BAB IV

IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi Sistem

Implementasi dilakukan setelah perancangan selesai dan selanjutnya akan diimplementasikan pada bahasa pemrograman yang akan digunakan. Tujuan implementasi untuk mengkonfirmasikan modul program perancangan pada para pelaku sistem sehingga user dapat memberi masukan kepada pembangun sistem.

4.1.1 Implementasi Data

Pembuatan *database* dilakukan dengan menggunakan aplikasi DBMS MySQL. Berikut ini adalah tabel-tabel yang terdapat dalam perangkat lunak yang dibangun.

Tabel 4.1 Tabel yang terdapat dalam perangkat lunak

No	Nama Tabel	Nama File
1	pegawai	pegawai.sql
2	jabatan	jabatan.sql
3	bagian	bagian.sql
4	jatah_cuti	jatah_cuti.sql
5	detail_jatah	detail_jatah.sql
6	rekapabsensi	rekapabsensi.sql
7	gaji	gaji.sql
8	tunjangan	tunjangan.sql
9	detail_tunjangan	detail_tunjangan.sql
10	pinjaman	pinjaman.sql
11	detail_pinjaman	detail_pinjaman.sql
12	jamsostek	jamsostek.sql

Implementasi data untuk masing-masing tabel dalam bahasa SQL adalah sebagai berikut:

1. Tabel pegawai

Implementasi *query* yang digunakan untuk membuat tabel pegawai adalah sebagai berikut:

Tabel 4.2 Tabel pegawai

```
1  CREATE TABLE `pegawai` (
2  `no_rfid` INT(11) NOT NULL,
3  `nip` VARCHAR(20) NULL DEFAULT NULL,
4  `nama_pegawai` VARCHAR(30) NULL DEFAULT NULL,
5  `kode_jabatan` VARCHAR(20) NULL DEFAULT NULL,
6  `alamat` VARCHAR(50) NULL DEFAULT NULL,
7  `jenis_kelamin` VARCHAR(10) NULL DEFAULT NULL,
8  `no_telp` INT(20) NULL DEFAULT NULL,
9  `username` VARCHAR(20) NULL DEFAULT NULL,
10 `password` VARCHAR(20) NULL DEFAULT NULL,
11 `level` VARCHAR(20) NULL DEFAULT NULL,
12 PRIMARY KEY (`no_rfid`),
13 INDEX `FK_pegawai_jabatan` (`kode_jabatan`),
14 CONSTRAINT `FK_pegawai_jabatan` FOREIGN KEY
15 (`kode_jabatan`) REFERENCES `jabatan` (`kode_jabatan`)
16 )
17 COLLATE='latin1_swedish_ci'
18 ENGINE=InnoDB;
```

2. Tabel jabatan

Implementasi *query* yang digunakan untuk membuat tabel jabatan adalah sebagai berikut:

Tabel 4.3 Tabel jabatan

```
1  CREATE TABLE `jabatan` (
2  `kode_jabatan` VARCHAR(20) NOT NULL,
3  `jabatan` VARCHAR(30) NULL DEFAULT NULL,
4  `kode_bagian` VARCHAR(20) NULL DEFAULT NULL,
5  `tunjangan` INT(20) NULL DEFAULT NULL,
6  PRIMARY KEY (`kode_jabatan`),
7  INDEX `FK_jabatan_bagian` (`kode_bagian`),
8  CONSTRAINT `FK_jabatan_bagian` FOREIGN KEY
9  (`kode_bagian`) REFERENCES `bagian` (`kode_bagian`)
10 )
11 COLLATE='latin1_swedish_ci'
12 ENGINE=InnoDB;
```

3. Tabel bagian

Implementasi *query* yang digunakan untuk membuat tabel bagian adalah sebagai berikut:

Tabel 4.4 Tabel bagian

```
1  CREATE TABLE `bagian` (
2  `kode_bagian` VARCHAR(20) NOT NULL,
3  `bagian` VARCHAR(20) NULL DEFAULT NULL,
4  PRIMARY KEY (`kode_bagian`)
5  )
6  COLLATE='latin1_swedish_ci'
7  ENGINE=InnoDB;
```

4. Tabel jatah_cuti

Implementasi *query* yang digunakan untuk membuat tabel jatah_cuti adalah sebagai berikut:

Tabel 4.5 Tabel jatah_cuti

```
1  CREATE TABLE `jatah_cuti` (
2  `id_jatah` INT(10) NOT NULL,
3  `nama_jatah` VARCHAR(20) NULL DEFAULT NULL,
4  `jumlah_jatah` INT(10) NULL DEFAULT NULL,
5  PRIMARY KEY (`id_jatah`)
6  )
7  COLLATE='latin1_swedish_ci'
8  ENGINE=InnoDB;
```

5. Tabel detail_jatah

Implementasi *query* yang digunakan untuk membuat tabel detail_jatah adalah sebagai berikut:

Tabel 4.6 Tabel detail_jatah

```
1  CREATE TABLE `detail_jatah` (
2  `id_dj` INT(10) NOT NULL,
3  `no_rfid` INT(11) NULL DEFAULT NULL,
4  `id_jatah` INT(10) NULL DEFAULT NULL,
5  `tanggal_mulai` DATE NULL DEFAULT NULL,
6  `tanggal_selesai` DATE NULL DEFAULT NULL,
7  PRIMARY KEY (`id_dj`),
8  INDEX `FK_detail_jatah_pegawai` (`no_rfid`),
9  INDEX `FK_detail_jatah_jatah_cuti` (`id_jatah`),
10 CONSTRAINT `FK_detail_jatah_pegawai` FOREIGN KEY
11 (`no_rfid`) REFERENCES `pegawai` (`no_rfid`),
```

```
12 CONSTRAINT `FK_detail_jatah_jatah_cuti` FOREIGN KEY
13 (`id_jatah`) REFERENCES `jatah_cuti` (`id_jatah`)
14 )
15 COLLATE='latin1_swedish_ci'
16 ENGINE=InnoDB;
```

6. Tabel rekapabsensi

Implementasi *query* yang digunakan untuk membuat tabel rekapabsensi adalah sebagai berikut:

Tabel 4.7 Tabel rekapabsensi

```
1  CREATE TABLE `rekapabsensi` (
2  `no_absensi` INT(5) NOT NULL,
3  `no_rfid` INT(11) NULL DEFAULT NULL,
4  `s` INT(10) NULL DEFAULT NULL,
5  `i` INT(10) NULL DEFAULT NULL,
6  `a` INT(10) NULL DEFAULT NULL,
7  `c` INT(10) NULL DEFAULT NULL,
8  `jumlah` INT(10) NULL DEFAULT NULL,
9  PRIMARY KEY (`no_absensi`),
10 INDEX `FK_rekapabsensi_pegawai` (`no_rfid`),
11 CONSTRAINT `FK_rekapabsensi_pegawai` FOREIGN KEY
12 (`no_rfid`) REFERENCES `pegawai` (`no_rfid`)
13 )
14 COLLATE='latin1_swedish_ci'
15 ENGINE=InnoDB;
```

7. Tabel gaji

Implementasi *query* yang digunakan untuk membuat tabel gaji adalah sebagai berikut:

Tabel 4.8 Tabel gaji

```
1  CREATE TABLE `gaji` (
2  `no_gaji` VARCHAR(10) NOT NULL,
3  `no_absensi` INT(5) NULL DEFAULT NULL,
4  `tanggal` DATE NULL DEFAULT NULL,
5  `gaji_pokok` INT(20) NULL DEFAULT NULL,
6  `no_detailtunjangan` INT(10) NULL DEFAULT NULL,
7  `id_dpinjaman` INT(10) NULL DEFAULT NULL,
8  `jumlah` INT(20) NULL DEFAULT NULL,
9  `no_kpj` INT(10) NULL DEFAULT NULL,
10 PRIMARY KEY (`no_gaji`),
11 INDEX `FK_gaji_rekapabsensi` (`no_absensi`),
12 INDEX `FK_gaji_detail_tunjangan`
13 (`no_detailtunjangan`),
14 INDEX `FK_gaji_detail_pinjaman` (`id_dpinjaman`),
15 INDEX `FK_gaji_jamsostek` (`no_kpj`),
```

```
16 CONSTRAINT `FK_gaji_detail_pinjaman` FOREIGN KEY
17 (`id_dpinjaman`) REFERENCES `detail_pinjaman`
18 (`id_dpinjaman`),
19 CONSTRAINT `FK_gaji_detail_tunjangan` FOREIGN KEY
20 (`no_detailtunjangan`) REFERENCES `detail_tunjangan`
21 (`no_detailtunjangan`),
22 CONSTRAINT `FK_gaji_jamsostek` FOREIGN KEY (`no_kpj`)
23 REFERENCES `jamsostek` (`no_kpj`),
24 CONSTRAINT `FK_gaji_rekapabsensi` FOREIGN KEY
25 (`no_absensi`) REFERENCES `rekapabsensi` (`no_absensi`)
26 )
27 COLLATE='latin1_swedish_ci'
28 ENGINE=InnoDB;
```

8. Tabel tunjangan

Implementasi *query* yang digunakan untuk membuat tabel tunjangan adalah sebagai berikut:

Tabel 4.9 Tabel tunjangan

```
1  CREATE TABLE `tunjangan` (
2  `no_tunjangan` INT(10) NOT NULL,
3  `tunjangan_makan` VARCHAR(20) NULL DEFAULT NULL,
4  `tunjangan_transport` VARCHAR(20) NULL DEFAULT NULL,
5  `kode_jabatan` VARCHAR(20) NULL DEFAULT NULL,
6  PRIMARY KEY (`no_tunjangan`),
7  INDEX `FK_tunjangan_jabatan` (`kode_jabatan`),
8  CONSTRAINT `FK_tunjangan_jabatan` FOREIGN KEY
9  (`kode_jabatan`) REFERENCES `jabatan` (`kode_jabatan`)
10 )
11 COLLATE='latin1_swedish_ci'
12 ENGINE=InnoDB;
```

9. Tabel detail_tunjangan

Implementasi *query* yang digunakan untuk membuat tabel detail_tunjangan adalah sebagai berikut:

Tabel 4.10 Tabel detail_tunjangan

```
1  CREATE TABLE `detail_tunjangan` (
2  `no_detailtunjangan` INT(10) NOT NULL,
3  `no_tunjangan` INT(10) NULL DEFAULT NULL,
4  `no_rfid` INT(11) NULL DEFAULT NULL,
5  `tanggal` DATE NULL DEFAULT NULL,
6  `jumlah` INT(20) NULL DEFAULT NULL,
7  PRIMARY KEY (`no_detailtunjangan`),
8  INDEX `FK_detail_tunjangan_pegawai` (`no_rfid`),
9  INDEX `FK_detail_tunjangan_tunjangan` (`no_tunjangan`),
10 CONSTRAINT `FK_detail_tunjangan_identitaspegawai`
```

```
11 FOREIGN KEY (`no_rfid`) REFERENCES `pegawai`
12 (`no_rfid`),
13 CONSTRAINT `FK_detail_tunjangan_tunjangan` FOREIGN KEY
14 (`no_tunjangan`) REFERENCES `tunjangan`
15 (`no_tunjangan`)
16 )
17 COLLATE='latin1_swedish_ci'
18 ENGINE=InnoDB;
```

10. Tabel pinjaman

Implementasi *query* yang digunakan untuk membuat tabel pinjaman adalah sebagai berikut:

Tabel 4.11 Tabel pinjaman

```
1  CREATE TABLE `pinjaman` (
2  `no_pinjaman` VARCHAR(20) NOT NULL,
3  `no_rfid` INT(11) NULL DEFAULT NULL,
4  `tanggal` DATE NULL DEFAULT NULL,
5  `jumlah` FLOAT NULL DEFAULT NULL,
6  PRIMARY KEY (`no_pinjaman`),
7  INDEX `FK_pinjaman_pegawai` (`no_rfid`),
8  CONSTRAINT `FK_pinjaman_pegawai` FOREIGN KEY
9  (`no_rfid`) REFERENCES `pegawai` (`no_rfid`)
10 )
11 COLLATE='latin1_swedish_ci'
12 ENGINE=InnoDB;
```

11. Tabel detail_pinjaman

Implementasi *query* yang digunakan untuk membuat tabel detail_pinjaman adalah sebagai berikut:

Tabel 4.12 Tabel detail_pinjaman

```
1  CREATE TABLE `detail_pinjaman` (
2  `id_dpinjaman` INT(10) NOT NULL,
3  `no_pinjaman` VARCHAR(20) NULL DEFAULT NULL,
4  `cicilan_1` INT(10) NULL DEFAULT NULL,
5  `cicilan_2` INT(10) NULL DEFAULT NULL,
6  `cicilan_3` INT(10) NULL DEFAULT NULL,
7  `cicilan_4` INT(10) NULL DEFAULT NULL,
8  `cicilan_5` INT(10) UNSIGNED NULL DEFAULT NULL,
9  `cicilan_6` INT(10) NULL DEFAULT NULL,
10 PRIMARY KEY (`id_dpinjaman`),
11 INDEX `FK_detail_pinjaman_pinjaman` (`no_pinjaman`),
12 CONSTRAINT `FK_detail_pinjaman_pinjaman` FOREIGN KEY
13 (`no_pinjaman`) REFERENCES `pinjaman` (`no_pinjaman`))
14 COLLATE='latin1_swedish_ci'
15 ENGINE=InnoDB;
```

12. Tabel jamsostek

Implementasi *query* yang digunakan untuk membuat tabel jamsostek adalah sebagai berikut:

Tabel 4.13 Tabel jamsostek

```
1  CREATE TABLE `jamsostek` (
2  `no_kpj` INT(10) NOT NULL,
3  `no_rfid` INT(11) NULL DEFAULT NULL,
4  `jkk` INT(10) NULL DEFAULT NULL,
5  `jkm` INT(10) NULL DEFAULT NULL,
6  `jht` INT(10) NULL DEFAULT NULL,
7  `total` INT(10) NULL DEFAULT NULL,
8  PRIMARY KEY (`no_kpj`),
9  INDEX `FK_jamsostek_pegawai` (`no_rfid`),
10 CONSTRAINT `FK_jamsostek_pegawai` FOREIGN KEY
11 (`no_rfid`) REFERENCES `pegawai` (`no_rfid`)
12 )
13 COLLATE='latin1_swedish_ci'
14 ENGINE=InnoDB;
```

4.1.2 Implementasi Antarmuka

Implementasi antarmuka dilakukan dengan setiap tampilan program yang dibangun dan pengkodeannya dalam bentuk *file* program. Berikut ini adalah implementasi antarmuka untuk pengguna dapat dilihat dilihat pada tabel 4.14. Adapun untuk tampilan implementasi pembangunan sistem informasi kepegawaian di PT. Cilegon Raya Utama Motor akan dilampirkan pada halaman lampiran

Tabel 4.14 Implementasi antar muka

Menu	Deskripsi	Nama File
Halaman Utama	Digunakan untuk menampilkan menu-menu pengolahan data	UnMenu.pas
Halaman Login	Digunakan sebagai halaman login pengguna	UnLogin.pas
Data Pegawai	Digunakan sebagai halaman untuk mengolah data pegawai dan menampilkan informasi data pegawai	UnPegawai.pas
Data Jabatan	Digunakan sebagai halaman untuk mengolah data jabatan dan menampilkan informasi data jabatan	UnJabatan.pas

Data Bagian	Digunakan sebagai halaman untuk mengolah data bagian dan menampilkan informasi data bagian	UnBagian.pas
Data Master Absensi	Digunakan sebagai halaman untuk mengolah data absensi	UnMasAbsensi.pas
Data Master Cuti	Digunakan sebagai halaman untuk mengolah data cuti	UnMasCuti.pas
Data Master Gaji	Digunakan sebagai halaman untuk mengolah data gaji	UnMasGaji.pas
Data Permohonan Cuti	Digunakan sebagai halaman untuk mengolah data permohonan cuti	UnPermCuti.pas
Data Permohonan Pinjaman	Digunakan sebagai halaman untuk mengolah data permohonan pinjaman	UnPermPinj.pas
Data Absensi	Digunakan sebagai halaman untuk mengolah data absensi dan menampilkan informasi data absensi	UnAbsensi.pas
Data Cuti	Digunakan sebagai halaman untuk mengolah data cuti dan menampilkan informasi data cuti	UnCuti.pas
Data Gaji	Digunakan sebagai halaman untuk mengolah data gaji dan menampilkan informasi data gaji	UnGaji.pas
Ganti Password	Digunakan sebagai halaman untuk mengubah data password	UnGPass.pas
Tentang	Digunakan sebagai halaman untuk menampilkan informasi tentang sistem informasi	UnTentang.pas
Tambah data RFID	Digunakan sebagai halaman untuk menambah data RFID dan menampilkan informasi data RFID	UnTestRFID.pas

4.2 Pengujian Sistem

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 tersebut.

Pengujian perangkat lunak ini menggunakan pengujian *black box* yang berfokus pada persyaratan fungsional perangkat lunak dan teknik yang digunakan adalah *equivalence partitioning* dimana pengujian perangkat lunak dilakukan dengan

cara membagi data masukan dari unit perangkat lunak menjadi beberapa partisi data dari mana test *case* dapat diturunkan. Pada prinsipnya, uji kasus dirancang untuk menutupi setiap partisi minimal sekali. Teknik ini mencoba mendefinisikan kasus uji yang mengungkap kelas kesalahan, sehingga mengurangi jumlah kasus uji yang harus di kembangkan.

4.2.1 Rencana Pengujian

Rencana pengujian yang akan dilakukan dengan menguji sistem. Skenario pengujian untuk bagian kepegawaian dapat dilihat pada tabel 4.15

Tabel 4.15 Skenario pengujian untuk bagian kepegawaian

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Isi data login	*Black Box*
	Verifikasi *username* dan *password*	*Black Box*
Permohonan Cuti	Menampilkan data permohonan cuti	*Black Box*
	Melakukan permohonan cuti	*Black Box*
Permohonan Pinjaman	Menampilkan data permohonan cuti	*Black Box*
	Melakukan permohonan cuti	*Black Box*
Cuti	Menampilkan jatah dan sisa Cuti	*Black Box*
Gaji	Menampilkan data Gaji	*Black Box*
Ganti Password	Menampilkan *form* ubah *password*	*Black Box*
	Verifikasi *password* lama	*Black Box*
	Verifikasi *password* baru	*Black Box*
	Verifikasi konfirmasi *password*	*Black Box*
Tentang	Menampilkan tentang program	*Black Box*
Logout	Menghapus *session user*	*Black Box*

Skenario pengujian untuk bagian keuangan terdiri dari lima kelas dan beberapa butir uji. Adapun jenis pengujiannya menggunakan metode *Black Box*, untuk lebih jelasnya dapat dilihat pada tabel 4.16

Tabel 4.16 Skenario pengujian untuk bagian keuangan

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Isi data login	*Black Box*
	Verifikasi *username* dan *password*	*Black Box*
Master Gaji	Tampilkan data gaji	*Black Box*
	Tambah data Gaji	*Black Box*
	Cari data Gaji	*Black Box*
	Ubah data Gaji	*Black Box*
	Hapus data Gaji	*Black Box*
Data Pinjaman	Tampilkan data Pinjaman	*Black Box*
	Tambah data Pinjaman	*Black Box*
	Cari data Pinjaman	*Black Box*
	Ubah data Pinjaman	*Black Box*
	Hapus data Pinjaman	*Black Box*
Ganti Password	Menampilkan *form* ubah *password*	*Black Box*
	Verifikasi *password* lama	*Black Box*
	Verifikasi *password* baru	*Black Box*
	Verifikasi konfirmasi *password*	*Black Box*
Logout	Menghapus *session operator*	*Black Box*

Skenario pengujian untuk admin terdiri dari sembilan kelas dan beberapa butir uji. Adapun jenis pengujiannya menggunakan metode *Black Box,* untuk lebih jelasnya dapat dilihat pada tabel 4.17

Tabel 4.17 Skenario pengujian untuk admin

Menu	Deskripsi	Nama File
Login	Isi data login	*Black Box*
	Verifikasi *username* dan *password*	*Black Box*
Data Pegawai	Tampilkan data Pegawai	*Black Box*
	Tambah data Pegawai	*Black Box*
	Cari data Pegawai	*Black Box*
	Ubah data Pegawai	*Black Box*
	Hapus data Pegawai	*Black Box*
Data Jabatan	Tampilkan data Jabatan	*Black Box*
	Tambah data Jabatan	*Black Box*
	Cari data Jabatan	*Black Box*
	Ubah data Jabatan	*Black Box*
	Hapus data Jabatan	*Black Box*
Data Bagian	Tampilkan data Bagian	*Black Box*
	Tambah data Bagian	*Black Box*
	Cari data Bagian	*Black Box*
	Ubah data Bagian	*Black Box*
	Hapus data Bagian	*Black Box*
Data Master Absensi	Tampilkan data Absensi	*Black Box*
	Tambah data Absensi	*Black Box*

	Cari data Absensi	*Black Box*
	Ubah data Absensi	*Black Box*
	Hapus data Absensi	*Black Box*
Data Master Cuti	Tampilkan data Cuti	*Black Box*
	Tambah data Cuti	*Black Box*
	Cari data Cuti	*Black Box*
	Ubah data Cuti	*Black Box*
	Hapus data Cuti	*Black Box*
Ganti Password	Menampilkan *form* ubah *password*	*Black Box*
	Verifikasi *password* lama	*Black Box*
	Verifikasi *password* baru	*Black Box*
	Verifikasi konfirmasi *password*	*Black Box*
Tambah data RFID	Menampilkan *form* tambah data RFID	*Black Box*
	Menambah data hasil *scanning* Tag RFID	*Black Box*
Logout	Menghapus *session admin*	*Black Box*

4.2.2 Pengujian *Blackbox*

Hasil implementasi perangkat lunak sistem informasi kepegawaian ini diuji pada pengguna bagian kepegawaian, bagian keuangan maupun admin di PT. Cilegon Raya Utama Motor yang berbasis *desktop* melalui spesifikasi perangkat sistem agar sistem dapat berjalan dengan baik.

Pengujian perangkat lunak sistem informasi kepegawaian menggunakan data uji berdasarkan form yang terdapat didalam aplikasi.yaitu sebagai berikut :

4.2.2.1 *Equivalence Partitioning* Untuk Bagian Kepegawaian

4.2.2.1.1 Pengujian Login Pegawai

Berikut dapat dilihat pengujian dari *login* pegawai dengan kasus dan hasil uji kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.18 :

Tabel 4.18 Pengujian login pegawai

Kasus dan Hasil Uji (Valid Class)			
Data Masukan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
username contoh masukan: username : elief	Data username yang dimasukan akan tampil pada masing-masing isian	Data yang username yang dimasukan tampil	[√] Diterima [ ] Ditolak
password contoh masukan: password : qwerty	Data password yang dimasukan akan tampil pada masing-masing isian	Data yang password yang dimasukan tampil	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Invalid Class)			
Data Masukan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
username contoh masukan: username : 12hxuw8@hsdj	Tampilkan pesan kesalahan “Maaf username tidak ditemukan Masukan username dengan benar”	Dapat menampilkan pesan kesalahan “Maaf username tidak ditemukan Masukan username dengan benar”	[√] Diterima [ ] Ditolak
password contoh masukan: password : o0oqww122	Tampilkan pesan kesalahan “Maaf password tidak ditemukan Masukan password dengan benar”	Dapat menampilkan pesan kesalahan “Maaf password tidak ditemukan Masukan password dengan benar”	[√] Diterima [ ] Ditolak

4.2.2.1.2 Pengujian Permohonan Cuti

Berikut dapat dilihat pengujian dari permohonan cuti dengan kasus dan hasil uji kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.19 :

Tabel 4.19 Pengujian permohonan cuti

Kasus dan Hasil Uji (Valid Class)			
Data Masukan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
No Cuti : 31952 Jenis Cuti : Tahunan Mulai : 19 Juli 2012 Selesai : 21 Juli 2012	Jika form terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses tambah data permohonan cuti	Semua form terisi atau benar dalam pengisian, lalu dapat melanjutkan proses tambah data permohonan cuti	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Invalid Class)			
Data Masukan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
No Cuti : 31952 Jenis Cuti : Tahunan Mulai : 19 Juli 2012 Selesai :	Tampilkan pesan kesalahan " Permohonan cuti tidak boleh kosong"	Dapat menampilkan pesan kesalahan " Permohonan cuti tidak boleh kosong"	[√] Diterima [ ] Ditolak

4.2.2.1.3 Pengujian Permohonan Pinjaman

Berikut dapat dilihat pengujian dari permohonan pinjaman dengan kasus dan hasil uji kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.20 :

Tabel 4.20 Pengujian permohonan pinjaman

Kasus dan Hasil Uji (Valid Class)			
Data Masukan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
No Pinjaman : 20019 NIP : IV-12.12-01003 Nama : Elief Jabatan : Kabag Jumlah : 1750000	Jika form terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses tambah data pinjaman	Semua form terisi atau benar dalam pengisian, lalu dapat melanjutkan proses tambah data pinjaman	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Invalid Class)			
Data Masukan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
No Pinjaman : 20019 NIP : IV-12.12-01003	Tampilkan pesan kesalahan " Data Pinjaman tidak boleh kosong"	Dapat menampilkan pesan kesalahan " Data Pinjaman tidak boleh kosong"	[√] Diterima [ ] Ditolak

Nama : Elief Jabatan : Kabag Jumlah :			

4.2.2.1.4 Pengujian Informasi Cuti

Berikut dapat dilihat pengujian dari informasi cuti dengan kasus dan hasil uji kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.21 :

Tabel 4.21 Pengujian informasi cuti

Kasus dan Hasil Uji (Valid Class)			
Data Masukan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
No RFID : 427139018	Data yang dimasukan akan menampilkan informasi cuti	Data yang dimasukan tampil	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Invalid Class)			
Data Masukan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
No RFID : 412131314	Tampilkan pesan kesalahan “ Data RFID belum terdaftar”	Dapat menampilkan pesan kesalahan “Data RFID belum terdaftar”	[√] Diterima [ ] Ditolak

4.2.2.1.5 Pengujian Informasi Gaji

Berikut dapat dilihat pengujian dari informasi gaji dengan kasus dan hasil uji kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.22 :

Tabel 4.22 Pengujian informasi gaji

Kasus dan Hasil Uji (Valid Class)			
Data Masukan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
No RFID : 427139018	Data yang dimasukan akan menampilkan informasi gaji	Data yang dimasukan tampil	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Invalid Class)			
Data Masukan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
No RFID : 412131314	Tampilkan pesan kesalahan “ Data RFID belum	Dapat menampilkan pesan kesalahan	[√] Diterima [ ] Ditolak

	terdaftar”	“Data RFID belum terdaftar”	

4.2.2.1.6 Pengujian Ganti Password

Berikut dapat dilihat pengujian dari ganti password dengan kasus dan hasil uji kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.23 :

Tabel 4.23 Pengujian ganti password

Kasus dan Hasil Uji (Valid Class)			
Data Masukan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
Password lama: qwerty Password Baru: 12345 Validasi: 12345	Jika form terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses ganti password	Semua form terisi atau benar dalam pengisian, dapat melanjutkan proses ganti password	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Invalid Class)			
Data Masukan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
Password lama: qwerty Password Baru: 12345 Validasi:	Tampilkan pesan kesalahan “Data tidak boleh kosong”	Dapat menampilkan pesan kesalahan “Data tidak boleh kosong”	[√] Diterima [ ] Ditolak

4.2.2.2 *Equivalence Partitioning* Untuk Bagian Keuangan

4.2.2.2.1 Pengujian Login Keuangan

Berikut dapat dilihat pengujian dari login keuangan dengan kasus dan hasil uji kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.25 :

Tabel 4.24 Pengujian login keuangan

Kasus dan Hasil Uji (Valid Class)			
Data Masukan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
username contoh masukan: username : gimin	Data username yang dimasukan akan tampil pada masing-masing isian	Data yang username yang dimasukan tampil	[√] Diterima [ ] Ditolak
password contoh masukan: password : qwerty	Data password yang dimasukan akan tampil pada masing-masing isian	Data yang password yang dimasukan tampil	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Invalid Class)			
Data Masukan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
username contoh masukan: username : 121efw11	Tampilkan pesan kesalahan "Maaf username tidak ditemukan Masukan username dengan benar"	Dapat menampilkan pesan kesalahan "Maaf username tidak ditemukan Masukan username dengan benar"	[√] Diterima [ ] Ditolak
password contoh masukan: password : 124x234-_2e	Tampilkan pesan kesalahan "Maaf password tidak ditemukan Masukan password dengan benar"	Dapat menampilkan pesan kesalahan "Maaf password tidak ditemukan Masukan password dengan benar"	[√] Diterima [ ] Ditolak

4.2.2.2.2 Pengujian Master Gaji

Berikut dapat dilihat pengujian dari master gaji dengan kasus dan hasil uji kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.25 :

Tabel 4.25 Pengujian master gaji

Kasus dan Hasil Uji (Valid Class)			
Data Masukan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nomor Gaji : 001 Tanggal : 27	Jika form terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses	Semua form terisi atau benar dalam pengisian, lalu dapat	[√] Diterima [ ] Ditolak

Juni 2012 Gaji Pokok : 1500000 Tunjangan : 300000 Lembur : 150000 Potongan : 0 Jumlah : 1950000	tambah data master gaji	melanjutkan proses tambah data master gaji	
Kasus dan Hasil Uji (Invalid Class)			
Data Masukan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nomor Gaji : 001 Tanggal : 27 Juni 2012 Gaji Pokok : 1500000 Tunjangan : 300000 Lembur : 150000 Potongan : 0 Jumlah :	Tampilkan pesan kesalahan " Master gaji tidak boleh kosong"	Dapat menampilkan pesan kesalahan " Master gaji tidak boleh kosong"	[√] Diterima [ ] Ditolak

4.2.2.2.3 Pengujian Data Pinjaman

Berikut dapat dilihat pengujian dari data pinjaman dengan kasus dan hasil uji kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.26 :

Tabel 4.26 Pengujian data pinjaman

Kasus dan Hasil Uji (Valid Class)			
Data Masukan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nomor Pinjaman : 001 NIP : IV-12.12-01003 Tanggal : 20 Juli 2012 Jumlah : 500000	Jika form terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses tambah data pinjaman	Semua form terisi atau benar dalam pengisian, lalu dapat melanjutkan proses tambah data pinjaman	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Invalid Class)			
Data Masukan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nomor Pinjaman : NIP : IV-12.12-01003 Tanggal : 20 Juli 2012 Jumlah :	Tampilkan pesan kesalahan “ pinjaman tidak boleh kosong”	Dapat menampilkan pesan kesalahan “ pinjaman tidak boleh kosong”	[√] Diterima [ ] Ditolak

4.2.2.3 *Equivalence Partitioning* Untuk Admin

4.2.2.3.1 Pengujian Login Admin

Berikut dapat dilihat pengujian dari login admin dengan kasus dan hasil uji kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.27 :

Tabel 4.27 Pengujian login admin

Kasus dan Hasil Uji (Valid Class)			
Data Masukan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
username contoh masukan: username : wahidin	Data username yang dimasukan akan tampil pada masing-masing isian	Data yang username yang dimasukan tampil	[√] Diterima [ ] Ditolak
password contoh masukan: password : qwerty	Data password yang dimasukan akan tampil pada masing-masing isian	Data yang password yang dimasukan tampil	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Invalid Class)			
Data Masukan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
username contoh masukan: username : 124eqwde	Tampilkan pesan kesalahan “Maaf username tidak ditemukan Masukan username dengan benar”	Dapat menampilkan pesan kesalahan “Maaf username tidak ditemukan Masukan username dengan benar”	[√] Diterima [ ] Ditolak

password contoh masukan: password : 123x-=12	Tampilkan pesan kesalahan “Maaf password tidak ditemukan Masukan password dengan benar”	Dapat menampilkan pesan kesalahan “Maaf password tidak ditemukan Masukan password dengan benar”	[√] Diterima [ ] Ditolak

4.2.2.3.2 Pengujian Data Pegawai

Berikut dapat dilihat pengujian dari data pegawai dengan kasus dan hasil uji kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.28 :

Tabel 4.28 Pengujian data pegawai

Kasus dan Hasil Uji (Valid Class)			
Data Masukan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
No RFID : 432702142 NIP : IV-12.12-01005 Nama : Nanang Kelamin : Laki-laki Telepon : 0254232402 Bagian : Kepegawaian Username : nanang Password : qwerty Level : Pegawai	Jika form terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses tambah data pegawai	Semua form terisi atau benar dalam pengisian, lalu dapat melanjutkan proses tambah data pegawai	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Invalid Class)			
Data Masukan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
No RFID : 432702142 NIP : IV-12.12-01005 Nama : Kelamin : Laki-laki	Tampilkan pesan kesalahan “ data pegawai tidak boleh kosong”	Dapat menampilkan pesan kesalahan “ data pegawai tidak boleh kosong”	[√] Diterima [ ] Ditolak

Telepon : 0254232402 Bagian : Kepegawaian Username : nanang Password : qwerty Level :			

4.2.2.3.3 Pengujian Data Jabatan

Berikut dapat dilihat pengujian dari data jabatan dengan kasus dan hasil uji kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.29 :

Tabel 4.29 Pengujian data jabatan

Kasus dan Hasil Uji (Valid Class)			
Data Masukan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
Kode Jabatan : 02 Jabatan : Kepala Bagian Tunjangan : 300000 Bagian : Kepegawaian	Jika form terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses tambah data jabatan	Semua form terisi atau benar dalam pengisian, lalu dapat melanjutkan proses tambah data jabatan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Invalid Class)			
Data Masukan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
Kode Jabatan : 02 Jabatan : Tunjangan : 300000 Bagian :	Tampilkan pesan kesalahan “ data jabatan tidak boleh kosong”	Dapat menampilkan pesan kesalahan “ data jabatan tidak boleh kosong”	[√] Diterima [ ] Ditolak

4.2.2.3.4 Pengujian Data Bagian

Berikut dapat dilihat pengujian dari data bagian dengan kasus dan hasil uji kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.30 :

Tabel 4.30 Pengujian data bagian

Kasus dan Hasil Uji (Valid Class)			
Data Masukan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
Kode Bagian : 002 Bagian : Kepegawaian	Jika form terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses tambah data bagian	Semua form terisi atau benar dalam pengisian, lalu dapat melanjutkan proses tambah data bagian	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Invalid Class)			
Data Masukan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
Kode Bagian : 002 Bagian :	Tampilkan pesan kesalahan “ data bagian tidak boleh kosong”	Dapat menampilkan pesan kesalahan “ data bagian tidak boleh kosong”	[√] Diterima [ ] Ditolak

4.2.2.3.5 Pengujian Master Cuti

Berikut dapat dilihat pengujian dari master cuti dengan kasus dan hasil uji kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.31 :

Tabel 4.31 Pengujian master cuti

Kasus dan Hasil Uji (Valid Class)			
Data Masukan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
No Cuti : 010 Jenis Cuti : Tahunan Tanggal Mulai : 07 Juli 2012 Tanggal Selesai 09 Juli 2012	Jika form terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses master cuti	Semua form terisi atau benar dalam pengisian, lalu dapat melanjutkan proses master cuti	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Invalid Class)			
Data Masukan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
No Cuti : Jenis Cuti : Tahunan Tanggal Mulai : 07 Juli 2012 Tanggal Selesai	Tampilkan pesan kesalahan “master cuti tidak boleh kosong”	Dapat menampilkan pesan kesalahan “ master cuti tidak boleh kosong”	[√] Diterima [ ] Ditolak

4.2.2.3.6 Pengujian Data RFID

Berikut dapat dilihat pengujian dari data RFID dengan kasus dan hasil uji kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.32 :

Tabel 4.32 Pengujian data RFID

Kasus dan Hasil Uji (Valid Class)			
Data Masukan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
No RFID : 423702312	Nomor RFID yang dimasukan akan tampil pada isian	Nomor RFID yang dimasukan tampil	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Invalid Class)			
Data Masukan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
No RFID : 412424849	Data RFID yang dimasukan tidak tampil pada isian	Data RFID yang dimasukan tidak tampil	[√] Diterima [ ] Ditolak

4.2.2.4 *Cause Effect Testing* Untuk Bagian Kepegawaian

Teknik ini merupakan suplemen dari untuk *equivalence testing* menyediakan cara untuk memilih kombinasi data *input*, melibatkan kondisi *input (Cause)* dan kondisi *output (effect)* untuk mencegah pendefinisian kasus uji yang terlalu banyak. Adapun langkah-langkahnya membagi spesifikasi kebutuhan menjadi bagian yang memiliki kemungkinan kerja kemudian definisikan *cause effect* berdasarkan spesifikasi

kebutuhan dan membuat tabel keputusan, kolom untuk *test case* dan baris untuk *cause/effect*, symbol 1 untuk mewakili ya symbol 0 untuk mewakili tidak. Berikut ini *cause effect testing* untuk pegawai :

4.2.2.4.1 *Cause effect testing* Login pegawai

Input States :

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

Output States :

A. Login berhasil

B. Menampilkan pesan kesalahan dari username yang diinputkan

C. Menampilkan pesan kesalahan dari password yang diinputkan

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari login pegawai. Tabel *cause effect testing* untuk login pegawai dapat dilihat pada tabel 4.33

Tabel 4.33 *Cause effect* testing login pegawai

Test case / Cause effect	1	2	3	4
1	1	0	1	0
2	1	1	0	0
A	1	0	0	0
B	0	1	0	1
C	0	0	1	1

4.2.2.4.2 *Cause effect testing* Permohonan cuti

Input States :

1. No Cuti memenuhi kondisi yang telah ditentukan
2. Jenis Cuti memenuhi kondisi yang telah ditentukan

3. Mulai memenuhi kondisi yang telah ditentukan
4. Selesai memenuhi kondisi yang telah ditentukan

Output States :

A. Data disimpan kedalam database
B. Menampilkan pesan kesalahan data yang diinputkan

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari permohonan cuti. Tabel *cause effect testing* untuk permohonan cuti dapat dilihat pada tabel 4.34

Tabel 4.34 *Cause effect* testing permohonan cuti

Test case / Cause effect	1	2	3	4	5
1	1	0	1	1	1
2	1	1	0	1	1
3	1	1	1	0	1
4	1	1	1	1	0
A	1	0	0	0	0
B	0	1	1	1	1

4.2.2.4.3 *Cause effect testing* Permohonan pinjaman

Input States :

1. No Pinjaman memenuhi kondisi yang telah ditentukan
2. NIP memenuhi kondisi yang telah ditentukan
3. Nama memenuhi kondisi yang telah ditentukan
4. Jabatan memenuhi kondisi yang telah ditentukan
5. Jumlah memenuhi kondisi yang telah ditentukan

Output States :

A. Data disimpan kedalam database

B. Menampilkan pesan kesalahan data yang diinputkan

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari permohonan pinjaman. Tabel *cause effect testing* untuk permohonan pinjaman dapat dilihat pada tabel 4.35

Tabel 4.35 *Cause effect* testing permohonan pinjaman

Test case / Cause effect	1	2	3	4	5	6
1	1	0	1	1	1	1
2	1	1	0	1	1	1
3	1	1	1	0	1	1
4	1	1	1	1	0	1
5	1	1	1	1	1	0
A	1	0	0	0	0	0
B	0	1	1	1	1	1

4.2.2.4.4 *Cause effect testing* Absensi

Input States :

1. No RFID memenuhi kondisi yang telah ditentukan

Output States :

A. Data disimpan kedalam database

B. Menampilkan pesan kesalahan data yang diinputkan

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari absensi. Tabel *cause effect testing* untuk absensi dapat dilihat pada tabel 4.36

Tabel 4.36 *Cause effect* testing absensi

Test case / Cause effect	1	2
1	1	0
A	1	0
B	0	1

4.2.2.4.5 *Cause effect testing* Informasi Cuti

Input States :

1. No RFID memenuhi kondisi yang telah ditentukan

Output States :

A. Data disimpan kedalam database

B. Menampilkan pesan kesalahan data yang diinputkan

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari informasi cuti. Tabel *cause effect testing* untuk informasi cuti dapat dilihat pada tabel 4.37

Tabel 4.37 *Cause effect* testing informasi cuti

Test case / Cause effect	1	2
1	1	0
A	1	0
B	0	1

4.2.2.4.6 *Cause effect testing* Informasi Gaji

Input States :

1. No RFID memenuhi kondisi yang telah ditentukan

Output States :

A. Data disimpan kedalam database

B. Menampilkan pesan kesalahan data yang diinputkan

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari informasi gaji. Tabel *cause effect testing* untuk informasi gaji dapat dilihat pada tabel 4.38

Tabel 4.38 *Cause effect* testing informasi gaji

Test case / Cause effect	1	2
1	1	0
A	1	0
B	0	1

4.2.2.4.7 *Cause effect testing* Ganti password

Input States :

1. Password lama memenuhi kondisi yang telah ditentukan
2. Password Baru memenuhi kondisi yang telah ditentukan
3. Validasi memenuhi kondisi yang telah ditentukan

Output States :

A. Data disimpan kedalam database

B. Menampilkan pesan kesalahan dari password lama yang diinputkan

C. Menampilkan pesan kesalahan dari password baru yang diinputkan

D. Menampilkan pesan kesalahan dari validasi yang diinputkan

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari ganti password. Tabel *cause effect testing* untuk ganti password dapat dilihat pada tabel 4.39

Tabel 4.39 *Cause effect* testing ganti password

Test case / Cause effect	1	2	3	4
1	1	0	1	1
2	1	1	0	1
3	1	1	1	0
A	1	0	0	0
B	0	1	0	0
C	0	0	1	0
D	0	0	0	1

4.2.2.5 *Cause Effect Testing* Untuk Bagian Keuangan

Teknik ini merupakan suplemen dari untuk *equivalence testing* menyediakan cara untuk memilih kombinasi data *input*, melibatkan kodisi *input (Cause)* dan kodisi *output (effect)* untuk mencegah pendefinisian kasus uji yang terlalu banyak. Berikut ini *cause effect testing* untuk Keuangan:

4.2.2.5.1 *Cause effect testing* Login Keuangan

Input States :

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

Output States :

A. Login berhasil

B. Menampilkan pesan kesalahan dari username yang diinputkan

C. Menampilkan pesan kesalahan dari password yang diinputkan

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari login keuangan. Tabel *cause effect testing* untuk login keuangan dapat dilihat pada tabel 4.40

Tabel 4.40 *Cause effect* testing login keuangan

Test case / Cause effect	1	2	3	4
1	1	0	1	0
2	1	1	0	0
A	1	0	0	0
B	0	1	0	1
C	0	0	1	1

4.2.2.5.2 *Cause effect testing* Master Gaji

Input States :

1. Nomor Gaji memenuhi kondisi yang telah ditentukan
2. Tanggal memenuhi kondisi yang telah ditentukan
3. Gaji Pokok memenuhi kondisi yang telah ditentukan
4. Tunjangan memenuhi kondisi yang telah ditentukan
5. Lembur memenuhi kondisi yang telah ditentukan
6. Potongan memenuhi kondisi yang telah ditentukan
7. Jumlah memenuhi kondisi yang telah ditentukan

Output States :

A. Data disimpan kedalam database

B. Menampilkan pesan kesalahan data yang diinputkan

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari master gaji. Tabel *cause effect testing* untuk master gaji dapat dilihat pada tabel 4.41

Tabel 4.41 *Cause effect* testing master gaji

Test case / Cause effect	1	2	3	4	5	6	7	8
1	1	0	1	1	1	1	1	1
2	1	1	0	1	1	1	1	1
3	1	1	1	0	1	1	1	1
4	1	1	1	1	0	1	1	1
5	1	1	1	1	1	0	1	1
6	1	1	1	1	1	1	0	1
7	1	1	1	1	1	1	1	0
A	1	0	0	0	0	0	0	0
B	0	1	1	1	1	1	1	1

4.2.2.5.3 *Cause effect testing* Data Pinjaman

Input States :

1. Nomor Pinjaman memenuhi kondisi yang telah ditentukan
2. NIP memenuhi kondisi yang telah ditentukan
3. Tanggal memenuhi kondisi yang telah ditentukan
4. Jumlah memenuhi kondisi yang telah ditentukan

Output States :

A. Data disimpan kedalam database

B. Menampilkan pesan kesalahan data yang diinputkan

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari data pinjaman. Tabel *cause effect testing* untuk data pinjaman dapat dilihat pada tabel 4.42

Tabel 4.42 *Cause effect* testing data pinjaman

Test case / Cause effect	1	2	3	4	5
1	1	0	1	1	1
2	1	1	0	1	1
3	1	1	1	0	1
4	1	1	1	1	0

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

4.2.2.6 *Cause Effect Testing* Untuk Admin

Teknik ini merupakan suplemen dari untuk *equivalence testing* menyediakan cara untuk memilih kombinasi data *input*, melibatkan kodisi *input (Cause)* dan kodisi *output (effect)* untuk mencegah pendefinisian kasus uji yang terlalu banyak. Berikut ini *cause effect testing* untuk Admin:

4.2.2.6.1 *Cause effect testing* Login Admin

Input States :

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

Output States :

Login berhasil

A. Menampilkan pesan kesalahan dari username yang diinputkan
B. Menampilkan pesan kesalahan dari password yang diinputkan

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari login admin. Tabel *cause effect testing* untuk login admin dapat dilihat pada tabel 4.43

Tabel 4.43 *Cause effect* testing login admin

Test case / Cause effect	**1**	**2**	**3**	**4**
1	1	0	1	0
2	1	1	0	0
A	1	0	0	0
B	0	1	0	1
C	0	0	1	1

4.2.2.6.2 *Cause effect testing* Data Pegawai

Input States :

1. No RFID memenuhi kondisi yang telah ditentukan
2. NIP memenuhi kondisi yang telah ditentukan
3. Nama memenuhi kondisi yang telah ditentukan
4. Kelamin memenuhi kondisi yang telah ditentukan
5. Telepon memenuhi kondisi yang telah ditentukan
6. Bagian memenuhi kondisi yang telah ditentukan
7. Username memenuhi kondisi yang telah ditentukan
8. Password memenuhi kondisi yang telah ditentukan
9. Level memenuhi kondisi yang telah ditentukan

Output States :

A. Data disimpan kedalam database

B. Menampilkan pesan kesalahan data yang diinputkan

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari data pegawai. Tabel *cause effect testing* untuk data pegawai dapat dilihat pada tabel 4.44

Tabel 4.44 *Cause effect* testing data pegawai

Test case / Cause effect	1	2	3	4	5	6	7	8	9	10
1	1	0	1	1	1	1	1	1	1	1
2	1	1	0	1	1	1	1	1	1	1
3	1	1	1	0	1	1	1	1	1	1
4	1	1	1	1	0	1	1	1	1	1
5	1	1	1	1	1	0	1	1	1	1
6	1	1	1	1	1	1	0	1	1	1
7	1	1	1	1	1	1	1	0	1	1

8	1	1	1	1	1	1	1	1	0	1
9	1	1	1	1	1	1	1	1	1	0
A	1	0	0	0	0	0	0	0	0	0
B	0	1	1	1	1	1	1	1	1	1

4.2.2.6.3 *Cause effect testing* Data Jabatan

Input States :

1. Kode Jabatan memenuhi kondisi yang telah ditentukan
2. Jabatan memenuhi kondisi yang telah ditentukan
3. Tunjangan memenuhi kondisi yang telah ditentukan
4. Bagian memenuhi kondisi yang telah ditentukan

Output States :

A. Data disimpan kedalam database

B. Menampilkan pesan kesalahan data yang diinputkan

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari data jabatan. Tabel *cause effect testing* untuk data jabatan dapat dilihat pada tabel 4.45

Tabel 4.45 *Cause effect* testing data jabatan

Test case / Cause effect	**1**	**2**	**3**	**4**	**5**
1	1	0	1	1	1
2	1	1	0	1	1
3	1	1	1	0	1
4	1	1	1	1	0
A	1	0	0	0	0
B	0	1	1	1	1

4.2.2.6.4 *Cause effect testing* Data Bagian

Input States :

1. Kode Bagian memenuhi kondisi yang telah ditentukan

2. Bagian memenuhi kondisi yang telah ditentukan

Output States :

A. Data disimpan kedalam database

B. Menampilkan pesan kesalahan data yang diinputkan

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari data bagian. Tabel *cause effect testing* untuk data bagian dapat dilihat pada tabel 4.46

Tabel 4.46 *Cause effect* testing data bagian

Test case / Cause effect	1	2	3
1	1	0	1
2	1	1	0
A	1	0	0
B	0	1	1

4.2.2.6.5 *Cause effect testing* Data Master Absensi

Input States :

1. No Absensi memenuhi kondisi yang telah ditentukan
2. Tanggal memenuhi kondisi yang telah ditentukan
3. Masuk memenuhi kondisi yang telah ditentukan
4. Keluar memenuhi kondisi yang telah ditentukan

Output States :

A. Data disimpan kedalam database

B. Menampilkan pesan kesalahan data yang diinputkan

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah

membaca hasil dari uji atau testing dari master absensi. Tabel *cause effect testing* untuk master absensi dapat dilihat pada tabel 4.47

Tabel 4.47 *Cause effect* testing login pegawai

Test case / Cause effect	1	2	3	4	5
1	1	0	1	1	1
2	1	1	0	1	1
3	1	1	1	0	1
4	1	1	1	1	0
A	1	0	0	0	0
B	0	1	1	1	1

4.2.2.6.6 *Cause effect testing* Data Master Cuti

Input States :

1. No Cuti memenuhi kondisi yang telah ditentukan
2. Jenis Cuti memenuhi kondisi yang telah ditentukan
3. Tanggal Mulai memenuhi kondisi yang telah ditentukan
4. Tanggal Selesai memenuhi kondisi yang telah ditentukan

Output States :

A. Data disimpan kedalam database

B. Menampilkan pesan kesalahan data yang diinputkan

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari master cuti. Tabel *cause effect testing* untuk master cuti dapat dilihat pada tabel 4.48

Tabel 4.48 *Cause effect* testing master cuti

Test case / Cause effect	1	2	3	4	5
1	1	0	1	1	1
2	1	1	0	1	1

3	1	1	1	0	1
4	1	1	1	1	0
A	1	0	0	0	0
B	0	1	1	1	1

4.2.2.6.7 *Cause effect testing* Data RFID

Input States :

1. No RFID memenuhi kondisi yang telah ditentukan

Output States :

A. Data disimpan kedalam database

B. Menampilkan pesan kesalahan data yang diinputkan

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari data RFID. Tabel *cause effect testing* untuk data RFID dapat dilihat pada tabel 4.49

Tabel 4.49 *Cause effect* testing data RFID

Test case / Cause effect	**1**	**2**
1	1	0
A	1	0
B	0	1

4.2.3 Pengujian Beta

Pengujian beta merupakan pengujian yang dilakukan secara objektif dimana diuji secara langsung ke lapangan yaitu instansi yang bersangkutan dengan membuat kuesioner mengenai kepuasan *user* dengan kandungan poin syarat *user friendy* untuk selanjutnya dibagikan kepada sebagian *user*. Dari hasil kuesioner tersebut akan

dilakukan perhitungan untuk dapat diambil kesimpulan terhadap penilaian penerapan sistem yang baru.

4.2.3.1 Wawancara Pengujian Beta Untuk Admin

1. Apakah Sistem Informasi Kepegawaian ini dapat membantu dalam pengolahan data *admin*?
2. Apakah anda setuju dengan adanya Sistem Informasi kepegawaian ini, Bagian kepegawaian PT. Cilegon Raya Utama Motor memberikan solusi layanan terbaik untuk pegawai?
3. Apakah anda mengerti dengan instruksi yang terdapat pada Sistem Informasi kepegawaian ini?
4. Apakah anda setuju dengan tampilan Sistem Informasi kepegawaian admin untuk pengolahan data pegawai?

4.2.3.2 Hasil Pengujian Beta Untuk Admin

1. Apakah Sistem Informasi Kepegawaian ini dapat membantu dalam pengolahan data *admin*?

   Jawaban : Sangat membantu, karena memudahkan kami untuk mengelolanya.
2. Apakah anda setuju dengan adanya Sistem Informasi kepegawaian ini, Bagian kepegawaian PT. Cilegon Raya Utama Motor memberikan solusi layanan terbaik untuk pegawai?

   Jawaban : Setuju, karena dengan adanya Sistem Informasi kepegawaian ini bagian kepegawaian PT. Cilegon Raya Utama Motor semakin berkembang karena memberikan solusi layanan yang terbaik untuk pegawai..

3. Apakah anda mengerti dengan instruksi yang terdapat pada Sistem Informasi kepegawaian ini?

   Jawaban : Setelah dijelaskan oleh anda, saya dapat mengerti bagaimana cara mengelolanya, ternyata sangat mudah.

4. Apakah anda setuju dengan tampilan Sistem Informasi kepegawaian admin untuk pengolahan data pegawai?

   Jawaban : Sangat setuju.

4.2.3.3 Wawancara Pengujian Beta Untuk Kepegawaian

1. Apakah Sistem Informasi kepegawaian ini dapat membantu anda dalam pengolahan data kepegawaian?
2. Apakah Sistem Informasi kepegawaian ini dapat membantu dalam pembuatan laporan absensi dan cuti?
3. Apakah anda mengerti dengan instruksi yang terdapat pada Sistem Informasi kepegawaian ini?
4. Apakah anda setuju dengan tampilan Sistem Informasi kepegawaian pada halaman bagian kepegawaian?

4.2.3.4 Hasil Pengujian Beta Untuk Kepegawaian

1. Apakah Sistem Informasi kepegawaian ini dapat membantu anda dalam pengolahan data kepegawaian?

   Jawaban : Sangat membantu saya, karena dengan adanya sistem informasi kepegawaian ini saya lebih mudah untuk mengelola seluruh data kepegawaian.

2. Apakah Sistem Informasi kepegawaian ini dapat membantu dalam pembuatan laporan absensi dan cuti?

Jawaban : Tentu.

3. Apakah anda mengerti dengan instruksi yang terdapat pada Sistem Informasi kepegawaian ini?

   Jawaban : Ya, saya sangat mengerti.

4. Apakah anda setuju dengan tampilan Sistem Informasi kepegawaian pada halaman bagian kepegawaian?

   Jawaban : Ya, Setuju.

4.2.3.5 Wawancara Pengujian Beta Untuk Keuangan

1. Apakah Sistem Informasi kepegawaian ini dapat membantu anda dalam pengolahan data penggajian?
2. Apakah Sistem Informasi kepegawaian ini dapat membantu dalam pembuatan laporan penggajian?
3. Apakah anda mengerti dengan instruksi yang terdapat pada Sistem Informasi kepegawaian ini?
4. Apakah anda setuju dengan tampilan Sistem Informasi kepegawaian pada halaman bagian keuangan?

4.2.3.6 Hasil Pengujian Beta Untuk Keuangan

1. Apakah Sistem Informasi kepegawaian ini dapat membantu anda dalam pengolahan data penggajian?

   Jawaban : Sangat membantu kerja saya agar lebih mudah untuk mengelola data penggajian.

2. Apakah Sistem Informasi kepegawaian ini dapat membantu dalam pembuatan laporan penggajian?

Jawaban : Ya, membantu.

3. Apakah anda mengerti dengan instruksi yang terdapat pada Sistem Informasi kepegawaian ini?

   Jawaban : Ya saya mengerti setelah dijelaskan oleh anda.

4. Apakah anda setuju dengan tampilan Sistem Informasi kepegawaian pada halaman bagian keuangan?

   Jawaban : Setuju.

4.2.3.7 Kuisioner Pengujian Beta Untuk Pegawai

1. Apakah anda setuju bahwa tampilan Sistem Informasi Kepegawaian ini cukup menarik dan mudah dipahami?

   A. Sangat Setuju

   B. Setuju

   C. Cukup Setuju

   D. Kurang Setuju

   E. Tidak Setuju

2. Apakah anda setuju bahwa Sistem Informasi Kepegawaian ini memudahkan anda dalam melakukan absensi?

   A. Sangat Setuju

   B. Setuju

   C. Cukup Setuju

   D. Kurang Setuju

   E. Tidak Setuju

3. Apakah anda setuju bahwa Sistem Informasi Kepegawaian ini memudahkan anda dalam melakukan permohonan cuti?

   A. Sangat Setuju

   B. Setuju

   C. Cukup Setuju

   D. Kurang Setuju

   E. Tidak Setuju

4. Apakah anda setuju bahwa Sistem Informasi Kepegawaian ini memudahkan anda dalam mengetahui informasi penggajian?

   A. Sangat Setuju

   B. Setuju

   C. Cukup Setuju

   D. Kurang Setuju

   E. Tidak Setuju

5. Apakah anda setuju bahwa tampilan Sistem Informasi Kepegawaian ini mudah dipahami / *user friendly*?

   A. Sangat Setuju

   B. Setuju

   C. Cukup Setuju

   D. Kurang Setuju

   E. Tidak Setuju

Berdasarkan data hasil kusioner tersebut, dapat dicari prosentase 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 *Black Box* dengan menggunakan rumus diatas. Kuisioner ini diujikan kepada 15 orang.

1. Apakah anda setuju bahwa tampilan Sistem Informasi Kepegawaian ini cukup menarik dan mudah dipahami?

Pertanyaan	**No.**	**Keterangan**	**Responden**	**Prosentase (%)**
1	1	Sangat Setuju	4	26,67
	2	Setuju	11	73,33
	3	Cukup Setuju	0	0
	4	Kurang Setuju	0	0
	5	Tidak Setuju	0	0
Jumlah			15	100

Berdasarkan hasil presentase diatas maka dapat disimpulkan sebanyak 4 orang atau 26,67% menyatakan sangat setuju dan 11 atau 73,33% menyatakan setuju, bahwa tampilan sistem informasi kepegawaian ini cukup menarik dan mudah untuk dipahami.

2. Apakah anda setuju bahwa Sistem Informasi Kepegawaian ini memudahkan anda dalam melakukan absensi?

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
2	1	Sangat Setuju	5	33,33
	2	Setuju	10	66,67
	3	Cukup Setuju	0	0
	4	Kurang Setuju	0	0
	5	Tidak Setuju	0	0
Jumlah			15	100

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 5 orang atau 33,33 % menyatakan sangat setuju, 10 orang atau 66,67% menyatakan setuju bahwa sistem informasi kepegawaian ini memudahkan pegawai dalam melakukan absensi.

3. Apakah anda setuju bahwa Sistem Informasi Kepegawaian ini memudahkan anda dalam melakukan permohonan cuti?

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
3	1	Sangat Setuju	2	13,33
	2	Setuju	13	86,67
	3	Cukup Setuju	0	0
	4	Kurang Setuju	0	0
	5	Tidak Setuju	0	0
Jumlah			15	100

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 2 atau 13,33% menyatakan sangat setuju dan 13 orang atau 86,67% menyatakan setuju bahwa sistem informasi kepegawaian ini memudahkan pegawai dalam melakukan permohonan cuti.

4. Apakah anda anda setuju bahwa Sistem Informasi Kepegawaian ini memudahkan anda dalam mengetahui informasi penggajian?

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
4	1	Sangat Setuju	6	40
	2	Setuju	9	60
	3	Cukup Setuju	0	0
	4	Kurang Setuju	0	0
	5	Tidak Setuju	0	0
Jumlah			15	100

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 6 orang atau 40% menyatakan sangat setuju dan 9 orang atau 60% menyatakan setuju bahwa sistem informasi kepegawaian ini memudahkan pegawai dalam mengetahui informasi penggajian.

5. Apakah anda setuju bahwa tampilan Sistem Informasi Kepegawaian ini mudah dipahami / *user friendly*?

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
5	1	Sangat Setuju	8	53,33
	2	Setuju	7	46,67
	3	Cukup Setuju	0	0
	4	Kurang Setuju	0	0
	5	Tidak Setuju	0	0
Jumlah			15	100

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 8 orang atau 53,33% menyatakan sangat setuju dan 7 orang atau 46,66 % menyatakan setuju bahwa tampilan sistem informasi kepegawaian ini mudah dipahami / *user friendly*.

4.2.4 Kesimpulan Pengujian

Berdasarkan hasil prosentase diatas yang didapatkan dari pengujian pengguna, maka sistem informasi kepegawaian ini sudah sesuai dengan tujuan yaitu dapat mempermudah pegawai dalam melakukan proses absensi, permohonan cuti serta informasi penggajian. Tampilannya *user friendly*, memudahkan dalam pengelolaan data kepegawaian dan memudahkan dalam membuat laporan penggajian dan laporan informasi cuti dari sistem informasi kepegawaian ini.