BAB 4
IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi Sistem

Implementasi sistem adalah tahap penerapan sistem yang telah dibuat pada tahap perancangan. Implementasi sistem bertujuan untuk menerjemahkan perancangan berdasarkan hasil analisis menjadi kode – kode bahasa pemrograman yang dapat dimengerti oleh mesin.

4.1.1 Perangkat Keras yang Digunakan

Spesifikasi perangkat keras yang digunakan untuk implementasi sistem monitoring ini dapat dilihat pada tabel 4.1 sebagai berikut.

Tabel 4. 1 Spesifikasi Perangkat Keras yang Digunakan

No.	Perangkat Keras	Spesifikasi
1.	Monitor	17”
2.	Processor	Processor dual core berkecepatan 2,7 GHz
3.	Memori / RAM	2 GB
4.	Harddisk	320 GB
5.	VGA Card	VGA On-Board
6.	Keyboard	Standar
7.	Mouse	Standar
8.	Printer	Standar

4.1.2 Perangkat Lunak yang Digunakan

Spesifikasi perangkat lunak yang digunakan untuk implementasi sistem monitoring ini dapat dilihat pada tabel 4.2 sebagai berikut.

Tabel 4. 2 Spesifikasi Perangkat Lunak yang Digunakan

No.	Perangkat Lunak	Spesifikasi
1.	Sistem Operasi	Minimal Windows XP
2.	*Browser*	Mozila Firefox, Crome, Opera

4.1.3 Implementasi *Database*

Implementasi *database* diambil berdasarkan perancangan basis data yang dibuat pada bab sebelumnya. Tabel – tabel yang diimplementasikan dapat dilihat pada tabel 4.3.

Tabel 4. 3 Implementasi Data

No.	Nama Tabel	File Yang Berhubungan
1.	analisis	analisis.sql
2.	analisis_kriteria	analisis_kriteria.sql
3.	analisis_puskesmas_nilai	analisis_puskesmas_nilai.sql
4.	rencana_anggaran	rencana_anggaran.sql
5.	kriteria	kriteria.sql
6.	pegawai_dinas	pegawai_dinas.sql
7.	pegawai_puskesmas	pegawai_puskesmas.sql
8.	poa	poa.sql
9.	poa_rencana_kegiatan	Poa_rencana_kegiatan.sql
10.	lpj	lpj.sql
11.	puskesmas	puskesmas.sql

12.	puskesmas_nilai	puskesmas_nilai.sql

Database sistem yang dibangun pada aplikasi yang dibangun diimplementasikan pada *DBMS MySQL*. Implementasi tabel – tabel yang ada pada data sistem monitoring ini dapat dilihat pada tabel berikut.

1. Tabel analisis

   Tabel analisis terdiri dari id_analisis, id_pegawai_dinas, wkt_tulis. Rincian yang terdapat pada tabel analisis dapat dilihat pada tabel 4.4 sebagai berikut.

Tabel 4. 4 analisis

Tabel analisis

```
CREATE TABLE IF NOT EXISTS `analisis` (
  `id_analisis` int(11) NOT NULL AUTO_INCREMENT,
  `id_pegawai_dinas` int(11) NOT NULL,
  `wkt_tulis` int(11) NOT NULL,
  PRIMARY KEY (`id_analisis`),
  KEY `id_pegawai_dinas` (`id_pegawai_dinas`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;
```

2. Tabel Analisis_kriteria

   Tabel analisis_kriteria terdiri dari id_analisis_kriteria, id_kriteria, dan anggaran_dana. Rincian yang terdapat pada tabel analisis_poa dapat dilihat pada tabel 4.5 sebagai berikut.

Tabel 4. 5 Analisis_kriteria

Tabel anak

```
CREATE TABLE IF NOT EXISTS `analisis_kriteria` (
  `id_analisis_kriteria` int(11) NOT NULL AUTO_INCREMENT,
  `id_analisis` int(11) NOT NULL,
  `id_kriteria` int(11) NOT NULL,
  `anggaran_dana` double NOT NULL,
  PRIMARY KEY (`id_analisis_proposal`),
  KEY `id_analisis` (`id_analisis`),
  KEY `id_kriteria` (`id_poa`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;
```

3. Tabel Analisis_puskesmas_nilai

   Tabel analisis_puskesmas_nilai terdiri dari id, id_analisis, id_puskesmas, dan nilai. Rincian yang terdapat pada tabel analisis_puskesmas_nilai dapat dilihat pada tabel 4.6 sebagai berikut.

Tabel 4. 6 Analisis_puskesmas_nilai

Tabel analisis_puskesmas_nilai

```
CREATE TABLE IF NOT EXISTS `analisis_puskesmas_nilai` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `id_analisis` int(11) NOT NULL,
  `id_puskesmas_nilai` int(11) NOT NULL,
  `nilai` double NOT NULL,
  PRIMARY KEY (`id`),
  KEY `id_analisis` (`id_analisis`),
  KEY `id_puskesmas_nilai` (`id_puskesmas_nilai`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
```

4. Tabel Rencana kegiatan

Tabel rencana kegiatan terdiri dari id_rencana_kegiatan, id_pegawai_puskesmas, nama_kegiatan, bulam, indikator, bentuk_kegiatan,, sasaran, volume, satuan, keterangan, total, dan status, wkt_tulis, dan wkt_edit. Rincian yang terdapat pada tabel rencana kegiatan dapat dilihat pada tabel 4.7 sebagai berikut.

Tabel 4. 7 Rencana_kegiatan

Tabel anggaran

```
CREATE TABLE IF NOT EXISTS `rencana_kegiatan` (
  `id_rencana_kegiatan` int(11) NOT NULL AUTO_INCREMENT,
  `id_pegawai_puskesmas` int(11) NOT NULL,
  `nama_kegiatan` varchar(150) NOT NULL,
  `bulan` date NOT NULL,
  `indikator` text NOT NULL,
  `bentuk_kegiatan` varchar(200) NOT NULL,
  `sasaran` varchar(150) DEFAULT NULL,
  `volume` varchar(100) DEFAULT NULL,
  `satuan` varchar(50) DEFAULT NULL,
  `keterangan ` text NOT NULL,
  `total` bigint(20) NOT NULL,
  `status` varchar(15) NOT NULL COMMENT 'Tunda, Ajukan,
Setujui, Batal',
  `wkt_tulis` int(11) NOT NULL,
  `wkt_edit` int(11) NOT NULL,
  PRIMARY KEY (`id_poa`),
  KEY `id_pegawai_puskesmas` (`id_pegawai_puskesmas`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=7 ;
```

5. Tabel Kriteria

Tabel kriteria terdiri dari id_kriteria, dan nama_kriteria. Rincian yang terdapat pada tabel kriteria dapat dilihat pada tabel 4.8 sebagai berikut.

Tabel 4. 8 Kriteria

Tabel kriteria

CREATE TABLE IF NOT EXISTS `kriteria` (
  `id_kriteria` int(11) NOT NULL AUTO_INCREMENT,
  `nama_kriteria` varchar(100) NOT NULL,
  PRIMARY KEY (`id_kriteria`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=6 ;

6. Tabel Pegawai_dinas

   Tabel pegawai_dinas terdiri dari id_pegawai_dinas, username, password, tipe, nama, tanggal_lahir, alamat, no_tlpn, email, aktif, wkt_tulis, wkt_edit, dan id_input. Rincian yang terdapat pada tabel pegawai_dinas dapat dilihat pada tabel 4.9 sebagai berikut.

Tabel 4. 9 Pegawai_dinas

Tabel pegawai_dinas

CREATE TABLE IF NOT EXISTS `pegawai_dinas` (
  `id_pegawai_dinas` int(11) NOT NULL AUTO_INCREMENT,
  `username` varchar(100) NOT NULL,
  `password` varchar(150) NOT NULL,
  `tipe` enum('Teknis','Bendahara','Kepala') NOT NULL DEFAULT 'Teknis',
  `nama` varchar(100) NOT NULL,
  `tanggal_lahir` date NOT NULL,
  `alamat` text NOT NULL,
  `no_tlp` varchar(15) NOT NULL,
  `email` varchar(100) NOT NULL,
  `aktif` enum('Y','N') NOT NULL,
  `wkt_tulis` int(11) NOT NULL,
  `wkt_edit` int(11) NOT NULL,
  `id_input` int(11) NOT NULL,
  PRIMARY KEY (`id_pegawai_dinas`),
  UNIQUE KEY `username` (`username`),
  KEY `id_input` (`id_input`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=5 ;

7. Tabel pegawai_puskesmas

   Tabel pegawai_puskesmas terdiri dariid_pegawai_puskesmas, id_puskesmas, username, password, nama, tanggal_lahir, alamat, no_tlpn, email, aktif, wkt_tulis, wkt_edit, dan id_input. Rincian yang terdapat pada pegawai_puskesmas dapat dilihat pada tabel 4.10 berikut.

Tabel 4. 10 pegawai_puskesmas

Tabel pegawai_puskesmas

```
CREATE TABLE IF NOT EXISTS `pegawai_puskesmas` (
  `id_pegawai_puskesmas` int(11) NOT NULL AUTO_INCREMENT,
  `id_puskesmas` int(11) NOT NULL,
  `username` varchar(100) NOT NULL,
  `password` varchar(150) NOT NULL,
  `nama` varchar(100) NOT NULL,
  `tanggal_lahir` date NOT NULL,
  `alamat` text NOT NULL,
  `no_tlp` varchar(15) NOT NULL,
  `email` varchar(100) NOT NULL,
  `aktif` enum('Y','N') NOT NULL,
  `wkt_tulis` int(11) NOT NULL,
  `wkt_edit` int(11) NOT NULL,
  `id_input` int(11) NOT NULL,
  PRIMARY KEY (`id_pegawai_puskesmas`),
  UNIQUE KEY `username` (`username`),
  KEY `id_input` (`id_input`),
  KEY `id_puskesmas` (`id_puskesmas`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=4 ;
```

8. Tabel POA

Tabel poa terdiri dari id_poa, id_pegawai_puskesmas, tgl_pengajuan, total, tanggapan, wkt_tulis, wkt_edit, dan id_pegawai_dinas. Rincian yang terdapat pada tabel biaya anak dapat dilihat pada tabel 4.11 berikut.

Tabel 4. 11 Tabel POA

Tabel poa

```
CREATE TABLE IF NOT EXISTS `poa` (
  `id_poa` int(11) NOT NULL AUTO_INCREMENT,
  `id_pegawai_puskesmas` int(11) NOT NULL,
  `tgl_pengajuan` date NOT NULL,
  `total` bigint(20) NOT NULL,
  `tanggapan` text NOT NULL,
  `status` varchar(15) NOT NULL COMMENT 'Ajukan, Setujui,
Sukses',
  `wkt_tulis` int(11) NOT NULL,
  `wkt_edit` int(11) NOT NULL,
  `id_pegawai_dinas` int(11) DEFAULT NULL,
  PRIMARY KEY (`id_poa`),
  KEY `id_pegawai_puskesmas` (`id_pegawai_puskesmas`),
  KEY `id_pegawai_dinas` (`id_pegawai_dinas`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;
```

9. Tabel Poa_rencana_kegiatan

Tabel poa_rencana_kegiatan terdiri dari id_poa_rencana_kegiatan, id_poa, id_rencana_kegiatan, nama_kegiatan, indikator, sasaran, bentuk_kegiatan, volume, satuan, keterangan, total_biaya, status , wkt_tulis, dan wkt_edit . Rincian yang terdapat pada tabel poa_rencana_kegiatan dapat dilihat pada tabel 4.12 berikut.

Tabel 4. 12 Poa_rencana_kegiatan

Tabel poa_rencana_kegiatan

```
CREATE TABLE IF NOT EXISTS `poa_rencana_kegiatan ` (
  `id_poa_rencana_kegiatan     `     int(11)     NOT     NULL
AUTO_INCREMENT,
  `id_poa` int(11) NOT NULL,
  `id_rencana_kegiatan` int(11) NOT NULL,
  `nama_kegiatan` varchar(150) NOT NULL,
  `tgl_pengajuan` date NOT NULL,
  `indikator` text,
  `bentuk_kegiatan` varchar(200) NOT NULL,
  `sasaran` varchar(150) DEFAULT NULL,
  `volume` varchar(100) DEFAULT NULL,
  `satuan` varchar(50) DEFAULT NULL,
  `keterangan` text NOT NULL,
  `total` bigint(20) NOT NULL,
  `status` varchar(15) NOT NULL COMMENT 'Ajukan, Setujui,
Sukses',
  `keterangan` varchar(15) DEFAULT NULL COMMENT 'Lebih,
Pas, Kurang',
  `wkt_tulis` int(11) NOT NULL,
  `wkt_edit` int(11) NOT NULL,
  PRIMARY KEY (`id_poa_rencana_kegiatan`),
  KEY `id_proposal` (`id_poa`),
  KEY `id_anggaran` (`id_rencana_kegiatan`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=7 ;
```

10. Tabel lpj

Tabel lpj terdiri dari id_lpj, id_poa_rencana_kegiatan, id_pegawai_puskesmas, tgl_kegiatan, uraian_anggaran, lokasi, digunakan, keterangan, bukti_lampiran. Rincian yang terdapat pada tabel lpj dapat dilihat pada tabel 4.13 berikut.

Tabel 4. 13 Tabel lpj

Tabel lpj

```
CREATE TABLE IF NOT EXISTS ` lpj` (
  `id_lpj` int(11) NOT NULL AUTO_INCREMENT,
  `id_poa_rencana_kegiatan ` int(11) NOT NULL,
  `id_pegawai_puskesmas` int(11) NOT NULL,
  `tgl_kegiatan` date NOT NULL,
  `uraian_anggaran` varchar(200) DEFAULT NULL,
  `lokasi` varchar(100) DEFAULT NULL,
  `pelaksana ` bigint(20) NOT NULL,
  `keterangan` text NOT NULL,
  `bukti_lampiran` varchar(100) NOT NULL,
  `waktu` int(11) NOT NULL,
  PRIMARY KEY (`id_lpj`),
  KEY `id_poa_rencana_kegiatan ` (`id_proposal_anggaran`),
  KEY `id_pegawai_puskesmas` (`id_pegawai_puskesmas`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;
```

11. Tabel puskesmas

Tabel puskesmas terdiri dari id_puskesmas, id_pegawai_dinas, nama_puskesmas, alamat, status, wkt_tulis, dan wkt_edit. Rincian yang terdapat pada tabel puskesmas dapat dilihat pada tabel 4.14 berikut.

Tabel 4. 14 Puskesmas

Tabel puskesmas

```
CREATE TABLE IF NOT EXISTS `puskesmas` (
  `id_puskesmas` int(11) NOT NULL AUTO_INCREMENT,
  `id_pegawai_dinas` int(11) DEFAULT NULL,
  `nama_puskesmas` varchar(100) NOT NULL,
  `alamat` text NOT NULL,
  `status` char(1) NOT NULL,
  `wkt_tulis` int(11) NOT NULL,
  `wkt_edit` int(11) NOT NULL,
  PRIMARY KEY (`id_puskesmas`),
  KEY `id_pegawai_dinas` (`id_pegawai_dinas`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=4 ;
```

12. Tabel Puskesmas_nilai

Tabel puskesmas_nilai terdiri dari id_puskesmas_nilai, id_puskesmas, id_kriteria, dan nilai. Rincian yang terdapat pada tabel puskesmas_nilai dapat dilihat pada tabel 4.15 berikut.

Tabel 4. 15 Puskesmas_nilai

Tabel proposal_anggaran
CREATE TABLE IF NOT EXISTS `puskesmas_nilai` (`id_puskesmas_nilai` int(11) NOT NULL AUTO_INCREMENT, `id_puskesmas` int(11) NOT NULL, `id_kriteria` int(11) NOT NULL, `nilai` double NOT NULL, PRIMARY KEY (`id_puskesmas_nilai`), KEY `id_puskesmas` (`id_puskesmas`), KEY `id_kriteria` (`id_kriteria`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=16 ;

4.1.4 Implementasi Antarmuka

Implementasi antarmuka dilakukan untuk setiap tampilan program yang dibangun beserta pengkodeannya dalam bentuk file program. Implementasi antarmuka yang dibangun terdiri dari implementasi antarmuka Pegawai Teknis, implementasi antarmuka Kepala BOK, implementasi Bendahara dan implementasi antarmuka Pegawai Puskesmas. Implementasi antarmuka Pegawai teknis untuk dapat dilihat pada tabel 4.16.

Tabel 4. 16 Implementasi Antarmuka Pegawai Teknis

Menu	**Deskripsi**	**Nama File**
Login	Halaman untuk mengisi data *login* agar dapat masuk kedalam sistem	index.php
Lupa *Password*	Halaman yang digunakan jika lupa *password*	lupa_*password* .php
Pegawai Dinas	Halaman yang digunakan jika mengolah data pegawai dinas	Pegawai_dinas.php
Pegawai_puskesmas	Halaman yang digunakan jika mengolah pegawai_puskesmas	Pegawai_puskesmas.php
Puskesmas	Halaman yang digunakan untuk	Puskesmas.php

	mengelola data puskesmas.	
Analisis	Halaman yang digunakan untuk mengolah analisis.	Analisis.php
Kriteria	Halaman yang digunakan untuk mengolah kriteria	Kriteria.php
Profil	Halaman yang digunakan untuk melihat atau edit profil	Profil.php

Implementasi antarmuka yang kedua Kepala BOK dapat dilihat pada tabel 4.17.

Tabel 4. 17 Implementasi Antarmuka Kepala BOK

Menu	**Deskripsi**	**Nama File**
Login	Halaman untuk mengisi data *login* agar dapat masuk kedalam sistem	index.php
Monitoring	Halaman yang digunakan memonitoring anggaran dana dan penggunaan dana	Monitoring.php
Profil	Halaman yang digunakan untuk melihat atau edit profil	Profil.php

Implementasi antarmuka yang ketiga Bendahara dapat dilihat pada tabel 4.18.

Tabel 4. 18 Implementasi Antarmuka Bendahara

Menu	**Deskripsi**	**Nama File**
Login	Halaman untuk mengisi data *login* agar dapat masuk kedalam sistem	index.php
POA	Halaman yang digunakan jika melihat POA untuk untuk dibandingkan	Poa.php

	dengan Lpj	
Lpj	Halaman yang digunakan jika melihat lpj	Lpj.php
Profil	Halaman yang digunakan untuk melihat atau edit profil	Profil.php

Implementasi antarmuka yang keempat Pegawai_puskesmasdapat dilihat pada tabel 4.19.

Tabel 4. 19 Implementasi Antarmuka Pegawai Puskesmas

Menu	Deskripsi	Nama File
Login	Halaman untuk mengisi data *login* agar dapat masuk kedalam sistem	index.php
Rencana kegiatan	Halaman yang digunakan jika mengolah data rencana kegiatan	Rencana_kegiatan.php
POA	Halaman yang digunakan jika mengolah data poa	Poa.php
Profil	Halaman yang digunakan untuk melihat atau edit profil	Profil.php

4.2 Pengujian Sistem

Pengujian sistem dilakukan untuk menguji apakah sistem yang dibangun sudah layak untuk digunakan. Pengujian yang digunakan untuk menguji sistem monitoring ini adalah menggunakan pengujian *black box* dan pegujian *beta.* Pengujian *black box* ini berfokus pada persyaratan fungsional dari perangkat lunak yang dibangun.

4.2.1 Rencana Pengujian

Rencana pengujian terdiri dari rencana pengujian untuk pegawai teknis, kepala BOK, bendahara dan pengujian pegawai puskesmas.

4.2.1.1 Rencana Pengujian Halaman Pegawai Teknis

Rencana pengujian untuk halaman Pegawai Teknis dapat dilihat pada tabel 4.20 sebagai berikut.

Tabel 4. 20 Rencana Pengujian Halaman Pegawai Teknis

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Input data *login*	Blackbox
	Verifikasi	Blackbox
	Lupa *password*	Blackbox
Pegawai Dinas	Tambah Pegawai Dinas	Blackbox
Pegawai Puskesmas	Lihat Pegawai Dinas	Blackbox
	Tambah Pegawai Dinas	Blackbox
	Lihat Pegawai Puskesmas	Blackbox
Puskesmas	Tambah puksesmas	Blackbox
	Edit pukesmas	Blackbox
Analisis	Tambah analisis	Blackbox
	Edit analisis	Blackbox
Kriteria	Tambah kriteria	Blackbox
	Lihat kriteria	Blackbox
	Edit kriteria	Blackbox
Profil	Edit Profil	Blackbox
	Lihat profil	Blackbox

4.2.1.2 Rencana Pengujian Halaman Kepala BOK

Rencana pengujian untuk halaman Kepala BOK dapat dilihat pada tabel 4.21 sebagai berikut.

Tabel 4. 21 Rencana Pengujian Halaman Kepala BOK

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Input data *login*	Blackbox
	Verifikasi	Blackbox
	Lupa *password*	Blackbox
Monitoring	Lihat anggaran dana dan penggunaan dana puskesmas	Blackbox
		Blackbox
Profil	Edit profil	Blackbox
	Lihat profil	Blackbox

4.2.1.3 Rencana Pengujian Halaman Bendahara

Rencana pengujian untuk halaman Bendahara dapat dilihat pada tabel 4.22 sebagai berikut.

Tabel 4. 22 Rencana Pengujian Halaman Bendahara

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Input data *login*	Blackbox
	Verifikasi	Blackbox
	Lupa *password*	Blackbox
POA	Lihat poa	Blackbox
Lpj	Lihat Lpj	Blackbox
Profil	Edit profil	Blackbox
	Lihat Profil	Blackbox

4.2.1.4 Rencana Pengujian Halaman Pegawai Puskesmas

Rencana pengujian untuk halaman Pegawai Puskesmas dapat dilihat pada tabel 4.23 sebagai berikut.

Tabel 4. 23 Rencana Pengujian Halaman Pegawai Puskesmas

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Input data *login*	Blackbox
	Verifikasi	Blackbox
	Lupa *password*	Blackbox
Rencana kegiatan	Masukan rencana kegiatan	Blackbox
	Lihat rencana kegiatan	Blackbox
POA	Lihat poa	Blackbox
	Tambah poa	Blackbox
Profil	Edit profil	Blackbox
	Lihat Profil	Blackbox

4.3 Pengujian Black Box

Pengujian *black box* adalah pengujian yang berfokus kepada keluaran yang dihasilkan dari data perangkat lunak yang dibangun. Teknik yang digunakan dalam pengujian sistem monitoring ini adalah teknik *equivalence partitioning* yaitu masing – masing kelas *equivalensi partition* diproses dimana program yang akan memproses anggota kelas – kelas tersebut secara *equivalen*.

4.3.1 Pengujian Metode *Equivalence partitioning* Halaman Puskesmas

Pengujian metode *equivalence partitioning* pada halaman Puskesmas terdiri dari pengujian *login* dan poa.

1. **Pengujian *Login***

Pengujian dari *login* dengan data *valid* kemudian diamati dan menghasilkan sebuah kesimpulan. Pengujian *login* dapat dilihat pada tabel 4.24.

Tabel 4. 24 Pengujian *Login* dengan Data *Valid*

Kasus dan hasil uji (data *valid*)			
Data masukan	***Valid* class**	**Pengamatan**	**Kesimpulan**
*User*name Contoh masukan : Nita	[A-Z | a-z | 0-9]	Data *user*name dapat diterima jika input data : 1. Kurang dari 10 karakter 2. Boleh menggunakan angka, huruf atau kombinasi dari keduanya	[√] diterima [ ] ditolak
Password Contoh masukan : 123456	[A-Z | a-z | 0-9]	Data *password* dapat diterima jika input data : 1. Kurang dari 150 karakter 2. Boleh menggunakan angka, huruf atau kombinasi dari keduanya	[√] diterima [ ] ditolak

Pengujian dari *login* dengan data *invalid* kemudian diamati dan menghasilkan sebuah kesimpulan. Pengujian *login* dapat dilihat pada tabel 4.25.

Tabel 4. 25 Pengujian *Login* dengan Data *Invalid*

Kasus dan hasil uji (data *invalid*)			
Data masukan	***Valid* class**	**Pengamatan**	**Kesimpulan**
*User*name Contoh masukan : Nita	Data kosong	Dapat menampilkan pesan kesalahan bahwa *user*name harus diisi (tidak boleh kosong)	[ ] diterima [√] ditolak
Password Contoh masukan : 123456	Data kosong	Dapat menampilkan pesan kesalahan bahwa *password* harus diisi (tidak boleh kosong)	[ ] diterima [√] ditolak

2. Pengujian Rencana kegiatan

Pengujian dari rencana kegiatan dengan data *valid* kemudian diamati dan menghasilkan sebuah kesimpulan. Pengujian rencana kegiatan dapat dilihat pada tabel 4.26.

Tabel 4. 26 Pengujian Rencana kegiatan dengan Data *Valid*

Kasus dan hasil uji (data *valid*)			
Data masukan	***Valid* class**	**Pengamatan**	**Kesimpulan**
Nama kegiatan yang dimasukan Contoh masukan : Pelayanan gizi	[A-Z | a-z]	Data nama kegiatan dapat diterima jika input data : 1. Kurang dari 50 karakter 2. Boleh menggunakan huruf kapital maupun huruf kecil atau kombinasi dari keduanya	[√] diterima [ ] ditolak
Waktu kegiatan yang dimasukan Contoh masukan : Agustus	[date]	Data dapat diterima apabila memilih bulan rencana kegiatan	[√] diterima [ ] ditolak
Indikator yang dimasukan Contoh masukan : Balita gizi buruk	[A-Z | a-z]	Data indikator dapat diterima jika input data : 1. Kurang dari 50 karakter 2. Boleh menggunakan huruf capital maupun huruf kecil atau kombinasi dari keduanya	[√] diterima [ ] ditolak
Bentuk kegiatan yang dimasukan Contoh masukan : Pelacakan balita gizi buruk	[A-Z | a-z]	Data Bentuk kegiatan dapat diterima jika input data : 1. Kurang dari 50 karakter 2. Boleh	[√] diterima [ ] ditolak

		menggunakan huruf kapital maupun huruf kecil atau kombinasi dari keduanya	
Sasaran yang dimasukan Contoh masukan: Balita gizi buruk	[A-Z | a-z]	Data Sasaran dapat diterima jika input data : 1. Kurang dari 50 karakter 2. Boleh menggunakan huruf kapital maupun huruf kecil atau kombinasi dari keduanya	[√] diterima [ ] ditolak
Volume Kegiatan yang dimasukan Contoh masukan: 10 org x 2 kasus x Rp.200.000	[A-Z | a-z |0-9]	Data Volume kegiatan dapat diterima jika input data : 1. Kurang dari 50 karakter 2. Boleh menggunakan huruf kapital maupun huruf kecil atau kombinasi dari keduanya dan angka	[√] diterima [ ] ditolak
Total Biaya yang dimasukan Contoh masukan: Rp. 4.000.000	[0..9]	Data dapat diterima jika memasukan angka	[√] diterima [ ] ditolak

Pengujian dari rencana kegiatan dengan data *invalid* kemudian diamati dan menghasilkan sebuah kesimpulan. Pengujian rencana kegiatan dapat dilihat pada tabel 4.27.

Tabel 4. 27 Pengujian Rencana kegiatan dengan Data *Invalid*

Kasus dan hasil uji (data In*valid*)			
Data masukan	***Valid* class**	**Pengamatan**	**Kesimpulan**
Nama kegiatan yang dimasukan Contoh masukan : Pelayanan gizi	Data kosong	Data menampilkan masukan nama kegiatan	[√] diterima [ ] ditolak
Waktu kegiatan yang dimasukan Contoh masukan : Agustus	Data kosong	Data menampilkan masukan waktu kegiatan	[√] diterima [ ] ditolak
Indikator yang dimasukan Contoh masukan : Balita gizi buruk	Data kosong	Data menampilkan masukan indikator	[√] diterima [ ] ditolak
Bentuk kegiatan yang dimasukan Contoh masukan : Pelacakan balita gizi buruk	Data kosong	Data menampilkan masukan bentuk kegiatan	[√] diterima [ ] ditolak
Sasaran yang dimasukan Contoh masukan: Balita gizi buruk	Data kosong	Data menampilkan masukan sasaran	[√] diterima [ ] ditolak
Volume Kegiatan yang dimasukan Contoh masukan: 10 org x 2 kasus x Rp.200.000	Data kosong	Data menampilkan masukan volume kegiatan	[√] diterima [ ] ditolak
Total Biaya yang dimasukan Contoh masukan: Rp. 4.000.000	Data kosong	Data menampilkan masukan total biaya	[√] diterima [ ] ditolak

3. Pengujian Poa

Pengujian dari poa dengan data *valid* kemudian diamati dan menghasilkan sebuah kesimpulan. Pengujian pengolahan data proposal dapat dilihat pada tabel 4.28.

Tabel 4. 28 Poa dengan Data *Valid*

Kasus dan hasil uji (data *valid*)			
Data masukan	***Valid* class**	**Pengamatan**	**Kesimpulan**
Tanggal pengajuan Contoh masukan : 10-10-2014	[date]	Data Tanggal pengajuan dapat diterima jika input data berupa tanggal	[√] diterima [ ] ditolak

Pengujian dari poa dengan data *invalid* kemudian diamati dan menghasilkan sebuah kesimpulan. Pengujian poa dapat dilihat pada tabel 4.29.

Tabel 4. 29 Poa dengan Data *Invalid*

Kasus dan hasil uji (data *valid*)			
Data masukan	***Valid* class**	**Pengamatan**	**Kesimpulan**
Tanggal pengajuan Contoh masukan : 10-10-2014	Data kosong	Masukan tanggal pengajuan	[√] diterima [ ] ditolak

4.3.2 Pengujian Metode *Equivalence Pertitioning* Halaman Pegawai Teknis

Pengujian metode *equivalence partitioning* pada halaman Pegawai teknis terdiri dari pengujian *login*,puskesmas, analisis, kriteria, dan profil.

1. Pengujian *Login* Pegawai Teknis

Pengujian dari *login* Pegawai teknis dengan data *valid* kemudian diamati dan menghasilkan sebuah kesimpulan. Pengujian *login* Pegawai teknis dapat dilihat pada tabel 4.30.

Tabel 4. 30 Pengujian *Login* Pegawai teknis dengan Data *Valid*

Kasus dan hasil uji (data *valid*)			
Data masukan	***Valid* class**	**Pengamatan**	**Kesimpulan**
*User*name Contoh masukan : Dedi	[A-Z | a-z | 0-9]	Data *user*name dapat diterima jika input data : 1. Kurang dari 10 karakter 2. Boleh menggunakan angka, huruf atau kombinasi dari keduanya	[√] diterima [ ] ditolak
Password	[A-Z | a-z |	Data *password* dapat	[√]

Kasus dan hasil uji (data *valid*)			
Data masukan	***Valid* class**	**Pengamatan**	**Kesimpulan**
Contoh masukan : 123456	0-9]	diterima jika input data : 1. Kurang dari 150 karakter 2. Boleh menggunakan angka, huruf atau kombinasi dari keduanya	diterima [ ] ditolak

Pengujian dari *login* Pegawai teknis dengan data *invalid* kemudian diamati dan menghasilkan sebuah kesimpulan. Pengujian *login* Pegawai teknis dapat dilihat pada tabel 4.31.

Tabel 4. 31 Pengujian *Login* Pegawai Teknis Dengan Data *Invalid*

Kasus dan hasil uji (data *invalid*)			
Data masukan	***Valid* class**	**Pengamatan**	**Kesimpulan**
*User*name Contoh masukan : Dedi	Data kosong	Dapat menampilkan pesan kesalahan bahwa *user*name harus diisi (tidak boleh kosong)	[√] diterima [ ] ditolak
Password Contoh masukan : 123456	Data kosong	Dapat menampilkan pesan kesalahan bahwa *password* harus diisi (tidak boleh kosong)	[√] diterima [ ] ditolak

2. Pengujian Pengolahan Puskesmas

Pengujian dari pengolahan puskesmas dengan data *valid* kemudian diamati dan menghasilkan sebuah kesimpulan. Pengujian pengolahan puskesmas dapat dilihat pada tabel 4.32.

Tabel 4. 32 Pengujian Pengolahan Puskesmas dengan Data *Valid*

Kasus dan hasil uji (data *valid*)			
Data masukan	***Valid* class**	**Pengamatan**	**Kesimpulan**
Nama puskesmas yang ditambah Contoh masukan : Cibuntu	[A-Z | a-z]	Data nama puskesmas dapat diterima jika input data : 1. Kurang dari 50 karakter	[√] diterima [ ] ditolak

Kasus dan hasil uji (data *valid*)			
Data masukan	***Valid* class**	**Pengamatan**	**Kesimpulan**
		2. Boleh menggunakan huruf kapital maupun huruf kecil atau kombinasi dari keduanya	
Alamat puskesmas yang di tambah Contoh masukan : Dago	[A-Z | a-z |	Data alamat puskesmas dapat diterima jika input data : 1. Kurang dari 50 karakter 2. Boleh menggunakan huruf, simbol atau kombinasi dari keduanya	[√] diterima [ ] ditolak

Pengujian dari pengolahan puskesmas dengan data *invalid* kemudian diamati dan menghasilkan sebuah kesimpulan. Pengujian pengolahan puskesmas dapat di lihat pada tabel 4.33.

Tabel 4. 33 Pengujian Pengolahan Puskesmas dengan Data *Invalid*

Kasus dan hasil uji (data *valid*)			
Data masukan	***Valid* class**	**Pengamatan**	**Kesimpulan**
Nama puskesmas yang ditambah Contoh masukan : Cibuntu	Data kosong	Data menampilkan pesan bahwa Nama puskesmas harus di isi	[√] diterima [ ] ditolak
Alamat puskesmas yang di tambah Contoh masukan : Dago	Data kosong	Data menampilkan pesan bahwa alamat puskesmas harus di isi	[√] diterima [ ] ditolak

3. Pengujian Analisis

Pengujian analisis dengan data *valid* kemudian diamati dan menghasilkan sebuah kesimpulan. Pengujian analisis dapat dilihat pada tabel 4.34.

Tabel 4. 34 Pengujian Analisis dengan Data *Valid*

Kasus dan hasil uji (data *valid*)			
Data masukan	***Valid* class**	**Pengamatan**	**Kesimpulan**
Quota dana Contoh masukan : 5400000	[0..9]	Data dapat diterima jika memasukan angka	[√] diterima [ ] ditolak

Pengujian dari analisis dengan data *invalid* kemudian diamati dan menghasilkan sebuah kesimpulan. Pengujian analisis dapat dilihat pada tabel 4.35.

Tabel 4. 35 Pengujian Analisis dengan Data *Invalid*

Kasus dan hasil uji (data *valid*)			
Data masukan	***Valid* class**	**Pengamatan**	**Kesimpulan**
Quota dana Contoh masukan : 5400000	[0..9]	Masukan quota dana	[√] diterima [ ] ditolak

4. Pengujian Kriteria

Pengujian dari kriteria dengan data *valid* kemudian diamati dan menghasilkan sebuah kesimpulan. Pengujian kriteria dapat dilihat pada tabel 4.36

Tabel 4. 36 Pengujian Kriteria dengan Data *Valid*

Kasus dan hasil uji (data *valid*)			
Data masukan	***Valid* class**	**Pengamatan**	**Kesimpulan**
Kriteria yang di tambah Contoh masukan : Jumlah penduduk	[A-Z | a-z |	Data kriteria dapat diterima jika input data : 1. Kurang dari 50 karakter 2. Boleh menggunakan huruf, simbol atau kombinasi dari keduanya	[√] diterima [ ] ditolak

Pengujian dari kriteria dengan data *invalid* kemudian diamati dan menghasilkan sebuah kesimpulan. Pengujian kriteria dapat dilihat pada tabel 4.37.

Tabel 4. 37 Pengujian Kriteria dengan Data *Invalid*

Kasus dan hasil uji (data *valid*)			
Data masukan	***Valid* class**	**Pengamatan**	**Kesimpulan**
Kriteria yang di tambahContoh masukan : Jumlah penduduk	Data kosong	Masukan kriteria	[√] diterima [ ] ditolak

4.3.3 Pengujian Metode *Equivalence Pertitioning* Halaman Kepala BOK

Pengujian metode *equivalence partitioning* pada halaman Kepala BOK terdiri dari pengujian *login*, dan monitoring.

1. Pengujian *Login*

Pengujian dari *login* dengan data *valid* kemudian diamati dan menghasilkan sebuah kesimpulan. Pengujian *login* dapat dilihat pada tabel 4.38.

Tabel 4. 38 Pengujian *Login* dengan Data *Valid*

Kasus dan hasil uji (data *valid*)			
Data masukan	***Valid* class**	**Pengamatan**	**Kesimpulan**
*User*name Contoh masukan : Rahmat	[A-Z | a-z | 0-9]	Data *user*name dapat diterima jika input data : 1. Kurang dari 10 karakter 2. Boleh menggunakan angka, huruf atau kombinasi dari keduanya	[√] diterima [ ] ditolak
Password Contoh masukan : 123456	[A-Z | a-z | 0-9]	Data *password* dapat diterima jika input data : 1. Kurang dari 150 karakter 2. Boleh menggunakan angka, huruf atau kombinasi dari keduanya	[√] diterima [ ] ditolak

Pengujian dari *login* dengan data *invalid* kemudian diamati dan menghasilkan sebuah kesimpulan. Pengujian *login* dapat dilihat pada tabel 4.39.

Tabel 4. 39 Pengujian *Login* dengan Data *Invalid*

Kasus dan hasil uji (data *invalid*)			
Data masukan	***Valid* class**	**Pengamatan**	**Kesimpulan**
*User*name Contoh masukan : Rahmat	Data kosong	Dapat menampilkan pesan kesalahan bahwa *user*name harus diisi (tidak boleh kosong)	[ ] diterima [√] ditolak
Password Contoh masukan : 123456	Data kosong	Dapat menampilkan pesan kesalahan bahwa *password* harus diisi (tidak boleh kosong)	[ ] diterima [√] ditolak

2. Pengujian Monitoring

Pengujian monitoring dengan data *valid* kemudian diamati dan menghasilkan sebuah kesimpulan. Pengujian monitoring dapat dilihat pada tabel 4.40.

Tabel 4. 40 Pengujian Monitoring dengan Data *valid*

Kasus dan hasil uji (data *invalid*)			
Data masukan	***Valid* class**	**Pengamatan**	**Kesimpulan**
Monitoring Contoh masukan: Penggunaan dana	Memilih	Data penggunaan dana	[ ] diterima [√] ditolak

Pengujian monitoring dengan data *invalid* kemudian diamati dan menghasilkan sebuah kesimpulan. Pengujian monitoring dapat dilihat pada tabel 4.41.

Tabel 4. 41 Pengujian Monitoring dengan Data *Invalid*

Kasus dan hasil uji (data *invalid*)			
Data masukan	***Valid* class**	**Pengamatan**	**Kesimpulan**
Monitoring Contoh masukan: Penggunaan dana	Tidak memilih	Harus memilih data yang ingin di monitoring	[ ] diterima [√] ditolak

4.3.4 Pengujian Metode Equivalence Pertitioning Halaman Bendahara

Pengujian metode *equivalence partitioning* pada halaman Bendahara terdiri dari pengujian poa, dan lpj

1. Pengujian poa

Pengujian dari poa dengan data *valid* kemudian diamati dan menghasilkan sebuah kesimpulan. Pengujian poa dapat dilihat pada tabel 4.42.

Tabel 4. 42 Pengujian poa dengan Data *Valid*

Kasus dan hasil uji (data *valid*)			
Data masukan	***Valid* class**	**Pengamatan**	**Kesimpulan**
Pilih poa yang ingin dilihat Contoh masukan : Puskesmas Cibuntu	Memilih	Data Poa yang di pilih	[√] diterima [ ] ditolak

Pengujian dari poa dengan data *invalid* kemudian diamati dan menghasilkan sebuah kesimpulan. Pengujian poa dapat dilihat pada tabel 4.43.

Tabel 4. 43 Pengujian poa dengan Data *Invalid*

Kasus dan hasil uji (data *invalid*)			
Data masukan	***Valid* class**	**Pengamatan**	**Kesimpulan**
Pilih poa yang ingin dilihat Contoh masukan : Puskesmas Cibuntu	TidakMemilih	Harus memilih poa yang ingin dilihat	[√] diterima [ ] ditolak

2. Pengujian Lpj

Pengujian dari lpj dengan data *valid* kemudian diamati dan menghasilkan sebuah kesimpulan. Pengujian lpj dapat dilihat pada tabel 4.45.

Tabel 4. 45 Pengujian lpj dengan Data *Valid*

Kasus dan hasil uji (data *valid*)			
Data masukan	***Valid* class**	**Pengamatan**	**Kesimpulan**
Pilih bulan lpj yang ingin dilihat Contoh masukan : Januari - maret	Memilih	Data lpj yang di pilih	[√] diterima [ ] ditolak

Pengujian dari poa dengan data *invalid* kemudian diamati dan menghasilkan sebuah kesimpulan. Pengujian poa dapat dilihat pada tabel 4.44.

Tabel 4. 44 Pengujian lpj dengan Data *Invalid*

Kasus dan hasil uji (data *invalid*)			
Data masukan	***Valid* class**	**Pengamatan**	**Kesimpulan**
Pilih bulan lpj yang ingin dilihat Contoh masukan : Januari - maret	TidakMemilih	Harus memilih bulan lpj yang ingin dilihat	[√] diterima [ ] ditolak

4.3.5 Kesimpulan Hasil Pengujian

Berdasarkan hasil pengujian sistem yag dilakukan secara menyeluruh telah memberikan suatu kesimpulan bahwa proses pada sistem monitoring ini telah melalui tahap perbaikan dan sudah dioptimalkan terhadap proses – proses tersebut dan secara fungsional sistem monitoring ini sudah dapat digunakan dan menghasilkan respon sistem yang diharapkan.

4.4 Pengujian Beta

Pengujian beta merupakan pengujian yang dilakukan secara objektif dimana sistem yang dibangun dapat diuji secara langsung ke pengguna yaitu Kepala BOK dengan melakukan wawancara.

4.4.1 Wawancara

Wawancara dilakukan kepada Kepala BOK pada sistem informasi monitoring ini.

a. Pengujian Beta Wawancara dengan Kepala BOK

Berikut adalah pertanyaan yang diajukan kepada Kepala BOK dan jawaban yang telah diberikan.

1. Apakah Sistem Informasi Monitoring Bantuan Operasional Kesehatan (BOK) ini dapat membantu Dinas Kesehatan dalam menentukan anggaran dana terhadap masing-masing puskesmas yang ada di kota Bandung?

   Jawaban :

Ya, dan bermanfaat karena aplikasi memuat perhitungan untuk alokasikan pendanaan ke puskesmas.

2. Apakah pengolahan data pengguna dalam website Sistem Informasi Monitoring Bantuan Operasional Kesehatan (BOK) ini dapat membantu Dinas Kesehatan dalam hal pengumpulan informasi laporan pertanggungjawaban penggunaan dana BOK tiap-tiap puskesmas di kota Bandung?

   Jawaban :

   Ya, dalam aplikasi ini dibuat untuk bendahara BOK mengontrol kegiatan pertanggungjawaban penggunaan dana.

3. Apakah website Sistem Informasi Monitoring Bantuan Operasional Kesehatan (BOK) ini dapat membantu Dinas Kesehatan dalam memonitoring pencapaian target kegiatan yang dilakukan tiap-tiap puskesmas di Kota Bandung?

   Jawaban :

   Ya, sangat memudahkan untuk monitoring target pencapaian

4. Apakah website Sistem Informasi Monitoring Bantuan Operasional Kesehatan (BOK) ini secara keseluruhan mudah digunakan?

   Jawaban:

   Ya, mudah digunakan dan interface menarik, tools/ menu-menu yang ada cukup lengkap.

5. Apakah website Sistem Informasi Monitoring Bantuan Operasional Kesehatan (BOK) ini menarik untuk digunakan?

   Jawaban:

   Ya, sangat menarik untuk digunakan karena aplikasi ini sudah dilengkapi dengan menu monitoring

Dari hasil wawancara yang dilakukan dapat ditarik kesimpulan bahwa website Sistem Informasi Monitoring Bantuan Operasional Kesehatan (BOK) Puskesmas ini dapat membantu untuk melakukan menentukan anggaran dana yang di dapat oleh puskesmas, pertanggungjawaban penggunaan dana, monitoring kegiatan dan membantu dalam pengolahan puskesmas yang mudah digunakan.