BAB IV

IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi Sistem

Pada bab ini akan dilakukan implementasi dan pengujian terhadap web yang telah dirancang. Tahapan ini dilakukan setelah perancangan selesai dilakukan dan selanjutnya akan diimplementasikan. Tujuan Implementasi adalah untuk menerapkan hasil perancangan agar maksud dan tujuan pembuatan aplikasi dapat tercapai. Setelah implementasi dilakukan, maka tahap pengujian terhadap pembuatan aplikasi dalam seksi pengolahan data dan informasi dilakukan, guna mengetahui apakah maksud dan tujuan yang ingin dicapai telah terpenuhi sehingga dapat ditarik kesimpulan

4.1.1 Implementasi Perangkat Keras

Spesifikasi perangkat keras minimal yang dibutuhkan untuk mengimplementasikan perangkat keras antara lain terdapat pada tabel 4.1 sebagai berikut.

Tabel 4. 1 Perangkat Keras yang dibutuhkan

No	Perangkat Keras	Spesifikasi
1	Processor	Intel Core 2 Duo 3.00 GHz
2	Monitor	Montor LCD 17 inch
3	VGA	VGA On-Board 224 MB
4	Memori	1024 MB
5	Optical Drive	DVD – ROM
6	Keyboard	Genius
7	Mouse	Genius
8	Printer	Printer hitam-putih

4.1.2 Implementasi Perangkat Lunak

Spesifikasi perangkat lunak yang dibutuhkan untuk mengimplementasikan perangkat lunak antara lain terdapat pada tabel 4.2 sebagai berikut :

Tabel 4. 2 Perangkat Lunak yang dibutuhkan

No	Perangkat Keras	Spesifikasi
1	Processor	Intel Dual Core2.0 GHz
2	Monitor	Montor 15 inch
3	VGA	VGA On-Board 128 MB
4	Memori	1024 MB
5	Optical Drive	DVD-ROM
6	Keyboard	K-Zone
7	Mouse	K-Zone
8	Printer	Printer hitam-putih

4.1.3 Implementasi Basis Data

Pembuatan basis data dilakukan dengan menggunakan aplikasi pemrograman *MySQL*. Implementasi basis data yang ada yaitu :

1. Implementasi Database *"troubleticket"*

	Tabel	Aksi	Catatan	Jenis	Penyortiran	Ukuran	Kelebihan (Overhead)
☐	customer		5	InnoDB	latin1_swedish_ci	48,0 KB	-
☐	departement		4	InnoDB	latin1_swedish_ci	16,0 KB	-
☐	penanganan		6	InnoDB	latin1_swedish_ci	48,0 KB	-
☐	pengguna		5	InnoDB	latin1_swedish_ci	32,0 KB	-
☐	problem		8	InnoDB	latin1_swedish_ci	16,0 KB	-
☐	ticket		12	InnoDB	latin1_swedish_ci	112,0 KB	-
	tabel 6	Jumlah	40	MyISAM	latin1_swedish_ci	272,0 KB	0 Bytes

2. Implementasi table *customer*

	Field	Jenis	Penyortiran	Atribut	Kosong	Default	Ekstra
☐	no	int(11)			Tidak	*None*	auto_increment
☐	id_customer	varchar(20)	latin1_swedish_ci		Tidak	*None*	
☐	nama_depan	varchar(20)	latin1_swedish_ci		Tidak	*None*	
☐	nama_akhir	varchar(20)	latin1_swedish_ci		Tidak	*None*	
☐	no_tlp	varchar(20)	latin1_swedish_ci		Tidak	*None*	
☐	email	varchar(255)	latin1_swedish_ci		Tidak	*None*	
☐	id_departement	int(11)			Tidak	*None*	

```
--
-- Struktur dari tabel `customer`
--

CREATE TABLE IF NOT EXISTS `customer` (
  `no` int(11) NOT NULL AUTO_INCREMENT,
  `id_customer` varchar(20) NOT NULL,
  `nama_depan` varchar(20) NOT NULL,
  `nama_akhir` varchar(20) NOT NULL,
  `no_tlp` varchar(20) NOT NULL,
  `email` varchar(255) NOT NULL,
  `id_departement` int(11) NOT NULL,
  PRIMARY KEY (`id_customer`),
  UNIQUE KEY `no` (`no`),
  KEY `id_departement` (`id_departement`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=10 ;

--
-- Constraints for table `customer`
--
ALTER TABLE `customer`
  ADD CONSTRAINT `customer_ibfk_1` FOREIGN KEY
(`id_departement`) REFERENCES `departement` (`id_departement`)
ON DELETE CASCADE ON UPDATE CASCADE;
```

3. implementasi table departement

	Field	Jenis	Penyortiran	Atribut	Kosong	Default	Ekstra
	id_departement	int(11)			Tidak	*None*	auto_increment
	nama_departement	varchar(55)	latin1_swedish_ci		Tidak	*None*	
	nama_gedung	varchar(55)	latin1_swedish_ci		Tidak	*None*	
	ruangan	varchar(55)	latin1_swedish_ci		Tidak	*None*	
	no_tlp_internal	varchar(20)	latin1_swedish_ci		Tidak	*None*	

```
--
-- Struktur dari tabel `departement`
--
CREATE TABLE IF NOT EXISTS `departement` (
  `id_departement` int(11) NOT NULL AUTO_INCREMENT,
  `nama_departement` varchar(55) NOT NULL,
  `nama_gedung` varchar(55) NOT NULL,
  `ruangan` varchar(55) NOT NULL,
  `no_tlp_internal` varchar(20) NOT NULL,
  PRIMARY KEY (`id_departement`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=5 ;
```

4. Implementasi tabel penanganan

	Field	Type	Collation	Attributes	Null	Default	Extra
	id_penanganan	int(11)			No	*None*	auto_increment
	id_ticket	varchar(20)	latin1_swedish_ci		No	*None*	
	id_problem	int(11)			No	*None*	
	id_pengguna	varchar(20)	latin1_swedish_ci		No	*None*	
	tindakan	text	latin1_swedish_ci		No	*None*	
	hasil_penanganan	enum('selesai','belum selesai','sedang dilakukan')	latin1_swedish_ci		No	*None*	

```
--
-- Struktur dari tabel `penanganan`
--
CREATE TABLE IF NOT EXISTS `penanganan` (
  `id_penanganan` int(11) NOT NULL AUTO_INCREMENT,
  `id_ticket` varchar(20) NOT NULL,
  `id_problem` int(11) NOT NULL,
  `id_pengguna` varchar(20) NOT NULL,
  `tindakan` text NOT NULL,
  `hasil_penanganan` enum('selesai','belum selesai','sedang
dilakukan') NOT NULL,
  PRIMARY KEY (`id_penanganan`),
  KEY `id_problem` (`id_problem`),
  KEY `id_pengguna` (`id_pengguna`),
  KEY `id_ticket` (`id_ticket`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=16 ;
--
-- Constraints for table `penanganan`
--
ALTER TABLE `penanganan`
  ADD CONSTRAINT `penanganan_ibfk_1` FOREIGN KEY (`id_problem`)
REFERENCES `problem` (`id_problem`) ON DELETE CASCADE ON UPDATE
CASCADE,
  ADD CONSTRAINT `penanganan_ibfk_2` FOREIGN KEY
(`id_pengguna`) REFERENCES `pengguna` (`id_pengguna`) ON DELETE
CASCADE ON UPDATE CASCADE,
  ADD CONSTRAINT `penanganan_ibfk_3` FOREIGN KEY (`id_ticket`)
REFERENCES `ticket` (`id_ticket`) ON DELETE CASCADE ON UPDATE
CASCADE;
```

5. Implementasi tabel pengguna

	Field	Type	Collation	Attributes	Null	Default	Extra
	no	int(11)			No	None	auto_increment
	id_pengguna	varchar(20)	latin1_swedish_ci		No	None	
	hak_akses	enum('helpdesk','teknisi','manager')	latin1_swedish_ci		No	None	
	password	varchar(255)	latin1_swedish_ci		No	None	
	nama_depan	varchar(20)	latin1_swedish_ci		No	None	
	nama_akhir	varchar(20)	latin1_swedish_ci		No	None	
	no_tlp	varchar(20)	latin1_swedish_ci		No	None	
	email	varchar(255)	latin1_swedish_ci		No	None	

```
--
-- Table structure for table `pengguna`
--
CREATE TABLE IF NOT EXISTS `pengguna` (
  `no` int(11) NOT NULL AUTO_INCREMENT,
  `id_pengguna` varchar(20) NOT NULL,
  `hak_akses` enum('helpdesk','teknisi','manager')
NOT NULL,
  `password` varchar(255) NOT NULL,
  `nama_depan` varchar(20) NOT NULL,
  `nama_akhir` varchar(20) NOT NULL,
  `no_tlp` varchar(20) NOT NULL,
  `email` varchar(255) NOT NULL,
  PRIMARY KEY (`id_pengguna`),
  UNIQUE KEY `no` (`no`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1
AUTO_INCREMENT=6 ;
```

6. Implementasi tabel problem

	Field	Type	Collation	Attributes	Null	Default	Extra
	id_problem	int(11)			No	None	auto_increment
	judul_problem	varchar(255)	latin1_swedish_ci		No	None	
	deskripsi_problem	text	latin1_swedish_ci		No	None	

```
--
-- Table structure for table `problem`
--
CREATE TABLE IF NOT EXISTS `problem` (
  `id_problem` int(11) NOT NULL AUTO_INCREMENT,
  `judul_problem` varchar(255) NOT NULL,
  `deskripsi_problem` text NOT NULL,
  PRIMARY KEY (`id_problem`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=9 ;
```

7. Implementasi tabel ticket

	Field	Type	Collation	Attributes	Null	Default	Extra
	id_ticket	varchar(20)	latin1_swedish_ci		No	None	
	tanggal_open	date			No	None	
	prioritas	enum('ya','tidak')	latin1_swedish_ci		No	None	
	status	enum('open','close')	latin1_swedish_ci		No	None	
	batas_waktu	int(11)			No	None	
	tanggal_close	date			No	None	
	keluhan	enum('telpon','email','fax')	latin1_swedish_ci		No	None	
	id_problem	int(11)			No	None	
	id_customer	varchar(20)	latin1_swedish_ci		No	None	
	id_pengguna	varchar(20)	latin1_swedish_ci		No	None	

```
--
-- Table structure for table `ticket`
--
CREATE TABLE IF NOT EXISTS `ticket` (
  `id_ticket` varchar(20) NOT NULL,
  `tanggal_open` date NOT NULL,
  `prioritas` enum('ya','tidak') NOT NULL,
  `status` enum('open','close') NOT NULL,
  `batas_waktu` int(11) NOT NULL,
  `tanggal_close` date NOT NULL,
  `keluhan` enum('telpon','email','fax') NOT NULL,
  `id_problem` int(11) NOT NULL,
  `id_customer` varchar(20) NOT NULL,
  `id_pengguna` varchar(20) NOT NULL,
  PRIMARY KEY (`id_ticket`),
  KEY `id_problem` (`id_problem`),
  KEY `id_customer` (`id_customer`),
  KEY `id_pengguna` (`id_pengguna`),
  KEY `tanggal_close` (`tanggal_close`),
  KEY `id_helpdesk` (`id_pengguna`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--
-- Constraints for table `ticket`
--
ALTER TABLE `ticket`
  ADD CONSTRAINT `ticket_ibfk_7` FOREIGN KEY
(`id_pengguna`) REFERENCES `pengguna` (`id_pengguna`) ON
DELETE CASCADE ON UPDATE CASCADE,
  ADD CONSTRAINT `ticket_ibfk_8` FOREIGN KEY
(`id_customer`) REFERENCES `customer` (`id_customer`) ON
DELETE CASCADE ON UPDATE CASCADE,
  ADD CONSTRAINT `ticket_ibfk_9` FOREIGN KEY
(`id_problem`) REFERENCES `problem` (`id_problem`);
```

8. Implementasi tabel keluhan

	Field	Type	Collation	Attributes	Null	Default	Extra
	id_keluhan	int(11)			No	None	auto_increment
	id_customer	varchar(20)	latin1_swedish_ci		No	None	
	prioritas	varchar(5)	latin1_swedish_ci		No	None	
	id_problem	int(11)			No	None	
	desk_keluhan	text	latin1_swedish_ci		No	None	
	tgl_masuk	date			No	None	

```
--

-- Table structure for table `keluhan`

--

CREATE TABLE IF NOT EXISTS `keluhan` (

  `id_keluhan` int(11) NOT NULL AUTO_INCREMENT,

  `id_customer` varchar(20) NOT NULL,

  `prioritas` varchar(5) NOT NULL,

  `id_problem` int(11) NOT NULL,

  `desk_keluhan` text NOT NULL,

  `tgl_masuk` date NOT NULL,

  PRIMARY KEY (`id_keluhan`),

  KEY `id_problem` (`id_problem`),

  KEY `id_customer` (`id_customer`)

) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=3
;

--

-- Constraints for table `keluhan`

--

ALTER TABLE `keluhan`

  ADD CONSTRAINT `keluhan_ibfk_1` FOREIGN KEY
(`id_customer`) REFERENCES `customer` (`id_customer`);
```

9. Implementasi tabel penugasan

	Field	Type	Collation	Attributes	Null	Default	Extra
	id_penugasan	int(11)			No	*None*	auto_increment
	id_ticket	varchar(20)	latin1_swedish_ci		No	*None*	
	id_teknisi	varchar(20)	latin1_swedish_ci		No	*None*	

```
--

-- Table structure for table `penugasan`

--

CREATE TABLE IF NOT EXISTS `penugasan` (

  `id_penugasan` int(11) NOT NULL AUTO_INCREMENT,

  `id_ticket` varchar(20) NOT NULL,

  `id_teknisi` varchar(20) NOT NULL,

  PRIMARY KEY (`id_penugasan`),

  KEY `id_ticket` (`id_ticket`),

  KEY `id_teknisi` (`id_teknisi`)

) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=22 ;

--

-- Constraints for table `penugasan`

--

ALTER TABLE `penugasan`

  ADD CONSTRAINT `penugasan_ibfk_3` FOREIGN KEY
(`id_ticket`) REFERENCES `ticket` (`id_ticket`) ON DELETE
CASCADE ON UPDATE CASCADE,

  ADD CONSTRAINT `penugasan_ibfk_4` FOREIGN KEY
(`id_teknisi`) REFERENCES `pengguna` (`id_pengguna`) ON
DELETE CASCADE ON UPDATE CASCADE;
```

4.1.4 Implementasi Antarmuka

Implementasi antarmuka dilakukan dengan setiap tampilan program yang dibangun dan pengkodeannya dalam bentuk *file* program. Berikut ini adalah implementasi antarmuka untuk pengguna. Adapun untuk tampilan implementasi memilki beberapa bagian tampilan menu dapat dilihat pada tabel berikut :

Tabel 4. 3 Implementasi Antarmuka Helpdesk

Menu	Deskripsi	Nama File
Login	Digunakan sebagai halaman login Helpdesk	index.php
Login Gagal	Digunakan sebagai halaman konfirmasi bila helpdesk gagal login	index.php
Customer	Digunakan sebagai halaman untuk pengolahan data customer	customer.php
Departement	Digunakan sebagai halaman untuk pengolahan data *departement*	departement.php
Tiket	Digunakan sebagai halaman untuk pengolahan data *tiket*	ticket.php
Gangguan	Digunakan sebagai halaman untuk pengolahan data gangguan	problem.php
Penanganan	Digunakan sebagai halaman untuk pengolahan data penanganan	penanganan.php
Logout	Digunakan oleh helpdesk untuk keluar dari aplikasi.	Logout.php

.

Tabel 4. 4 Implementasi Antarmuka Teknisi

Menu	Deskripsi	Nama File
Login	Digunakan sebagai halaman login Helpdesk	index.php
Login Gagal	Digunakan sebagai halaman konfirmasi bila helpdesk gagal login	index.php
View ticket	Digunakan sebagai halaman untuk melihat data *ticket*	detail_ticket.php
Edit ticket	Digunakan sebagai halaman untuk mengedit status *ticket*	edit_ticket.php
Pencarian *ticket*	Digunakan sebagai halaman untuk melakukan pencarian data *ticket*	pencarian_ticket.php
View penanganan	Digunakan sebagai halaman untuk melihat data penanganan	detail_penanganan.php

Edit penanganan	Digunakan sebagai halaman untuk mengedit data penanganan	edit_penanganan.php
Tambah penanganan	Digunakan sebagai halaman untuk menambah data penanganan	tambah_penanganan.php
Pencarian data penanganan	Digunakan sebagai halaman untuk pencarian data penanganan	pencarian_penanganan.php
Logout	Digunakan oleh teknisi untuk keluar dari aplikasi.	Logout.php

Tabel 4. 5 Implementasi Antarmuka Manager

Menu	Deskripsi	Nama File
Login	Digunakan sebagai halaman login Manager	index.php
Login Gagal	Digunakan sebagai halaman konfirmasi bila helpdesk gagal login	index.php
Pengguna	Digunakan sebagai halaman untuk pengolahan data pengguna	pengguna.php
View ticket	Digunakan sebagai halaman untuk melihat data *ticket*	detail_ticket.php
Pencarian *ticket*	Digunakan sebagai halaman untuk melakukan pencarian data *ticket*	pencarian_ticket.php
cetak	Digunakan untuk mencetak data ticket yang terdapat pada *view ticket*	cetak.php
laporan	Digunakan sebagai halaman untuk membuat laporan data ticket perbulan	view_laporan.php
grafik	Digunakan sebagai halaman untuk melihat data ticket lewat grafik	tampil_grafik.php
Logout	Digunakan oleh manager untuk keluar dari aplikasi.	Logout.php

Tabel 4. 6 Implementasi Antarmukan Customer

Menu	Deskripsi	Nama File
Form Keluhan	Digunakan sebagai halaman untuk mengisi form keluhan	Tambah_keluhan.php

4.2 Pengujian Perangkat Lunak

Pengujian sistem merupakan hal terpenting yang bertujuan untuk menemukan 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.

4.2.1 Pengujian Alpha

Pengujian alpha dilakukan dengan menggunakan metode *black box*. Untuk pengujian alpha ini yaitu pada pengujian sebagai pengguna.

4.2.2 Skenario Pengujian Alpha

Skenario pengujian alpha di Direktorat TIK UPI terdiri dari 4 pelaku sistem diantaranya teknisi, helpdesk , manager, dan customer.

Tabel 4. 7 Skenario Pengujian Aplikasi Helpdesk

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Isi data Login	*Black Box*
	Validasi *userid and password*	*Black Box*
Pengolahan Data Master	Pengolahan Data Customer	*Black Box*
	Pengolahan Data Departement	*Black Box*
	Pengolahan Data Tiket	*Black Box*
	Pengolahan Data Gangguan	*Black Box*
	Pengolahan Data Penanganan	*Black Box*
Pengolahan Data Customer	View Data Customer	*Black Box*
	Edit Data Customer	*Black Box*
	Hapus Data Customer	*Black Box*
	Tambah Data Customer	*Black Box*
	Pencarian Data Customer	*Black Box*
Pengolahan Data *Departement*	View Data Departement	*Black Box*
	Edit Data Departement	*Black Box*
	Hapus Data Departement	*Black Box*
	Tambah Data Departement	*Black Box*
	Pencarian Data Departement	*Black Box*
Pengolahan Data Tiket	View Data Tiket	*Black Box*
	Edit Data Tiket	*Black Box*
	Hapus Data Tiket	*Black Box*

	Tambah Data Tiket	*Black Box*
	Pencarian Data Tiket	*Black Box*
Pengolahan Data Gangguan	View Data Gangguan	*Black Box*
	Edit Data Gangguan	*Black Box*
	Hapus Data Gangguan	*Black Box*
	Tambah Data Gangguan	*Black Box*
	Pencarian Data Gangguan	*Black Box*
Pengolahan Data Penanganan	View Data Penanganan	*Black Box*
	Edit Data Penanganan	*Black Box*
	Hapus Data Penanganan	*Black Box*
	Tambah Data Penanganan	*Black Box*
	Pencarian Data Penanganan	*Black Box*
Logout	Menghapus *session* helpdesk	*Black Box*

Tabel 4. 8 Skenario Pengujian Aplikasi Teknisi

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Isi data Login	*Black Box*
	Validasi *userid and password*	*Black Box*
Tampil Data Ticket	View Data Ticket	*Black Box*
	Edit Status Ticket	*Black Box*
	Pencarian Data Ticket	*Black Box*
Tampil Data Penanganan	View Data Penanganan	*Black Box*
	Edit Data Penanganan	*Black Box*
	Tambah Data Penanganan	*Black Box*
	Pencarian Data Penanganan	*Black Box*
Logout	Menghapus *session* teknisi	*Black Box*

Tabel 4. 9 Skenario Pengujian Aplikasi Manager

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Isi data Login	*Black Box*
	Validasi *userid and password*	*Black Box*
Pengolahan Data Pengguna	View Data Pengguna	*Black Box*
	Edit Data Pengguna	*Black Box*
	Hapus Data Pengguna	*Black Box*
	Tambah Data Pengguna	*Black Box*
	Pencarian Data pengguna	*Black Box*
Tampil Data Ticket	View Data Ticket	*Black Box*
	Pencarian Data Ticket	*Black Box*
	Cetak Data Ticket	*Black Box*
Laporan	Laporan Data Ticket Perbulan	*Black Box*
Tampil Grafik	Grafik Data ticket	*Black Box*
Logout	Menghapus *session* manager	*Black Box*

Tabel 4. 10 Skenario Pengujian Aplikasi Customer

Kelas Uji	Butir Uji	Jenis Pengujian
Form keluhan	Input Data Keluhan	*Black Box*

4.2.3 Kasus Dan Hasil Pengujian

Butir pengujian dilakukan dengan menguji setiap proses *data flow diagram* dan kemungkinan kesalahan yang terjadi untuk setiap proses. Pengujian ini dilakukan secara *black box*, yaitu pengujian dilakukan dengan hanya memperhatikan masukan ke sistem dan keluaran ke sistem.

4.2.3.1 Pengujian Login

Berikut dapat dilihat pengujian login dapat dilihat pada tabel berikut:

Tabel 4. 11 Pengujian Login

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
User ID : H001 *Password* : Helpdesk	Tercantum pada text box *field User ID* dan *Password*	Dapat mengisi data login sesuai dengan yang diharapkan	[√] Diterima [ ] Ditolak
Data Login berupa *User ID* : H001 *Password* : helpdesk	Data login dicari dalam *database, userid* dan data *password* benar login valid dan dapat masuk ke menu sesuai hak akses	Tidak dapat login/masuk dan menampilkan pesan kesalahan sesuai yang diharapkan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji salah (Data Salah)			

User ID : H001 *Password* : 123	Tidak dapat login/masuk ke menu utama sesuai hak akses dan menampilkan pesan "Maaf anda tidak terdaftar"	Menampilkan pesan "Maaf anda tidak terdaftar"	[√] Diterima [ ] Ditolak

4.2.3.2 Pengujian Tambah Data Customer

Berikut dapat dilihat pengujian tambah data customer dapat dilihat pada tabel berikut:

Tabel 4. 12 Pengujian Tambah Data Customer

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah Customer	Menampilkan *form* tambah customer	Dapat menampilkan *form* tambah customer	[√] Diterima [ ] Ditolak
Pengisian data customer secara lengkap	Tercantum pada masing-masing field dan tersimpan kedalam database	Dapat mengisi masing-masing field dan tersimpan kedalam database	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji salah (Data Salah)			
Pengisian hanya pada salah satu field atau belum terisi	Data *customer* tidak tersimpan didalam *database* dan menampilkan pesan data field tidak boleh kosong	Data *customer* tidak tersimpan didalam *database* dan dapat menampilkan pesan "Data field tidak boleh kosong"	[√] Diterima [ ] Ditolak

4.2.3.3 Pengujian Edit Data Customer

Berikut dapat dilihat pengujian edit data customer dapat dilihat pada tabel berikut:

Tabel 4. 13 Pengujian Edit Data Customer

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Edit Customer	Menampilkan *form* edit customer	Dapat menampilkan *form edit* customer	[√] Diterima [ ] Ditolak
Pengisian data customer yang akan diubah dengan lengkap	Data yang lama akan berhasil diubah dengan yang baru	Pengubahan data customer berhasil dilakukan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji salah (Data Salah)			
Pengisian data customer yang akan diubah hanya pada salah satu field atau belum terisi	Data customer tidak dapat diubah dan disimpan dan akan tampil pesan	Dapat menampilkan pesan "Data field yang diubah harus sesuai dan tidak boleh boleh kosong"	[√] Diterima [ ] Ditolak

4.2.3.4 Pengujian Hapus Data Customer

Berikut dapat dilihat pengujian edit data customer dapat dilihat pada tabel berikut:

Tabel 4. 14 Pengujian Hapus Data Customer

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data *customer* yang akan dihapus	Tercantum data *customer* yang akan dihapus, setelah memilih	Menampilkan data *customer* yang akan dihapus, setelah memilih HAPUS	[√] Diterima [ ] Ditolak

	HAPUS maka tampil pesan konfirmasi dan apabila memilih OK maka data *customer* berhasil dihapus dan jika memilih Batal maka tidak akan menghapus data *customer*.	maka tampil pesan konfirmasi dan apabila memilih OK maka data *customer* berhasil dihapus dan jika memilih Batal maka tidak akan menghapus data *customer*.	

4.2.3.5 Pengujian Tambah Departemen

Tabel 4. 15 Pengujian Tambah Departemen

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah departmen	Menampilkan *form* tambah departmen	Dapat menampilkan *form* tambah departmen	[√] Diterima [ ] Ditolak
Pengisian data departmen secara lengkap	Tercantum pada masing-masing field dan tersimpan kedalam database	Dapat mengisi masing-masing field dan tersimpan kedalam database	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji salah (Data Salah)			
Pengisian hanya pada salah satu field atau belum terisi	Data departmen tidak tersimpan didalam *database* dan menampilkan pesan data field tidak boleh kosong	Data departmen tidak tersimpan didalam *database* dan dapat menampilkan pesan "Data field tidak boleh kosong"	[√] Diterima [ ] Ditolak

4.2.3.6 Pengujian Edit Data Departmen

Tabel 4. 16 Pengujian Edit Data Departmen

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Edit departmen	Menampilkan *form* edit departmen	Dapat menampilkan *form edit* departmen	[√] Diterima [ ] Ditolak
Pengisian data departmen yang akan diubah dengan lengkap	Data yang lama akan berhasil diubah dengan yang baru	Pengubahan data departmen berhasil dilakukan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji salah (Data Salah)			
Pengisian data departmen yang akan diubah hanya pada salah satu field atau belum terisi	Data departmen tidak dapat diubah dan disimpan dan akan tampil pesan	Dapat menampilkan pesan "Data field yang diubah harus sesuai dan tidak boleh boleh kosong"	[√] Diterima [ ] Ditolak

4.2.3.7 Pengujian Hapus Data Departmen

Tabel 4. 17 Pengujian Hapus Data Departmen

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data *departmen* yang akan dihapus	Tercantum data *departmen* yang akan dihapus, setelah memilih HAPUS maka tampil pesan konfirmasi dan apabila memilih OK maka data departmen berhasil dihapus dan jika memilih Batal maka tidak akan menghapus data *departmen*	Menampilkan data *departmen* yang akan dihapus, setelah memilih HAPUS maka tampil pesan konfirmasi dan apabila memilih OK maka data *departmen* berhasil dihapus dan jika memilih Batal maka tidak akan menghapus data *departmen* .	[√] Diterima [ ] Ditolak

4.2.3.8 Pengujian Tambah Data Ticket

Tabel 4. 18 Pengujian Tambah Data Ticket

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah ticket	Menampilkan *form* tambah ticket	Dapat menampilkan *form* tambah ticket	[√] Diterima [ ] Ditolak
Pengisian data departmen secara lengkap	Tercantum pada masing-masing field dan tersimpan kedalam database	Dapat mengisi masing-masing field dan tersimpan kedalam database	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji salah (Data Salah)			
Pengisian hanya pada salah satu field atau belum terisi	Data ticket tidak tersimpan didalam *database* dan menampilkan pesan data field tidak boleh kosong	Data ticket tidak tersimpan didalam *database* dan dapat menampilkan pesan "Data field tidak boleh kosong"	[√] Diterima [ ] Ditolak

4.2.3.9 Pengujian Edit Data Ticket

Tabel 4. 19 Pengujian Edit Data Ticket

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Edit ticket	Menampilkan *form* edit ticket	Dapat menampilkan *form edit* ticket	[√] Diterima [ ] Ditolak
Pengisian data ticket yang akan diubah dengan lengkap	Data yang lama akan berhasil diubah dengan yang baru	Pengubahan data ticket berhasil dilakukan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji salah (Data Salah)			

Pengisian data ticket yang akan diubah hanya pada salah satu field atau belum terisi	Data ticket tidak dapat diubah dan disimpan dan akan tampil pesan	Dapat menampilkan pesan "Data field yang diubah harus sesuai dan tidak boleh boleh kosong"	[√] Diterima [ ] Ditolak

4.2.3.10 Pengujian Hapus Data Ticket

Tabel 4. 20 Pengujian Hapus Data Ticket

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data *ticket* yang akan dihapus	Tercantum data *ticket* yang akan dihapus, setelah memilih HAPUS maka tampil pesan konfirmasi dan apabila memilih OK maka data *ticket* berhasil dihapus dan jika memilih Batal maka tidak akan menghapus data *ticket*	Menampilkan data *ticket* yang akan dihapus, setelah memilih HAPUS maka tampil pesan konfirmasi dan apabila memilih OK maka data *ticket* berhasil dihapus dan jika memilih Batal maka tidak akan menghapus data *ticket* .	[√] Diterima [ ] Ditolak

4.2.3.11 Pengujian Tambah Data Gangguan (*Problem*)

Tabel 4. 21 Pengujian Tambah Data Gangguan (*Problem*)

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah gangguan(*problem*)	Menampilkan *form* tambah gangguan(*problem*)	Dapat menampilkan *form* tambah gangguan(*problem*)	[√] Diterima [ ] Ditolak

Pengisian data gangguan(*problem*) secara lengkap	Tercantum pada masing-masing field dan tersimpan kedalam database	Dapat mengisi masing-masing field dan tersimpan kedalam database	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji salah (Data Salah)			
Pengisian hanya pada salah satu field atau belum terisi	Data gangguan(*problem*) tidak tersimpan didalam *database* dan menampilkan pesan data field tidak boleh kosong	Data gangguan(*problem*) tidak tersimpan didalam *database* dan dapat menampilkan pesan "Data field tidak boleh kosong"	[√] Diterima [ ] Ditolak

4.2.3.12 Pengujian Edit Data Gangguan (Problem)

Tabel 4. 22 Pengujian Edit Data Gangguan (*Problem*)

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Edit gangguan(*problem*)	Menampilkan *form* edit gangguan(*problem*)	Dapat menampilkan *form edit* gangguan(*problem*)	[√] Diterima [ ] Ditolak
Pengisian data gangguan(*problem*)yang akan diubah dengan lengkap	Data yang lama akan berhasil diubah dengan yang baru	Pengubahan data gangguan(*problem*) berhasil dilakukan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji salah (Data Salah)			
Pengisian data gangguan(*problem*) yang akan diubah hanya pada salah satu field atau belum terisi	Data gangguan(*problem*) tidak dapat diubah dan disimpan dan akan tampil pesan	Dapat menampilkan pesan "Data field yang diubah harus sesuai dan tidak boleh boleh kosong"	[√] Diterima [ ] Ditolak

4.2.3.13 Pengujian Hapus Data Gangguan (*Problem*)

Tabel 4. 23 Pengujian Hapus Data Gangguan (*Problem*)

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data gangguan(*problem)* yang akan dihapus	Tercantum data gangguan(*problem)* yang akan dihapus, setelah memilih HAPUS maka tampil pesan konfirmasi dan apabila memilih OK maka data gangguan(*problem)* berhasil dihapus dan jika memilih Batal maka tidak akan menghapus data gangguan(*problem)*	Menampilkan data gangguan(*problem)* yang akan dihapus, setelah memilih HAPUS maka tampil pesan konfirmasi dan apabila memilih OK maka data gangguan(*problem)* berhasil dihapus dan jika memilih Batal maka tidak akan menghapus data gangguan(*problem)* .	[√] Diterima [ ] Ditolak

4.2.3.14 Pengujian Tambah Data Penanganan

Tabel 4. 24 Pengujian Tambah Data Penanganan

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah Penanganan	Menampilkan *form* tambah Penanganan	Dapat menampilkan *form* tambah Penanganan	[√] Diterima [ ] Ditolak
Pengisian data Penanganan secara lengkap	Tercantum pada masing-masing field dan tersimpan kedalam database	Dapat mengisi masing-masing field dan tersimpan kedalam database	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji salah (Data Salah)			
Pengisian hanya pada salah satu field atau belum terisi	Data Penanganan tidak tersimpan didalam *database* dan menampilkan pesan data field tidak boleh kosong	Data Penanganan tidak tersimpan didalam *database* dan dapat menampilkan pesan "Data field tidak boleh kosong"	[√] Diterima [ ] Ditolak

4.2.3.15 Pengujian Edit Data Penanganan

Tabel 4. 25 Pengujian Edit Data Penanganan

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Edit Penanganan	Menampilkan *form* edit penanganan	Dapat menampilkan *form edit* penanganan	[√] Diterima [ ] Ditolak
Pengisian data penanganan yang akan diubah dengan lengkap	Data yang lama akan berhasil diubah dengan yang baru	Pengubahan data penanganan berhasil dilakukan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji salah (Data Salah)			
Pengisian data penanganan yang akan diubah hanya pada salah satu field atau belum terisi	Data penanganan tidak dapat diubah dan disimpan dan akan tampil pesan	Dapat menampilkan pesan "Data field yang diubah harus sesuai dan tidak boleh boleh kosong"	[√] Diterima [ ] Ditolak

4.2.3.16 Pengujian Hapus Data Penanganan

Tabel 4. 26 Pengujian Hapus Data Penanganan

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data penanganan yang akan dihapus	Tercantum data penanganan yang akan dihapus, setelah memilih HAPUS maka tampil pesan konfirmasi dan apabila memilih OK maka data penanganan berhasil dihapus dan jika memilih Batal maka tidak akan menghapus data penanganan	Menampilkan data penanganan yang akan dihapus, setelah memilih HAPUS maka tampil pesan konfirmasi dan apabila memilih OK maka data penanganan berhasil dihapus dan jika memilih Batal maka tidak akan menghapus data penanganan	[√] Diterima [ ] Ditolak

4.2.3.17 Pengujian Tambah Data Pengguna

Tabel 4. 27 Pengujian Tambah Data Pengguna

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah Pengguna	Menampilkan *form* tambah pengguna	Dapat menampilkan *form* tambah pengguna	[√] Diterima [ ] Ditolak
Pengisian data pengguna secara lengkap	Tercantum pada masing-masing field dan tersimpan kedalam database	Dapat mengisi masing-masing field dan tersimpan kedalam database	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji salah (Data Salah)			

Pengisian hanya pada salah satu field atau belum terisi	Data pengguna tidak tersimpan didalam *database* dan menampilkan pesan data field tidak boleh kosong	Data pengguna tidak tersimpan didalam *database* dan dapat menampilkan pesan "Data field tidak boleh kosong"	[√] Diterima [ ] Ditolak

4.2.3.18 Pengujian Edit Data Pengguna

Tabel 4. 28 Pengujian Edit Data Pengguna

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Edit Pengguna	Menampilkan *form* edit pengguna	Dapat menampilkan *form edit* pengguna	[√] Diterima [ ] Ditolak
Pengisian data pengguna yang akan diubah dengan lengkap	Data yang lama akan berhasil diubah dengan yang baru	Pengubahan data pengguna berhasil dilakukan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji salah (Data Salah)			
Pengisian data pengguna akan diubah hanya pada salah satu field atau belum terisi	Data pengguna tidak dapat diubah dan disimpan dan akan tampil pesan	Dapat menampilkan pesan "Data field yang diubah harus sesuai dan tidak boleh boleh kosong"	[√] Diterima [ ] Ditolak

4.2.3.19 Pengujian Hapus Data Pengguna

Tabel 4. 29 Pengujian Hapus Data Pengguna

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data pengguna yang akan dihapus	Tercantum data pengguna yang akan dihapus, setelah memilih HAPUS maka tampil pesan konfirmasi dan apabila memilih OK maka data pengguna berhasil dihapus dan jika memilih Batal maka tidak akan menghapus data pengguna .	Menampilkan data pengguna yang akan dihapus, setelah memilih HAPUS maka tampil pesan konfirmasi dan apabila memilih OK maka data pengguna berhasil dihapus dan jika memilih Batal maka tidak akan menghapus data pengguna .	[√] Diterima [ ] Ditolak

4.2.3.20 Pengujian Cetak Laporan

Tabel 4. 30 Pengujian Cetak Laporan

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Laporan per bulan Contoh : Bulan : Juni Tahun : 2011	Mencetak laporan data ticket pada bulan Juni dan tahun 2011 dengan format pdf	Dapat mencetak laporan sesuai yang diharapkan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji salah (Data Salah)			

Laporan per bulan Contoh : Bulan : Agustus Tahun : 2011	Gagal mencetak laporan data ticket dan menampilkan pesan peringatan bahwa tidak ada data di bulan agustus	Tidak dapat mencetak laporan data ticket sesuai yang diharapkan	[√] Diterima [ ] Ditolak

4.2.3.21 Pengujian Hapus Data Keluhan

Tabel 4. 31 Pengujian Hapus Data Keluhan

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data keluhan yang akan dihapus	Tercantum data keluhan yang akan dihapus, setelah memilih HAPUS maka tampil pesan konfirmasi dan apabila memilih OK maka data keluhan berhasil dihapus dan jika memilih Batal maka tidak akan menghapus data keluhan .	Menampilkan data keluhan yang akan dihapus, setelah memilih HAPUS maka tampil pesan konfirmasi dan apabila memilih OK maka data keluhan berhasil dihapus dan jika memilih Batal maka tidak akan menghapus data keluhan .	[√] Diterima [ ] Ditolak

4.2.4 Kesimpulan Hasil Pengujian Alpha

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

4.2.5 Pengujian Beta

Pengujian beta merupakan pengujian yang dilakukan secara objektif dimana diuji secara langsung ke lapangan yaitu instansi yang bersangkutan mengenai kepuasan user dengan kandungan poin yaitu pemenuhan kebutuhan dari tujuan awal pembangunan *trouble ticket system* dan tampilan antarmuka dari *trouble ticket system*. Pengujian beta dilakukan melalui dua teknik pengambilan data, yaitu melalui wawancara dan kuesioner.

Pengujian beta melalui wawancara dilakukan untuk halaman *helpdesk*, *teknisi* dan *manager* pada aplikasi *trouble ticket system*, karena terbatasnya jumlah pengguna untuk *helpdesk*, *teknisi* dan *manager*.

Pengujian beta melalui kuisioner dilakukan dengan membuat kuesioner mengenai kepuasan *customer* dengan kandungan poin syarat user friendly untuk selanjutnya dibagikan kepada sebagian *customer* dengan mengambil sample sebanyak 6 orang. Dari hasil kuesioner tersebut akan dilakukan perhitungan untuk dapat diambil kesimpulan terhadap penilaian penerapan sistem yang baru. Pengujian beta ini dilakukan dengan cara melakukan pengujian langsung di tempat penelitian.

4.2.5.1 Wawancara Pengguna

Wawancara dilakukan dengan menggunakan teknik kualitatif melalui pertanyaan terbuka. Dari hasil wawancara tersebut akan dilakukan pengambilan kesimpulan terhadap penilaian untuk penerapan aplikasi yang

baru. Wawancara terdiri dari 3 pertanyaan yang akan ditanyakan kepada narasumber. Berikut adalah wawancara yang dilakukan pada narasumber :

A. Wawancara Pengujian Beta Untuk Helpdesk

Wawancara dilakukan untuk pengujian beta terhadap aplikasi *trouble ticket system* kepada helpdesk dengan total narasumber sebanyak 1 orang. Berikut daftar pertanyaan yang diajukan kepada narasumber :

1. Menurut pendapat anda apakah dengan adanya halaman helpdesk dalam aplikasi trouble ticket system dapat membantu anda dalam pengelolaan keluhan gangguan layanan?
2. Menurut pendapat anda apakah dengan adanya halaman helpdesk dalam aplikasi trouble ticket system perkerjaan anda dalam melayani laporan gangguan dari customer lebih mudah?
3. Menurut pendapat Anda, apakah fungsionalitas dari halaman helpdesk ini mudah dipahami dan digunakan / *User friendly* ?

Berikut ini adalah masing-masing jawaban dari pertanyaan hasil wawancara tersebut.

1. Menurut pendapat anda apakah dengan adanya halaman helpdesk dalam aplikasi trouble ticket system dapat membantu anda dalam pengelolaan keluhan gangguan layanan?

   Jawaban :

Sudah dapat, karena dengan adanya halaman helpdesk dalam aplikasi trouble ticket system saya dapat dengan mudah mengelola keluhan-keluhan ketika ada gangguan layanan pada setiap departemen.

Berdasarkan hasil wawancara diatas maka dapat disimpulkan bahwa halaman helpdesk dalam aplikasi troubel ticket system sudah mampu memenuhi kebutuhan helpdesk dalam mengelolah keluhan-keluhan ketika ada gangguan layanan pada setiap departemen.

2. Menurut pendapat anda apakah dengan adanya halaman helpdesk dalam aplikasi trouble ticket system perkerjaan anda lebih mudah dalam melayani laporan gangguan dari customer ?

   Jawaban :

Sudah dapat, karena dengan adanya halaman helpdesk dalam aplikasi trouble ticket system perekerjaan saya lebih mudah melayani laporan gangguan dari customer.

Berdasarkan hasil wawancara diatas maka dapat disimpulkan bahwa halaman helpdesk dalam aplikasi troubel ticket system mempermudah helpdesk dalam melayani setiap laporan gangguan dari customer.

3. Menurut pendapat Anda, apakah fungsionalitas dari halaman helpdesk ini mudah dipahami dan digunakan / *User friendly* ?

   Jawaban :

Seluruh halaman helpdesk dalam aplikasi trouble ticket system dapat dengan mudah saya pahami dan saya gunakan.

   Berdasarkan hasil wawancara diatas maka dapat disimpulkan bahwa fungsionalitas dari halaman helpdesk ini mudah dipahami dan digunakan / *User friendly*.

B. Wawancara Pengujian Beta Untuk Teknisi

Wawancara dilakukan untuk pengujian beta terhadap aplikasi *trouble ticket system* kepada teknisi dengan total narasumber sebanyak 1 orang. Berikut daftar pertanyaan yang diajukan kepada narasumber :

1. Menurut pendapat anda dengan adanya halaman teknisi dalam aplikasi trouble ticket system dapat membantu perkerjaan anda menjadi lebih mudah ketika ada gangguan yang harus ditangani?
2. Menurut pendapat anda dengan adanya halaman teknisi dalam aplikasi trouble ticket system dapat mempermudah anda dalam melaporkan hasil penanganan yang telah anda lakukan ?
3. Menurut pendapat Anda, apakah fungsionalitas dari halaman teknisi ini mudah dipahami dan digunakan / *User friendly* ?

Berikut ini adalah masing-masing jawaban dari pertanyaan hasil wawancara tersebut.

1. Menurut pendapat anda dengan adanya halaman teknisi dalam aplikasi trouble ticket system dapat membantu perkerjaan anda menjadi lebih mudah ketika ada gangguan yang harus ditangani?

   Jawaban :

Sudah dapat, karena dengan adanya halaman teknisi dalam aplikasi trouble ticket system saya dengan mudah mengetahui ketika ada gangguan yang harus ditangani.

Berdasarkan hasil wawancara diatas maka dapat disimpulkan bahwa halaman teknisi dalam aplikasi troubel ticket system mempermudah teknisi ketika ada gangguan yang harus ditangani.

2. Menurut pendapat anda dengan adanya halaman teknisi dalam aplikasi trouble ticket system dapat mempermudah anda dalam melaporkan hasil penanganan yang telah anda lakukan ?

   Jawaban :

Sudah dapat, karena dengan adanya halaman teknisi saya dapat dengan mudah memberikan laporan tentang penanganan yang sudah saya lakukan.

Berdasarkan hasil wawancara diatas maka dapat disimpulkan bahwa halaman teknisi dalam aplikasi troubel ticket system mempermudah

teknisi dalam memberikan laporan tentang penanganan yang sudah teknisi lakukan.

3. Menurut pendapat Anda, apakah fungsionalitas dari halaman teknisi ini mudah dipahami dan digunakan / *User friendly* ?

   Jawaban :

Seluruh halaman teknisi dalam aplikasi trouble ticket system dapat dengan mudah saya pahami dan saya gunakan.

Berdasarkan hasil wawancara diatas maka dapat disimpulkan bahwa fungsionalitas dari halaman teknisi ini mudah dipahami dan digunakan / *User friendly*.

C. Wawancara Pengujian Beta Untuk Manager

Wawancara dilakukan untuk pengujian beta terhadap aplikasi *trouble ticket system* kepada manager dengan total narasumber sebanyak 1 orang. Berikut daftar pertanyaan yang diajukan kepada narasumber :

1. Menurut pendapat anda dengan adanya halaman manager dalam aplikasi trouble ticket system dapat mengelola data pengguna dengan mudah?
2. Menurut pendapat anda dengan adanya halaman manager dalam apliksai trouble ticket system dapat mempermudah anda dalam membuat laporan data ticket dan juga dapat melihat gangguan yang sering muncul lewat grafik ?

3. Menurut pendapat Anda, apakah fungsionalitas dari halaman teknisi ini mudah dipahami dan digunakan / *User friendly* ?

Berikut ini adalah masing-masing jawaban dari pertanyaan hasil wawancara tersebut.

1. Menurut pendapat anda dengan adanya halaman manager dalam aplikasi trouble ticket system dapat mengelola data pengguna dengan mudah?

Sudah dapat, karena dengan adanya halaman manager dalam aplikasi trouble ticket system saya dengan mudah mengelola data pengguna

Jawaban :

Berdasarkan hasil wawancara diatas maka dapat disimpulkan bahwa halaman manager dalam aplikasi troubel ticket system mempermudah manager dalam mengolah data pengguna.

2. Menurut pendapat anda dengan adanya halaman manager dalam apliksai trouble ticket system dapat mempermudah anda dalam membuat laporan data ticket dan juga dapat melihat gangguan yang sering muncul lewat grafik ?

Jawaban :

Sudah dapat, karena dengan adanya halaman manager saya dengan mudah membuat laporan data ticket dan saya juga dapat mengetahui gangguan yang sering terjadi dengan cara melihat grafik.

Berdasarkan hasil wawancara diatas maka dapat disimpulkan bahwa halaman manager dalam aplikasi troubel ticket system mempermudah manager dalam membuat laporan dan mengetahui gangguan yang sering terjadi dengan cara melihat grafik.

3. Menurut pendapat Anda, apakah fungsionalitas dari halaman teknisi ini mudah dipahami dan digunakan / *User friendly* ?

   Jawaban :

Seluruh halaman teknisi dalam aplikasi trouble ticket system dapat dengan mudah saya pahami dan saya gunakan.

Berdasarkan hasil wawancara diatas maka dapat disimpulkan bahwa fungsionalitas dari halaman manager ini mudah dipahami dan digunakan / *User friendly*.

4.2.5.2 Kuisioner Customer

Kuisioner bagi customer mempunyai lima pertanyaan dan tujuh pilihan jawaban. Berikut adalah detailnya:

1. Apakah anda setuju bahwa aplikasi *trouble ticket system* untuk menginputkan data keluhan membantu anda dengan cepat ketika ada gangguan yang terjadi?

No.	Keterangan
1.	Sangat Setuju
2.	Setuju
3.	Cukup Setuju
4.	Kurang Setuju
5	Tidak Setuju
6	Sangat Tidak Setuju

2. Apakah anda setuju dengan adanya aplikasi trouble ticket penanganan dapat dilakukan dengan cepat?

No.	Keterangan
1.	Sangat Setuju
2.	Setuju
3.	Cukup Setuju
4.	Kurang Setuju
5	Tidak Setuju
6	Sangat Tidak Setuju

3. Apakah anda setuju bahwa tampilan *website* ini *user friendly?*

No.	Keterangan
1.	Sangat Setuju
2.	Setuju
3.	Cukup Setuju
4.	Kurang Setuju
5	Tidak Setuju
6	Sangat Tidak Setuju

4. Apakah anda setuju bahwa bahasa yang digunakan mudah dimengerti?

No.	Keterangan
1.	Sangat Setuju
2.	Setuju
3.	Cukup Setuju
4.	Kurang Setuju
5	Tidak Setuju
6	Sangat Tidak Setuju

Kuesioner disebarkan menggunakan teknik sampling yaitu Simple Random Sampling yang disebarkan kepada 5 pengguna. Dari hasil kuesioner tersebut akan dilakukan perhitungan agar dapat diambil kesimpulan terhadap penilaian penerapan sistem yang baru. Kuisioner ini terdiri dari 5 pertanyaan (contoh kuisioner dapat dilihat pada lampiran) dengan menggunakan skala likert dengan skala 1 sampai 5.

Berdasarkan data hasil kuisioner tersebut, 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

Berikut ini adalah hasil persentase masing-masing jawaban yang sudah dihitung nilainya dengan menggunakan rumus diatas. Kuisioner ini diujikan kepada 5 orang.

1. Apakah anda setuju bahwa aplikasi *trouble ticket system* untuk meninputkan data keluhan membantu anda dengan cepat ketika ada gangguan yang terjadi?

Tabel 4. 32 Pertanyaan No.1

Pertanyaan	No	Keterangan	Responden	Persentase (%)
1	1	Sangat Setuju		
	2	Setuju	4	80%
	3	Cukup Setuju	1	20%
	4	Kurang Setuju		
	5	Tidak Setuju		
Jumlah			5	100%

Berdasarkan hasil persentase diatas maka dapat disimpulkan sebanyak 4 orang atau 80% menyatakan setuju, 1 orang atau 20% menyatakan cukup setuju bahwa sistem yang dibangun membantu *customer* menginputkan keluhan ketika ada gangguan.

2. Apakah anda setuju dengan adanya aplikasi *trouble ticket system* penanganan dapat dilakukan dengan cepat?

Tabel 4. 33 Pertanyaan No.2

Pertanyaan	No	Keterangan	Responden	Persentase (%)
2	1	Sangat Setuju		
	2	Setuju	3	60%
	3	Cukup Setuju	1	20%
	4	Kurang Setuju	1	20%
	5	Tidak Setuju		
Jumlah			5	100%

Berdasarkan hasil persentase diatas maka dapat disimpulkan sebanyak 3 orang atau 60% menyatakan setuju, 1 orang atau 20% menyatakan cukup setuju dan 1 orang atau 20% menyatakan kurang setuju bahwa sistem yang dibangun memudahkan *customer* dengan penanganan yang lebih cepat dari *helpdesk* yang ada di Direktori TIK UPI.

3. Apakah anda setuju bahwa tampilan website ini *user friendly?*

Tabel 4. 34 Pertanyaan No.3

Pertanyaan	No	Keterangan	Responden	Persentase (%)
3	1	Sangat Setuju		
	2	Setuju	5	100%
	3	Cukup Setuju		
	4	Kurang Setuju		
	5	Tidak Setuju		
Jumlah			5	100%

Berdasarkan hasil persentase diatas maka dapat disimpulkan sebanyak 5 orang atau 100% menyatakan setuju, bahwa tampilan aplikasi ini cukup *user frienly.*

4. Apakah anda setuju bahwa bahasa yang digunakan mudah dimengerti?

Tabel 4. 35 Pertanyaan No.4

Pertanyaan	No	Keterangan	Responden	Persentase (%)
4	1	Sangat Setuju	3	60%
	2	Setuju		
	3	Cukup Setuju	1	20%
	4	Kurang Setuju	1	20%
	5	Tidak Setuju		
Jumlah			5	100%

Berdasarkan hasil persentase diatas maka dapat disimpulkan sebanyak 3 orang atau 60% menyatakan sangat setuju, 1 orang atau 20% menyatakan cukup dan 1 orang menyatakan kurang setuju bahwa sistem yang dibangun mudah dimengerti oleh *customer*