BAB 4
IMPLEMENTASI DAN PENGUJIAN

4.1. Implementasi Sistem

Setelah sistem dianalisis dan didesain secara terperinci maka selanjutnya akan dilakukan tahap implementasi. Implementasi sistem merupakan tahap peletakan sistem sehingga dapat dioperasikan. Implementasi bertujuan untuk mengkonfirmasi modul-modul perancangan sehingga pengguna sistem dapat memberikan masukan-masukan terhadap pengembang sistem.

4.1.1. Lingkungan Implementasi

Untuk mendukung aplikasi yang akan diterapkan pada lingkungan implementasi, maka dalam hal ini harus memperhatikan perangkat keras dan perangkat lunak yang menunjang dalam aplikasi sistem informasi ini.

4.1.2. Implementasi Perangkat Lunak

Perangkat lunak yang digunakan untuk implementasi sistem informasi ini dapat dilihat pada table 4.1 berikut.

Tabel 4. 1 Implementasi Perangkat Lunak

No	***Perangkat Lunak***	***Keterangan***
1.	Sistem Operasi	Windows 7
2.	*Web Server*	WampServer 2.2e
3.	*DBMS*	MySQL 5
4.	*Web browser*	Mozilla Firefox 12
5.	*Code Editor*	Macromedia Dreamweaver CS6

4.1.3. Implementasi Perangkat Keras

Perangkat keras yang digunakan untuk implementasi sistem informasi ini dapat dilihat pada tabel 4.2 berikut.

Tabel 4. 2 Implementasi Perangkat Keras

	Perangkat Keras	Spesifikas
1.	Processor	Prosesor Core 2 Duo 1,5 GHz
2.	Monitor	Monitor17 inch
3.	VGA	VGA 512 Mb
4.	Memori	2 GB
5.	*Space Hardisk*	80 GB
6.	Jaringan	Internet pada Jaringan Dinkes
7.	Printer	Printer berwarna
8.	*Mouse* dan *Key Board*	Standar

4.1.4. Implementasi Basis Data

Pembangunan *database* dilakukan dengan menggunakan aplikasi *DBMS MySQL*. Implementasi *database* dalam bahasa *SQL* dan berikut adalah langkah-langkah dalam pembuatan tabel :

1. Tabel detail_kartu_stok

No	SQL
1	`CREATE TABLE IF NOT EXISTS `detail_kartu_stok` (`
2	`  `id_det_kartu_stok` int(11) NOT NULL,`
3	`  `tgl_kartu` date NOT NULL,`
4	`  `masuk` int(11) NOT NULL,`
5	`  `keluar` int(11) NOT NULL,`
6	`  `sisa` int(11) NOT NULL,`
7	`  `keterangan` text NOT NULL,`
8	`  `status` enum('ok','tunggu') NOT NULL DEFAULT`
9	`'tunggu',`
10	`  `no_kartu` int(11) NOT NULL,`
11	`  `id_det_pengeluaran` int(11) DEFAULT NULL,`
12	`  PRIMARY KEY (`id_det_kartu_stok`),`
13	`  KEY `no_kartu` (`no_kartu`),`
14	`  KEY `id_det_pengeluaran` (`id_det_pengeluaran`),`
15	`  CONSTRAINT `detail_kartu_stok_ibfk_1` FOREIGN KEY`
16	`(`no_kartu`) REFERENCES `kartu_stok` (`no_kartu`) ON`
17	`DELETE RESTRICT ON UPDATE RESTRICT,`
18	`  CONSTRAINT `detail_kartu_stok_ibfk_2` FOREIGN KEY`
19	`(`id_det_pengeluaran`) REFERENCES `detail_pengeluaran``
20	`(`id_det_pengeluaran`) ON DELETE RESTRICT ON UPDATE`
21	`RESTRICT ) ENGINE=InnoDB DEFAULT CHARSET=latin1;`

2. Tabel detail_pemakaian

NO	SQL
1	`CREATE TABLE IF NOT EXISTS `detail_pemakaian` (`
2	`  `id_det_pemakaian` int(11) NOT NULL,`
3	`  `jumlah_pemakaian` int(11) NOT NULL,`
4	`  `status` enum('ok','tunggu') NOT NULL DEFAULT`
5	`'tunggu',`
6	`  `no_kartu` int(11) NOT NULL,`
7	`  `id_pemakaian` int(11) NOT NULL,`
8	`  PRIMARY KEY (`id_det_pemakaian`),`
9	`  KEY `no_kartu` (`no_kartu`),`
10	`  KEY `id_pemakaian` (`id_pemakaian`),`
11	`  CONSTRAINT `detail_pemakaian_ibfk_1` FOREIGN KEY`
12	`(`no_kartu`) REFERENCES `kartu_stok` (`no_kartu`) ON`
13	`DELETE RESTRICT ON UPDATE RESTRICT,`
14	`  CONSTRAINT `detail_pemakaian_ibfk_2` FOREIGN KEY`
15	`(`id_pemakaian`) REFERENCES `pemakaian``
16	`(`id_pemakaian`) ON DELETE RESTRICT ON UPDATE RESTRICT`
17	`) ENGINE=InnoDB DEFAULT CHARSET=latin1;`

3. Tabel detail_pengeluaran

NO	SQL
1	`CREATE TABLE IF NOT EXISTS `detail_pengeluaran` (`
2	`  `id_det_pengeluaran` int(11) NOT NULL,`
3	`  `jumlah_pengeluaran` int(11) NOT NULL,`
4	`  `id_pengeluaran` int(11) NOT NULL,`
5	`  `id_det_permintaan` int(11) NOT NULL,`
6	`  PRIMARY KEY (`id_det_pengeluaran`),`
7	`  KEY `id_det_permintaan` (`id_det_permintaan`),`
8	`  KEY `id_pengeluaran` (`id_pengeluaran`),`
9	`  CONSTRAINT `detail_pengeluaran_ibfk_1` FOREIGN KEY`
10	`(`id_pengeluaran`) REFERENCES `pengeluaran``
11	`(`id_pengeluaran`) ON DELETE RESTRICT ON UPDATE`
12	`RESTRICT,`
13	`  CONSTRAINT `detail_pengeluaran_ibfk_2` FOREIGN KEY`
14	`(`id_det_permintaan`) REFERENCES `detail_permintaan``
15	`(`id_det_permintaan`) ON DELETE RESTRICT ON UPDATE`
16	`RESTRICT`
17	`) ENGINE=InnoDB DEFAULT CHARSET=latin1;`

4. Tabel detail_permintaan

NO	SQL
1	`CREATE TABLE IF NOT EXISTS `detail_permintaan` (`
2	`  `id_det_permintaan` int(11) NOT NULL,`
3	`  `jumlah_permintaan` int(11) NOT NULL,`
4	`  `status` enum('ok','tunggu') NOT NULL,`
5	`  `kode_obat` varchar(10) NOT NULL,`

```
6    `id_permintaan` int(11) NOT NULL,
7    PRIMARY KEY (`id_det_permintaan`),
8    KEY `kode_obat` (`kode_obat`),
9    KEY `id_permintaan` (`id_permintaan`),
10   CONSTRAINT `detail_permintaan_ibfk_1` FOREIGN KEY
11 (`kode_obat`) REFERENCES `obat` (`kode_obat`) ON DELETE
12 RESTRICT ON UPDATE RESTRICT,
13   ADD CONSTRAINT `detail_permintaan_ibfk_2` FOREIGN KEY
14 (`id_permintaan`) REFERENCES `permintaan`
15 (`id_permintaan`) ON DELETE RESTRICT ON UPDATE RESTRICT
16 ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5. Tabel kartu_stok

NO	SQL

```
1  CREATE TABLE IF NOT EXISTS `kartu_stok` (
2    `no_kartu` int(11) NOT NULL,
3    `stok` int(11) NOT NULL,
4    `lokasi` enum('gudang','puskesmas') NOT NULL,
5    `kode_obat` varchar(10) NOT NULL,
6    `id_kemasan` int(11) NOT NULL,
7    PRIMARY KEY (`no_kartu`)
8  ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

6. Tabel Kemasan

NO	SQL

```
1  CREATE TABLE IF NOT EXISTS `kemasan` (
2    `id_kemasan` int(11) NOT NULL,
3    `nama_kemasan` varchar(30) NOT NULL,
4    `id_satuan` int(11) NOT NULL,
5    PRIMARY KEY (`id_kemasan`),
6  KEY `id_satuan` (`id_satuan`),
7    CONSTRAINT `satuan_ibfk_1` FOREIGN KEY (`id_satuan`)
8  REFERENCES `satuan` (`id_satuan`) ON DELETE RESTRICT ON
9  UPDATE RESTRICT
10 ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

7. Tabel Satuan

NO	SQL

```
1  CREATE TABLE IF NOT EXISTS `satuan` (
2    `id_satuan` int(11) NOT NULL,
3    `nama_satuan` varchar(30) NOT NULL,
4    PRIMARY KEY (`id_satuan`)
5  ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

8. Tabel Obat

NO	SQL
1	`CREATE TABLE IF NOT EXISTS `obat` (`
2	`  `kode_obat` varchar(11) NOT NULL,`
3	`  `nama_obat` varchar(50) NOT NULL,`
4	`  `exp_date` date NOT NULL,`
5	`  `id_petugas` int(11) NOT NULL,`
6	`  PRIMARY KEY (`kode_obat`),`
7	`  KEY `id_petugas` (`id_petugas`),CONSTRAINT`
8	`` `obat_ibfk_1` FOREIGN KEY (`id_petugas`) REFERENCES ``
9	`` `petugas` (`id_petugas`) ON DELETE RESTRICT ON UPDATE ``
10	`RESTRICT`
11	`) ENGINE=InnoDB DEFAULT CHARSET=latin1;`

9. Tabel Pemakaian

NO	SQL
1	`CREATE TABLE IF NOT EXISTS `pemakaian` (`
2	`  `id_pemakaian` int(11) NOT NULL,`
3	`  `tgl_pemakaian` date NOT NULL,`
4	`  `kd_puskesmas` int(11) NOT NULL,`
5	`  `id_petugas` int(11) NOT NULL,`
6	`  PRIMARY KEY (`id_pemakaian`),`
7	`  KEY `kd_puskesmas` (`kd_puskesmas`),`
8	`  KEY `id_petugas` (`id_petugas`),`
9	`  CONSTRAINT `pemakaian_ibfk_1` FOREIGN KEY`
10	`(`id_petugas`) REFERENCES `petugas` (`id_petugas`) ON`
11	`DELETE RESTRICT ON UPDATE RESTRICT,`
12	`  CONSTRAINT `pemakaian_ibfk_2` FOREIGN KEY`
13	`(`kd_puskesmas`) REFERENCES `puskesmas``
14	`(`kd_puskesmas`) ON DELETE RESTRICT ON UPDATE RESTRICT`
15	`) ENGINE=InnoDB DEFAULT CHARSET=latin1;`

10. Tabel Pengeluaran

NO	SQL
1	`CREATE TABLE IF NOT EXISTS `pengeluaran` (`
2	`  `id_pengeluaran` int(11) NOT NULL,`
3	`  `tanggal_pengeluaran` date NOT NULL,`
4	`  `status` enum('waiting','ok') NOT NULL,`
5	`  `id_permintaan` int(11) NOT NULL,`
6	`  `id_petugas` int(11) NOT NULL,`
7	`  PRIMARY KEY (`id_pengeluaran`),`
8	`  KEY `id_permintaan` (`id_permintaan`),`
9	`  KEY `id_petugas` (`id_petugas`),`
10	`  CONSTRAINT `pengeluaran_ibfk_1` FOREIGN KEY`
11	`(`id_permintaan`) REFERENCES `permintaan``
12	`(`id_permintaan`) ON DELETE RESTRICT ON UPDATE`
13	`RESTRICT,`

```
  CONSTRAINT `pengeluaran_ibfk_2` FOREIGN KEY
(`id_petugas`) REFERENCES `petugas` (`id_petugas`) ON
DELETE RESTRICT ON UPDATE RESTRICT
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

11. Tabel permintaan

NO	SQL

```
CREATE TABLE IF NOT EXISTS `permintaan` (
  `id_permintaan` int(11) NOT NULL,
  `tanggal_permintaan` date NOT NULL,
  `status` enum('idle','checked') NOT NULL,
  `kd_puskesmas` int(11) NOT NULL,
  `id_petugas` int(11) NOT NULL,
  PRIMARY KEY (`id_permintaan`),
  UNIQUE KEY `id_permintaan` (`id_permintaan`),
  KEY `kd_puskesmas` (`kd_puskesmas`),
  KEY `id_petugas` (`id_petugas`),
  CONSTRAINT `permintaan_ibfk_1` FOREIGN KEY
(`kd_puskesmas`) REFERENCES `puskesmas`
(`kd_puskesmas`) ON DELETE RESTRICT ON UPDATE RESTRICT,
  ADD CONSTRAINT `permintaan_ibfk_2` FOREIGN KEY
(`id_petugas`) REFERENCES `petugas` (`id_petugas`) ON
DELETE RESTRICT ON UPDATE RESTRICT
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

12. Tabel Petugas

NO	SQL

```
CREATE TABLE IF NOT EXISTS `petugas` (
  `id_petugas` int(11) NOT NULL,
  `nama` varchar(30) NOT NULL,
  `username` varchar(30) NOT NULL,
  `password` varchar(150) NOT NULL,
  `role` enum('admin','gudang','puskesmas') NOT NULL,
  `tanggal_dibuat` datetime NOT NULL,
  `last_session` datetime DEFAULT NULL,
  `status` enum('on','off') NOT NULL DEFAULT 'off',
  `kd_puskesmas` int(11) DEFAULT NULL,
  PRIMARY KEY (`id_petugas`),
  KEY `kd_puskesmas` (`kd_puskesmas`),
  CONSTRAINT `petugas_ibfk_1` FOREIGN KEY
(`kd_puskesmas`) REFERENCES `puskesmas`
(`kd_puskesmas`) ON DELETE RESTRICT ON UPDATE RESTRICT
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

13. Tabel puskesmas

NO	SQL
1	`CREATE TABLE IF NOT EXISTS `puskesmas` (`
2	`  `kd_puskesmas` int(11) NOT NULL,`
3	`  `nama_puskesmas` varchar(30) NOT NULL,`
4	`  `alamat` text NOT NULL,`
5	`  PRIMARY KEY (`kd_puskesmas`)`
6	`) ENGINE=InnoDB DEFAULT CHARSET=latin1;`

4.1.5. Implementasi Antarmuka

Implementasi antarmuka merupakan tampilan dari aplikasi yang dibangun. Aplikasi pengolahan data proses memilki beberapa bagian tampilan menu dapat dilihat pada tabel 4.3 berikut:

Tabel 4. 3 Implementasi Antarmuka Administrator

Menu	Deskripsi	Nama File
Login	Digunakan sebagai halaman login admin	index.php
Beranda	Halaman yang digunakan untuk beranda admin	securepage.php
Pengolahan Data Petugas	Halaman yang digunakan untuk mengolah data petugas, seperti tambah,ubah dan hapus	petugas.php
Pengolahan Data Puskesmas	Halaman yang digunakan untuk mengolah data puskesmas ,seperti tambah,ubah dan hapus	puskesmas.php
Data Log	Halaman yang digunakan untuk menmpilkan data log	log.php
Keluar		logout.php

Tabel 4. 4 Implementasi Antarmuka (Petugas Gudang)

Menu	Deskripsi	Nama File
Login	Digunakan sebagai halaman login petugas	index.php
Beranda	Halaman yang digunakan untuk beranda petugas	index.php
Pengolahan Data Obat	Halaman yang digunakan	obat.php

	untuk mengolah data obat, seperti tambah, ubah, dan cari	
Pengolahan Data Kemasan	Halaman yang digunakan untuk mengolah data kemasan, seperti tambah, ubah dan hapus	jenis.php
Pengolahan Data satuan	Halaman yang digunakan untuk mengolah data satuan, seperti tambah, ubah dan hapus	Satuan.php
Pengolahan Data kartu_stok	Halaman yang digunakan untuk mengolah data kartu_stok , seperti tambah, ubah dan pencarian	kartu_stok.php
Pengolahan data detail kartu_stok	Halaman yang digunakan untuk mengolah data detail kartu_stok , seperti tambah, ubah	detail_kartu.php
Pengeluaran Obat	Halaman yang digunakan untuk mengolah data pengeluaran, seperti tambah,ubah, dan setujui	pengeluaran.php
Pengolahan Detail Pengeluaran	Halaman yang digunakan untuk mengolah data detail pengeluaran yang berisikan ubah data detail	detail_pengeluaran.php
Pengolahan data permintaan	Halaman yang digunakan untuk mengolah data permintaan, seperti menambah data pengeluaran melalui data permintaan	request.php
Pengolahan Ramalan	Halaman yang digunakan untuk mengolah data pemintaan untuk memprediksi stok barang	ramalan.php
Keluar		logout.php

Tabel 4. 5 Implementasi Antarmuka (Petugas Puskesmas)

Menu	Deskripsi	Nama File
Login	Digunakan sebagai halaman login petugas	index.php

Menu	Deskripsi	Nama File
Beranda	Halaman yang digunakan untuk beranda petugas	index.php
Stok Obat	Halaman yang menampilkan data stok obat	stok_obat.php
Pengolahan Data Pemakaian	Halaman yang digunakan untuk mengolah data pemakaian , seperti tambah, ubah	pemakaian.php
Pengolahan data detail pemakaian	Halaman yang digunakan untuk mengolah data detail pemakaian , seperti tambah, ubah, dan hapus	Detail_pemakaian.php
Pengolahan permintaan	Halaman yang digunakan untuk mengolah data permintaan tambah, ubah, hapus	request.php
Pengolahan Detail Permintaan	Halaman yang digunakan untuk mengolah data detail permintaan, seperti tambah, ubah, dan hapus	detail_permintaan.php
Pengolahan Laporan	Halaman yang digunakan untuk mengolah data laporan menjadi bentuk pdf, laporan tersebut terdiri dari laporan StockOpname, dan laporan pemakaian dan permintaan	StockOp.php LPLPO.php
Keluar		logout.php

4.2. Pengujian Sistem

Pengujian sistem merupakan hal terpenting yang bertujuan untuk menemukan kesalahan-kesalahan atau kekurangan-kekurangan pada sistem informasi yang diuji. Pengujian bermaksud untuk mengetahui sistem informasi yang dibuat telah memenuhi kinerja sesuai dengan tujuan perancangan.

Pengujian yang dipergunakan untuk menguji sistem yang baru adalah metode pengujian *black box*. Pengujian *black box* terfokus pada pengujian persyaratan fungsional sistem informasi.

Rencana pengujian yang akan dilakukan adalah dengan cara menguji sistem yang dibangun secara *alpha* dan beta. Pengujian sistem informasi distribusi obat di Dinas Kesehatan Kabupaten Bandung Barat menggunakan data uji berdasarkan data yang diberikan dari beberapa data yang diberikan petugas gudang farmasi.

4.2.1. Pengujian *Black Box*

Pengujian *black box* dilakukan dengan tujuan untuk menguji fungsi-fungsi khusus dari perangkat lunak yang dirancang sehingga dapat diketahui apakah data yang dimasukan sudah sesuai dengan apa yang diharapkan atau belum. Teknik pengujian black box yang digunakan adalah *Equivalence partitioning* dan *Cause-Effect Relationship*.

4.2.1.1. Skenario Pengujian *Black Box*

Skenario pengujian *Black box* di dinas kesehatan kabupaten bandung barat terdiri dari 3 pelaku sistem, yaitu Administrator (Kepala Bagian Informasi), Gudang Farmasi, dan Puskesmas. Berikut skenario pengujian aplikasi administrator dapat dilihat pada table 4.6.

Tabel 4. 6 Skenario Pengujian *Admin*

Kelas Uji	Butir Uji	Jenis Pengujian
Login	*username*	*Black Box*
	password	*Black Box*
Data Petugas	Nama	*Black Box*
	username	*Black Box*
	password	*Black Box*
Data Puskesmas	nama	*Black Box*
	alamat	*Black Box*

Tabel 4. 7 Skenario Pengujian Petugas Gudang

Kelas Uji	Butir Uji	Jenis Pengujian
Login	*username*	*Black Box*
	password	*Black Box*
	alamat	*Black Box*
Data Obat	Nama	*Black Box*
	Expire date	*Black Box*
Data Satuan	nama	*Black Box*
Data Kemasan	kemasan	*Black Box*
	satuan	*Black Box*
Data Kartu_stok	Nama obat	*Black Box*
	Kemasan	*Black Box*
Data detail_kartu_stok	Tanggal	*Black Box*
	masuk	*Black Box*
	Keterangan	*Black Box*
Data Pengeluaran	Tanggal	*Black Box*
Data Detail Pengeluaran	Nama obat	*Black Box*
	pengeluaran	*Black Box*

Tabel 4. 8 Skenario Pengujian Petugas Puskesmas

Kelas Uji	Butir Uji	Jenis Pengujian
Login	*username*	*Black Box*
	password	*Black Box*
Data Pemakaian	Tanggal	*Black Box*
Data Detail Pemakaian	Nama Obat	*Black Box*
	Kemasan	*Black Box*
	Jumlah pemakaian	*Black Box*
Data Detail Pengeluaran	Nama obat	*Black Box*
	pengeluaran	*Black Box*

Kelas Uji	Butir Uji	Jenis Pengujian
Data Permintaan	tanggal	*Black Box*
Data detail Pemintaan	Nama obat	*Black box*
	Kemasan	*Black box*
	Jumlah permintaan	*Black box*

4.2.1.2. Kasus dan Hasil Pengujian *Black Box*

Kasus dan hasil pengujian berisi pemaparan dari rencana pengujian yang telah disusun pada skenario pengujian. Teknik pengujian *black box* yang digunakan adalah *Equivalence partitioning* dan *Cause-Effect Relationship*. Berikut ini pemaparan dari setiap butir pengujian yang terdapat pada skenario pengujian dengan menggunakan *Equivalence partitioning* :

4.2.1.2.1. *Equivalence Partitioning Testing*

Teknik pengujian *Equivalence Partitioning* adalah teknik pengujian perangkat lunak yang dilakukan dengan mencoba beberapa kemungkinan nilai masukan sebagai kelas-kelas uji untuk mengetahui kesalahan apa saja yang terdapat pada sistem.

1. *Equivalence Partitioning Testing* Admin

a. Login

Tabel 4.9 merupakan skenario pengujian menggunakan teknik pengujian *Equivalence Partitioning Testing* yang dilakukan pada bagian login admin.

Tabel 4. 9 *Equivalence Partitioning Testing* - Login Admin

Kasus dan Hasil Uji (Data Valid)						
Butir Uji	**Masukan yang Valid**	**Masukan yang Tidak Valid**	**Data yang Dimasukkan**	**Keluaran yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Username	{text} [a-z, A-Z, 0-9, simbol]	-	admin	Dapat memasukkan username pada field username	Dapat memasukkan username pada field username	[✓] Diterima [ ] Ditolak
Password	{text} [a-z, A-Z, 0-9, simbol]	-	admindinkes	Dapat memasukkan password pada field password	Dapat memasukkan password pada field password	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Tidak Valid)						
Butir Uji	**Masukan yang Valid**	**Masukan yang Tidak Valid**	**Data yang Dimasukkan**	**Keluaran yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Username	{text} [a-z, A-Z, 0-9, simbol]	-	Null	Menampilkan pesan "*username dan password tidak terdaftar*"	Menampilkan pesan "*username dan password tidak terdaftar*"	[✓] Diterima [ ] Ditolak
Password	{text} [a-z, A-Z, 0-9, simbol]	-	Null	Menampilkan pesan "*username dan password tidak terdaftar*"	Menampilkan pesan "*username dan password tidak terdaftar*"	[✓] Diterima [ ] Ditolak

b. Pengolahan Data Petugas

Tabel 4.10 merupakan skenario pengujian menggunakan teknik pengujian *Equivalence Partitioning Testing* yang dilakukan pada bagian Pengolahan Data Petugas.

Tabel 4. 10 *Equivalence Partitioning Testing* - Data Petugas

Kasus dan Hasil Uji (Data Valid)						
Butir Uji	**Masukan yang Valid**	**Masukan yang Tidak Valid**	**Data yang Dimasukkan**	**Keluaran yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama	{text} [a-z, A-Z, 0-9]	-	dadang	Dapat memasukkan nama pada field nama	Dapat memasukkan nama pada field nama	[✓] Diterima [ ] Ditolak
Username	{text} [a-z, A-Z, 0-9, simbol]	-	dadangjuga	Dapat memasukkan username pada field username	Dapat memasukkan username pada field username	[✓] Diterima [ ] Ditolak
Password	{text} [a-z, A-Z, 0-9, simbol]	-	Dadang47	Dapat memasukkan password pada field password	Dapat memasukkan password pada field password	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Tidak Valid)						
Butir Uji	**Masukan yang Valid**	**Masukan yang Tidak Valid**	**Data yang Dimasukkan**	**Keluaran yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama	{text} [a-z, A-Z, 0-9]	-	Null	Dapat menampilkan pesan "*Belum Diisi*"	Dapat menampilkan pesan "*Belum Diisi*"	[✓] Diterima [ ] Ditolak
Username	{text} [a-z, A-Z, 0-9, simbol]	-	Null	Dapat menampilkan pesan "*Belum Diisi*"	Dapat menampilkan pesan "*Belum Diisi*"	[✓] Diterima [ ] Ditolak
Password	{text} [a-z, A-Z, 0-9, simbol]	-	Null	Dapat menampilkan pesan "*Belum Diisi*"	Dapat menampilkan pesan "*Belum Diisi*"	[✓] Diterima [ ] Ditolak

c. Data Puskesmas

Tabel 4.11 merupakan skenario pengujian menggunakan teknik pengujian *Equivalence Partitioning Testing* yang dilakukan pada bagian Pengolahan Data Puskesmas.

Tabel 4. 11 *Equivalence Partitioning Testing* - Data Petugas

Kasus dan Hasil Uji (Data Valid)						
Butir Uji	**Masukan yang Valid**	**Masukan yang Tidak Valid**	**Data yang Dimasukkan**	**Keluaran yang Diharapkan**	**Pengamatan**	**Kesimpulan**
nama	{text} [a-z, A-Z, 0-9]	-	cirata	Dapat memasukkan nama pada field nama	Dapat memasukkan nama pada field nama	[✓] Diterima [ ] Ditolak
Alamat	{text} [a-z, A-Z, 0-9, simbol]	-	Jln.cirata no.25	Dapat memasukkan alamat pada field alamat	Dapat memasukkan alamat pada field alamat	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Tidak Valid)						
Butir Uji	**Masukan yang Valid**	**Masukan yang Tidak Valid**	**Data yang Dimasukkan**	**Keluaran yang Diharapkan**	**Pengamatan**	**Kesimpulan**
nama	{text} [a-z, A-Z, 0-9]	-	Null	Dapat menampilkan pesan "*Belum Diisi*"	Dapat menampilkan pesan "*Belum Diisi*"	[✓] Diterima [ ] Ditolak
Alamat	{text} [a-z, A-Z, 0-9, simbol]	-	Null	Dapat menampilkan pesan "*Belum Diisi*"	Dapat menampilkan pesan "*Belum Diisi*"	[✓] Diterima [ ] Ditolak

2. *Equivalence Partitioning Testing* Petugas Gudang

a. Data Obat

Tabel 4.12 merupakan skenario pengujian menggunakan teknik pengujian *Equivalence Partitioning Testing* yang dilakukan pada bagian Pengolahan Data Obat.

Tabel 4. 12 *Equivalence Partitioning Testing* - Data obat

Kasus dan Hasil Uji (Data Valid)						
Butir Uji	**Masukan yang Valid**	**Masukan yang Tidak Valid**	**Data yang Dimasukkan**	**Keluaran yang Diharapkan**	**Pengamatan**	**Kesimpulan**
nama	{text} [a-z, A-Z, 0-9]	-	Abendazol	Dapat memasukkan nama pada field nama obat	Dapat memasukkan nama pada field nama obat	[✓] Diterima [ ] Ditolak
Expire date	{text} [a-z, A-Z, 0-9]	-	25-02-2014	Dapat memasukkan *Expire date* pada field *Expire date*	Dapat memasukkan *Expire date* pada field *Expire date*	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Tidak Valid)						
Butir Uji	**Masukan yang Valid**	**Masukan yang Tidak Valid**	**Data yang Dimasukkan**	**Keluaran yang Diharapkan**	**Pengamatan**	**Kesimpulan**
nama	{text} [a-z, A-Z, 0-9]	-	Null	Dapat menampilkan pesan "*Belum Diisi*"	Dapat menampilkan pesan "*Belum Diisi*"	[✓] Diterima [ ] Ditolak
Expire date	{text} [a-z, A-Z, 0-9]	-	Null	Dapat menampilkan pesan "*Belum Diisi*"	Dapat menampilkan pesan "*Belum Diisi*"	[✓] Diterima [ ] Ditolak

b. Data Satuan

Tabel 4.13 merupakan skenario pengujian menggunakan teknik pengujian *Equivalence Partitioning Testing* yang dilakukan pada bagian Pengolahan Data Data Satuan.

Tabel 4. 13 Equivalence Partitioning Testing - Data Satuan

Kasus dan Hasil Uji (Data Valid)						
Butir Uji	**Masukan yang Valid**	**Masukan yang Tidak Valid**	**Data yang Dimasukkan**	**Keluaran yang Diharapkan**	**Pengamatan**	**Kesimpulan**
nama	{text} [a-z, A-Z, 0-9]	-	Tablet	Dapat memasukkan nama pada field nama satuan	Dapat memasukkan nama pada field nama satuan	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Tidak Valid)						
Butir Uji	**Masukan yang Valid**	**Masukan yang Tidak Valid**	**Data yang Dimasukkan**	**Keluaran yang Diharapkan**	**Pengamatan**	**Kesimpulan**
nama	{text} [a-z, A-Z, 0-9]	-	Null	Dapat menampilkan pesan "*Belum Diisi*"	Dapat menampilkan pesan "*Belum Diisi*"	[✓] Diterima [ ] Ditolak

c. Data Kemasan

Tabel 4.14 merupakan skenario pengujian menggunakan teknik pengujian *Equivalence Partitioning Testing* yang dilakukan pada bagian Pengolahan Data Data Kemasan.

Tabel 4. 14 *Equivalence Partitioning Testing* - Data Kemasan

Kasus dan Hasil Uji (Data Valid)						
Butir Uji	**Masukan yang Valid**	**Masukan yang Tidak Valid**	**Data yang Dimasukkan**	**Keluaran yang Diharapkan**	**Pengamatan**	**Kesimpulan**
kemasan	{text} [a-z, A-Z, 0-9,simbol]	-	Strip @ 10	Dapat memasukkan Kemasan pada field kemasan	Dapat memasukkan Kemasan pada field kemasan	[✓] Diterima [ ] Ditolak
satuan	{text} [a-z, A-Z, 0-9]	-	Tablet	Dapat memasukkan satuan pada field satuan	Dapat memasukkan satuan pada field satuan	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Tidak Valid)						
Butir Uji	**Masukan yang Valid**	**Masukan yang Tidak Valid**	**Data yang Dimasukkan**	**Keluaran yang Diharapkan**	**Pengamatan**	**Kesimpulan**
kemasan	{text} [a-z, A-Z, 0-9,simbol]	-	Null	Dapat menampilkan pesan "*Belum Diisi*"	Dapat menampilkan pesan "*Belum Diisi*"	[✓] Diterima [ ] Ditolak
satuan	{text} [a-z, A-Z, 0-9]	-	Null	Dapat menampilkan pesan "*Belum Diisi*"	Dapat menampilkan pesan "*Belum Diisi*"	[✓] Diterima [ ] Ditolak

d. Data Kartu Stok

Tabel 4.15 merupakan skenario pengujian menggunakan teknik pengujian *Equivalence Partitioning Testing* yang dilakukan pada bagian Pengolahan Data Kartu Stok.

Tabel 4. 15 *Equivalence Partitioning Testing* - Data Kartu Stok

Kasus dan Hasil Uji (Data Valid)						
Butir Uji	**Masukan yang Valid**	**Masukan yang Tidak Valid**	**Data yang Dimasukkan**	**Keluaran yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama Obat	{text} [a-z, A-Z, 0-9]	-	Abendazol	Dapat memasukkan Nama Obat pada field Nama Obat	Dapat memasukkan Nama Obat pada field Nama Obat	[✓] Diterima [ ] Ditolak
Kemasan	{text} [a-z, A-Z, 0-9,simbol]	-	Botol	Dapat memasukkan Kemasan pada field Kemasan	Dapat memasukkan Kemasan pada field Kemasan	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Tidak Valid)						
Butir Uji	**Masukan yang Valid**	**Masukan yang Tidak Valid**	**Data yang Dimasukkan**	**Keluaran yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama Obat	{text} [a-z, A-Z, 0-9]	-	Null	Dapat menampilkan pesan "*Belum Diisi*"	Dapat menampilkan pesan "*Belum Diisi*"	[✓] Diterima [ ] Ditolak
Kemasan	{text} [a-z, A-Z, 0-9,simbol]	-	Null	Dapat menampilkan pesan "*Belum Diisi*"	Dapat menampilkan pesan "*Belum Diisi*"	[✓] Diterima [ ] Ditolak

e. Data Detail Kartu Stok

Tabel 4.16 merupakan skenario pengujian menggunakan teknik pengujian *Equivalence Partitioning Testing* yang dilakukan pada bagian Pengolahan Data Detail Kartu Stok.

Tabel 4. 16 *Equivalence Partitioning Testing* - Data Detail Kartu Stok

Kasus dan Hasil Uji (Data Valid)						
Butir Uji	**Masukan yang Valid**	**Masukan yang Tidak Valid**	**Data yang Dimasukkan**	**Keluaran yang Diharapkan**	**Pengamatan**	**Kesimpulan**
tanggal	{text} [a-z, A-Z, 0-9]	-	24-02-2014	Dapat memasukkan tanggal pada field tanggal	Dapat memasukkan tanggal pada field tanggal	[✓] Diterima [ ] Ditolak
Masuk	{numeric} [0-9]	{alfabet} [a-z, A-Z]	500000	Dapat memasukkan masuk pada field masuk	Dapat memasukkan masuk pada field masuk	[✓] Diterima [ ] Ditolak
keterangan	{text} [a-z, A-Z, 0-9,simbol]	-	Barang masuk dari supplier	Dapat memasukkan keterangan pada field keterangan	Dapat memasukkan keterangan pada field keterangan	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Tidak Valid)						
Butir Uji	**Masukan yang Valid**	**Masukan yang Tidak Valid**	**Data yang Dimasukkan**	**Keluaran yang Diharapkan**	**Pengamatan**	**Kesimpulan**
tanggal	{text} [a-z, A-Z, 0-9]	-	Null	Dapat menampilkan pesan "*Belum Diisi*"	Dapat menampilkan pesan "*Belum Diisi*"	[✓] Diterima [ ] Ditolak
Masuk	{numeric} [0-9]	{alfabet} [a-z, A-Z]	Null	Dapat menampilkan pesan "*Belum Diisi*"	Dapat menampilkan pesan "*Belum Diisi*"	[✓] Diterima [ ] Ditolak
keterangan	{text} [a-z, A-Z, 0-9,simbol]	-	Null	Dapat menampilkan pesan "*Belum Diisi*"	Dapat menampilkan pesan "*Belum Diisi*"	[✓] Diterima [ ] Ditolak

f. Data Pengeluaran

Tabel 4.17 merupakan skenario pengujian menggunakan teknik pengujian *Equivalence Partitioning Testing* yang dilakukan pada bagian Pengolahan Data Pengeluaran.

Tabel 4. 17 *Equivalence Partitioning Testing* - Data Pengeluaran

Kasus dan Hasil Uji (Data Valid)						
Butir Uji	**Masukan yang Valid**	**Masukan yang Tidak Valid**	**Data yang Dimasukkan**	**Keluaran yang Diharapkan**	**Pengamatan**	**Kesimpulan**
tanggal	{text} [a-z, A-Z, 0-9]	-	25-02-2014	Dapat memasukkan *tanggal* pada field *tanggal*	Dapat memasukkan *tanggal* pada field *tanggal*	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Tidak Valid)						
Butir Uji	**Masukan yang Valid**	**Masukan yang Tidak Valid**	**Data yang Dimasukkan**	**Keluaran yang Diharapkan**	**Pengamatan**	**Kesimpulan**
tanggal	{text} [a-z, A-Z, 0-9]	-	Null	Dapat menampilkan pesan "*Belum Diisi*"	Dapat menampilkan pesan "*Belum Diisi*"	[✓] Diterima [ ] Ditolak

g. Data Detail Pengeluaran

Tabel 4.18 merupakan skenario pengujian menggunakan teknik pengujian *Equivalence Partitioning Testing* yang dilakukan pada bagian Pengolahan Data Detail Pengeluaran.

Tabel 4. 18 *Equivalence Partitioning Testing* - Data Detail Pengeluaran

Kasus dan Hasil Uji (Data Valid)						
Butir Uji	**Masukan yang Valid**	**Masukan yang Tidak Valid**	**Data yang Dimasukkan**	**Keluaran yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama Obat	{text} [a-z, A-Z, 0-9]	-	Abendazol	Dapat memasukkan Nama obat pada field nama obat	Dapat memasukkan Nama obat pada field nama obat	[✓] Diterima [ ] Ditolak
Jumlah Pengeluaran	{numeric} [0-9]	{alfabet} [a-z, A-Z]	4500	Dapat memasukkan jumlah pengeluaran pada field jumlah pengeluaran	Dapat memasukkan jumlah pengeluaran pada field jumlah pengeluaran	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Tidak Valid)						
Butir Uji	**Masukan yang Valid**	**Masukan yang Tidak Valid**	**Data yang Dimasukkan**	**Keluaran yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama Obat	{text} [a-z, A-Z, 0-9]	-	Null	Dapat menampilkan pesan "*Belum Diisi*"	Dapat menampilkan pesan "*Belum Diisi*"	[✓] Diterima [ ] Ditolak
Jumlah Pengeluaran	{numeric} [0-9]	{alfabet} [a-z, A-Z]	Null	Dapat menampilkan pesan "*Belum Diisi*"	Dapat menampilkan pesan "*Belum Diisi*"	[✓] Diterima [ ] Ditolak

3. *Equivalence Partitioning Testing* Petugas Puskesmas

a. Data Permintaan

Tabel 4.19 merupakan skenario pengujian menggunakan teknik pengujian *Equivalence Partitioning Testing* yang dilakukan pada bagian Pengolahan Data Permintaan.

Tabel 4. 19 *Equivalence Partitioning Testing* - Data Permintaan

Kasus dan Hasil Uji (Data Valid)						
Butir Uji	**Masukan yang Valid**	**Masukan yang Tidak Valid**	**Data yang Dimasukkan**	**Keluaran yang Diharapkan**	**Pengamatan**	**Kesimpulan**
tanggal	{text} [a-z, A-Z, 0-9]	-	25-02-2014	Dapat memasukkan *tanggal* pada field *tanggal*	Dapat memasukkan *tanggal* pada field *tanggal*	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Tidak Valid)						
Butir Uji	**Masukan yang Valid**	**Masukan yang Tidak Valid**	**Data yang Dimasukkan**	**Keluaran yang Diharapkan**	**Pengamatan**	**Kesimpulan**
tanggal	{text} [a-z, A-Z, 0-9]	-	Null	Dapat menampilkan pesan "*Belum Diisi*"	Dapat menampilkan pesan "*Belum Diisi*"	[✓] Diterima [ ] Ditolak

b. Data Detail Permintaan

Tabel 4.18 merupakan skenario pengujian menggunakan teknik pengujian *Equivalence Partitioning Testing* yang dilakukan pada bagian Pengolahan Data Permintaan.

Tabel 4. 20 *Equivalence Partitioning Testing* - Data Detail Permintaan

Kasus dan Hasil Uji (Data Valid)						
Butir Uji	**Masukan yang Valid**	**Masukan yang Tidak Valid**	**Data yang Dimasukkan**	**Keluaran yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama Obat	{text} [a-z, A-Z, 0-9]	-	Abendazol	Dapat memasukkan Nama obat pada field nama obat	Dapat memasukkan Nama obat pada field nama obat	[✓] Diterima [ ] Ditolak
Jumlah Permintaan	{numeric} [0-9]	{alfabet} [a-z, A-Z]	4500	Dapat memasukkan jumlah permintaan pada field jumlah permintaan	Dapat memasukkan jumlah permintaan pada field jumlah permintaan	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Tidak Valid)						
Butir Uji	**Masukan yang Valid**	**Masukan yang Tidak Valid**	**Data yang Dimasukkan**	**Keluaran yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama Obat	{text} [a-z, A-Z, 0-9]	-	Null	Dapat menampilkan pesan "*Belum Diisi*"	Dapat menampilkan pesan "*Belum Diisi*"	[✓] Diterima [ ] Ditolak
Jumlah Permintaan	{numeric} [0-9]	{alfabet} [a-z, A-Z]	Null	Dapat menampilkan pesan "*Belum Diisi*"	Dapat menampilkan pesan "*Belum Diisi*"	[✓] Diterima [ ] Ditolak

c. Data Pemakaian

Tabel 4.21 merupakan skenario pengujian menggunakan teknik pengujian *Equivalence Partitioning Testing* yang dilakukan pada bagian Pengolahan Data Pemakaian.

Tabel 4. 21 *Equivalence Partitioning Testing* - Data Pemakaian

Kasus dan Hasil Uji (Data Valid)						
Butir Uji	**Masukan yang Valid**	**Masukan yang Tidak Valid**	**Data yang Dimasukkan**	**Keluaran yang Diharapkan**	**Pengamatan**	**Kesimpulan**
tanggal	{text} [a-z, A-Z, 0-9]	-	25-02-2014	Dapat memasukkan *tanggal* pada field *tanggal*	Dapat memasukkan *tanggal* pada field *tanggal*	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Tidak Valid)						
Butir Uji	**Masukan yang Valid**	**Masukan yang Tidak Valid**	**Data yang Dimasukkan**	**Keluaran yang Diharapkan**	**Pengamatan**	**Kesimpulan**
tanggal	{text} [a-z, A-Z, 0-9]	-	Null	Dapat menampilkan pesan "*Belum Diisi*"	Dapat menampilkan pesan "*Belum Diisi*"	[✓] Diterima [ ] Ditolak

d. Data Detail Permintaan

Tabel 4.22 merupakan skenario pengujian menggunakan teknik pengujian *Equivalence Partitioning Testing* yang dilakukan pada bagian Pengolahan Data Permintaan.

Tabel 4. 22 *Equivalence Partitioning Testing* - Data Detail Permintaan

Kasus dan Hasil Uji (Data Valid)						
Butir Uji	**Masukan yang Valid**	**Masukan yang Tidak Valid**	**Data yang Dimasukkan**	**Keluaran yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama Obat	{text} [a-z, A-Z, 0-9]	-	Abendazol	Dapat memasukkan Nama obat pada field nama obat	Dapat memasukkan Nama obat pada field nama obat	[✓] Diterima [ ] Ditolak
Jumlah Pemakaian	{numeric} [0-9]	{alfabet} [a-z, A-Z]	4500	Dapat memasukkan jumlah Pemakaian pada field jumlah pemakaian	Dapat memasukkan jumlah Pemakaian pada field jumlah pemakaian	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Tidak Valid)						
Butir Uji	**Masukan yang Valid**	**Masukan yang Tidak Valid**	**Data yang Dimasukkan**	**Keluaran yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama Obat	{text} [a-z, A-Z, 0-9]	-	Null	Dapat menampilkan pesan "*Belum Diisi*"	Dapat menampilkan pesan "*Belum Diisi*"	[✓] Diterima [ ] Ditolak
Jumlah Pemakaian	{numeric} [0-9]	{alfabet} [a-z, A-Z]	Null	Dapat menampilkan pesan "*Belum Diisi*"	Dapat menampilkan pesan "*Belum Diisi*"	[✓] Diterima [ ] Ditolak

4.2.1.2.2. *Cause-Effect Relationship Testing*

1. *Cause-Effect Relationship Testing* Admin

a. Login

Input states :

1. *Username* memenuhi kondisi yang telah ditentukan
2. *Password* memenuhi kondisi yang telah ditentukan

Output States :

A. Berhasil melakukan login dan dapat masuk ke halaman admin
B. Menampilkan pesan kesalahan dari *username* yang dimasukkan
C. Menampilkan pesan kesalahan dari *password* yang dimasukkan

Tabel 4. 23 *Cause-Efect Testing - Login Admin*

Test Case / ***Cause Effect***	1	2	3
1	1	1	0
2	1	0	1
A	1	0	0
B	0	1	0
C	0	0	1

b. Data Petugas

Input states :

1. Nama memenuhi kondisi yang telah ditentukan
2. *Username* memenuhi kondisi yang telah ditentukan
3. *Password* memenuhi kondisi yang telah ditentukan

Output States :

A. Data tersimpan ke *Database*
B. Menampilkan pesan kesalahan dari nama yang dimasukan
C. Menampilkan pesan kesalahan dari *username* yang dimasukkan
D. Menampilkan pesan kesalahan dari *password* yang dimasukkan

Tabel 4. 24 *Cause-Effect Testing* - Data Petugas

Test Case / ***Cause Effect***	1	2	3	4
1	1	1	1	0
2	1	1	0	1

3	1	0	1	1
A	1	0	0	0
B	0	0	0	1
C	0	0	1	0
D	0	1	0	0

c. Data Puskesmas

Input states :

1. Nama memenuhi kondisi yang telah ditentukan
2. Alamat memenuhi kondisi yang telah ditentukan

Output States :

A. Data tersimpan ke *Database*
B. Menampilkan pesan kesalahan dari nama yang dimasukan
C. Menampilkan pesan kesalahan dari alamat yang dimasukkan

Tabel 4. 25 *Cause-Effect Testing* - Data Puskesmas

Test Case / ***Cause Effect***	1	2	3
1	1	1	1
2	1	1	0
A	1	0	1
B	0	0	1
C	0	1	0

2. *Cause-Effect Relationship Testing* Petugas Gudang

a. Login

Input states :

1. *Username* memenuhi kondisi yang telah ditentukan
2. *Password* memenuhi kondisi yang telah ditentukan
3. *Role* memenuhi kondisi yang telah ditentukan

Output States :

A. Berhasil melakukan login dan dapat masuk ke halaman petugas gudang
B. Menampilkan pesan kesalahan dari *username* yang dimasukkan
C. Menampilkan pesan kesalahan dari *password* yang dimasukkan
D. Menampilkan pesan kesalahan dari *role* yang dipilih

Tabel 4. 26 *Cause-Effect Testing - Login* Petugas Gudang

Test Case / ***Cause Effect***	1	2	3	4
1	1	1	1	0
2	1	1	0	1
3	1	0	1	1
A	1	0	0	0
B	0	0	0	1
C	0	0	1	0
D	0	1	0	0

b. Data obat

Input states :

1. Nama Obat memenuhi kondisi yang telah ditentukan
2. *Expire Date* memenuhi kondisi yang telah ditentukan

Output States :

A. Berhasil melakukan login dan dapat masuk ke halaman admin

B. Menampilkan pesan kesalahan dari nama obat yang dimasukkan
C. Menampilkan pesan kesalahan dari *expire date* yang dimasukkan

Tabel 4. 27 *Cause-Effect Testing* – Data Obat

Test Case / ***Cause Effect***	1	2	3
1	1	1	0
2	1	0	1
A	1	0	0
B	0	1	0
C	0	0	1

c. Data Satuan

Input states :

1. Nama Satuan memenuhi kondisi yang telah ditentukan

Output States :

A. Data disimpan ke *database*
B. Menampilkan pesan kesalahan dari nama satuan yang dimasukkan

Tabel 4. 28 *Cause-Effect Testing - Satuan*

Test Case / ***Cause Effect***	1	2
1	1	0
A	1	0
B	0	1

d. Data Kemasan

Input states :

1. Nama kemasan memenuhi kondisi yang telah ditentukan
2. Nama satuan memenuhi kondisi yang telah ditentukan

Output States :

A. Data disimpan ke *database*
B. Menampilkan pesan kesalahan dari nama kemasan yang dimasukkan
C. Menampilkan pesan kesalahan dari nama satuan yang dipilih

Tabel 4. 29 *Cause-Effect Testing* – Data Kemasan

Test Case / ***Cause Effect***	1	2	3
1	1	1	0
2	1	0	1
A	1	0	0
B	0	1	0
C	0	0	1

e. Data Kartu Stok

Input states :

1. Nama obat memenuhi kondisi yang telah ditentukan

2. Nama kemasan memenuhi kondisi yang telah ditentukan

Output States :

A. Data disimpan ke *database*
B. Menampilkan pesan kesalahan dari nama obat yang dimasukkan
C. Menampilkan pesan kesalahan dari nama kemasan yang dipilih

Tabel 4. 30 *Cause-Effect Testing* – Data Kartu Stok

Test Case / ***Cause Effect***	1	2	3
1	1	1	0
2	1	0	1
A	1	0	0
B	0	1	0
C	0	0	1

f. Data Detail Kartu Stok

Input states :

1. Tanggal memenuhi kondisi yang telah ditentukan
2. Jumlah barang masuk memenuhi kondisi yang telah ditentukan
3. keterangan memenuhi kondisi yang telah ditentukan

Output States :

A. Data disimpan ke *database*
B. Menampilkan pesan kesalahan dari tanggal yang dimasukkan
C. Menampilkan pesan kesalahan dari jumlah barang masuk yang dimasukan
D. Menampilkan pesan kesalahan dari keterangan yang dimasukan

Tabel 4. 31 *Cause-Effect Testing* – Data Detail Kartu Stok

Test Case / *Cause Effect*	1	2	3	4
1	1	0	1	1
2	1	1	0	1
3	1	1	1	0
A	1	0	0	0
B	0	1	0	0
C	0	0	1	0
D	0	0	0	1

g. Data Pengeluaran

Input states :

1. Tanggal memenuhi kondisi yang telah ditentukan

Output States :

A. Data disimpan ke *database*

B. Menampilkan pesan kesalahan dari tanggal yang dimasukkan

Tabel 4. 32 *Cause-Effect Testing* – Data Pengeluaran

Test Case / *Cause Effect*	1	2
1	1	0
A	1	0
B	0	1

h. Data Detail Pengeluaran

Input states :

1. Nama Obat memenuhi kondisi yang telah ditentukan
2. Nama Kemasan memenuhi kondisi yang telah ditentukan
3. Jumlah pengeluaran memenuhi kondisi yang telah ditentukan

Output States :

A. Data disimpan ke *database*
B. Menampilkan pesan kesalahan dari nama obat yang dimasukkan
C. Menampilkan pesan kesalahan dari nama kemasan yang dimasukan
D. Menampilkan pesan kesalahan dari jumlah pengeluaran yang dimasukan

Tabel 4. 33 *Cause-Effect Testing* – Data Detail Pengeluaran

Test Case / ***Cause Effect***	1	2	3	4
1	1	1	1	0
2	1	1	0	1
3	1	0	1	1
A	1	0	0	0
B	0	0	0	1
C	0	0	1	0
D	0	1	0	0

3. *Cause-Effect Relationship Testing* Petugas Gudang

a. Login

Input states :

1. *Username* memenuhi kondisi yang telah ditentukan
2. *Password* memenuhi kondisi yang telah ditentukan
3. *Role* memenuhi kondisi yang telah ditentukan

Output States :

A. Berhasil melakukan login dan dapat masuk ke halaman petugas gudang
B. Menampilkan pesan kesalahan dari *username* yang dimasukkan
C. Menampilkan pesan kesalahan dari *password* yang dimasukkan
D. Menampilkan pesan kesalahan dari *role* yang dipilih

Tabel 4. 34 *Cause-Effect Testing - Login* Petugas Puskesmas

Test Case / ***Cause Effect***	1	2	3	4
1	1	1	1	0
2	1	1	0	1
3	1	0	1	1
A	1	0	0	0
B	0	0	0	1
C	0	0	1	0
D	0	1	0	0

b. Data pemakaian

Input states :

1. tanggal memenuhi kondisi yang telah ditentukan

Output States :

A. Data disimpan ke *database*
B. Menampilkan pesan kesalahan dari *tanggal* yang dimasukkan

Tabel 4. 35 *Cause-Effect Testing* – Data Pemakaian

Test Case / ***Cause Effect***	1	2
1	1	0
A	1	0
B	0	1

c. Data Detail Pemakaian

Input states :

1. Nama obat memenuhi kondisi yang telah ditentukan
2. Jumlah pemakaian memenuhi kondisi yang telah ditentukan

Output States :

A. Data disimpan ke *database*
B. Menampilkan pesan kesalahan dari nama obat yang dimasukkan
C. Menampilkan pesan kesalahan dari jumlah pemakaian yang dimasukkan

Tabel 4. 36 *Cause-Effect Testing* – Data Detail Pemakaian

Test Case / ***Cause Effect***	1	2	3
1	1	1	0
2	1	0	1

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

d. Data Permintaan

Input states :

1. tanggal memenuhi kondisi yang telah ditentukan

Output States :

A. Data disimpan ke *database*
B. Menampilkan pesan kesalahan dari tanggal yang dimasukkan

Tabel 4. 37 *Cause-Effect Testing* – Data Detail Pemakaian

Test Case / ***Cause Effect***	1	2
1	1	0
A	1	0
B	0	1

e. Data Detail Permintaan

Input states :

1. Nama obat memenuhi kondisi yang telah ditentukan
2. Jumlah permintaan memenuhi kondisi yang telah ditentukan

Output States :

A. Data disimpan ke *database*
B. Menampilkan pesan kesalahan dari nama obat yang dimasukkan

C. Menampilkan pesan kesalahan dari jumlah permintaan yang dimasukkan

Tabel 4. 38 *Cause-Effect Testing* – Data Detail Permintaan

Test Case / *Cause Effect*	1	2	3
1	1	1	0
2	1	0	1
A	1	0	0
B	0	1	0
C	0	0	1

4.2.1.3. Kesimpulan Hasil Pengujian *Black Box*

Berdasarkan hasil pengujian aplikasi yang dilakukan maka dapat disimpulkan bahwa Sistem Informasi yang dibangun sudah berjalan sesuai dengan yang diharapkan baik itu dari segi validasi maupun proses penanganan kesalahan.

4.2.2. Pengujian Beta

Pengujian beta merupakan pengujian yang bersifat langsung di lingkungan yang sebenarnya. Pengguna melakukan penilaian terhadap perangkat lunak dengan menggunakan metode wawancara. Dari hasil wawancara tersebut maka dapat ditarik kesimpulan apakah perangkat lunak yang dibangun telah sesuai dengan tujuan atau tidak.

Wawancara dilakukan terhadap pengujian *beta* untuk halaman setiap pengguna yang terdiri dari Admin, Petugas Farmasi. Wawancara mengenai halaman Admin dilakukan terhadap kepala seksi informasi dan penelitian kesehatan , wawancara mengenai halaman petugas Farmasi dilakukan terhadap Petugas pengelola obat yang berada di gudang.

Pengujian *beta* ini dilakukan dengan cara melakukan pengujian langsung di tempat penelitian. Penelitian ini dilakukan di Dinas Kesehatan Kabupaten

Bandung Barat, agar dapat mengetahui sejauh mana sistem yang dibangun dapat menjadi solusi dari penyelesaian permasalahan yang telah dijelaskan sebelumnya.

4.2.2.1. Hasil Pengujian Beta

Wawancara dilakukan dengan menggunakan teknik kualitatif melalui pertanyan terbuka. Wawancara mengenai backend dilakukan terhadap 2 orang pegawai Dinas Kesehatan Kabupaten Bandung Barat. Dari hasil wawancara tersebut akan dilakukan pengambilan kesimpulan terhadap penilaian penerapan sistem yang baru. Wawancara terdiri dari 5 pertanyaan terbuka yang akan ditanyakan kepada narasumber.

Wawancara dilakukan untuk pengujian *beta* terhadap *website* Sistem Informasi Distribusi Obat di Dinas Kesehatan Kabupaten Bandung Barat. Adapun daftar narasumber dan waktu pelaksanaan wawancara dipaparkan didalam tabel 4.30.

Tabel 4. 39 Daftar Narasumber

Nama Narasumber	Jabatan	Waktu	Tempat
Sugiartono , SKM, M.Kes	Kepala Seksi Informasi dan Penelitian Kesehatan	Pagi	Dinas Kesehatan Kabupaten Bandung Barat
Rizky	Petugas Pengelola Obat (Gudang Farmasi)	Siang	Gudang Farmasi

Berikut daftar pertanyaan yang diajukan kepada narasumber :

1. Apakah dengan adanya *web* Sistem Informasi Distribusi Obat ini memudahkan Dinas Kesehatan dalam memperoleh informasi stok obat ? Alasannya ?
2. Apakah dengan adanya web sistem informasi Distribusi Obat ini memudahkan anda dalam memperoleh informasi permintaan dan pengeluaran obat ? Alasannya ?

3. Apakah dengan adanya *web* Sistem Informasi Distribusi Obat ini memudahkan anda dalam membuat laporan bulanan atau laporan *stock opname* ? Alasannya?
4. Apakah *web* Sistem Informasi Distribusi Obat ini mudah digunakan ? Alasannya ?
5. Apakah fitur *web* Sistem Informasi Distribusi Obat ini sudah sesuai dengan kebutuhan anda dalam pekerjaan? Alasannya ?

Adapun rincian hasil dari wawancara terhadap narasumber untuk pertanyaan pertama terdapat pada tabel 4.31 hingga tabel 4.35.

Tabel 4. 40 Hasil Wawancara Pertanyaan ke-1

Apakah dengan adanya *web* Sistem Informasi Distribusi Obat ini memudahkan Dinas Kesehatan dalam memperoleh informasi stok obat ? Alasannya ?	
Narasumber 1	Dengan adanya *web* Sistem Informasi Distribusi Obat ini memudahkan dalam memperoleh data stok seluruh obat setiap saat.
Narasumber 2	Memudahkan dalam pencarian data obat.

Tabel 4. 41 Hasil Wawancara Pertanyaan ke-2

Apakah dengan adanya web sistem informasi Distribusi Obat ini memudahkan anda dalam memperoleh informasi permintaan dan pengeluaran obat ? Alasannya ?	
Narasumber 1	Sangat memudahkan untuk memperolah informasi permintaan dan pengeluaran obat karena datanya sudah tersedia di *web* dan prosesnya cepat.
Narasumber 2	Tersedianya *web* Sistem Informasi Distribusi Obat sangat

	memudahkan karena data yang tersedia lengkap mulai dari data puskesmas, permintaan obat, data obat beserta jumlahnya.

Tabel 4. 42 Hasil Wawancara Pertanyaan ke-3

Apakah dengan adanya *web* Sistem Informasi Distribusi Obat ini memudahkan anda dalam membuat laporan bulanan atau laporan *stock opname* ? Alasannya?	
Narasumber 1	Memudahkan dalam pembuatan laporan karena hanya memilih tanggal dan klik pdf sudah menghasilkan data yang diperlukan.
Narasumber 2	Sangat mudah dan membantu dalam membuat laporan yang dibutuhkan dan proses pembuatan laporan lebih cepat.

Tabel 4. 43 Hasil Wawancara Pertanyaan ke-5

Apakah *web* Sistem Informasi Distribusi Obat ini mudah digunakan ? Alasannya ?	
Narasumber 1	Sangat mudah dan implementasinya tidak menyulitkan.
Narasumber 2	Menu yang tersedia lengkap, jelas dan mudah dipahami.

Tabel 4. 44 Hasil Wawancara Pertanyaan ke-5

Apakah fitur *web* Sistem Informasi Distribusi Obat ini sudah sesuai dengan kebutuhan anda dalam pekerjaan? Alasannya ?	
Narasumber 1	*Web* Sistem Informasi Distribusi Obat sudah sesuai dengan kebutuhan Dinas Kesehatan sebagai alat kontrol internal kegiatan pengolahan data obat melalui log.
Narasumber 2	*Web* Sistem Informasi distribusi Obat dapat membantu dan memudahkan dalam pendataan barang masuk dan keluar disertai pembuatan laporan stock barang setiap saat.

4.2.2.2. Kesimpulan Hasil Pengujian Beta

Setelah dilakukannya kegiatan wawancara terhadap 2 orang pegawai Dinas Kesehatan Kabupaten Bandung Barat, dapat diambil kesimpulan bahwa penerapan sistem informasi distribusi obat ini sudah sesuai dengan tujuan yaitu:

1. Sistem Informasi Distribusi Obat di Dinas Kesehatan Kabupaten Bandung Barat ini dapat berjalan sesuai dengan tujuannya, yaitu membantu memudahkan Dinas Kesehatan Kabupaten Bandung Barat dalam pengolahan data obat
2. Dengan adanya sistem informasi distribusi obat ini,memudahkan Dinas Kesehatan Kabupaten Bandung Barat dalam pembuatan Laporan.
3. Petugas Gudang dimudahkan dalam pendataan barang masuk dan keluar.
4. Memberikan informasi secara cepat dan lengkap mengenai data obat-obatan.