BAB 4

IMPLEMENTASI DAN PENGUJIAN

Pada bab ini akan dilakukan implementasi dan pengujian terhadap aplikasi sistem informasi kalibrasi yang akan dibangun. Tahapan ini dilakukan setelah perancangan selesai dilakukan dan selanjutnya akan diimplementasikan ke dalam bahasa pemograman. Setelah implementasi maka dilakukan pengujian terhadap aplikasi yang baru dan akan dilihat kekurangan-kekurangan pada aplikasi yang baru untuk pengembangan sistem selanjutnya.

4.1 Implementasi

Setelah sistem dianalisis dan didesain secara rinci, maka akan menuju tahap implementasi. Implementasi sistem merupakan tahap meletakkan sistem sehingga siap untuk dioperasikan.Implementasi bertujuan untuk mengkonfirmasi modul-modul perancangan, sehingga pengguna dapat memberi masukan kepada pembangun sistem.

4.1.1 Implementasi Perangkat Keras

Perangkat keras yang digunakan untuk mengimplementasikan sistem adalah sebagai berikut :

1. Processor core two duo
2. RAM 2 GB,
3. Hard Disk 200 GB,
4. CD ROM Drive,
5. Monitor 15’,
6. Keyboard dan mouse.
7. Lan Card
8. Switch
9. Hub
10. Kabel UTP

4.1.2 Implementasi Perangkat Lunak

Perangkat lunak yang dipasang pada sistem computer yang akan digunakan untuk membangun aplikasi ini adalah sebagai berikut :

2. Sistem Operasi Windows XP / Windows 7
3. Software database MySQL, sebagai *engine* pengolah data yang digunakan untuk menyimpan dan melakukan beberapa fungsi yang lazim dilakukan dalam sebuah *database*.
4. Software Programming Macromedia Dreamweaver.
5. Browser Mozzila Firefox dan Google Chrome.

4.1.3 Implementasi Basis Data

Pembuatan basis data dilakukan dengan menggunakan MYSQL. Berikut akan dijelaskan langkah-langkah pembuatan *database* aplikasi pengolahan data aset.

Tabel 4.1 Tabel Implementasi Basis Data

Tabel User

```
CREATE TABLE IF NOT EXISTS `tb_user` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `username` varchar(50) NOT NULL,
  `password` varchar(100) NOT NULL,
  `nama` varchar(50) NOT NULL,
  `divisi` varchar(50) NOT NULL,
  `created_by` varchar(50) NOT NULL,
  `date_created` datetime NOT NULL,
  `last_modified` datetime NOT NULL,
  `status` char(1) NOT NULL,
  `modified_by` varchar(50) NOT NULL,
  `jabatan` varchar(25) NOT NULL,
  PRIMARY KEY (`id`)
  CONSTRAINT `FK_nama`
  FOREIGN KEY (`divisi`) REFERENCES
`tb_bagian`(`nama`) ON DELETE CASCADE
ON UPDATE CASCADE
)     ENGINE=InnoDB          DEFAULT     CHARSET=latin1
AUTO_INCREMENT=38 ;
```

Tabel Permohonan

```
CREATE TABLE IF NOT EXISTS `tb_permohonan` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
```

```
`id_detail` int(11) NOT NULL,
  `tgl_permohonan` date NOT NULL,
  `no_permohonan` varchar(25) NOT NULL,
  `tgl_penerimaan` date NOT NULL,
  `no_penerimaan` varchar(25) NOT NULL,
  `bagian` varchar(50) NOT NULL,
  `kontak` varchar(50) NOT NULL,
  `tempat` enum('0','1') NOT NULL DEFAULT '0',
  `lab` enum('0','1') NOT NULL DEFAULT '0',
  `dtawal` date NOT NULL,
  `dtakhir` date NOT NULL,
  `userProduksi` int(11) NOT NULL,
  `stProduksi` enum('0','1') NOT NULL,
  `userQA` int(11) NOT NULL,
  `stQA` enum('0','1') NOT NULL,
  `userKalibrasi` int(11) NOT NULL,
  `stKalibrasi` enum('0','1') NOT NULL,
  `catatan` text NOT NULL,
`id_user` int(11) NOT NULL,
  PRIMARY KEY (`id`)
CONSTRAINT `FK_user`
  FOREIGN KEY (`id_user`) REFERENCES
`tb_user`(`id`) ON DELETE CASCADE
ON UPDATE CASCADE
  CONSTRAINT `FK_detail`
  FOREIGN KEY (`id_detail`) REFERENCES
`tb_detail_permohonan`(`id`) ON DELETE CASCADE
  ON UPDATE CASCADE
)     ENGINE=InnoDB          DEFAULT     CHARSET=latin1
AUTO_INCREMENT=34 ;
```

Tabel Detail Permohonan

```
CREATE TABLE IF NOT EXISTS `tb_detail_permohonan` (
  `id_detail` int(11) NOT NULL AUTO_INCREMENT,
  `no_permohonan` varchar(25) NOT NULL,
  `kd_alat` varchar(11) NOT NULL,
  `lokasi` varchar(100) NOT NULL,
  `status` enum('0','1') NOT NULL,
  `keterangan` varchar(100) NOT NULL,
  `A` enum('0','1') NOT NULL COMMENT 'visual',
  `B` enum('0','1') NOT NULL COMMENT 'perlengkapan',
  PRIMARY KEY (`id_detail`),
  CONSTRAINT `FK_alat`
  FOREIGN KEY (`kd_alat`) REFERENCES
`tb_alat`(`kd_alat`) ON DELETE CASCADE
  ON UPDATE CASCADE
)     ENGINE=InnoDB          DEFAULT     CHARSET=latin1
```

```
AUTO_INCREMENT=75 ;
```

Tabel Alat

```
CREATE TABLE IF NOT EXISTS `tb_alat` (
  `kd_alat` varchar(11) NOT NULL,
  `nama` varchar(100) NOT NULL,
  `merk` varchar(30) NOT NULL,
  `kd_kelompok` int(11) NOT NULL,
  `exp_date` date NOT NULL,
  `status` enum('0','1') NOT NULL,
  `date_created` datetime NOT NULL,
  `created_by` varchar(50) NOT NULL,
  `date_modified` datetime NOT NULL,
  `modified_by` varchar(50) NOT NULL,
  `stdel` enum('0','1') NOT NULL,
  `no_seri` varchar(50) NOT NULL,
  `sensor` varchar(50) NOT NULL,
  `skala` varchar(50) NOT NULL,
  `digunakan` varchar(50) NOT NULL,
  `bagian` int(3) NOT NULL,
  `rangex` varchar(25) NOT NULL,
  `lokasi` varchar(25) NOT NULL,
  PRIMARY KEY (`kd_alat`),
  CONSTRAINT `FK_kelompok`
  FOREIGN KEY (`kd_kelompok`) REFERENCES
`tb_kelompok_alat`(`kd_kelompok`) ON DELETE CASCADE
  ON UPDATE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

Tabel Kelompok Alat

```
CREATE TABLE IF NOT EXISTS `tb_kelompok_alat` (
  `kd_kelompok` int(11) NOT NULL AUTO_INCREMENT,
  `nama` varchar(100) NOT NULL,
  `date_created` datetime NOT NULL,
  `created_by` varchar(50) NOT NULL,
  `date_modified` datetime NOT NULL,
  `modified_by` varchar(50) NOT NULL,
  PRIMARY KEY (`kd_kelompok`)
)     ENGINE=InnoDB          DEFAULT     CHARSET=latin1
AUTO_INCREMENT=10 ;
```

Tabel Kalibrator

```
CREATE TABLE IF NOT EXISTS `tb_kal` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `kd_alat` varchar(11) NOT NULL,
  `nama` varchar(100) NOT NULL,
  `merk` varchar(30) NOT NULL,
  `kd_kelompok` int(11) NOT NULL,
```

```
  `exp_date` date NOT NULL,
  `status` enum('0','1') NOT NULL,
  `date_created` datetime NOT NULL,
  `created_by` varchar(50) NOT NULL,
  `date_modified` datetime NOT NULL,
  `modified_by` varchar(50) NOT NULL,
  `stdel` enum('0','1') NOT NULL,
  `no_seri` varchar(50) NOT NULL,
  PRIMARY KEY (`id`),
  CONSTRAINT `FK_kelompok`
  FOREIGN KEY (`kd_kelompok`) REFERENCES
`tb_kelompok_alat`(`kd_kelompok`) ON DELETE CASCADE
  ON UPDATE CASCADE
)     ENGINE=InnoDB          DEFAULT     CHARSET=latin1
AUTO_INCREMENT=22 ;
```

Tabel Bagian

```
CREATE TABLE IF NOT EXISTS `tb_bagian` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `nama` varchar(50) NOT NULL,
  `date_created` datetime NOT NULL,
  `created_by` varchar(50) NOT NULL,
  `date_modified` datetime NOT NULL,
  `modified_by` varchar(50) NOT NULL,
  PRIMARY KEY (`id`),
  UNIQUE KEY `nama` (`nama`)
)       ENGINE=InnoDB       DEFAULT       CHARSET=latin1
AUTO_INCREMENT=501 ;
```

Tabel Laporan

```
CREATE TABLE IF NOT EXISTS `tb_lap` (
  `id` int(10) NOT NULL AUTO_INCREMENT,
  `jenis_lap` varchar(50) NOT NULL,
  `no_laporan` varchar(50) NOT NULL,
  `kd_alat` varchar(50) NOT NULL,
  `spek1` int(11) NOT NULL,
  `spek2` int(11) NOT NULL,
  `spek3` int(11) NOT NULL,
  `f1` varchar(50) NOT NULL,
  `f2` varchar(50) NOT NULL,
  `f3` varchar(50) NOT NULL,
  `f4` varchar(50) NOT NULL,
  `f5` varchar(50) NOT NULL,
  `f6` varchar(50) NOT NULL,
  `f7` varchar(50) NOT NULL,
  `f8` varchar(50) NOT NULL,
  `f9` varchar(50) NOT NULL,
  `f10` varchar(50) NOT NULL,
```

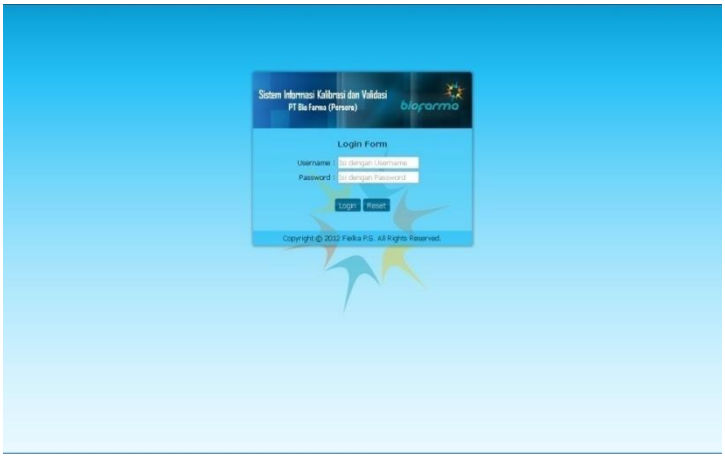
```
  `f11` varchar(50) NOT NULL,
  `f12` varchar(50) NOT NULL,
  `f13` varchar(50) NOT NULL,
  `f14` varchar(50) NOT NULL,
  `f15` varchar(50) NOT NULL,
  `f16` varchar(50) NOT NULL,
  `f17` varchar(50) NOT NULL,
  `f18` varchar(50) NOT NULL,
  `f19` varchar(50) NOT NULL,
  `f20` varchar(50) NOT NULL,
  `f21` varchar(50) NOT NULL,
  `f22` varchar(50) NOT NULL,
  `f23` varchar(50) NOT NULL,
  `f24` varchar(50) NOT NULL,
  `f25` varchar(50) NOT NULL,
  `f26` varchar(50) NOT NULL,
  `f27` varchar(50) NOT NULL,
  `f28` varchar(50) NOT NULL,
  `f29` varchar(50) NOT NULL,
  `f30` varchar(50) NOT NULL,
  `f31` varchar(50) NOT NULL,
  `f32` varchar(50) NOT NULL,
  `f33` varchar(50) NOT NULL,
  `f34` varchar(50) NOT NULL,
  `f35` varchar(50) NOT NULL,
  `f36` varchar(50) NOT NULL,
  `f37` varchar(50) NOT NULL,
  `f38` varchar(50) NOT NULL,
  `f39` varchar(50) NOT NULL,
  `f40` varchar(50) NOT NULL,
  `f41` varchar(50) NOT NULL,
  `f42` varchar(50) NOT NULL,
  `f43` varchar(50) NOT NULL,
  `f44` varchar(50) NOT NULL,
  `f45` varchar(50) NOT NULL,
  `f46` varchar(50) NOT NULL,
  `f47` varchar(50) NOT NULL,
  `f48` varchar(50) NOT NULL,
  `f49` varchar(50) NOT NULL,
  `f50` varchar(50) NOT NULL,
  `f51` varchar(50) NOT NULL,
  `f52` varchar(50) NOT NULL,
  `f53` varchar(50) NOT NULL,
  `f54` varchar(50) NOT NULL,
  `f55` varchar(50) NOT NULL,
  `f56` varchar(50) NOT NULL,
  `f57` varchar(50) NOT NULL,
```

```
  `f58` varchar(50) NOT NULL,
  `f59` varchar(50) NOT NULL,
  `f60` varchar(50) NOT NULL,
  `f61` varchar(50) NOT NULL,
  `f62` varchar(50) NOT NULL,
  `f63` varchar(50) NOT NULL,
  `f64` varchar(50) NOT NULL,
  `f65` varchar(50) NOT NULL,
  `f66` varchar(50) NOT NULL,
  `f67` varchar(50) NOT NULL,
  `f68` varchar(50) NOT NULL,
  `f69` varchar(50) NOT NULL,
  `f70` varchar(50) NOT NULL,
  `f71` varchar(50) NOT NULL,
  `f72` varchar(50) NOT NULL,
  `f73` varchar(50) NOT NULL,
  `f74` varchar(50) NOT NULL,
  `f75` varchar(50) NOT NULL,
  `f76` varchar(50) NOT NULL,
  `f77` varchar(50) NOT NULL,
  `f78` varchar(50) NOT NULL,
  `f79` varchar(50) NOT NULL,
  `f80` varchar(50) NOT NULL,
  PRIMARY KEY (`id`),
CONSTRAINT `FK_alat`
  FOREIGN KEY (`kd_alat`) REFERENCES
`tb_alat`(`kd_alat`) ON DELETE CASCADE
  ON UPDATE CASCADE
)     ENGINE=InnoDB          DEFAULT     CHARSET=latin1
AUTO_INCREMENT=41 ;
```

4.1.4 Implementasi Antarmuka

Implementasi antarmuka dilakukan dengan setiap halaman web yang dibuat. Implementasi antarmuka yang dibuat unutk semua pengguna, yaitu *administrator, validasi dan kalibrasi, quality assurance* dan pemilik alat. Untuk sisanya bisa dilihat dilampiran anatamuka.

1. Halaman Utama



Gambar 4.1 Halaman Utama

2. Halaman *Home Administrator*



Gambar 4.2 Halaman *Home Administrator*

3. Halaman *Home Validasi dan Kalibrasi*



Gambar 4.3 Halaman *Home Validasi dan Kalibrasi*

4. Halaman *Home Quality Assurance*



Gambar 4.4 Halaman *Home Quality Assurance*

5. Halaman *Home* Pemilik Alat



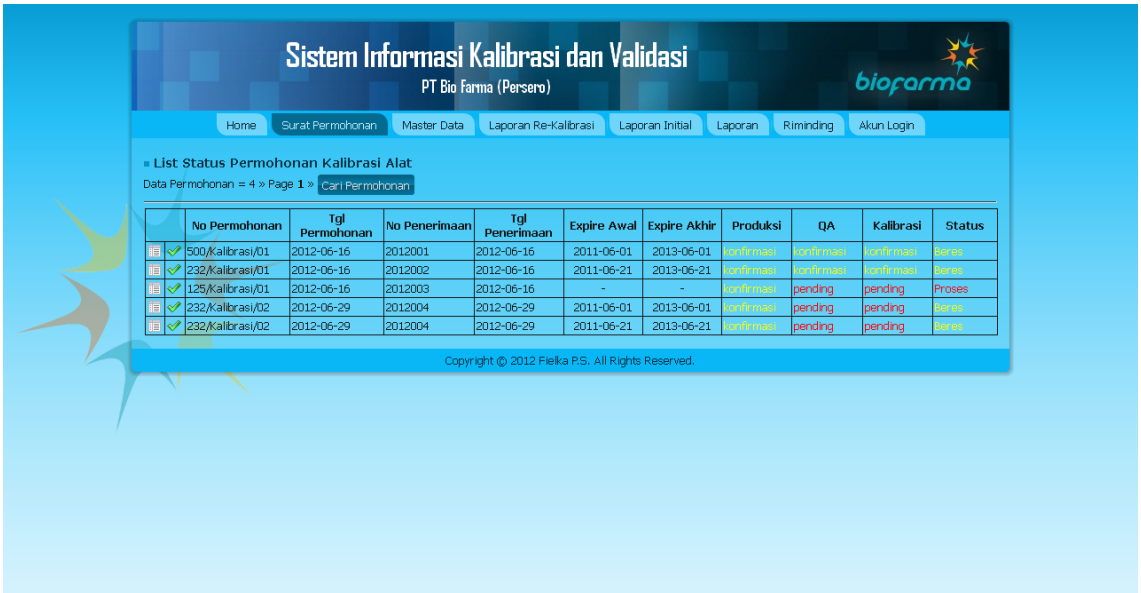
Gambar 4.5 Halaman *Home* Pemilik Alat

6. Halaman List Surat Permohonan *Administrator*



Gambar 4.6 Halaman List Surat Permohonan *Administrator*

7. Halaman List Surat Permohonan *Validasi dan Kalibrasi*



Gambar 4.7 Halaman List Surat Permohonan *Validasi dan Kalibrasi*

8. Halaman List Surat Permohonan *Quality Assurance*



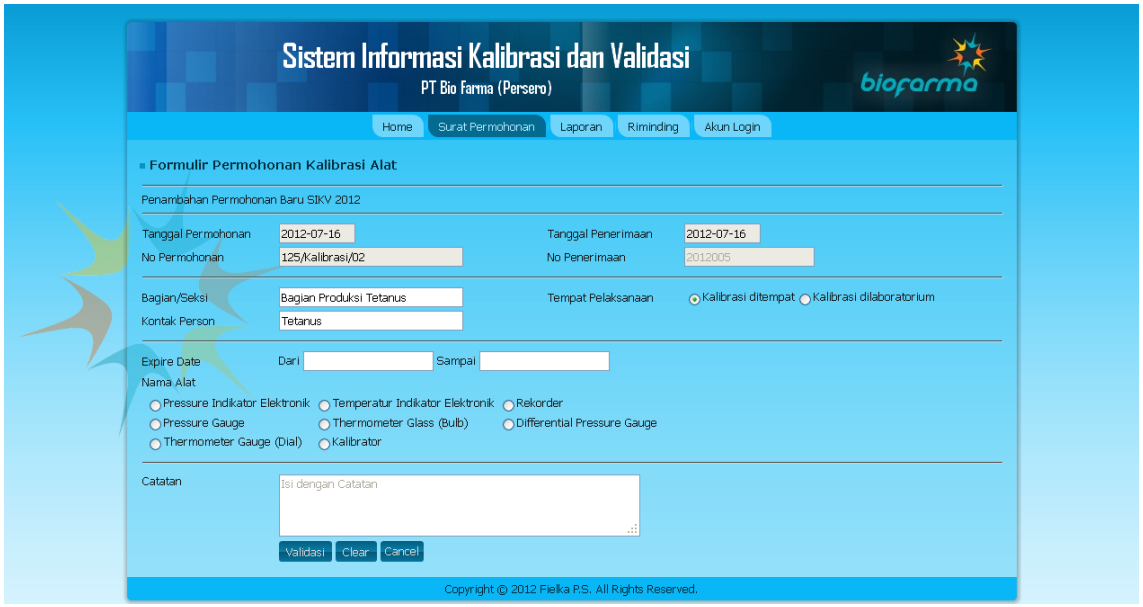
Gambar 4.8 Halaman List Surat Permohonan *Quality Assurance*

9. Halaman List Surat Permohonan Pemilik Alat



Gambar 4.9 Halaman List Surat Permohonan Pemilik Alat

10. Halaman Tambah Surat Permohonan



Gambar 4.10 Halaman Tambah Surat Permohonan

11. Halaman Validasi Tambah Surat Permohonan



Gambar 4.11 Halaman Validasi Surat Permohonan

4.2 Pengujian

Pengujian merupakan bagian yang penting dalam siklus pembangunan perangkat lunak. Pengujian dilakukan untuk menjamin kualitas dan juga mengetahui kelemahan dari perangkat lunak. Tujuan pengujian ini adalah untuk menjamin bahwa perangkat lunak yang dibangun memiliki kualitas yang handal, yaitu mampu mempresentasikan kajian pokok dan spesifikasi, analisis perancangan dan pengkodean dari perangkat lunak itu sendiri.

Pengujian yang dilakukan yaitu pengujian secara fungsional *(alpha)* dan betha. Metode pengujian *alpha* yang digunakan adalah pengujian *blackbox* yang berfokus pada persyaratan fungsional dari sistem yang dibangun. Pengujian ini memungkinkan perekayasa perangkat lunak mendapatkan serangkaian kondisi input yang sepenuhnya semua persyaratan fungsional untuk suatu program.

4.2.1 Rancangan Pengujian

Pengujian perangkat lunak ini menggunakan metode pengujian *BlackBox*. Pengujian *BlackBox* berfokus pada persyaratan fungsional perangkat lunak yang dibuat.

Pengujian sistem informasi kalibrasi berikut menggunakan data uji berupa sebuah data dan masukan dari user.

Tabel 4.2 Rancangan Pengujian

Item Uji	Detail Pengujian	Metode Uji
Login	Verifikasi username dan password	*Black Box*
Mengelola Data User	Tambah User	*Black Box*
	Ubah User	*Black Box*
	Hapus User	*Black Box*
Mengelola Data Bagian	Tambah Bagian	*Black Box*
	Ubah Bagian	*Black Box*
	Hapus Bagian	*Black Box*
Mengelola Data Kelompok Alat	Tambah Kelompok Alat	*Black Box*
	Ubah Kelompok Alat	*Black Box*
	Hapus Kelompok Alat	*Black Box*
Mengelola Data Kalibrator	Tambah Kalibrator	*Black Box*
	Ubah Kalibrator	*Black Box*
	Hapus Kalibrator	*Black Box*
Mengelola Surat Permohonan	Tambah permohonan	*Black Box*
	Ubah permohonan	*Black Box*
	Hapus permohonan	*Black Box*
Mengelola Data Laporan	Tambah laporan initial	*Black Box*
	Tambah Laporan Re-Kalibrasi	*Black Box*
	Ubah Laporan	*Black Box*
	Hapus Hapus Laporan	*Black Box*
Mengelola Rekap Laporan	Buat Rekap Laporan	*Black Box*
Mengelola Reminding	Lihat Reminding	*Black Box*
MengelolaPassword	Ganti Password	*Black Box*

4.2.1.1 Kasus dan Hasil Pengujian

Berdasarkan rencana pengujian yang telah disusun, diperoleh hasil pengujian seperti berikut :

1. Pengujian *Login*

Berikut dapat dilihat pengujian dari *login* dapat dilihat pada tabel 4.3 berikut ini :

Tabel 4.3 Pengujian *Login*

Kasus dan Hasil Uji (data normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Username : admin *Password* : admin	Tercantum pada text *field**username*dan*passwor**d*	Dapat mengisi data login sesuai yang diharapkan	Diterima
Data Login berupa *Username* : admin *Password* : admin	Data login dicari dalam *database*, *username* dan data *password* benar login valid dan dapat masuk ke menu home admin.	*username* dan *password* benar dapat masuk ke menu home admin	Diterima
Kasus dan Hasil Uji (data salah)			
Username : admin *Password* : 1234	Tidak dapat masuk dan menampilkan pesan diantaranya yaitu “username dan password belum terdaftar. Harap hubungi administrator !”.	Tidak dapat login / masuk dan menampilkan pesan kesalahan sesuai yang diharapkan	Diterima

2. Pengujian Tambah *User*

Berikut ini dapat dilihat pengujian dari tambah *user* dapat dilihat pada tabel 4.4 berikut ini :

Tabel 4.4 Pengujian Tambah *User*

Kasus dan Hasil Uji (data normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tambah *user*	Menampilkan *form*tambah *user*	Dapat menampilkan *form* user	Diterima
Mengisi data *user* Contoh : *Username* : admin	Tercantum pada masing-masing *field* kemudian data akan	Dapat mengisi masing-masing *field* dan data berhasil	Diterima

Password : admin Jabatan : admin	tersimpan didalam *database* dan menampilkan pesan”Menambahkan Data User Berhasil.”	disimpan ke dalam *database* dan menampilkan pesan sesuai yang diharapkan.	
Kasus dan Hasil Uji (data salah)			
Mengisi data *user* Contoh : *Username* : admin *Password* : 1234	Data user tidak tersimpan di dalam *database* dan menampilkan pesan “ Username sudah terdaftar. Ganti username!”	Data user tidak tersimpan di dalam *database* dan menampilkan pesan sesuai yang diharapkan	Diterima

3. Pengujian Ubah *User*

Berikut ini dapat dilihat pengujian dari ubah *user* dapat dilihat pada tabel 4.5 berikut ini :

Tabel 4.5 Pengujian Ubah *User*

Kasus dan Hasil Uji (data normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Ubah *user*	Menampilkan data*user*	Dapat menampilkan data *user*	Diterima
Data *user*lama Contoh : *Username* : admin *Password* : admin Jabatan : admin	Tercantum data *user* berupa *username, password* dan bagian yang lama pada text *fielduser*	Dapat menampilkan *field* data yang akan diubah dengan benar	Diterima
Data *user*yang akan diubah Contoh : *Username* : fielka *Password* : admin Jabatan : admin	Dapat mengisi *field* data yang akan diubah dengan benar dan data berhasil diubah dan disimpan dalam *database* dan menampilkan pesan “Ubah Data User Telah Berhasil.”	Dapat mengisi *field* data yang akan diubah dendgan benar dan data berhasil diubah dan disimpan dalam *database* dan menampilkan pesan sesuai dengan yang diharapkan.	Diterima
Kasus dan Hasil Uji (data salah)			
Data *user*lama Contoh : *Username* : (kosong) *Password* : 1234	Proses ubah *user* gagal dilakukan karena data kosong/tidak terisi menampilkan	Proses ubah *user* gagal dilakukan dikarenakan data yang diinputkan kosong.	Diterima

	peringatan “Isi isiian ini”.	Menampilkan peringatan “Isi isiian ini”.	

4. Pengujian Hapus User

Berikut ini dapat dilihat pengujian dari hapus *user* dapat dilihat pada tabel 4.6 berikut ini :

Tabel 4.6 Pengujian Hapus *User*

Kasus dan Hasil Uji (data normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pilih Data *user* yang akan dihapus. Contoh : *Username* : admin Jabatan : admin	Tercantum data *user* yang akan dihapus berupa username dan jabatan, setelah memilih ikon hapus maka tampil form hapus dan apabila memilik OK maka data *user* berhasil dihapus dan jika Cancel maka tidak akan menghapus data *user*	Data *user* yang akan dihapus berada di daftar user dan apabila menekan ikon hapus maka akan tampil form hapus dan jika memilih OK maka data *user* berhasil dihapus dan jika memilih Cancel maka tidak akan menghapus data *user*	Diterima

5. Pengujian Tambah Bagian

Berikut ini dapat dilihat pengujian dari tambah bagian dapat dilihat pada tabel 4.7 berikut ini :

Tabel 4.7 Pengujian Tambah Bagian

Kasus dan Hasil Uji (data normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tambahbagian	Menampilkan *form* tambah bagian	Dapat menampilkan *form* bagian	Diterima
Mengisi data bagian Contoh : Kode Bagian : 114 Nama Bagian : Bagian Produksi Vaksin Campak	Tercantum pada masing-masing *field* kemudian data akan tersimpan didalam *database* dan muncul pesan “Penambahan bagian baru telah berhasil”serta data	Dapat mengisi masing-masing *field* dan data berhasil disimpan ke dalam *database* dan menampilkan pesan sesuai yang diharapkan serta data	Diterima

	bagian yang baru ditambahkan tercantum juga di form tambah laporan initial, tambah laporan re-kalibrasi, tambah user, buat reminding dan buat rekap laporan	bagian yang baru tampil di form tambah laporan initial, tambah laporan re-kalibrasi, tambah user, buat reminding dan buat rekap laporan	
Kasus dan Hasil Uji (data salah)			
Mengisi data bagian Contoh : Kode Bagian : 111 Nama Bagian : Bagian Produksi Vaksin Campak	Data bagian tidak tersimpan di dalam *database* dan menampilkan pesan "nama bagian sudah terdaftar"	Data bagian tidak tersimpan di dalam *database* dan menampilkan pesan kesalahan sesuai yang diharapkan	Diterima

6. Pengujian Ubah Bagian

Berikut ini dapat dilihat pengujian dari tambah bagian dapat dilihat pada tabel 4.8 berikut ini :

Tabel 4.8 Pengujian Ubah Bagian

Kasus dan Hasil Uji (data normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Ubahbagian	Menampilkan data bagian	Dapat menampilkan data bagian	Diterima
Data bagianlama Contoh : Nama Bagian : Quality Assurance	Tercantum data bagian berupa nama bagiandan bagian yang lama pada text *field*bagian	Dapat menampilkan *field* data bagian yang akan diubah dengan benar	Diterima
Data bagianyang baru Contoh : Nama Bagian : QA	Dapat mengisi *field* data yang akan diubah dengan benar dan data berhasil diubah dan disimpan dalam *database* dan menampilkan pesan "Ubah Data Bagian Telah Berhasil."	Dapat mengisi *field* data yang akan diubah dengan benar dan data berhasil diubah dan disimpan dalam *database* dan menampilkan pesan sesuai dengan yang diharapkan.	Diterima
Kasus dan Hasil Uji (data salah)			
Data bagianyang baru	Proses ubah bagian gagal dilakukan	Proses ubah bagian gagal dilakukan	Diterima

Contoh : Nama Bagian : QA	karena data sudah terdaftar menampilkan peringatan “nama bagian sudah terdaftar”.	dikarenakan data yang diinputkan sudah terdaftar. Menampilkan peringatan “bagian sudah terdaftar”.	

7. Pengujian Hapus Bagian

Berikut ini dapat dilihat pengujian dari hapus bagian dapat dilihat pada tabel 4.9 berikut ini :

Tabel 4.9 Pengujian Hapus Bagian

Kasus dan Hasil Uji (data normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pilih Data bagian yang akan dihapus. Contoh : Kode bagian : 114 Nama bagian : Bagian Produksi Vaksin Campak	Tercantum data bagianyang akan dihapus berupa kode bagian dan nama bagian, setelah memilih ikon hapus maka tampil pesan “ Apakah anda yakin akan menghapus bagian ini ?” dan apabila memilik OK maka data bagian berhasil dihapus dan jika Cancel maka tidak akan menghapus data bagian.	Data bagianyang akan dihapus berada di daftar bagian dan apabila menekan ikon hapus maka akan tampil pesan yang diharapkan dan jika memilih OK maka data bagian berhasil dihapus dan jika memilih Cancel maka tidak akan menghapus data bagian.	Diterima

8. Pengujian Tambah Kelompok Alat

Berikut ini dapat dilihat pengujian dari tambah kelompok alat dapat dilihat pada tabel 4.10 berikut ini :

Tabel 4.10 Pengujian Tambah Kelompok Alat

Kasus dan Hasil Uji (data normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tambahkelompok alat	Menampilkan *form* tambah kelompok alat	Dapat menampilkan *form* kelompok alat	Diterima

Mengisi data kelompok alat Contoh : Nama Kelompok Alat : Thermometer Gauge (TID).	Tercantum pada masing-masing *field* kemudian data akan tersimpan didalam *database* dan menampilkan pesan "Penambahan kelompok alat baru telah berhasil" serta data kelompok alat yang baru ditambahkan tercantum juga di form tambah surat, tambah laporan initial, tambah laporan re-kalibrasi.	Dapat mengisi masing-masing *field* dan data berhasil disimpan ke dalam *database* dan menampilkan pesan yang diharapkan serta data kelompok alat yang baru tampil di form tambah surat, tambah laporan initial, tambah laporan re-kalibrasi.	Diterima
Kasus dan Hasil Uji (data salah)			
Mengisi data kelompok alat Contoh : Kode Bagian : 111 Nama Bagian : Bagian Produksi Vaksin Campak	Data kelompok alat tidak tersimpan di dalam *database* dan menampilkan pesan "Nama kelompok alat sudah terdaftar"	Data kelompok alat tidak tersimpan di dalam *database* dan menampilkan pesan kesalahan sesuai yang diharapkan	Diterima

9. Pengujian Ubah Kelompok Alat

Berikut ini dapat dilihat pengujian dari ubah kelompok alat dapat dilihat pada tabel 4.11 berikut ini :

Tabel 4.11 Pengujian Ubah Kelompok Alat

Kasus dan Hasil Uji (data normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Ubahkelompok alat	Menampilkan data kelompok alat	Dapat menampilkan data kelompok alat	Diterima
Data kelompok alat yanglama Contoh : Nama Kelompok Alat : TID	Tercantum data bagian berupa nama kelompok alatdan nama kelompok alat yang lama pada text *field* kelompok alat.	Dapat menampilkan *field* data kelompok alat yang akan diubah dengan benar	Diterima
Data kelompok alatyang baru Contoh :	Dapat mengisi *field* data yang akan diubah dengan benar	Dapat mengisi *field* data yang akan diubah dengan benar	Diterima

Nama Kelompok Alat : *Thermometer Gauge* (TID).	dan data berhasil diubah dan disimpan dalam *database* dan menampilkan pesan "Ubah Data Kelompok Alat Telah Berhasil."	dan data berhasil diubah dan disimpan dalam *database* dan menampilkan pesan sesuai dengan yang diharapkan.	
Kasus dan Hasil Uji (data salah)			
Data kelompok alatyang baru Contoh : Nama Kelompok Alat : *Thermometer Gauge* (TID).	Proses ubah kelompok alat gagal dilakukan karena data sudah terdaftar menampilkan peringatan "nama kelompok alat sudah terdaftar".	Proses ubah kelompok alat gagal dilakukan dikarenakan data yang diinputkan sudah terdaftar. Menampilkan peringatan "nama kelompok alat sudah terdaftar".	Diterima

10. Pengujian Hapus Kelompok Alat

Berikut ini dapat dilihat pengujian dari hapus kelompok alat dapat dilihat pada tabel 4.12 berikut ini :

Tabel 4.12 Pengujian Hapus Kelompok Alat

Kasus dan Hasil Uji (data normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pilih Data kelompok alat yang akan dihapus. Contoh : Nama Kelompok Alat : Bagian QA	Tercantum data kelompok alatyang akan dihapus berupa nama kelompok alat, setelah memilih ikon hapus maka tampil pesan " Apakah anda yakin akan menghapus kelompok ini ?" dan apabila memilik OK maka data kelompok alat berhasil dihapus dan jika Cancel maka tidak akan menghapus data kelompok alat.	Data kelompok alatyang akan dihapus berada di daftar kelompok alat dan apabila menekan ikon hapus maka akan tampil pesan yang diharapkan dan jika memilih OK maka data kelompok alat berhasil dihapus dan jika memilih Cancel maka tidak akan menghapus data kelompok alat.	Diterima

11. Pengujian Tambah *Kalibrator*

Berikut ini dapat dilihat pengujian dari tambah *kalibrator* dapat dilihat pada tabel 4.13 berikut ini :

Tabel 4.13 Pengujian Tambah *Kalibrator*

Kasus dan Hasil Uji (data normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tambah *kalibrator*	Menampilkan *form* tambah *kalibrator*	Dapat menampilkan *formkalibrator*	Diterima
Mengisi data *kalibrator* Contoh : Kode : 232-Mcal-01 Nama : Multifunction Calibrator Merk : Beamex No Seri : 1234. Kelompok : Kalibrator *Expire Date* : 07/19/2012	Tercantum pada masing-masing *field* kemudian data akan tersimpan didalam *database* dan menampilkan pesan "Penambahan *kalibrator* baru telah berhasil" serta data *kalibrator* yang baru ditambahkan tercantum juga di form tambah surat, tambah laporan initial, tambah laporan re-kalibrasi.	Dapat mengisi masing-masing *field* dan data berhasil disimpan ke dalam *database* dan menampilkan pesan yang diharapkan serta data *kalibrator* yang baru tampil di form tambah surat, tambah laporan initial, tambah laporan re-kalibrasi,	Diterima
Kasus dan Hasil Uji (data salah)			
Mengisi data *kalibrator* Contoh : Kode : 232-Mcal-01 Nama : Multifunction Calibrator Merk : Beamex No Seri : 1234. Kelompok : Kalibrator *Expire Date* : 07/19/2012	Data *kalibrator* tidak tersimpan di dalam *database* dan menampilkan pesan "Kode *kalibrator* sudah terdaftar"	Data *kalibrator* tidak tersimpan di dalam *database* dan menampilkan pesan kesalahan sesuai yang diharapkan	Diterima

12. Pengujian Ubah *Kalibrator*

Berikut ini dapat dilihat pengujian dari ubah *kalibrator* dapat dilihat pada tabel 4.14 berikut ini :

Tabel 4.14 Pengujian Ubah *Kalibrator*

Kasus dan Hasil Uji (data normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Ubah*kalibrator*	Menampilkan data*kalibrator*	Dapat menampilkan data *kalibrator*	Diterima
Mengisi data *kalibrator* Contoh : Kode : 232-Mcal-01 Nama : Multifunction Calibrator Merk : Beamex No Seri : 1234. Kelompok : Kalibrator *Expire Date* : 07/19/2012	Tercantum pada masing-masing *field* kemudian data akan tersimpan didalam *database* dan menampilkan pesan "Ubah *kalibrator* baru telah berhasil" serta data *kalibrator* yang baru ditambahkan tercantum juga di form tambah surat, tambah laporan initial, tambah laporan re-kalibrasi.	Dapat menampilkan *field* data kelompok alat yang akan diubah dengan benar	Diterima
Data *kalibrator*yang baru Contoh : Kode : 232-Mcal-02 Nama : Multifunction Calibrator Merk : Beamex No Seri : 1234. Kelompok : Kalibrator *Expire Date* : 07/19/2012	Dapat mengisi *field* data yang akan diubah dengan benar dan data berhasil diubah dan disimpan dalam *database* dan menampilkan pesan "Ubah Data Kalibrator Telah Berhasil."	Dapat mengisi *field* data yang akan diubah dengan benar dan data berhasil diubah dan disimpan dalam *database* dan menampilkan pesan sesuai dengan yang diharapkan.	Diterima
Kasus dan Hasil Uji (data salah)			
Data *kalibrator*yang baru Contoh : Kode : 232-Mcal-	Proses ubah *kalibrator* gagal dilakukan karena data sudah terdaftar menampilkan	Proses ubah *kalibrator* gagal dilakukan dikarenakan data yang diinputkan	Diterima

02 Nama : Multifunction Calibrator Merk : Beamex No Seri : 1234. Kelompok : Kalibrator *Expire Date* : 07/19/2012	peringatan “Kode Alat Kalibrator sudah terdaftar”.	sudah terdaftar. Menampilkan peringatan “Kode Alat Kalibrator sudah terdaftar”.	

13. Pengujian Hapus *Kalibrator*

Berikut ini dapat dilihat pengujian dari hapus *kalibrator* dapat dilihat pada tabel 4.15 berikut ini :

Tabel 4.15 Pengujian Hapus *Kalibrator*

Kasus dan Hasil Uji (data normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pilih Data *kalibrator* yang akan dihapus. Contoh : Kode : 232-Mcal-02 Nama : Multifunction Calibrator Merk : Beamex No Seri : 1234. Kelompok : Kalibrator *Expire Date* : 07/19/2012	Tercantum data *kalibrator*yang akan dihapus berupa nama Kode alat, nama, merk, no seri, kelompok dan *expire date*. Setelah memilih ikon hapus maka tampil pesan “ Apakah anda yakin akan menghapus kalibrator ini ?” dan apabila memilik OK maka data *kalibrator* berhasil dihapus dan jika Cancel maka tidak akan menghapus data *kalibrator*.	Data *kalibrator*yang akan dihapus berada di daftar *kalibrator* dan apabila menekan ikon hapus maka akan tampil pesan yang diharapkan dan jika memilih OK maka data *kalibrator* berhasil dihapus dan jika memilih Cancel maka tidak akan menghapus data *kalibrator*.	Diterima

14. Pengujian Tambah Permohonan

Berikut ini dapat dilihat pengujian dari tambah permohonan dapat dilihat pada tabel 4.16 berikut ini :

Tabel 4.16 Pengujian Tambah Permohonan

Kasus dan Hasil Uji (data normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tambahsurat	Menampilkan *form* tambah surat	Dapat menampilkan *form* surat	Diterima
Mengisi data tambah surat Contoh : *Expire date*, Dari : 2012/07/11 Sampai : 2012/07/21 Kelompok Alat : *Thermometer Gauge* (TID) Catatan : (kosong) Mengisi data validasi : Contoh : Kode alat : 232-TID-01 Lokasi : Gedung Teknik Keterangan : (kosong)	Tercantum pada masing-masing *field* kemudian data akan di validasi dan menampilkan data sesuai dengan data yang telah diisikan pada *form* tambah surat pada *formvalidasi*. Tercantum pada masing-masing *field* kemudian data di *form* tambah surat dan di *form validasi* akan tersimpan didalam *database* dan menampilkan pesan "Permohonan Kalibrasi Berhasil Disimpan"serta tampil di list surat permohonan.	Dapat mengisi masing-masing *field* dan menampilkan data sesuai dengan data yang telah diisikan di *form* tambah surat dan menampilkannya di *formvalidasi.* Dapat mengisi masing-masing *field* dan data berhasil disimpan ke dalam *database* dan menampilkan pesan yang diharapkan serta data permohoan yang baru tampil di list surat permohonan	Diterima
Kasus dan Hasil Uji (data salah)			
Mengisi data surat Contoh : *Expire date*, Dari : (kosong) Sampai : (kosong) Kelompok Alat : *Thermometer Gauge* (TID) Catatan : (kosong)	Tidak tampil *form validasi* dikarenakan data *expire date* belum diisi dan menampilkan peringatan "isi isiian ini"	*Form validasi* tidak tampil dan menampilkan peringatan kesalahan sesuai yang diharapkan	Diterima

15. Pengujian Ubah Permohonan

Berikut ini dapat dilihat pengujian dari ubah permohonan dapat dilihat pada tabel 4.17 berikut ini :

Tabel 4.17 Pengujian Ubah Permohonan

Kasus dan Hasil Uji (data normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Ubahpermohonan	Menampilkan data permohonan	Dapat menampilkan data permohonan	Diterima
Memilih data permohonan yang akan diubah. Contoh : *Expire date*, Dari : 2012/07/11 Sampai : 2012/07/21 Catatan : (kosong) Mengisi data validasi : Contoh : Kode alat : 232-TID-01 Lokasi : Gedung Teknik Keterangan : (kosong)	Tercantum pada masing-masing *field*data yang akan diubah yang diambil dari *database* dan	Dapat menampilkan *field* data permohonan yang akan diubah dengan benar	Diterima
Data permohonanyang baru Contoh : *Expire date*, Dari : 2012/07/11 Sampai : 2012/07/21 Catatan : (kosong) Mengisi data validasi : Contoh : Kode alat : 232-TID-02 Lokasi : Gedung Teknik	Dapat mengubah *field* data permohonan yang dipilih dan diambil dari databasedan data berhasil diubah dan disimpan dalam *database* dan menampilkan pesan "Ubah Data Permohonan Telah Berhasil."	Dapat mengubah *field* data permohonan yang dipilih dan diambil dari database dan data berhasil diubah dan disimpan dalam *database*dan menampilkan pesan sesuai dengan yang diharapkan.	Diterima

Keterangan : (kosong)			
Kasus dan Hasil Uji (data salah)			
Data permohonanyang baru Contoh : *Expire date*, Dari : 2012/07/11 Sampai : 2012/07/21 Catatan : (kosong) Mengisi data validasi : Contoh : Kode alat : (kosong) Lokasi : Gedung Teknik Keterangan : (kosong)	Proses ubah permohonan gagal dilakukan karena kode alat kosong dan menampilkan peringatan “isi isiian ini”.	Proses ubah permohonan gagal dilakukan karena kode alat kosong dan menampilkan peringatan yang diharapkan.	Diterima

16. Pengujian Hapus Permohonan

Berikut ini dapat dilihat pengujian dari hapus permohonan dapat dilihat pada tabel 4.18 berikut ini :

Tabel 4.18 Pengujian Hapus Permohonan

Kasus dan Hasil Uji (data normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pilih Data permohonan yang akan dihapus. Contoh : No Permohonan : 232/Kalibrasi/01 Tgl Permohonan : 2012/06/16 No Penerimaan : 2012001 Tgl Penerimaan : 2012/06/16	Tercantum data permohonanyang akan dihapus berupa nama no permohonan, tgl permohonan, no pernerimaan dan tgl penerimaan. Setelah memilih ikon hapus maka tampil pesan “ Apakah anda yakin akan menghapus permohonan ini ?”	Data permohonanyang akan dihapus berada di daftar permohonan dan apabila menekan ikon hapus maka akan tampil pesan yang diharapkan dan jika memilih OK maka data permohonan berhasil dihapus dan jika memilih Cancel	Diterima

	dan apabila memilik OK maka data permohonanberhasil dihapus dan jika Cancel maka tidak akan menghapus data permohonan.	maka tidak akan menghapus data permohonan.	

17. Pengujian Tambah Laporan *Initial*

Berikut ini dapat dilihat pengujian dari tambah laporan *initial* dapat dilihat pada tabel 4.19 berikut ini :

Tabel 4.19 Pengujian Tambah Laporan *Initial*

Kasus dan Hasil Uji (data normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tambahlaporan *initial*	Menampilkan *form* tambah laporan *initial*	Dapat menampilkan *form* laporan *initial*	Diterima
Mengisi identifikasi pada data tambah laporan *initial* Contoh : Tanggal Kalibrasi : 2012-07-15 No Laporan : TID 201202 No ID : 125-TID-1 Bagian : Tetanus Kelompok Alat : Thermometer Gauge (TID) Lokasi : Koridor Tetanus Merk/Type : Wika No seri : 123AC23 Tipe Sensor : Dial Skala Divisi : 1	Proses *input* data selesai dilakukan. Setelah semua data terisi dengan benar kemudian tekan tombol simpan ke database dan muncul pesan "Menambah Laporan *Initial* Berhasil". Laporan *initial* tampil di list laporan, dan laporan re-kalibrasi.	Dapat mengisi masing-masing *field* dan data berhasil disimpan ke dalam *database* dan menampilkan pesan yang diharapkan serta data laporan *initial* tampil di list laporan, dan laporan re-kalibrasi.	Diterima

Range : 0 s/d 150 Digunakan : Killing Tank Memilih data kalibrator. Sumber Suhu 1 : No ID : 232-LBath-01 Thermometer Standar : No ID : 232-TIES-01 Data kalibrasi Test Point : 0-1-2-3-4 Sumber Suhu : 0.0-1.0-2.0-3.0-4.0 Thermometer Standar : 0.1234-0.9871-1.8976-3.1234-4.1423 Nilai Aktual : 0-2-1-3-4 Kelembaban : 67% Suhu : 26 ^{0}C Memilih Petugas : Fielka			
Kasus dan Hasil Uji (data salah)			
Mengisi identifikasi pada data tambah laporan *initial* Contoh : Tanggal Kalibrasi : 2012-07-15 No Laporan : TID 201202 No ID : 125-TID-1	Data laporan *initial* tidak tersimpan di dalam *database*dikarenakan bagian (kosong) dan menampilkan pesan "pilih bagian"	Data laporan *initial* tidak tersimpan di dalam *database* dan menampilkan pesan kesalahan sesuai yang diharapkan	Diterima

Bagian : (kosong) Kelompok Alat : Thermometer Gauge (TID) Lokasi : Koridor Tetanus Merk/Type : Wika No seri : 123AC23 Tipe Sensor : Dial Skala Divisi : 1 Range : 0 s/d 150 Digunakan : Killing Tank Memilih data kalibrator. Sumber Suhu 1 : No ID : 232-LBath-01 Thermometer Standar : No ID : 232-TIES-01 Data kalibrasi Test Point : 0-1-2-3-4 Sumber Suhu : 0.0-1.0-2.0-3.0-4.0 Thermometer Standar : 0.1234-0.9871-1.8976-3.1234-4.1423 Nilai Aktual : 0-2-1-3-4 Kelembaban : 67% Suhu :			

26 ^{0}C Memilih Petugas : Fielka			

18. Pengujian Tambah Laporan *Re-Kalibrasi*

Berikut ini dapat dilihat pengujian dari tambah laporan *re-kalibrasi* dapat dilihat pada tabel 4.20 berikut ini :

Tabel 4.20 Pengujian Tambah Laporan *Re-Kalibrasi*

Kasus dan Hasil Uji (data normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tambahlaporan *re-kalibrasi*	Menampilkan *form* tambah laporan *re-kalibrasi*	Dapat menampilkan *form* laporan *re-kalibrasi*	Diterima
Mengisi identifikasi pada data tambah laporan *initial* Contoh : Tanggal Kalibrasi : 2012-07-15 No Laporan : TID 201202 No ID : 125-TID-1 Bagian : Tetanus Kelompok Alat : Thermometer Gauge (TID) Lokasi : Koridor Tetanus Merk/Type : Wika No seri : 123AC23 Tipe Sensor : Dial Skala Divisi : 1 Range : 0 s/d 150 Digunakan :	Pilih bagian dan pilih no id alat yang akan dibuat laporan re-kalibrasi. Isikan data actual dan data petugas, prroses *input* data selesai dilakukan. Setelah semua data terisi dengan benar kemudian tekan tombol simpan ke database dan muncul pesan "Menambah Laporan *Re-Kalibrasi* Berhasil". Laporan *re-kalibrasi* tampil di list laporan.	Dapat menampilkan dan mengisi masing-masing *field*. Data berhasil disimpan ke dalam *database* dan menampilkan pesan yang diharapkan serta data laporan *re-kaliibrasi* yang baru tampil di list laporan.	Diterima

Killing Tank Memilih data kalibrator. Sumber Suhu 1 : No ID : 232-LBath-01 Thermometer Standar : No ID : 232-TIES-01 Data kalibrasi Test Point : 0-1-2-3-4 Sumber Suhu : 0.0-1.0-2.0-3.0-4.0 Thermometer Standar : 0.1234-0.9871-1.8976-3.1234-4.1423 Nilai Aktual : 0-2-1-3-4 Kelembaban : 67% Suhu : 26 ^{0}C Memilih Petugas : Fielka			
Kasus dan Hasil Uji (data salah)			
Mengisi identifikasi pada data tambah laporan *initial* Contoh : Tanggal Kalibrasi : 2012-07-15 No Laporan : TID 201202 No ID : 125-TID-1 Bagian : (kosong) Kelompok Alat :	Data laporan *initial* tidak tersimpan di dalam *database* dikarenakan bagian (kosong) dan menampilkan pesan “pilih bagian”	Data laporan *initial* tidak tersimpan di dalam *database* dan menampilkan pesan kesalahan sesuai yang diharapkan	Diterima

Thermometer Gauge (TID) Lokasi : Koridor Tetanus Merk/Type : Wika No seri : 123AC23 Tipe Sensor : Dial Skala Divisi : 1 Range : 0 s/d 150 Digunakan : Killing Tank Memilih data kalibrator. Sumber Suhu 1 : No ID : 232-LBath-01 Thermometer Standar : No ID : 232-TIES-01 Data kalibrasi Test Point : 0-1-2-3-4 Sumber Suhu : 0.0-1.0-2.0-3.0-4.0 Thermometer Standar : 0.1234-0.9871-1.8976-3.1234-4.1423 Nilai Aktual : 0-2-1-3-4 Kelembaban : 67% Suhu : 26 ^{0}C Memilih Petugas : Fielka			

19. Pengujian Ubah Laporan

Berikut ini dapat dilihat pengujian dari ubah laporandapat dilihat pada tabel 4.21 berikut ini :

Tabel 4.21 Pengujian Ubah Laporan

Kasus dan Hasil Uji (data normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Ubahlaporan	Menampilkan data laporan	Dapat menampilkan data laporan	Diterima
Mengisi identifikasi pada data tambah laporan *initial* Contoh : Tanggal Kalibrasi : 2012-07-15 No Laporan : TID 201202 No ID : 125-TID-1 Bagian : Tetanus Kelompok Alat : Thermometer Gauge (TID) Lokasi : Koridor Tetanus Merk/Type : Wika No seri : 123AC23 Tipe Sensor : Dial Skala Divisi : 1 Range : 0 s/d 150 Digunakan : Killing Tank Memilih data kalibrator. Sumber Suhu 1 : No ID : 232-LBath-01	Menampilkan data laporan yang akan diubah dengan benar.	Dapat menampilkan *field* data kelompok alat yang akan diubah dengan benar	Diterima

Thermometer Standar : No ID : 232-TIES-01 Data kalibrasi Test Point : 0-1-2-3-4 Sumber Suhu : 0.0-1.0-2.0-3.0-4.0 Thermometer Standar : 0.1234-0.9871-1.8976-3.1234-4.1423 Nilai Aktual : 0-2-1-3-4 Kelembaban : 67% Suhu : 26 ^{0}C Memilih Petugas : Fielka			
Mengisi identifikasi pada data tambah laporan *initial* Contoh : Tanggal Kalibrasi : 2012-07-15 No Laporan : TID 201202 No ID : 125-TID-1 Bagian : Tetanus Kelompok Alat : Thermometer Gauge (TID) Lokasi : Koridor Tetanus Merk/Type : Wika No seri : 123AC23	Dapat mengubah *field* data yang akan diubah dengan benar dan data berhasil diubah dan disimpan dalam *database* dan menampilkan pesan "Ubah Data Laporan Telah Berhasil."	Dapat mengisi *field* data yang akan diubah dengan benar dan data berhasil diubah dan disimpan dalam *database* dan menampilkan pesan sesuai dengan yang diharapkan.	Diterima

Tipe Sensor : Dial Skala Divisi : 1 Range : 0 s/d 150 Digunakan : Killing Tank Memilih data kalibrator. Sumber Suhu 1 : No ID : 232-LBath-01 Thermometer Standar : No ID : 232-TIES-01 Data kalibrasi Test Point : 0-1-2-3-4 Sumber Suhu : 0.0-1.0-2.0-3.0-4.0 Thermometer Standar : 0.1234-1.1121-1.8976-3.1234-4.1423 Nilai Aktual : 0-2-1-3-4 Kelembaban : 67% Suhu : 26 ^{0}C Memilih Petugas : Fielka			
Kasus dan Hasil Uji (data salah)			
Mengisi identifikasi pada data tambah laporan *initial* Contoh : Tanggal Kalibrasi : 2012-07-15	Proses ubah laporan gagal dilakukan karena salah satu kolom nilai actual (kosong) dan menampilkan peringatan “isi isiian	Proses ubah laporan gagal dilakukan dikarenakan salah satu kolom nilai actual (kosong) dan menampilkan peringatan sesuai	Diterima

No Laporan : TID 201202 No ID : 125-TID-1 Bagian : Tetanus Kelompok Alat : Thermometer Gauge (TID) Lokasi : Koridor Tetanus Merk/Type : Wika No seri : 123AC23 Tipe Sensor : Dial Skala Divisi : 1 Range : 0 s/d 150 Digunakan : Killing Tank Memilih data kalibrator. Sumber Suhu 1 : No ID : 232-LBath-01 Thermometer Standar : No ID : 232-TIES-01 Data kalibrasi Test Point : 0-1-2-3-4 Sumber Suhu : 0.0-1.0-2.0-3.0-4.0 Thermometer Standar : 0.1234-1.1121-1.8976-3.1234-4.1423 Nilai Aktual : 0-2-1-3-4	ini”.	yang diharapkan.	

Kelembaban : 67% Suhu : 26 ^{0}C Memilih Petugas : Fielka			

20. Pengujian Hapus Laporan

Berikut ini dapat dilihat pengujian dari hapus laporan dapat dilihat pada tabel 4.22 berikut ini :

Tabel 4.22 Pengujian Hapus Laporan

Kasus dan Hasil Uji (data normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pilih Data permohonan yang akan dihapus. Contoh : No Permohonan : 232/Kalibrasi/01 Tgl Permohonan : 2012/06/16 No Penerimaan : 2012001 Tgl Penerimaan : 2012/06/16	Tercantum data laporanyang akan dihapus berupa nama no laporan, jenis laporan, no ID, bagian, merk, tgl kalibrasi dan hasil. Setelah memilih ikon hapus maka tampil pesan " Apakah anda yakin akan menghapus laporan ini ?" dan apabila memilik OK maka data laporanberhasil dihapus dan jika Cancel maka tidak akan menghapus data laporan.	Data laporanyang akan dihapus berada di daftar laporan dan apabila menekan ikon hapus maka akan tampil pesan yang diharapkan dan jika memilih OK maka data laporan berhasil dihapus dan jika memilih Cancel maka tidak akan menghapus data laporan.	Diterima

21. Pengujian Buat Rekap Laporan

Berikut ini dapat dilihat pengujian dari buat rekap laporan dapat dilihat pada tabel 4.23 berikut ini :

Tabel 4.23 Pengujian Buat Rekap Laporan

Kasus dan Hasil Uji (data normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan

Pilih bagian dan waktu yang akan dibuat rekap laporan nya Contoh : Bagian : Bagian Kalibrasi dan Validasi Tgl : 2012/06/16 Sampai Tgl : 2012/06/16	Tercantum data bagian dan tanggalyang akan dibuat rekap laporan Setelah memilih tombol buat rekap laporan menampilkan data rekap laporan sesuai dengan bagian dan tanggal tang dipilih	Rekap laporan yang dibuat tampil sesuai yang diharapkan	Diterima

22. Pengujian Lihat Reminding

Berikut ini dapat dilihat pengujian dari lihat reminding dapat dilihat pada tabel 4.24 berikut ini :

Tabel 4.24 Pengujian Lihat Reminding

Kasus dan Hasil Uji (data normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pilih bulan reminding yang akan dilihat Contoh : Pilih Bulan Reminding : Juli	Tercantum data alat yang masa berlaku pada bulan yang dipih dan muncul tombol simpan reminding ke ms.excel	Reminding yang ingin dilihat tampil sesuai dengan data yang diminta dan tombol simpan reminding tampil sesuai yang diharapkan	Diterima

23. Pengujian Ganti *Password*

Berikut ini dapat dilihat pengujian dari ganti *password* dapat dilihat pada tabel 4.25 berikut ini :

Tabel 4.25 Pengujian Ganti *Password*

Kasus dan Hasil Uji (data normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Masukkan data ganti password Contoh : Username : Admin Password Lama : 1234	Dapat mengisikan setiap field data yang akan diubah dan setelah menekan tombol simpan maka data yang baru tersimpan ke	Dapat mengisikan setiap *field* data yang akan diubah dan data berhasil ke *database* serta menampilkan pesan yang diharapkan.	Diterima

Password Baru : admin Ketik Ulang Password Baru : admin klik tombol simpan	*database* dan muncul pesan "Password anda telah berhasil diubah. Gunakan Password yang baru untuk login ke sistem"		

4.2.1.2 Kesimpulan Hasil Pengujian *Alpha*

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.

4.3 Pengujian *Beta*

Pengujian *beta* merupakan pengujian langsung kepada pengguna untuk mencoba aplikasi yang baru dan mengisi kuisioner mengenai kepuasan pengguna. Dari kuisioner tersebut akan dilakukan perhitungan untuk dapat mengambil kesimpulan terhadap penilaian dari aplikasi yang baru dibuat. (Lembaran kuisioner dilampirkan).

Berdasarkan data hasil kuisioner, dapat dicari persentase masing-masing jawaban dengan menggunakan rumus:

$$Y = P/Q * 100\%$$

Keterangan :

P = Banyaknya jawaban responden tiap soal

Q = Jumlah responden

Y = Nilai Persentase

Pengujian dilakukan dengan cara memberikan pertanyaan dalam bentuk kuisioner kepada enam orang user yang berada di Bio Farma Bandung yaitu 23 (Dua Puluh Tiga) orang karyawan bagian produksi, 2(Dua) orang karyawan bagian QA dan 3(Tiga) orang karyawan bagian Kalibrasi dan Validasi.

4.3.1 Kuisioner Terhadapa Kepuasan Pengguna

4.3.1.1 Kuisioner Untuk *Administrator*

1. Pertanyaan no 1 : “Menurut anda, apakah bahasa pemograman yang dipakai untuk sistem informasi ini anda menguasainya ? “.

Tabel 4.26 Pengujian Kuisioner No 1 Untuk *Administrator*

No	**Keterangan**	**Responden**	**Persentase (%)**
1	Sangat Menguasai	0	0
2	Menguasai	0	0
3	Cukup Menguasai	5	100
4	Tidak Menguasai	0	0

2. Pertanyaan no 2 : “Apakah anda mengerti struktur data dalam sistem informasi ini? “.

Tabel 4.27 Pengujian Kuisioner No 2 Untuk *Administrator*

No	**Keterangan**	**Responden**	**Persentase (%)**
1	Sangat Mengerti	1	20
2	Mengerti	4	80
3	Cukup Mengerti	0	0
4	Tidak Mengerti	0	0

3. Pertanyaan no 3 : “ Apakah database yang digunakan untuk sistem informasi ini, anda menguasainya ? “.

Tabel 4.28 Pengujian Kuisioner No 3 Untuk *Administrator*

No	**Keterangan**	**Responden**	**Persentase (%)**
1	Sangat Menguasai	0	0
2	Menguasai	2	40

3	Cukup Menguasai	3	60
4	Tidak Menguasai	0	0

4. Pertanyaan no 4 : " Apakah sistem informasi ini dapat dikembangkan lagi untuk kedepannya ? ".

Tabel 4.29Pengujian Kuisioner No 4 Untuk *Administrator*

No	**Keterangan**	**Responden**	**Persentase (%)**
1	Sangat Bisa	2	40
2	Bisa	3	60
3	Cukup Bisa	0	0
4	Tidak Bisa	0	0

5. Pertanyaan no 5 : " Apakah fungsi – fungsi pada sistem informasi ini sesuai dengan kegunaannya ? ".

Tabel4.30 Pengujian Kuisioner No 5 Untuk *Administrator*

No	**Keterangan**	**Responden**	**Persentase (%)**
1	Sesuai	4	80
2	Cukup Sesuai	1	20
3	Kurang Sesuai	0	0

4.3.1.2 Kuisioner Untuk *Validasi dan Kalibrasi, Quality Assurancce* dan Pemilik Alat

1. Pertanyaan no.1 : "Menurut Anda, Apakah tampilan (antarmuka) sistem informasi ini terlihat menarik?"

Tabel 4.31 Pengujian Kuisioner Pertanyaan No. 1

No	**Keterangan**	**Responden**	**Persentase (%)**
1	Sangat Menarik	5	20

2	Menarik	15	60
3	Cukup Menarik	5	20
4	Biasa	0	0
5	Kurang Menarik	0	0
6	Tidak menarik	0	0
7	Sangat Tidak Menarik	0	0

Berdasarkan hasil persentase pada Tabel 4.31 maka dapat disimpulkan bahwa 20% responden menyatakan tampilan sistem informasi ini sangat menarik, 60% responden menyatakan tampilan sistem informasi ini menarik dan 20% responden menyatakan tampilan sistem informasi ini cukup menarik.

2. Pertanyaan no.2 : "Menurut Anda, Apakah sistem informasi ini dapat membantu mempercepat proses kalibrasi?"

Tabel 4.32 Pengujian Kuisioner Pertanyaan No. 2

No	**Keterangan**	**Responden**	**Persentase (%)**
1	Sangat Membantu	8	32
2	Membantu	10	40
3	Cukup Membantu	7	28
4	Biasa	0	0
5	Kurang Membantu	0	0
6	Tidak Membantu	0	0
7	Sangat Tidak Membantu	0	0

Berdasarkan hasil persentase pada Tabel 4.32 maka dapat disimpulkan bahwa 32% responden menyatakan sistem informasi ini sangat membantu mempercepat proses kalibrasi d,40%responden menyatakan aplikasi ini membantu mempercepat proses kalibrasi dan 28%responden menyatakan aplikasi ini cukup membantu mempercepat proses kalibrasi

3. Pertanyaan no.3 : “Menurut Anda, apakah fungsionalitas sistem informasi ini memenuhi kebutuhan?”

Tabel 4.33 Pengujian Kuisioner Pertanyaan No. 3

No	Keterangan	Responden	Persentase (%)
1	Sangat Memenuhi	8	32
2	Memenuhi	6	24
3	Cukup Memenuhi	11	44
4	Biasa	0	0
5	Kurang Memenuhi	0	0
6	Tidak Memenuhi	0	0
7	Sangat Tidak Memenuhi	0	0

Berdasarkan hasil persentase pada Tabel 4.33 maka dapat disimpulkanbahwa 32% responden menyatakan fungsionalitas sistem informasi ini sangat memenuhi kebutuhan,24% responden menyatakan fungsionalitas aplikasi ini memenuhi kebutuhan dan 44% responden menyatakan fungsionalitas aplikasi ini cukup memenuhi kebutuhan

4. Pertanyaan no.4 : “Menurut Anda, Apakah sistem informasi ini mudah digunakan?”

Tabel 4.34 Pengujian Kuisioner Pertanyaan No. 4

No	Keterangan	Responden	Persentase (%)
1	Sangat Mudah	0	0
2	Mudah	10	40
3	Cukup Mudah	15	60
4	Biasa	0	0
5	Kurang Mudah	0	0
6	Tidak Mudah	0	0

7	Sangat Tidak Mudah	0	0

Berdasarkan hasil persentase pada Tabel 4.34 maka dapat disimpulkanbahwa 40 % responden menyatakan sistem informasi ini mudah digunakan dan 60% responden menyatakan sistem informasi inicukup mudah digunakan

5. Pertanyaan no.5 : “Menurut Anda, apakah sistem informasi ini bermanfaat?”

Tabel 4.35 Pengujian Kuisioner Pertanyaan No. 5

No	**Keterangan**	**Responden**	**Persentase (%)**
1	Sangat Bermanfaat	8	32
2	Bermanfaat	11	44
3	Cukup Bermanfaat	6	24
4	Biasa	0	0
5	Kurang Bermanfaat	0	0
6	Tidak Bermanfaat	0	0
7	Sangat Tidak Bermanfaat	0	0

Berdasarkan hasil persentase pada Tabel 4.35 maka dapat disimpulkanbahwa 32% responden menyatakan sistem informasi ini sangat bermanfaat, 44% responden menyatakan sistem informasi ini bermanfaat dan 24% responden menyatakan sistem informasi ini sangat bermanfaat.

4.3.2 Kesimpulan Hasil Pengujian Beta

Dari hasil pengujian *beta* yang telah dilakukan yaitu dengan pengujian perhitungan pilihan kategori jawaban dari kuisioner yang telah dibagikan di lapangan, maka dapat disimpulkan bahwa Sistem InformasiKalibrasi ini memudahkan *validasi dan kalibrasi* dalam proses pembuatan laporan, merekap laporan dan penyimpanan laporan kalibrasimenjadi lebih cepat dan

efisien.Sistem informasi ini juga memudahkan pemilik alat dalam melakukan membuat permohonan kalibrasi serta pemilik alat dan *quality assurance* bisa lebih cepat mendapat informasi tentang alat yang sedang dalam proses kalibrasi.