BAB IV
IMPLEMENTASI DAN PENGUJIAN SISTEM

IV.1. Implementasi Sistem

Tahap implementasi merupakan tahap pembangunan perangkat lunak, tahap kelanjutan dari kegiatan perancangan sistem.Tahap ini merupakan tahap dimanasistem siap untuk dioperasikan, yang terdiri dari penjelasan mengenai lingkungan implementasi, dan implementasi program.

IV.1.1. Impelementasi Perangkat Keras yang Digunakan

Spesifikasi perangkat keras yang digunakan dalam pembangunan aplikasi yang dibangun dapat dilihat pada tabel IV.1.

Tabel IV.1 Perangkat Keras yang Digunakan

No.	Perangkat Keras	Spesifikasi
1	Processor	*Speed* 2,5 GHz
2	RAM	2GB
3	Harddisk	500GB
4	Monitor	Resolusi 1024x768
5	VGA	256 Mb

IV.1.2. Implementasi Perangkat Lunak yang Digunakan

Spesifikasi perangkat lunak yang digunakan dalam aplikasi yang dibangun dapat dilihat pada tabel IV.2.

Tabel IV.2 Perangkat Lunak yang Digunakan

No.	Perangkat Lunak	Spesifikasi
1	Sistem Operasi	Windows 7 Ultimate
2	Bahasa Pemrograman	PHP
3	*DBMS*	MySQL 5.5.16
4	*Web Browser*	Mozilla Firefox dan Google Chorme
5	*Code Editor*	Adobe Dreamweaver CS 3
6	*Web Server*	WAMP 2.1

IV.2. Implementasi *Database*

Pembuatan *database* dilakukan dengan menggunakan aplikasi *DBMSMySQL*. Implementasi *database* dalam bahasa *SQL* adalah sebagai berikut.

1. Tabel Petugas

```
CREATE TABEL IF NOT EXISTS `petugas` (
`nip` varchar(20) NOT NULL,
`nama_petugas` varchar(45) DEFAULT NULL,
`alamat_petugas` varchar(45) DEFAULT NULL,
`jabatan` varchar(45) DEFAULT NULL,
`user` varchar(25) NOT NULL,
`password` varchar(45) DEFAULT NULL,
`id_kantor` int(11) DEFAULT NULL,
PRIMARY KEY (`nip`),
FOREIGN KEY (`id_kantor`) REFERENCES `kantor`
(`id_kantor`) ON DELETE CASCADE ON UPDATE CASCADE)
ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

2. Tabel Kantor

```
CREATE TABEL IF NOT EXISTS `kantor`(
`id_kantor` int(11) NOT NULL AUTO_INCREMENT,
`nama_kantor` varchar(45) DEFAULT NULL,
`alamat_kantor` varchar(45) DEFAULT NULL,
`telp` varchar(45) DEFAULT NULL,
`kelurahan` varchar(45) DEFAULT NULL,
`kecamatan` varchar(45) DEFAULT NULL,
`kota` varchar(45) DEFAULT NULL,
`id_pusat` int(11) DEFAULT NULL,
PRIMARY KEY (`id_kantor`),
KEY `fk_kantor_kantor1` (`id_pusat`))
ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
```

3. Tabel Obat

```
CREATE TABEL IF NOT EXISTS `obat`(
`id_obat` int(11) NOT NULL AUTO_INCREMENT,
`id_jenis` int(11) NOT NULL,
`nama_obat` varchar(45) DEFAULT NULL,
`satuan` varchar(6) DEFAULT NULL,
`kemasan` varchar(45) NOT NULL,
`harga` decimal(10,2) DEFAULT NULL,
PRIMARY KEY (`id_obat`),
KEY `fk_obat_jenis_obat1` (`id_jenis`)
ALTER TABEL `obat`
ADD CONSTRAINT `obat_ibfk_1` FOREIGN KEY (`id_jenis`)
REFERENCES `obat` (`id_obat`) ON DELETE CASCADE ON UPDATE
CASCADE)
ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
```

4. Tabel Detail_Obat

```
CREATE TABEL IF NOT EXISTS `detail_obat`(
`id_kantor` int(11) NOT NULL,
`id_obat`
int(11) NOT NULL,
`stok` int(11) DEFAULT NULL,
KEY `fk_detail_obat_tabel1`(`id_kantor`),
KEY `fk_detail_obat_obat1`(`id_obat`)
ALTER TABEL `detail_obat`
ADD CONSTRAINT `detail_obat_ibfk_1`
FOREIGN KEY (`id_kantor`) REFERENCES `kantor`
(`id_kantor`)
ON DELETE CASCADE ON UPDATE CASCADE)
ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5. Tabel Pemakaian

```
CREATE TABEL IF NOT EXISTS `pemakaian`(
`id_pemakaian` int(11) NOT NULL AUTO_INCREMENT,
`id_kantor` int(11) NOT NULL,
`tgl_pemakaian` date DEFAULT NULL,
PRIMARY KEY (`id_pemakaian`),
KEY `fk_pemakaian_kantor1` (`id_kantor`)
ALTER TABEL `pemakaian`
ADD CONSTRAINT `pemakaian_ibfk_1` FOREIGN KEY
(`id_kantor`)
REFERENCES `kantor` (`id_kantor`)
ON DELETE CASCADE ON UPDATE CASCADE)
ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=1;
```

6. Tabel Detail_Pemakaian

```
CREATE TABEL IF NOT EXISTS `detail_pemakaian` (
`id_pemakaian` int(11) NOT NULL,
`id_obat` int(11) NOT NULL,
`jumlah` int(11) DEFAULT NULL,
KEY `fk_detail_pemakaian_obat1` (`id_obat`),
KEY `fk_detail_pemakaian_pemakaian1` (`id_pemakaian`))
ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

7. Tabel Permintaan

```
CREATE TABEL IF NOT EXISTS `permintaan` (
`id_permintaan` int(11) NOT NULL AUTO_INCREMENT,
`id_kantor` int(11) NOT NULL,
`tgl_permintaan` date DEFAULT NULL,
`status` int(2) NOT NULL,
PRIMARY KEY (`id_permintaan`),
FOREIGN KEY (`id_kantor`) REFERENCES
`detail_obat` (`id_kantor`) ON DELETE CASCADE ON UPDATE
CASCADE)
ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=1;
```

8. Tabel Detail_Permintaan

```
CREATE TABEL IF NOT EXISTS `detail_permintaan` (
`id_permintaan` int(11) NOT NULL,
`id_obat` int(11) NOT NULL,
`jumlah` int(11) DEFAULT NULL,
KEY `fk_detail_permintaan_permintaan` (`id_permintaan`),
KEY `fk_detail_permintaan_obat1` (`id_obat`))
ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

9. Tabel Pengiriman

```
CREATE TABEL IF NOT EXISTS `pengiriman` (
`id_pengiriman` int(11) NOT NULL AUTO_INCREMENT,
`id_kantor` int(11) NOT NULL,
`tgl_pengiriman` date NOT NULL,
`status` int(2) NOT NULL,
PRIMARY KEY (`id_pengiriman`),
FOREIGN KEY (`id_kantor`) REFERENCES`kantor`(`id_kantor`)
ON DELETE CASCADE ON UPDATE CASCADE)
ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
```

10. Tabel Detail_Pengiriman

```
CREATE TABEL IF NOT EXISTS `detail_pengiriman`(
`id_pengiriman` int(11) NOT NULL,
`id_obat` int(11) NOT NULL,
`jumlah` int(11) NOT NULL,
FOREIGN KEY (`id_pengiriman`) REFERENCES `pengiriman`
(`id_pengiriman`)ON DELETE CASCADE ON UPDATE CASCADE,
FOREIGN KEY (`id_obat`) REFERENCES `obat`(`id_obat`)
ON DELETE CASCADE ON UPDATE CASCADE)
ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

11. Tabel Penerimaan

```
CREATE TABEL IF NOT EXISTS `penerimaan` (
`id_penerimaan` int(11) NOT NULL AUTO_INCREMENT,
`id_kantor` int(11) NOT NULL,
`tgl_penerimaan` date NOT NULL,
PRIMARY KEY (`id_penerimaan`),
FOREIGN KEY (`id_kantor`) REFERENCES `kantor` (`id_kantor`)
ON DELETE CASCADE ON UPDATE CASCADE)
ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
```

12. Tabel Detail_Penerimaan

```
CREATE TABEL IF NOT EXISTS `detail_penerimaan` (
`id_penerimaan` int(11) NOT NULL,
`id_obat` int(11) NOT NULL,
`jumlah` varchar(45) DEFAULT NULL,
FOREIGN KEY (`id_penerimaan`) REFERENCES `penerimaan`
(`id_penerimaan`) ON DELETE CASCADE ON UPDATE CASCADE,
ADD CONSTRAINT `detail_penerimaan_ibfk_2`
FOREIGN KEY (`id_obat`) REFERENCES `obat`(`id_obat`)
ON DELETE CASCADE ON UPDATE CASCADE)
ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

13. Tabel Eoq

```
CREATE TABEL IF NOT EXISTS `eoq` (
`id_eoq` int(11) NOT NULL AUTO_INCREMENT,
`id_kantor` int(11) NOT NULL,
`tgl_eoq` date NOT NULL,
PRIMARY KEY (`id_eoq`))
ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=1;
```

14. Tabel Detail_Eoq

```
CREATE TABEL IF NOT EXISTS `detail_eoq` (
`id_eoq` int(11) NOT NULL,
`id_obat` int(11) NOT NULL,
`rata` decimal(10,0) NOT NULL,
`safety_stok` int(11) NOT NULL,
`jumlah` int(11) NOT NULL,
FOREIGN KEY (`id_eoq`) REFERENCES `eoq` (`id_eoq`)
ON DELETE CASCADE ON UPDATE CASCADE)
ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

15. Tabel Jenis Obat

```
CREATE TABEL IF NOT EXISTS `jenis_obat` (
`id_jenis` int(11) NOT NULL AUTO_INCREMENT,
`nama_jenis` varchar(45) DEFAULT NULL,
PRIMARY KEY (`id_jenis`))
ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
```

IV.3. Implementasi Antarmuka

Implementasi antar muka dilakukan dengan setiap tampilan program yang dibangun dan pengkodeannya dalam bentuk *file* program. Berikut ini adalah implementasi antarmuka untuk pengguna dapat dilihat pada Tabel IV.3.

Tabel IV.3 Implementasi Antarmuka Untuk Farmasi Dinkes

No.	Menu	Deskripsi	Nama File
1.	Beranda	Penyajian informasi data obat	index.php
2.	Tambah Data Petugas	Digunakan untuk menambah data petugas	inputpetugas.php
3.	Data Petguas	Penyajian informasi data petugas	datapetugas.php
4.	Tambah Data Obat	Digunakan untuk menambah data obat	inputobat.php
5.	Data Obat	Penyajian informasi data obat.	Dataobat.php
6.	Tambah Jenis Obat	Digunakan untuk menambah data jenis obat	inputobat.php
7.	Data Jenis Obat	Penyajian informasi data jenis obat	datajenisobat.php
8.	Tambah Data Puskesmas	Digunakan untuk menambah data puskesmas.	inputpuskesmas.php
9.	Data Puskemas	Penyajian Informasi data puskesmas	datapuskesmas.php
10.	Pemakaian Obat	Digunakan untuk menambah data pemakaian obat	pemakaian.php
11.	Data Permintaan Obat	Digunakan untuk melihat permintaan dari puskesmas	datapermintaan.php
12.	Pengiriman Obat	Digunakan untuk menambah data pengiriman obat	pengiriman.php
13.	Data Pemakaian Obat	Penyajian informasi data pemakaian obat	datapemakaian.php
14.	Data Pengiriman Obat	Penyajian informasi data pengiriman obat	datapengiriman.php
15.	Laporan Pemakaian	Penyajian informasi cetak laporan pemakaian	lappemakaian.php
16.	Laporan Permintaan	Penyajian informasi cetak laporan permintaan	lappermintaan.php
17.	Laporan Pengiriman	Penyajian informasi cetak laporan pengiriman	lappengiriman.php

Tabel IV.4 Implementasi Antarmuka Untuk Farmasi Puskesmas

No.	Menu	Deskripsi	Nama File
1.	Beranda	Penyajian informasi data obat	index.php
2.	Data Obat	Penyajian informasi data obat.	dataobat.php
3.	Pemakaian Obat	Digunakan untuk menambah data pemakaian obat	pemakaian.php
4.	Data Permintaan Obat	Digunakan untuk menghitung jumlah permintaan obat ke Dinkes	permintaan.php
5.	Penerimaan Obat	Digunakan untuk menambah data penerimaan obat	penerimaan.php
6.	Data Pemakaian Obat	Penyajian informasi data pemakaian obat	datapemakaian.php
7.	Data Permintaan Obat	Penyajian informasi data permintaan obat	datapermintaan.php
8.	Data Penerimaan Obat	Penyajian informasi data penerimaan obat	datapenerimaan.php
9.	Laporan Pemakaian	Penyajian informasi cetak laporan pemakaian	lappemakaian.php
10.	Laporan Permintaan	Penyajian informasi cetak laporan permintaan	lappermintaan.php
11.	Laporan Pengiriman	Penyajian informasi cetak laporan pengiriman	lappenerimaan.php

IV.4. Pengujian Perangkat Lunak

Pengujian sistem merupakan hal terpenting yang bertujuan untukmenemukan kesalahan–kesalahan atau kekurangan – kekurangan pada perangkat lunak yang diuji.Pengujian bermaksud untuk mengetahui perangkat lunak yang dibuat sudah memenuhi kriteria yang sesuai dengan tujuan perancangan perangkat lunak tersebut.

Pengujian perangkat lunak ini menggunakan pengujian *black box*.Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak.

IV.4.1. Skenario Pengujian *Black Box*

Pengujian *black box* menjelaskan skenario pengujian sistem dijelaskan pada tabel IV.5.

Tabel IV.5 Tabel Skenario Pengujian

Kasus Uji	Detail Uji	Jenis Pengujian
Login	Login pengguna	*Black Box*
Data Petugas	Penambahan Pengubahan	*Black Box*
Data Obat	Penambahan Pengubahan	*Black Box*
Data Jenis Obat	Penambahan Pengubahan	*Black Box*
Data Puskesmas	Penambahan Pengubahan	*Black Box*
Proses Pemakaian	Penambahan	*Black Box*
Proses Permintaan	Penambahan	*Black Box*
Proses Pengiriman	Penambahan	*Black Box*
Proses Penerimaan	Penambahan	*Black Box*
Data Proses Pemakaian	Tampilkan	*Black Box*
Data Proses Permintaan	Tampilkan	*Black Box*
Data Pengiriman	Tampilkan	*Black Box*
Laporan Pemakaian	Menampilkan form download	*Black Box*
Laporan Permintaan	Menampilkan form download	*Black Box*
Laporan Pengiriman	Menampilkan form download	*Black Box*
Laporan Penerimaan	Menampilkan form download	*Black Box*

IV.4.2. Kasus dan Hasil Pengujian *Black Box*

Pengujian dilakukan dengan menguji setiap proses untuk kemungkinan kesalahan yang terjadi.

1. Pengujian Login

Login digunakan untuk penentuan hak akses pengguna sebagai farmasi puskesmas atau farmasi dinas kesehatan. Dapat dilihat pengujian data benar login pada tabel IV.6.

Tabel IV.6. Pengujian Login Data Benar

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Username **Password** Contoh Masukan 1. Data login farmasi puskemas Username : puskes Password : puskes 2. Data login bidan	Mengisikan data login yang sudah terdaftar.	Data login benar dan akan masuk ke masing-masing *interface* yang telah ditentukan sesuai dengn data login.	[√]diterima [ ]ditolak

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
puskesmas Username : bidan Password : bidan 3. Data login farmasi dinas kesehatan Username : dinkes Password : dinkes			

Pengujian data salah login maka dapat dilihat pada tabel IV.7.

Tabel IV.7 Pengujian Login Data Salah

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Username **Password** Contoh Masukan 1. Data login farmasi puskesmas Username : puskes Password : (kosong) 2. Data login bidan puskesmas Username : bidan Password : (kosong) 3. Data login farmasi dinas kesehatan Username : dinkes Password : (kosong)	Menampilkan pesan "Maaf Login Gagal, Silahkan Masukkan Kembali Nip dan Password anda".	Dapat menampilkan pesan kesalahan ""Maaf Login Gagal, Silahkan Masukkan Kembali Nip dan Password anda".	[√]diterima [ ]ditolak
Username **Password** Contoh Masukan 1. Data login farmasi puskesmas Username : 123456 Password : puskes 2. Data login bidan puskesmas Username : 1234abc Password : bidan 3. Data login farmasi dinas kesehatan Username : abcd123 Password : dinkes	Menampilkan pesan "Maaf Login Gagal, Silahkan Masukkan Kembali Nip dan Password anda".	Dapat menampilkan pesan kesalahan "Maaf Login Gagal, Silahkan Masukkan Kembali Nip dan Password anda".	[√]diterima [ ]ditolak

2. Pengujian Penambahan Data Petugas

Pengujian penambahan data petugas dilakukan oleh farmasi dinas kesehatan pada tabel IV.8.

Tabel IV.8 Pengujian Penambahan Data Petugas Data Benar

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Tambah Petugas Contoh Masukan : NIP : 101086666 Nama : Ryan Alamat : Gunung Batu Jabatan : Petugas Kantor : Sukajadi User : Ryan Password : 4r3e2w1q	Mengisikan data petugas dan menyimpan data.	Data petugas dapat tersimpan.	[√]diterima [ ]ditolak

Pengujian data salah penambahan data petugas dapat dilihat pada tabel IV.9.

Tabel IV.9 Pengujian Penambahan Data Petugas Data Salah

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Tambah Petugas Contoh Masukan : NIP : (kosong) Nama : (kosong) Alamat : (kosong) Jabatan : (kosong) Kantor : (kosong) User : (kosong) Password :	Menampilkan pesan “Maaf data gagal ditambahkan! Data Masukkan tidak boleh kosong atau data masukan tidak valid.”	Dapat menampilkan pesan kesalahan “Maaf data gagal ditambahkan! Data Masukkan tidak boleh kosong atau data masukan tidak valid.”	[√]diterima [ ]ditolak

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
(kosong)			
Data Tambah Petugas Contoh Masukan : NIP : 10108666 Nama : 122 Alamat : 212 Jabatan : (kosong) Kantor : (kosong) User : (kosong) Password : (kosong)	Menampilkan pesan "Maaf data gagal ditambahkan! Data Masukkan tidak boleh kosong atau data masukan tidak valid."	Dapat menampilkan pesan kesalahan " Maaf data gagal ditambahkan! Data Masukkan tidak boleh kosong atau data masukan tidak valid."	[√]diterima [ ]ditolak

3. Pengujian Penambahan Data Obat

Pengujian penambahan data obat dilakukan oleh farmasi dinas kesehatan pada tabel IV.10.

Tabel IV.10. Pengujian Penambahan Data Obat Data Benar

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Tambah Obat Contoh Masukan : Nama Obat : Amoksilin Jenis Obat : Anti Satuan : Tabelt Kemasan : 100 Persediaan : 200 Harga : 2000	Mengisikan data obat dan menyimpan data.	Data obat dapat tersimpan.	[√]diterima [ ]ditolak

Pengujian data salah penambahan data obat dapat dilihat pada tabel IV.11.

Tabel IV.11 Pengujian Penambahan Data Obat Data Salah

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Tambah Obat Contoh Masukan : Nama Obat : (kosong) Jenis Obat : (kosong) Satuan : (kosong) Kemasan : (kosong) Persediaan : (kosong) Harga : (kosong)	Menampilkan pesan "Maaf data gagal ditambahkan! Data Masukkan tidak boleh kosong atau data masukan tidak valid."	Dapat menampilkan pesan kesalahan "Maaf data gagal ditambahkan! Data Masukkan tidak boleh kosong atau data masukan tidak valid."	[√]diterima [ ]ditolak
Data Tambah Obat Contoh Masukan : Nama Obat : 200 Jenis Obat : antiinfeksi Satuan : 200 Kemasan : kk Persediaan : 11kk Harga : aa	Menampilkan pesan "Maaf data gagal ditambahkan! Data Masukkan tidak boleh kosong atau data masukan tidak valid."	Dapat menampilkan pesan kesalahan " Maaf data gagal ditambahkan! Data Masukkan tidak boleh kosong atau data masukan tidak valid."	[√]diterima [ ]ditolak

4. Pengujian Penambahan Data Jenis Obat

Pengujian penambahan data Jenis obat dilakukan oleh farmasi dinas kesehatan pada tabel IV.12.

Tabel IV.12 Pengujian Penambahan Data Jenis Obat Data Benar

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Tambah Jenis Obat Contoh Masukan :	Mengisikan data jenis obat dan menyimpan data.	Data jenis obat dapat tersimpan.	[√]diterima [ ]ditolak

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama Jenis Obat : Anti Infeksi			

Pengujian data salah penambahan data jenis obat dapat dilihat pada tabel IV.13.

Tabel IV.13 Pengujian Penambahan Data Jenis Obat Data Salah

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Tambah Jenis Obat Contoh Masukan : Nama Jenis Obat : (Kosong)	Menampilkan pesan "Maaf data gagal ditambahkan! Data Masukkan tidak boleh kosong atau data masukan tidak valid."	Dapat menampilkan pesan kesalahan "Maaf data gagal ditambahkan! Data Masukkan tidak boleh kosong atau data masukan tidak valid."	[√]diterima [ ]ditolak
Data Tambah Jenis Obat Contoh Masukan : Nama Obat : 11111	Menampilkan pesan "Maaf data gagal ditambahkan! Data Masukkan tidak boleh kosong atau data masukan tidak valid."	Dapat menampilkan pesan kesalahan " Maaf data gagal ditambahkan! Data Masukkan tidak boleh kosong atau data masukan tidak valid."	[√]diterima [ ]ditolak

5. Pengujian Penambahan Data Puskesmas

Pengujian penambahan data puskesmas dilakukan oleh farmasi dinas kesehatan pada tabel IV.14.

Tabel IV.14 Pengujian Penambahan Data Puskesmas Data Benar

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Tambah Puskesmas Contoh Masukan : Nama Puskesmas : Salam Alamat :	Mengisikan data puskesmas dan menyimpan data.	Data puskesmas dapat tersimpan.	[√]diterima [ ]ditolak

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Kiaracondong Telepon : 022-1238899 Kelurahan : Kiaracondong Kecamatan : Kiaracondong Kota : Bandung Pusat : Dinas Kesehatan			

Pengujian data salah penambahan data puskesmas dapat dilihat pada tabel IV.15.

Tabel IV.15 Pengujian Penambahan Data Puskesmas Data Salah

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Tambah Puskesmas Contoh Masukan : Nama Puskesmas : (kosong) Alamat : (kosong) Telepon : (kosong) Kelurahan : (kosong) Kecamatan : (kosong) Kota : (kosong) Pusat : (kosong)	Menampilkan pesan "Maaf data gagal ditambahkan! Data Masukkan tidak boleh kosong atau data masukan tidak valid."	Dapat menampilkan pesan kesalahan "Maaf data gagal ditambahkan! Data Masukkan tidak boleh kosong atau data masukan tidak valid."	[√]diterima [ ]ditolak
Data Tambah Puskesmas Contoh Masukan : Nama Puskesmas : 1233 Alamat : 11sss Telepon : dsad	Menampilkan pesan "Maaf data gagal ditambahkan! Data Masukkan tidak boleh kosong atau data masukan tidak valid."	Dapat menampilkan pesan kesalahan " Maaf data gagal ditambahkan! Data Masukkan tidak boleh kosong atau data masukan tidak valid."	[√]diterima [ ]ditolak

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Kelurahan : 123d Kecamatan : dq Kota : 33ddd Pusat : 23ccc			

6. Pengujian Penambahan Data Pemakaian

Pengujian penambahan data pemakaian dilakukan oleh farmasi puskesmas dan farmasi dinas kesehatan pada tabel IV.16.

Tabel IV.16 Pengujian Penambahan Data Pemakaian Data Benar

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Tambah Pemakaian Contoh Masukan : Periode : 12-12-2012 Nama Obat : Amoksilin Jumlah : 1234	Mengisikan data pemakaian dan menyimpan data.	Data pemakaian dapat tersimpan.	[√]diterima [ ]ditolak

Pengujian data salah penambahan data pemakaian dapat dilihat pada tabel IV.17.

Tabel IV.17 Pengujian Penambahan Data Pemakaian Data Salah

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Tambah Pemakaian Contoh Masukan : Periode : (kosong) Nama Obat : (kosong) Jumlah: (kosong)	Menampilkan pesan "Maaf data gagal ditambahkan! Data Masukkan tidak boleh kosong atau data masukan tidak valid."	Dapat menampilkan pesan kesalahan "Maaf data gagal ditambahkan! Data Masukkan tidak boleh kosong atau data masukan tidak valid."	[√]diterima [ ]ditolak
Data Tambah Pemakaian	Menampilkan pesan	Dapat menampilkan	[√]diterima

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Contoh Masukan : Periode : 13ss Nama Obat : Amok22 Jumlah: a122	"Maaf data gagal ditambahkan! Data Masukkan tidak boleh kosong atau data masukan tidak valid."	pesan kesalahan " Maaf data gagal ditambahkan! Data Masukkan tidak boleh kosong atau data masukan tidak valid."	[ ]ditolak

7. Pengujian Penambahan Data Permintaan

Pengujian penambahan data permintaan dilakukan oleh farmasi puskesmas pada tabel IV.18.

Tabel IV.18 Pengujian Penambahan Data Permintaan Data Benar

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Tambah Permintaan Contoh Masukan : Periode : 12-12-2012	Menghitung jumlah permintaan obat dan menyimpan data.	Peramalan permintaan obat dapat tersimpan.	[√]diterima [ ]ditolak

Pengujian data salah penambahan data permintaan dapat dilihat pada tabel IV.19.

Tabel IV.19 Pengujian Penambahan Data Permintaan Data Salah

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Tambah Permintaan Contoh Masukan : Periode : (kosong)	Menampilkan pesan "Maaf data gagal ditambahkan! Data Masukkan tidak boleh kosong atau data masukan tidak valid."	Dapat menampilkan pesan kesalahan "Maaf data gagal ditambahkan! Data Masukkan tidak boleh kosong atau data masukan tidak valid."	[√]diterima [ ]ditolak
Data Tambah Permintaan	Menampilkan pesan "Maaf data gagal	Dapat menampilkan pesan kesalahan "	[√]diterima [ ]ditolak

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Contoh Masukan : Periode : 13ss	ditambahkan! Data Masukkan tidak boleh kosong atau data masukan tidak valid.”	Maaf data gagal ditambahkan! Data Masukkan tidak boleh kosong atau data masukan tidak valid.”	

8. Pengujian Penambahan Data Pengiriman

Pengujian penambahan data pengiriman dilakukan oleh farmasi dinas kesehatan pada tabel IV.20.

Tabel IV.20 Pengujian Penambahan Data Pengiriman Data Benar

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Tambah Pengiriman Contoh Masukan : Periode : 12-12-2012 Nama Kantor : Salam Nama Obat : Amoksilin Jumlah : 1234	Mengisikan data pengiriman dan menyimpan data.	Data pengiriman akaian dapat tersimpan.	[√]diterima [ ]ditolak

Pengujian data salah penambahan data pengiriman dapat dilihat pada tabel IV.21.

Tabel IV.21 Pengujian Penambahan Data Pengiriman Data Salah

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Tambah Pengiriman Contoh Masukan : Periode : (kosong) Nama Kantor : (kosong) Nama Obat : (kosong)	Menampilkan pesan “Maaf data gagal ditambahkan! Data Masukkan tidak boleh kosong atau data masukan tidak valid.”	Dapat menampilkan pesan kesalahan “Maaf data gagal ditambahkan! Data Masukkan tidak boleh kosong atau data	[√]diterima [ ]ditolak

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Jumlah: (kosong)		masukan tidak valid.”	
Data Tambah Pengiriman Contoh Masukan : Periode : 13ss Nama Kantor : sal4 Nama Obat : Amok22 Jumlah: a122	Menampilkan pesan “Maaf data gagal ditambahkan! Data Masukkan tidak boleh kosong atau data masukan tidak valid.”	Dapat menampilkan pesan kesalahan " Maaf data gagal ditambahkan! Data Masukkan tidak boleh kosong atau data masukan tidak valid.”	[√]diterima [ ]ditolak

9. Pengujian Penambahan Data Penerimaan

Pengujian penambahan data penerimaan dilakukan oleh farmasi puskesmas pada tabel IV.22.

Tabel IV.22 Pengujian Penambahan Data Penerimaan Data Benar

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Tambah Penerimaan Contoh Masukan : Periode : 12-12-2012 Nama Obat : Amoksilin Jumlah : 1234	Mengisikan data pemakaian dan menyimpan data.	Data pemakaian dapat tersimpan.	[√]diterima [ ]ditolak

Pengujian data salah penambahan data penerimaan dapat dilihat pada tabel IV.23.

Tabel IV.23 Pengujian Penambahan Data Penerimaan Data Salah

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Tambah Penerimaan Contoh Masukan : Periode : (kosong)	Menampilkan pesan “Maaf data gagal ditambahkan! Data Masukkan tidak boleh	Dapat menampilkan pesan kesalahan “Maaf data gagal ditambahkan! Data	[√]diterima [ ]ditolak

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama Obat : (kosong) Jumlah: (kosong)	kosong atau data masukan tidak valid.”	Masukkan tidak boleh kosong atau data masukan tidak valid.”	
Data Tambah Penerimaan Contoh Masukan : Periode : 13ss Nama Obat : Amok22 Jumlah: a122	Menampilkan pesan “Maaf data gagal ditambahkan! Data Masukkan tidak boleh kosong atau data masukan tidak valid.”	Dapat menampilkan pesan kesalahan " Maaf data gagal ditambahkan! Data Masukkan tidak boleh kosong atau data masukan tidak valid.”	[√]diterima [ ]ditolak

10. Pengujian Cetak Laporan

Pengujian Cetak Laporan dilakukan oleh farmasi puskesmas, Farmasi dan Dinas Kesehatan pada tabel IV.24.

Tabel IV.24 Pengujian Cetak Laporan Data Benar

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Tambah Penerimaan Contoh Masukan : Periode : 12-12-2012	Menampilkan form download	Menampilkan form download	[√]diterima [ ]ditolak

Pengujian cetak laporan data salah dapat dilihat pada tabel IV.25.

Tabel IV.25 Pengujian Cetak Laporan Data Salah

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Tambah Penerimaan Contoh Masukan : Periode : (kosong)	Menampilkan pesan “Periode Tidak Boleh Kosong”	Dapat menampilkan pesan kesalahan “Periode Tidak Boleh Kosong”	[√]diterima [ ]ditolak

IV.4.3. Kesimpulan Pengujian *Black Box*

Berdasarkan hasil pengujian dengan kasus *sample* uji yang telah dilakukan memberikan kesimpulan bahwa pada proses sudah benar. Penyaringan kesalahan proses dalam bentuk arahan tampilan halaman pesan sudah cukup maksimal. Secara fungsional sistem sudah dapat menghasilkan *output* yang diharapkan.

IV.5. Pengujian Beta

Pengujian beta merupakan pengujian yang dilakukan secara objektif dimana diuji secara langsung ke lapangan, dengan mengadakan wawancara mengenai kepuasan *user* terhadap aplikasi yang telah dibangun, dengan kandungan point syarat *user friendly* serta apakah sesuai dengan yang diharapkan.

IV.5.1. Skenario Pengujian Beta

Pengujian beta ini dilakukan dengan cara melakukan pengujian langsung ditempat penelitian. Penelitian ini dilakukan di lingkungan salah satu puskesmas di bandung, sebagai berikut.

1. Lingkungan Farmasi Puskesmas

   Penelitian di lingkungan farmasi puskesmas, dilakukan supaya dapat mengetahui sejauh mana sistem yang dibangun dapat memberikan kemudahan pada farmasi puskesmas untuk melakukan pendataan dan pengiriman data permintaan..

2. Lingkungan Farmasi Dinas Kesehatan Kota Bandung

   Penelitian di lingkungan peserta Farmasi Dinas Kesehatan Bandung yang akan melakukan pendataan pengiriman obat.

IV.5.2. Daftar Pertanyaan

Proses wawancara ini akan ditanyakan lima pertanyaan untuk satu orang farmasi puskesmas sukajadi sebagai user di puskesmas dan lima pertanyaan untuk satu orang farmasi dinas kesehatan bandung sebagai user di Dinas Kesehatan Bandung.

a. Pertanyaan yang akan diberikan untuk Farmasi Puskesmas adalah sebagai berikut.

1. Apakah anda setuju bahwa tampilan sistem cukup menarik dan mudah dipahami ?

   Jawaban : Tampilan sistem ini menarik dan mudah dipahami bagi pengguna baru.

2. Apakah sistem yang dibangun sesuai dengan kebutuhan Puskesmas dalam mengendalikan persediaan obat ?

   Jawaban : Sistem ini sudah sesuai dalam proses pengendalian obat dan sudah sesuai dengan kebutuhan puskesmas.

3. Apakah sistem yang dibangun dapat membantu dalam memprediksi jumlah kebutuhan yang akan datang ?

   Jawaban : Sistem ini dapat membantu dalam memprediksi kebutuhan obat puskesmas yang akan datang secara otomatis.

4. Apakah anda setuju pengelolaan data pemakaian, permintaan dan penerimaan mudah dilakukan ?

   Jawaban : Sistem ini cukup membantu dalam pengelolaan data pemakaian, permintaan dan penerimaan.

5. Apakah sistem yang dibangun dapat mempermudah dalam mengecek persediaan dan mencetak laporan ?

   Jawaban : Sistem ini dapat mempermudah untuk melihat dan mengecek obat yang tersedia, dan sangat membantu dalam pembuatan laporan.

b. Pertanyaan yang akan diberikan untuk Farmasi Dinas Kesehatan Bandung

1. Apakah anda setuju bahwa tampilan sistem yang dibangun ini cukup menarik dan mudah dipahami ?

   Jawaban : Tampilan sistem ini sudah cukup menarik dan praktis.

2. Apakah anda setuju bahwa pengecekan data permintaan mudah dilakukan?

   Jawaban : Sistem ini sesuai dengan hitungan stok optimum.

3. Apakah sistem ini membantu dalam pengelolaan data obat ?

   Jawaban : Sistem ini sangat membantu dalam pengelolaan data obat.

4. Apakah sistem ini membantu dalam menambah data petugas puskesmas ?

   Jawaban : Sistem ini cukup membantu dalam memprediksi kebutuhan obat yang akan datang.

5. Apakah anda setuju bahwa sistem ini membantu dalam menentukan obat yang harus dikirim ?

   Jawaban : Sistem ini dapat otomatis memilih obat yang harus disediakan beserta kauntitinya

IV.5.3. Kesimpulan Pengujian Beta

Kesimpulan hasil wawancara yang telah dilakukan adalah sebagai berikut.

1. Tampilan (*Interface*) Sistem informasi ini cukup menarik dan mudah dipahami.
2. Sistem informasi ini membantu dalam mengendalikan persediaan, membantu dalam menentukan jumlah permintaan .
3. Pendokumentasian di sistem informasi cukup membantu dalam pelaporan.
4. Sistem informasi cukup membantu bidan dalam membuat laporan.
5. Sistem informasi ini juga memudahkan farmasi puskesmas untuk pendataan dan pengiriman permintaan obat ke pihak dinas kesehatan bandung.