Tambah Kategori	*Black Box*
	Ubah Kategori	*Black Box*
	Hapus Kategori	*Black Box*
Pengolahan data Barang	Cari Barang	*Black Box*
	Tambah Barang	*Black Box*
	Ubah Barang	*Black Box*
	Hapus Barang	*Black Box*
Pengolahan data Supplier	Cari *Supplier*	*Black Box*
	Tambah *Supplier*	*Black Box*
	Ubah *Supplier*	*Black Box*
	Hapus *Supplier*	*Black Box*
Pengolahan data Petugas	Cari Petugas	*Black Box*
	Tambah Petugas	*Black Box*
	Ubah Petugas	*Black Box*
	Hapus Petugas	*Black Box*
Pengolahan Data Pemesanan	Cari Pemesanan	*Black Box*
	Ubah Pemesanan	*Black Box*
	Hapus Pemesanan	*Black Box*
Pengolahan Data Penerimaan	Cari Penerimaan	*Black Box*
	Ubah Penerimaan	*Black Box*
	Hapus Penerimaan	*Black Box*
Pengolahan Data Pengeluaran	Cari Pengeluaran	*Black Box*
	Ubah Pengeluaran	*Black Box*
	Hapus Pengeluaran	*Black Box*
Logout	Menghapus *session* admin	*Black Box*

Tabel 4.10 Skenario Pengujian Petugas

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Isi data *login*	*Black Box*
	Validasi *username* dan *password*	*Black Box*
Pengolahan data Barang	Cari Barang	*Black Box*
Pengolahan Data Pemesanan	Cari Pemesanan	*Black Box*
	Tambah Pemesanan	*Black Box*
Pengolahan Data Penerimaan	Cari Penerimaan	*Black Box*
	Tambah Penerimaan	*Black Box*
Pengolahan Data Pengeluaran	Cari Pengeluaran	*Black Box*
	Tambah Pengeluaran	*Black Box*
Rencana Inventori	Hitung Rencana inventori	*Black Box*
Laporan Pemesanan	Cari Laporan Pemesanan	*Black Box*
	Cetak Laporan Pemesanan	*Black Box*
Laporan Penerimaan	Cari Laporan Penerimaan	*Black Box*
	Cetak Laporan Penerimaan	*Black Box*
Laporan Pengeluaran	Cari Laporan Pengeluaran	*Black Box*
	Cetak Laporan Pengeluaran	*Black Box*
Logout	Menghapus *session* petugas	*Black Box*

4.2.2 Kasus dan Hasil Pengujian

Kasus dan hasil pengujian mendeskripsikan antara fungsi yang dikelola oleh sistem dengan hasil pengujian dari fungsi tersebut.

4.2.2.1 Pengujian *Login*

Pengujian *login* dapat dilihat pada tabel berikut.

Tabel 4.11 Pengujian *Login*

Kasus dan Hasil Uji (Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data *Login* berupa *Username* : admin *Password* : 1234	Ketika memilih tombol *Login*, data *login* dicari dalam *database*, *username* dan data *password* benar *login* valid dan dapat masuk ke menu utama	*username* dan *password* benar dapat masuk ke menu utama	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Salah)			
Data *Login* berupa *Username* : admin *Password* : 10sa	Ketika memilih tombol *Login*, tidak dapat *login*/masuk dan menampilkan pesan "*Username*/*Password* tidak sesuai"	Tidak dapat *login*/masuk dan menampilkan pesan kesalahan sesuai yang diharapkan	[√] Diterima [ ] Ditolak
Data *Login* berupa *Username* : admin *Password* : Atau *Username* : *Password* : 1234	Tidak dapat *login*, tampil pesan kesalahan "*Password* atau *Username* tidak boleh kosong"	Tidak dapat *login*/masuk dan menampilkan pesan kesalahan sesuai yang diharapkan	[√] Diterima [ ] Ditolak
Data *Login* kosong *Username* : *Password* :	Tidak dapat *login*, tampil pesan kesalahan "*Password* dan *Username* tidak boleh kosong"	Tidak dapat *login*/masuk dan menampilkan pesan kesalahan sesuai yang diharapkan	[√] Diterima [ ] Ditolak

4.2.2.2 Pengujian Pengolahan Data Barang

Pengujian pengolahan data barang terdiri dari proses cari, tambah, ubah, dan hapus barang.

1. Pengujian Cari Barang

Pengujian cari barang dapat dilihat pada tabel berikut ini.

Tabel 4.12 Pengujian Cari Barang

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data cari berupa Nama Barang : MTR Kategori : Transformator	Ketika menekan tombol Enter, tampil data barang yang dicari	Dapat menampilkan data barang yang dicari	[√] Diterima [ ] Ditolak

2. Pengujian Tambah Barang

Pengujian tambah data barang dapat dilihat pada tabel sebagai berikut.

Tabel 4.13 Pengujian Tambah Barang

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Nama Barang : MTR Kategori : Transformator Satuan : Unit Stok : 200 Stok Minimal : 2	Ketika memilih tombol Simpan, data barang dapat tersimpan dalam *database.*	Ketika memilih tombol Simpan, menampilkan data tambah dan data tersimpan pada *database*	[√] Diterima [ ] Ditolak
Kasus Dan Hasil Uji Salah (Data Salah)			
Nama Barang : MTR Kategori : Transformator Satuan : Unit Stok : a2ff Stok Minimal : 1	Ketika memilih tombol Simpan, tampil pesan kesalahan "Data barang MTR sudah ada, Stok harus berupa integer." Kemudian data tidak tersimpan di *database*	Ketika memilih tombol Simpan, tampil pesan kesalahan sesuai validasi masing-masing *field input* dan data *input*	[√] Diterima [ ] Ditolak

Tabel 4.14 Pengujian Tambah Barang (Lanjutan)

Kasus Dan Hasil Uji Salah (Data Salah)			
Nama Barang : MCB Kategori : Alat Ukur Satuan : Stok : -2 Stok Minimal :	Data barang tidak tersimpan ke *database* dan tampil pesan kesalahan "Satuan tidak boleh kosong, Stok terlalu kecil (minimum 0), Stok Minimal tidak boleh kosong"	Ketika memilih tombol Simpan, tampil pesan kesalahan sesuai validasi masing-masing *field input* dan data *input*	[√] Diterima [ ] Ditolak

3. Pengujian Ubah Barang

Pengujian ubah data barang dapat dilihat pada tabel berikut ini.

Tabel 4.15 Pengujian Ubah Barang

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data barang lama Nama Barang : MTR Kategori : Transformator Satuan : Unit Stok : 200 Stok Minimal : 2	Ketika memilih tombol Simpan, data barang dapat tersimpan dalam *database.*	Ketika memilih tombol Simpan, menampilkan data hasil ubah dan data tersimpan pada *database*	[√] Diterima [ ] Ditolak
Data barang baru Nama Barang : CONN Kategori : Switchgear Satuan : Unit Stok : 155 Stok Minimal : 1			
Kasus Dan Hasil Uji Salah (Data Salah)			
Data barang baru Nama Barang : MCB Kategori : Satuan : Unit Stok : 155 Stok Minimal : a	Ketika tombol simpan diklik, data tidak tersimpan ke dalam *database.* Tampil pesan "Nama Barang MCB sudah ada", "Kategori tidak boleh kosong ", Stok Minimal harus berupa integer "	Ketika memilih tombol simpan, data tersebut tidak tersimpan ke dalam *database* tampil pesan kesalahan sesuai validasi masing-masing *field input* dan validasi data *input*	[√] Diterima [ ] Ditolak

4. Pengujian Hapus Barang

Pengujian hapus data barang dapat dilihat pada tabel berikut ini.

Tabel 4.16 Pengujian Hapus Barang

Kasus dan Hasil Uji			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data yang akan dihapus Contoh: Nama Barang : MTR Kategori : Transformator Satuan : Unit Stok : 200 Stok Minimal : 2	Setelah memilih tombol hapus tampil pesan konfirmasi penghapusan data	Setelah memilih tombol hapus tampil pesan konfirmasi penghapusan data, jika *user* pilih Ok maka data barang akan dihapus dan tampil pesan data berhasil dihapus. Dan Jika memilih Cancel, maka penghapusan dibatalkan	[√] Diterima [ ] Ditolak
Kasus Dan Hasil Uji			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data yang akan dihapus dan data sudah digunakan oleh data transaksi Contoh: Nama Barang : MCB Kategori : Alat Ukur Satuan : Buah Stok : 200 Stok Minimal : 2	Setelah memilih tombol hapus tampil pesan konfirmasi penghapusan data, serta data tidak terhapus di *database*	Setelah memilih tombol hapus tampil pesan konfirmasi penghapusan data, jika *user* pilih Ok tampil pesan "Data gagal dihapus. Data barang sudah digunakan di data transaksi" Dan Jika memilih *Cancel*, maka penghapusan dibatalkan	[√] Diterima [ ] Ditolak

4.2.2.3 Pengujian Pengolahan Data Kategori

Pengujian pengolahan data kategori terdiri dari proses cari, tambah, ubah, dan hapus kategori.

1. Pengujian Cari Kategori

Pengujian cari data kategori dapat dilihat pada tabel sebagai berikut.

Tabel 4.17 Pengujian Cari Kategori

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Nama Satuan : Alat Ukur	Ketika menekan tombol Enter, tampil data barang yang dicari	Dapat menampilkan data satuan yang dicari	[√] Diterima [ ] Ditolak

2. Pengujian Tambah Kategori

Pengujian tambah data kategori dapat dilihat pada tabel sebagai berikut.

Tabel 4.18 Pengujian Tambah Kategori

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Nama Kategori: Transformator	Ketika memilih tombol Simpan, data barang dapat tersimpan dalam *database.*	Ketika memilih tombol Simpan, menampilkan data tambah dan data tersimpan pada *database*	[√] Diterima [ ] Ditolak
Kasus Dan Hasil Uji Salah (Data Salah)			
Data input berupa Contoh: Nama Kategori: Transformator Atau Nama Kategori:	Nama kategori “Transformator “ sudah ada di *database*. Ketika memilih tombol Simpan, tampil pesan peringatan data yang dimasukan sudah ada atau *field* tidak boleh kosong	Ketika memilih tombol Simpan, tampil pesan kesalahan sesuai validasi *field input* dan data *input*	[√] Diterima [ ] Ditolak

3. Pengujian Ubah Kategori

Pengujian ubah data kategori dapat dilihat pada tabel berikut ini.

Tabel 4.19 Pengujian Ubah Kategori

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data lama; Nama Kategori : Transformator	Ketika memilih tombol Simpan, data berhasil disimpan ke dalam *database.*	Ketika memilih tombol Simpan, menampilkan data hasil ubah dan data tersimpan pada *database*	[√] Diterima [ ] Ditolak
Data baru; Nama Kategori: Switchgear			
Kasus Dan Hasil Uji Salah (Data Salah)			
Data baru; Nama Kategori : Alat Ukur Atau Nama Kategori :	Nama kategori "Alat Ukur " sudah ada di *database.* Ketika memilih tombol Simpan, tampil pesan peringatan data yang dimasukan sudah ada atau *field* tidak boleh kosong	Ketika memilih tombol Simpan, tampil pesan kesalahan sesuai validasi *field input* dan data *input*	[√] Diterima [ ] Ditolak

4. Pengujian Hapus Kategori

Pengujian hapus data satu kategori dan dapat dilihat pada tabel berikut ini.

Tabel 4.20 Pengujian Hapus Kategori

Kasus dan Hasil Uji			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data yang akan dihapus Contoh: Nama Kategori: Menara & Tiang	Setelah memilih tombol hapus tampil pesan konfirmasi . Jika memilih Ok, data akan dihapus. Jika memilih *Cancel* maka data tidak dihapus.	Setelah memilih tombol hapus tampil pesan konfirmasi. Jika memilih *Cancel* maka data tidak dihapus. Jika memilih Ok, data dihapus dan muncul pesan data berhasil dihapus.	[√] Diterima [ ] Ditolak
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data yang akan dihapus sudah digunakan oleh data barang Contoh: Nama Kategori: Transformator	Setelah memilih tombol hapus tampil pesan konfirmasi . Jika memilih Ok, data tidak terhapus. Jika memilih *Cancel* maka hapus data dibatalkan.	Setelah memilih tombol hapus tampil pesan konfirmasi. Jika memilih *Cancel* maka hapus data dibatalkan. Jika memilih Ok, data tidak terhapus dan muncul pesan data	[√] Diterima [ ] Ditolak

		gagal dihapus.	

4.2.2.4 Pengujian Pengolahan Data *Supplier*

Pengujian pengolahan data *supplier* terdiri dari proses cari, tambah, ubah, dan hapus *supplier*.

1. Pengujian Cari *Supplier*

Pengujian cari data *supplier* dapat dilihat pada tabel sebagai berikut.

Tabel 4.21 Pengujian Cari *Supplier*

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Nama Supplier : Dirga Supplier No Telp: 0226018256	Ketika menekan tombol Enter, tampil data barang yang dicari	Dapat menampilkan data satuan yang dicari	[√] Diterima [ ] Ditolak

2. Pengujian Tambah *Supplier*

Pengujian tambah data *Supplier* dapat dilihat pada tabel sebagai berikut.

Tabel 4.22 Pengujian Tambah *Supplier*

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Nama Supplier : C.V Jatayu Supplier No Telp: 0226018213 Alamat: Jl. Komud Supadio No. 97 Bandung	Ketika memilih tombol Simpan, data disimpan ke *database* dan tampil pesan databerhasil disimpan	Ketika memilih tombol Simpan, tersimpan pada *database*, menampilkan data hasil tambah dan pesan data berhasil disimpan	[√] Diterima [ ] Ditolak
Kasus Dan Hasil Uji Salah (Data Salah)			
Data input berupa Contoh: Nama Supplier : C.V Jatayu Supplier No Telp: 0226018467 Alamat:	Nama Supplier "C.V Jatayu Supplier" sudah ada di *database*. Ketika memilih tombol Simpan, tampil pesan peringatan data yang dimasukan sudah ada atau *field* tidak boleh kosong	Ketika memilih tombol Simpan, tampil pesan kesalahan sesuai validasi masing-masing *field input* dan data *input*	[√] Diterima [ ] Ditolak

3. Pengujian Ubah *Supplier*

Pengujian ubah data *supplier* dapat dilihat pada tabel berikut ini.

Tabel 4.23 Pengujian Ubah *Supplier*

<table>
<tr><td colspan="4">Kasus dan Hasil Uji (Data Benar)</td></tr>
<tr><td>Data Masukan</td><td>Yang diharapkan</td><td>Pengamatan</td><td>Kesimpulan</td></tr>
<tr><td>Data lama;
Nama Supplier : C.V Jatayu Supplier
No Telp: 0226018213
Alamat: Jl. Komud Supadio No. 97 Bandung</td><td rowspan="2">Ketika memilih tombol Simpan, data berhasil disimpan ke dalam database.</td><td rowspan="2">Ketika memilih tombol Simpan, menampilkan data hasil ubah dan data tersimpan pada database</td><td rowspan="2">[√] Diterima
[ ] Ditolak</td></tr>
<tr><td>Data baru;
Nama Supplier : C.V Pixma Power Indo
No Telp: 0226018213
Alamat: Jl. Komud Supadio No. 97 Bandung</td></tr>
<tr><td colspan="4">Kasus Dan Hasil Uji Salah (Data Salah)</td></tr>
<tr><td>Data baru;
Nama Supplier : C.V Pixma Power Indo
No Telp: 0226018213
Alamat:</td><td>Nama supplier “Dirga Supplier “ sudah ada di database. Ketika memilih tombol Simpan, tampil pesan peringatan data yang dimasukan sudah ada atau field tidak boleh kosong</td><td>Ketika memilih tombol Simpan, tampil pesan kesalahan sesuai validasi masing-masing field input dan data input</td><td>[√] Diterima
[ ] Ditolak</td></tr>
</table>

4. Pengujian Hapus *Supplier*

Pengujian hapus data satu *supplier* dan dapat dilihat pada tabel berikut ini.

Tabel 4.24 Pengujian Hapus *Supplier*

Kasus dan Hasil Uji			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data yang akan dihapus Contoh: Nama Supplier : C.V Edielson Mark No Telp: 0225532210 Alamat: Jl. Brajamusti No. 005 Bandung	Setelah memilih tombol hapus tampil pesan konfirmasi . Jika memilih Ok, data akan dihapus. Jika memilih *Cancel* maka data tidak dihapus.	Setelah memilih tombol hapus tampil pesan konfirmasi. Jika memilih *Cancel* maka data tidak dihapus. Jika memilih Ok, data dihapus dan muncul pesan data berhasil dihapus.	[√] Diterima [ ] Ditolak
Data yang akan dihapus digunakan oleh data transaksi Contoh: Nama Supplier : C.V Jatayu Supplier No Telp: 0226018213 Alamat: Jl. Komud Supadio No. 97 Bandung	Setelah memilih tombol hapus tampil pesan konfirmasi . Jika memilih Ok, data tidak terhapus. Jika memilih *Cancel* maka hapus data dibatalkan.	Setelah memilih tombol hapus tampil pesan konfirmasi. Jika memilih *Cancel* maka hapus data dibatalkan. Jika memilih Ok, data tidak terhapus dan muncul pesan data gagal dihapus.	[√] Diterima [ ] Ditolak

4.2.2.5 Pengujian Pengolahan Data Petugas

Pengujian pengolahan data petugas terdiri dari proses cari, tambah, ubah, dan hapus petugas.

1. Pengujian Cari Petugas

Pengujian cari data petugas dapat dilihat pada tabel sebagai berikut.

Tabel 4.25 Pengujian Cari Petugas

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Nama Petugas : bima NIP:10107240	Ketika menekan tombol Enter, tampil data barang yang dicari	Dapat menampilkan data satuan yang dicari	[√] Diterima [ ] Ditolak

2. Pengujian Tambah Petugas

Pengujian tambah data petugas dapat dilihat pada tabel sebagai berikut.

Tabel 4.26 Pengujian Tambah Petugas

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
NIP:10103010 Nama Petugas: dimas Kategori Petugas:petugas Password: bandung89	Ketika memilih tombol Simpan, data disimpan ke *database* dan tampil pesan data berhasil disimpan	Ketika memilih tombol Simpan, tersimpan pada *database*, menampilkan data hasil tambah dan pesan data berhasil disimpan	[√] Diterima [ ] Ditolak
Kasus Dan Hasil Uji Salah (Data Salah)			
NIP: Nama Petugas: dimas Kategori Petugas: Password: bandung89	Nama Petugas "dimas" sudah ada di *database.* Ketika memilih tombol Simpan, tampil pesan peringatan data yang dimasukan sudah ada dan atau *field* tidak boleh kosong	Ketika memilih tombol Simpan, tampil pesan kesalahan sesuai validasi masing-masing *field input* dan data *input*	[√] Diterima [ ] Ditolak

3. Pengujian Ubah Petugas

Pengujian ubah data petugas dapat dilihat pada tabel berikut ini.

Tabel 4.27 Pengujian Ubah Petugas

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data lama; NIP:10103010 Nama Petugas: dimas Password: bandung89	Ketika memilih tombol Simpan, data berhasil disimpan ke dalam *database.*	Ketika memilih tombol Simpan, data disimpan ke *database*. Tampil data hasil ubah dan pesan data berhasil disimpan	[√] Diterima [ ] Ditolak
Data baru; NIP:10103021 Nama Petugas: dimas Password: 89bandung32			

Tabel 4.28 Pengujian Ubah Petugas (Lanjutan)

Kasus Dan Hasil Uji Salah (Data Salah)			
Data baru; NIP:10107240 Nama Petugas: bima Password: bandung89	Nama petugas "bima " sudah ada di *database*. Ketika memilih tombol Simpan, tampil pesan peringatan data yang dimasukan sudah ada atau *field* tidak boleh kosong	Ketika memilih tombol Simpan, tampil pesan kesalahan sesuai validasi masing-masing *field input* dan data *input*	[√] Diterima [ ] Ditolak

4. Pengujian Hapus Petugas

Pengujian hapus data satu petugas dan dapat dilihat pada tabel berikut ini.

Tabel 4.29 Pengujian Hapus Petugas

Kasus dan Hasil Uji			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data yang akan dihapus Contoh: NIP:10103010 Nama Petugas: dimas Kategori Petugas:petugas Password: bandung89	Setelah memilih tombol hapus tampil pesan konfirmasi . Jika memilih Ok, data akan dihapus. Jika memilih *Cancel* maka data tidak dihapus.	Setelah memilih tombol hapus tampil pesan konfirmasi. Jika memilih *Cancel* maka data tidak dihapus. Jika memilih Ok, data dihapus dan muncul pesan data berhasil dihapus.	[√] Diterima [ ] Ditolak
Data yang akan dihapus berelasi dengan data transaksi Contoh: NIP:1010720 Nama Petugas: bima Kategori Petugas:petugas Password: 1234	Setelah memilih tombol hapus tampil pesan konfirmasi . Jika memilih Ok, data tidak terhapus. Jika memilih *Cancel* maka hapus data dibatalkan.	Setelah memilih tombol hapus tampil pesan konfirmasi. Jika memilih *Cancel* maka hapus data dibatalkan. Jika memilih Ok, data tidak terhapus dan muncul pesan data gagal dihapus.	[√] Diterima [ ] Ditolak

4.2.2.6 Pengujian Pengolahan Data Pemesanan

Pengujian pengolahan data pemesanan terdiri dari proses cari, tambah, ubah, dan hapus pemesanan.

1. Pengujian Cari Pemesanan

Pengujian cari data pemesanan dapat dilihat pada tabel sebagai berikut.

Tabel 4.30 Pengujian Cari Pemesanan

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
No Pemesanan: PSN120602001 Tgl Pemesanan:2012-06-02 Petugas:admin	Ketika menekan tombol Enter, tampil data barang yang dicari	Dapat menampilkan data satuan yang dicari	[√] Diterima [ ] Ditolak

2. Pengujian Tambah Pemesanan

Pengujian tambah data pemesanan dapat dilihat pada tabel sebagai berikut.

Tabel 4.31 Pengujian Tambah Pemesanan

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Contoh; Tgl Pemesanan:2012-07-16 Supplier: C.V. Jatayu Supplier Nama Barang: MCB Kategori: Alat Ukur Satuan: Buah Jumlah Pesan: 100	Ketika memilih tombol Simpan, data disimpan ke *database* dan tampil pesan data berhasil disimpan	Ketika memilih tombol Simpan, tersimpan pada *database*, menampilkan data hasil tambah dan pesan data berhasil disimpan	[√] Diterima [ ] Ditolak
Kasus Dan Hasil Uji Salah (Data Salah)			
Contoh; Tgl Pemesanan:2012-07-16 Supplier: Nama Barang: MCB Kategori: Alat Ukur Satuan: Buah Jumlah Pesan:	Ketika memilih tombol Simpan, tampil pesan peringatan "Supplier tidak boleh kosong", "Maaf Jumlah belum diisi"	Ketika memilih tombol Simpan, tampil pesan kesalahan sesuai validasi masing-masing *field input* dan data *input*	[√] Diterima [ ] Ditolak

3. Pengujian Ubah Pemesanan

Pengujian ubah data pemesanan dapat dilihat pada tabel berikut ini.

Tabel 4.32 Pengujian Ubah Pemesanan

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data lama; Tgl Pemesanan:2012-07-16 Supplier: C.V. Jatayu Supplier Nama Barang: MCB Kategori: Alat Ukur Satuan: Buah Jumlah Pesan: 100	Ketika memilih tombol Simpan, data berhasil disimpan ke dalam *database.*	Ketika memilih tombol Simpan, data disimpan ke *database*. Tampil data hasil ubah dan pesan data berhasil disimpan	[√] Diterima [ ] Ditolak
Data baru; Tgl Pemesanan:2012-07-17 Supplier: C.V. Jatayu Supplier Nama Barang: MCB Kategori: Alat Ukur Satuan: Buah Jumlah Pesan: 60			
Kasus Dan Hasil Uji Salah (Data Salah)			
Data baru; Tgl Pemesanan:2012-07-17 Supplier: Nama Barang: MCB Kategori: Alat Ukur Satuan: Buah Jumlah Pesan: aa34	Ketika memilih tombol Simpan, tampil pesan peringatan "Supplier tidak boleh kosong", "Maaf nilai yang dimasukan salah"	Ketika memilih tombol Simpan, data tidak disimpan di *database* lalu tampil pesan kesalahan sesuai validasi masing-masing *field input* dan data *input*	[√] Diterima [ ] Ditolak

4. Pengujian Hapus Pemesanan

Pengujian hapus data satu petugas dan dapat dilihat pada tabel berikut ini.

Tabel 4.33 Pengujian Hapus Petugas

Kasus dan Hasil Uji			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data yang akan dihapus Contoh: Tgl Pemesanan:2012-07-16 Supplier: C.V. Jatayu Supplier Nama Barang: MCB Kategori: Alat Ukur Satuan: Buah Jumlah Pesan: 100	Setelah memilih tombol hapus tampil pesan konfirmasi . Jika memilih Ok, data akan dihapus. Jika memilih *Cancel* maka data tidak dihapus.	Setelah memilih tombol hapus tampil pesan konfirmasi. Jika memilih *Cancel* maka data tidak dihapus. Jika memilih Ok, data dihapus dan muncul pesan data berhasil dihapus.	[√] Diterima [ ] Ditolak
Data yang akan dihapus berelasi dengan data penerimaan Contoh: Tgl Pemesanan:2013-05-01 Supplier: C.V Star Vision Nama Barang: MCB Kategori: Alat Ukur Satuan: Buah Jumlah Pesan: 500	Setelah memilih tombol hapus tampil pesan konfirmasi . Jika memilih Ok, data tidak terhapus. Jika memilih *Cancel* maka hapus data dibatalkan.	Setelah memilih tombol hapus tampil pesan konfirmasi. Jika memilih *Cancel* maka hapus data dibatalkan. Jika memilih Ok, data tidak terhapus dan muncul pesan data gagal dihapus.	[√] Diterima [ ] Ditolak

4.2.2.7 Pengujian Pengolahan Data Penerimaan

Pengujian pengolahan data penerimaan terdiri dari proses cari, tambah, ubah, dan hapus penerimaan

1. Pengujian Cari Penerimaan

Pengujian cari data penerimaan dapat dilihat pada tabel sebagai berikut.

Tabel 4.34 Pengujian Cari Penerimaan

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
No Penerimaan: TRM120602001 Tgl Peneirmaan:2012-06-02 Petugas:admin No Faktur:0001	Ketika menekan tombol Enter, tampil data barang yang dicari	Dapat menampilkan data satuan yang dicari	[√] Diterima [ ] Ditolak

2. Pengujian Tambah Penerimaan

Pengujian tambah data penerimaan dapat dilihat pada tabel sebagai berikut.

Tabel 4.35 Pengujian Tambah Penerimaan

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Contoh; Tgl Penerimaan:2012-07-16 Supplier: C.V. Jatayu Supplier Nama Barang: MCB Kategori: Alat Ukur Satuan: Buah Jumlah: 100	Ketika memilih tombol Simpan, data disimpan ke *database* dan tampil pesan data berhasil disimpan	Ketika memilih tombol Simpan, tersimpan pada *database*, menampilkan data hasil tambah dan pesan data berhasil disimpan	[√] Diterima [ ] Ditolak
Kasus Dan Hasil Uji Salah (Data Salah)			
Contoh; Tgl Penerimaan:2012-07-16 Supplier: Nama Barang: MCB Kategori: Alat Ukur Satuan: Buah Jumlah:	Ketika memilih tombol Simpan, tampil pesan peringatan "Supplier tidak boleh kosong", "Maaf Jumlah belum diisi"	Ketika memilih tombol Simpan, tampil pesan kesalahan sesuai validasi masing-masing *field input* dan data *input*	[√] Diterima [ ] Ditolak

3. Pengujian Ubah Penerimaan

Pengujian ubah data penerimaan dapat dilihat pada tabel berikut ini.

Tabel 4.36 Pengujian Ubah Penerimaan

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data lama; Tgl Penerimaan:2012-07-16 Supplier: C.V. Jatayu Supplier Nama Barang: MCB Kategori: Alat Ukur Satuan: Buah Jumlah: 100	Ketika memilih tombol Simpan, data berhasil disimpan ke dalam *database.*	Ketika memilih tombol Simpan, data disimpan ke *database*. Tampil data hasil ubah dan pesan data berhasil disimpan	[√] Diterima [ ] Ditolak
Data baru; Tgl Penerimaan:2012-07-17 Supplier: C.V. Jatayu Supplier Nama Barang: MCB Kategori: Alat Ukur Satuan: Buah Jumlah: 60			
Kasus Dan Hasil Uji Salah (Data Salah)			
Data baru; Tgl Pemesanan:2012-07-17 Supplier: Nama Barang: MCB Kategori: Alat Ukur Satuan: Buah Jumlah: aa34	Ketika memilih tombol Simpan, tampil pesan peringatan "Supplier tidak boleh kosong", "Maaf nilai yang dimasukan salah"	Ketika memilih tombol Simpan, data tidak disimpan di *database* lalu tampil pesan kesalahan sesuai validasi masing-masing *field input* dan data *input*	[√] Diterima [ ] Ditolak

4. Pengujian Hapus penerimaan

Pengujian hapus data penerimaan dapat dilihat pada tabel berikut ini.

Tabel 4.37 Pengujian Hapus Penerimaan

Kasus dan Hasil Uji			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data yang akan dihapus Contoh: No Penerimaan: TRM120602001 Tgl Penerimaan:2012-07-16 Supplier: C.V. Jatayu Supplier Nama Barang: MCB Kategori: Alat Ukur Satuan: Buah Jumlah: 100	Setelah memilih tombol hapus tampil pesan konfirmasi . Jika memilih Ok, data akan dihapus. Jika memilih *Cancel* maka data tidak dihapus.	Setelah memilih tombol hapus tampil pesan konfirmasi. Jika memilih *Cancel* maka data tidak dihapus. Jika memilih Ok, sistem melakukan *update* jumlah sisa di data detail pemesanan dan data barang, lalu data dihapus dan muncul pesan data berhasil dihapus.	[√] Diterima [ ] Ditolak

4.2.2.8 Pengujian Pengolahan Data Pengeluaran

Pengujian pengolahan data pengeluaran terdiri dari proses cari, tambah, ubah, dan hapus pengeluaran

1. Pengujian Cari Pengeluaran

Pengujian cari data pengeluaran dapat dilihat pada tabel sebagai berikut.

Tabel 4.38 Pengujian Cari Pengeluaran

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
No Faktur: KLR120602001 Tgl Pengeluaran:2012-06-01 Petugas:admin No Permintaan:0001 Kantor Cabang: PT.PLN Persero UPJ Pangalengan	Ketika menekan tombol Enter, tampil data barang yang dicari	Dapat menampilkan data satuan yang dicari	[√] Diterima [ ] Ditolak

2. Pengujian Tambah Pengeluaran

Pengujian tambah data pengeluaran dapat dilihat pada tabel sebagai berikut.

Tabel 4.39 Pengujian Tambah Pengeluaran

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Contoh; Tgl Pengeluaran:2012-07-16 Kacab: PT.PLN Persero UPJ Kertasari No Telepon: 022576223 Petugas Kacab: Maman Nama Barang: MCB Kategori: Alat Ukur Satuan: Buah Jumlah: 100	Ketika memilih tombol Simpan, data disimpan ke *database* dan tampil pesan data berhasil disimpan	Ketika memilih tombol Simpan, sistem melakukan *update* stok barang tersimpan pada *database*, menampilkan data hasil tambah dan pesan data berhasil disimpan	[√] Diterima [ ] Ditolak
Kasus Dan Hasil Uji Salah (Data Salah)			
Contoh; Tgl Pengeluaran:2012-07-16 Kantor Cabang: Petugas Kacab: No Telepon: 022576223 Nama Barang: MCB Kategori: Alat Ukur Satuan: Buah Jumlah:	Ketika memilih tombol Simpan, tampil pesan peringatan "Kantor Cabang tidak boleh kosong", "Petugas Kacab tidak boleh kosong", dan "Maaf Jumlah belum diisi"	Ketika memilih tombol Simpan, tampil pesan kesalahan sesuai validasi masing-masing *field input* dan data *input*	[√] Diterima [ ] Ditolak

3. Pengujian Ubah Pengeluaran

Pengujian ubah data penerimaan dapat dilihat pada tabel berikut ini.

Tabel 4.40 Pengujian Ubah Penerimaan

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data lama; Tgl Pengeluaran:2012-07-16 Kantor Cabang: PT.PLN Persero UPJ Kertasari Petugas Kacab: Maman Nama Barang: MCB Kategori: Alat Ukur Satuan: Buah Jumlah: 100	Ketika memilih tombol Simpan, data berhasil disimpan ke dalam *database.*	Ketika memilih tombol Simpan, sistem melakukan *update* stok barang lalu data disimpan ke *database*. Tampil data hasil ubah dan pesan data berhasil disimpan	[√] Diterima [ ] Ditolak
Data baru; Tgl Pengeluaran:2012-07-18 Kantor Cabang: PT.PLN Persero UPJ Kertasari Petugas Kacab: Maman Nama Barang: MCB Kategori: Alat Ukur Satuan: Buah Jumlah: 90			
Kasus Dan Hasil Uji Salah (Data Salah)			
Data baru; Tgl Pengeluaran:2012-07-19 Kantor Cabang: PT.PLN Persero UPJ Kertasari Petugas Kacab: Nama Barang: MCB Kategori: Alat Ukur Satuan: Buah Jumlah:Asx112	Ketika memilih tombol Simpan, tampil pesan peringatan "Kantor cabang tidak boleh kosong", "Petugas Kacab tidak boleh kosong","Maaf nilai yang dimasukan salah"	Ketika memilih tombol Simpan, data tidak disimpan di *database* lalu tampil pesan kesalahan sesuai validasi masing-masing *field input* dan data *input*	[√] Diterima [ ] Ditolak

4. Pengujian Hapus penerimaan

Pengujian hapus data penerimaan dapat dilihat pada tabel berikut ini.

Tabel 4.41 Pengujian Hapus Penerimaan

Kasus dan Hasil Uji			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data yang akan dihapus Contoh: No Faktur: KLR120718001 Pengeluaran:2012-07-18 Kantor Cabang: PT.PLN Persero UPJ Kertasari Petugas Kacab: Dahlan Nama Barang: MCB Kategori: Alat Ukur Satuan: Buah Jumlah: 90	Setelah memilih tombol hapus tampil pesan konfirmasi . Jika memilih Ok, data akan dihapus. Jika memilih *Cancel* maka data tidak dihapus.	Setelah memilih tombol hapus tampil pesan konfirmasi. Jika memilih *Cancel* maka data tidak dihapus. Jika memilih Ok, sistem melakukan *update* stok barang, lalu data dihapus dan muncul pesan data berhasil dihapus.	[√] Diterima [ ] Ditolak

4.2.2.9 Pengujian Hitung Rencana Inventori

Pengujian menu hitung rencana inventori dapat dilihat pada tabel berikut ini.

Tabel 4.42 Pengujian Hitung Rencana Inventori

Kasus dan Hasil Uji (Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Barang: KWH 2 Periode 1: Januari Periode 2: Juni Tahun: Ongkos Pesan: 1000000 Ongkos Simpan: 5000 Ongkos Kekurangan: 5000 Waktu Estimasi: 2	Ketika tombol Rencana dipilih, akan tampil data pengeluaran selama periode yang ditentukan dan hasil perhitungan (kebijakan inventori)	Dapat melakukan perhitungan rencana inventori , tampil data pengeluaran dan hasil perhitungan (kebijakan inventori)	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Salah)			
Barang: Periode 1: Januari Periode 2: Tahun: Ongkos Pesan: asd1 Ongkos Simpan: 5000 Ongkos Kekurangan: -600 Waktu Estimasi: 15	Menampilkan pesan kesalahan berdasarkan diisi atau tidaknya *field input* serta pesan kesalahan dari data input yang salah	Dapat menampilkan pesan kesalahan berdasarkan diisi atau tidaknya *field input* serta pesan kesalahan dari data input yang salah, tidak dapat melakukan perhitungan dan tidak tampil hasil perhitungan (kebijakan inventori)	[√] Diterima [ ] Ditolak

4.2.2.10 Pengujian Pengolahan Laporan

Pengujian pengolahan laporan dilakukan terhadap tiga jenis menu laporan, yaitu laporan pemesanan, penerimaan, dan laporan pengeluaran.

1. Pengujian Laporan Pemesanan

Pengujian terhadap menu laporan pemesanan dapat dilihat pada tabel beikut.

Tabel 4.43 Pengujian Laporan Pemesanan

Kasus dan Hasil Uji (Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tgl Pemesanan: 2012-01-01 s.d 2012-07-30 Nama Barang: MCB 220v Kategori: Alat Ukur	Ketika tombol cari dipilih, tampil hasil pencarian laporan. Ketika tombol Cetak dipilih, tampil halaman cetak dan dialog *printer*	Dapat menampilkan hasil pencarian, halaman cetak dan dialog *printer*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Salah)			
Tgl Pemesanan: 2012-01-01 s.d 2012-07-30 Nama Barang: MCB 220v Kategori: Alat Ukur	Bila memang data pemesanan tidak ada selama waktu cari yang ditentukan, maka akan tampil pesan data tidak ditemukan	Dapat menampilkan pesan data tidak ditemukan	[√] Diterima [ ] Ditolak

2. Pengujian Laporan Penerimaan

Pengujian terhadap menu laporan penerimaan dapat dilihat pada tabel beikut.

Tabel 4.44 Pengujian Laporan Penerimaan

Kasus dan Hasil Uji (Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tgl Penerimaan: 2012-01-01 s.d 2012-07-30 Nama Barang: MCB 220v Kategori: Alat Ukur	Ketika tombol cari dipilih, tampil hasil pencarian laporan. Ketika tombol Cetak dipilih, tampil halaman cetak dan dialog *printer*	Dapat menampilkan hasil pencarian, halaman cetak dan dialog *printer*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Salah)			
Tgl Penerimaan: 2012-01-01 s.d 2012-07-30 Nama Barang: MCB 220v Kategori: Alat Ukur	Bila memang data penerimaan tidak ada selama waktu cari yang ditentukan, maka akan tampil pesan data tidak ditemukan	Dapat menampilkan pesan data tidak ditemukan	[√] Diterima [ ] Ditolak

3. Pengujian Laporan Pengeluaran

Pengujian terhadap menu laporan pengeluaran dapat dilihat pada tabel beikut.

Tabel 4.45 Pengujian Laporan Penerimaan

Kasus dan Hasil Uji (Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tgl Pengeluaran: 2012-01-01 s.d 2012-07-30 Nama Barang: MCB 220v Kategori: Alat Ukur	Ketika tombol cari dipilih, tampil hasil pencarian laporan. Ketika tombol Cetak dipilih, tampil halaman cetak dan dialog *printer*	Dapat menampilkan hasil pencarian, halaman cetak dan dialog *printer*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Salah)			
Tgl Pengeluaran: 2012-01-01 s.d 2012-07-30 Nama Barang: MCB 220v Kategori: Alat Ukur	Bila memang data pengeluaran tidak ada selama waktu cari yang ditentukan, maka akan tampil pesan data tidak ditemukan	Dapat menampilkan pesan data tidak ditemukan	[√] Diterima [ ] Ditolak

4.2.2.11 Pengujian *Logout*

Pengujian *logout* dapat dilihat pada tabel berikut.

Tabel 4.46 Pengujian *Logout*

Kasus dan Hasil Uji (Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tombol Logout	Dapat keluar dan menghapus *session* pengguna	Pengguna (*user*) dapat keluar dari aplikasi	[√] Diterima [ ] Ditolak

4.2.3 Kesimpulan Hasil Pengujian *Black Box*

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

4.2.4 Pengujian Beta

Pengujian beta merupakan pengujian yang dilakukan secara objektif dimana pengujian dilakukan secara langsung ke lapangan yaitu instansi yang bersangkutan mengenai kepuasan *user* dengan kandungan poin yaitu pemenuhan kebutuhan dari tujuan awal pembangunan sistem informasi *inventory control* dan tampilan antarmuka dari sistem informasi tersebut. Pengujian beta dilakukan melalui teknik pengambilan data melalui wawancara.

Pengujian beta melalui wawancara dilakukan untuk halaman admin dan halaman petugas pada sistem informasi *inventory control*, karena terbatasnya jumlah pengguna untuk admin dan petugas. Wawancara mengenai halaman admin dilakukan terhadap *supervisor* logistik, sedangkan untuk wawancara mengenai halaman petugas dilakukan terhadap *junior* logistik.

4.2.4.1 Wawancara Pengguna

Wawancara dilakukan dengan menggunakan teknik kualitatif melalui pertanyaan terbuka. Wawancara mengenai halaman admin dilakukan kepada satu orang yaitu petugas *supervisor* logistik, sedangkan untuk wawancara mengenai halaman petugas dilakukan terhadap satu orang yaitu *junior* logistik. Dari hasil wawancara tersebut akan dilakukan pengambilan kesimpulan terhadap penilaian penerapan sistem yang baru.

Wawancara terdiri dari beberapa pertanyaan terbuka yang akan ditanyakan kepada narasumber.

4.2.4.1.1 Wawancara Pengujian Beta Untuk Halaman Admin

Wawancara untuk halaman Admin ini dilakukan untuk pengujian beta terhadap sistem informasi *inventory control* kepada *supervisor* logistik dengan total narasumber sebanyak 1 orang. Masing-masing jawaban dari pertanyaan hasil wawancara tersebut adalah sebagai berikut.

1. Pertanyaan : Menurut Anda, apakah halaman untuk admin pada sistem informasi ini membantu anda dalam menentukan rencana kebutuhan persediaan untuk periode mendatang?

   Jawaban : *Supervisor* logistik beranggapan bahwa sistem yang dibangun cukup membantu dengan memberikan gambaran tentang jumlah rencana kebutuhan persediaan untuk waktu atau periode mendatang.

Berdasarkan hasil wawancara di atas maka dapat disimpulkan bahwa halaman admin pada sistem informasi mampu memenuhi kebutuhan *supervisor* logistik dalam menentukan rencana kebutuhan persediaan untuk periode mendatang.

2. Pertanyaan : Menurut Anda, apakah halaman untuk admin pada sistem informasi ini membantu dalam membuat kebijakan persediaan dengan menentukan jumlah pemesanan yang efektif?

   Jawaban : *Supervisor* logistik menjawab bahwa sistem yang dibangun cukup membantu dalam menentukan jumlah atau kuantitas pemesanan yang efektif sehingga kebijakan persediaan yang dibuat bisa lebih baik.

Berdasarkan hasil wawancara di atas maka dapat disimpulkan bahwa halaman admin pada sistem informasi ini dapat membantu *supervisor* logistik dalam membuat kebijakan persediaan dengan cara menentukan kuantitas pemesanan yang efektif.

3. Pertanyaan : Menurut Anda, apakah halaman untuk admin pada sistem informasi ini membantu dalam pengendalian persediaan sehingga persediaan efektif terhadap ketidakpastian pada permintaan atau pengeluaran barang?

   Jawaban : *Supervisor* logistik menjawab bahwa sistem yang dibangun membantu dalam keefektifan persediaan terhadap tingkat permintaan atau pengeluaran barang sehingga persediaan yang ada dapat lebih terkendali.

Berdasarkan hasil wawancara di atas maka dapat disimpulkan bahwa sistem informasi ini dapat membantu *supervisor* logistik dalam keefektifan persediaan terhadap ketidakpastian pada tingkat permintaan atau pengeluaran barang sehingga persediaan dapat lebih terkendali.

4. Pertanyaan : Menurut Anda, apakah sistem informasi ini dapat membantu menigkatkan keefektifan dan kestabilan kinerja divisi logistik?

   Jawaban : *Supervisor* logistik menjawab bahwa dengan sistem yang dibangun pengolahan berbagai dapat dilakukan lebih cepat dan akurat sehingga dapat meningkatkan keefektifan dan kestabilan kinerja divisi logistik.

Berdasarkan hasil wawancara di atas maka dapat disimpulkan bahwa sistem informasi ini dapat meningkatkan keefektifan dan kestabilan kinerja divisi logistik.

4.2.4.1.2 Wawancara Pengujian Beta Untuk Halaman Petugas

Wawancara untuk halaman petugas ini dilakukan untuk pengujian beta terhadap sistem informasi *inventory control* kepada *junior* logistik dengan total narasumber sebanyak 1 orang. Jawaban dari masing-masing pertanyaan hasil wawancara tersebut adalah sebagai berikut.

1. Pertanyaan : Menurut Anda, apakah sistem informasi yang dibangun mempermudah dalam melakukan pengolahan data transaksi?

   Jawaban : *Junior* logistik menjawab mampu dengan alasan bahwa sistem informasi ini membuat proses kerja yang dilakukannya bisa lebih cepat dan mudah dalam menangani transaksi yang ada di gudang.

Berdasarkan hasil wawancara di atas maka dapat disimpulkan bahwa halaman petugas pada sistem informasi ini membantu *junior* logistik melakukan pengolahan data transaksi.

2. Pertanyaan : Apakah sistem informasi ini mampu mencegah kesalahan dalam pencatatan atau pengolahan data transaksi serta keefektifan dan keakuratan data terjaga?

   Jawaban : *Junior* logistik menjawab mampu dengan alasan bahwa sistem informasi ini membuat proses kerja yang dilakukannya bisa lebih cepat dan mudah dalam menangani transaksi.

Berdasarkan hasil wawancara di atas dapat disimpulkan bahwa halaman petugas pada sistem informasi ini bisa mencegah kesalahan pencatatan atau pengolahan data transaksi serta cukup mampu membuat data lebih efektif dan akurat.

3. Pertanyaan : Apakah halaman untuk petugas pada sistem informasi ini dapat mempermudah Anda dalam pembuatan laporan?

   Jawaban : *Junior* logistik beranggapan bahwa sistem yang dibangun mampu mempermudah dan mempercepat membuat laporan.

Berdasarkan hasil wawancara di atas maka dapat disimpulkan bahwa halaman petugas pada sistem informasi memudahkan *junior* logistik dalam membuat laporan.

4. Pertanyaan : Menurut Anda, apakah secara keseluruhan tampilan fungsionalitas halaman petugas pada sistem informasi ini mudah dipahami dan digunakan?

   Jawaban : *Junior* logistik menyatakan tidak mengalami kesulitan untuk memahami dan menggunakan halaman petugas pada sistem yang dibangun.

Berdasarkan hasil wawancara di atas maka dapat disimpulkan bahwa halaman petugas pada sistem informasi ini cukup mudah dipahami dan digunakan.

4.2.5 Kesimpulan Hasil Pengujian Beta

Berdasarkan hasil wawancara yang didapatkan dari pengujian beta terhadap *supervisor* logistik dan *junior* logistik diperoleh kesimpulan bahwa sistem informasi *inventory control* ini membantu berbagai pengolahan data yang dibutuhkan pada divisi logistik sehingga cukup mambantu meningkatkan keefektifan dan kestabilan kinerja divisi tersebut. Sistem yang dibagun juga dapat membantu dalam menentukan tingkat persediaan yang efektif untuk periode atau waktu yang akan datang sehingga persediaan yang ada di gudang lebih terkendali. Secara keseluruhan sistem informasi *inventory control* yang dibangun sudah memenuhi kebutuhan divisi logistik, mudah dipahami, dipelajari dan digunakan.