BAB IV

IMPLEMENTASI DAN PENGUJIAN

Pada bab ini akan dilakukan implementasi dan pengujian terhadap perangkat lunak Aplikasi Akademik Mahasiswa Universitas Komputer Indonesia berbasis RFID, yaitu tahap penerjemahan kebutuhan pembangunan aplikasi kedalam representasi kebutuhan perangkat lunak. Setelah implementasi maka dilakukan pengujian sistem yang dibangun untuk melihat kekurangan-kekurangan yang ada pada aplikasi.

4.1 Implementasi Sistem

Setelah aplikasi dianalisis dan dirancang secara rinci, tahap selanjutnya adalah implementasi. Tujuan dari implementasi adalah untuk mengkonfirmasi modul-modul perancangan sehingga pengguna dapat memberikan masukan kepada pengembangan sistem.

4.1.1 Implementasi Perangkat Keras

Perangkat keras yang digunakan untuk mengimplementasikan sistem adalah sebagai berikut :

1. *Processor* : Intel Pentium Dual-Core 2.00 GHz
2. *Harddisk* : 160 GB

3. RAM : DDR2 1 GB
4. LCD : 14.1”
5. RFID *Reader* : RFID ID-12
6. *Keyboard*
7. *Mouse*

1.1.2 Implementasi Perangkat Lunak

Perangkat lunak yang digunakan untuk membangun dan mengimplementasikan sistem adalah sebagai berikut :

1. Sistem Operasi Windows 7 Home Premium
2. MySQL 5.5.16
3. WampServer 2.2
4. Visual Studio 2010

1.1.3 Implementasi Basis Data

Implementasi basis data diambil berdasarkan perancangan basis data yang dibuat sebelumnya. Secara fisik, implementasi basis data diimplementasikan menggunakan perangkat lunak DBMS MySQL. Berikut adalah *query* sql untuk setiap tabel.

1. Tabel Absensi

Query sql untuk tabel Absensi adalah sebagai berikut:

```
CREATE TABLE `absensi` (
  `kodekelasperkuliahan` int(11) NOT NULL,
  `nim` varchar(8) NOT NULL,
  `m1` tinyint(4) default '0',
  `m2` tinyint(4) default '0',
  `m3` tinyint(4) default '0',
  `m4` tinyint(4) default '0',
  `m5` tinyint(4) default '0',
  `m6` tinyint(4) default '0',
  `m7` tinyint(4) default '0',
  `m8` tinyint(4) default '0',
  `m9` tinyint(4) default '0',
  `m10` tinyint(4) default '0',
  `m11` tinyint(4) default '0',
  `m12` tinyint(4) default '0',
  `m13` tinyint(4) default '0',
  `m14` tinyint(4) default '0',
  `m15` tinyint(4) default '0',
  `m16` tinyint(4) default '0',
  PRIMARY KEY  (`kodekelasperkuliahan`,`nim`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
ALTER TABLE `absensi`
  ADD CONSTRAINT `absensi_ibfk_2` FOREIGN KEY (`nim`)
REFERENCES `mahasiswa` (`NIM`),
  ADD CONSTRAINT `absensi_ibfk_1` FOREIGN KEY
(`kodekelasperkuliahan`) REFERENCES `absensi` (`kodekelasperkuliahan`) ON
DELETE CASCADE ON UPDATE CASCADE;
```

2. Tabel Mahasiswa

Query sql untuk tabel Mahasiswa adalah sebagai berikut:

```
. CREATE TABLE `mahasiswa` (
  `NIM` varchar(8) NOT NULL default '',
  `Nama` varchar(100) NOT NULL default '',
  `Tahun` varchar(4) NOT NULL default '',
```

```
  `Jalur` char(1) NOT NULL default '',
  `Status` char(1) NOT NULL default '',
  `KodeKelas` varchar(7) default NULL,
  `Jenis` char(1) default NULL,
  `Pass` varchar(50) default NULL,
  `Email` varchar(50) default NULL,
  `kddsn` int(11) default NULL,
  `SemesterMasuk` varchar(5) default NULL,
  `Kekhususan` varchar(20) default NULL,
  PRIMARY KEY  (`NIM`),
  UNIQUE KEY `IdxEMail` (`Email`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
ALTER TABLE `mahasiswa`
 ADD CONSTRAINT `mahasiswa_ibfk_1` FOREIGN KEY (`KodeKelas`)
REFERENCES `kelas` (`KodeKelas`) ON UPDATE CASCADE;
```

3. Tabel Matakuliah

Query sql untuk tabel Matakuliah adalah sebagai berikut:

```
CREATE TABLE `matakuliah` (
  `KodeProdi` varchar(3) default NULL,
  `KodeMataKuliah` varchar(10) NOT NULL default '',
  `MATAKULIAH` varchar(100) default NULL,
  `SKS` tinyint(1) unsigned default NULL,
  `SEMESTER` tinyint(1) unsigned default NULL,
  `PROGRAM` char(2) NOT NULL default 'S1',
  `WP` char(1) NOT NULL default 'W',
  `ALIAS` varchar(30) default NULL,
  `TLG` char(1) NOT NULL default 'T',
  `KURIKULUM` varchar(4) NOT NULL default '',
  `KETERANGAN` varchar(30) default NULL,
  `KodeKelompokMK` varchar(1) default NULL,
  `Disurvey` enum('Y','T') default 'Y',
  `Kekhususan` varchar(20) default NULL,
  PRIMARY KEY  (`KodeMataKuliah`,`KURIKULUM`),
  KEY `KURIKULUM` (`KURIKULUM`),
  KEY `MATAKULIAH` (`MATAKULIAH`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
ALTER TABLE `matakuliah`ADD CONSTRAINT `matakuliah_ibfk_1`
FOREIGN KEY (`KodeProdi`) REFERENCES `prodi` (`KodeProdi`);
```

4. FRS

Query sql untuk tabel FRS adalah sebagai berikut:

```
CREATE TABLE `frs` (
  `semester` varchar(5) NOT NULL,
  `nim` varchar(8) NOT NULL,
  `kodematakuliah` varchar(10) NOT NULL,
  `kurikulum` varchar(4) NOT NULL,
  `kodekelas` varchar(7) default NULL COMMENT 'tidak digunakan',
  `KodeKelasPerkuliahan` int(11) default NULL,
  `waktu_isi` datetime default NULL,
  `disetujui` enum('T','Y') default 'T',
  `waktu_setuju` datetime default NULL,
  `jenis_perwalian` enum('FRS','PRS') default 'FRS',
  PRIMARY KEY  (`semester`,`nim`,`kodematakuliah`,`kurikulum`),
  KEY `IdxKelasPerkuliahan` (`KodeKelasPerkuliahan`),
  KEY `idxsemnim` (`semester`,`nim`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
ALTER TABLE `frs`
  ADD CONSTRAINT `frs_ibfk_3` FOREIGN KEY (`kurikulum`)
REFERENCES `matakuliah` (`KURIKULUM`) ON UPDATE CASCADE,
  ADD CONSTRAINT `frs_ibfk_1` FOREIGN KEY (`nim`) REFERENCES
`mahasiswa` (`NIM`) ON UPDATE CASCADE,
  ADD CONSTRAINT `frs_ibfk_2` FOREIGN KEY (`kodematakuliah`)
REFERENCES `matakuliah` (`KodeMataKuliah`) ON UPDATE CASCADE
```

5. Tabel Ruang

Query sql untuk tabel Ruang adalah sebagai berikut

```
CREATE TABLE `ruang` (
  `KodeRuang` varchar(20) NOT NULL default '',
  `JenisPerkuliahan` varchar(1) default NULL,
  PRIMARY KEY  (`KodeRuang`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

6. Tabel Kelas

Query sql untuk tabel Kelas adalah sebagai berikut:

```
CREATE TABLE `kelas` (
  `KodeKelas` varchar(10) NOT NULL,
  `KodeProdi` varchar(10) default NULL,
  `Nama` varchar(20) default NULL COMMENT 'Nama Kelas',
  `Tahun` varchar(4) default NULL COMMENT 'Tahun Masuk',
  `KodeJenisKelas` varchar(1) default NULL COMMENT 'Jenis Kelas
(R)eguler/(K) Karyawan',
  `KodeDosenWali` int(11) default NULL,
  PRIMARY KEY  (`KodeKelas`),
  KEY `FK_kelas_prodi` (`KodeProdi`),
  KEY `FK_kelas_jeniskelas` (`KodeJenisKelas`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
ALTER TABLE `kelas`
  ADD CONSTRAINT `kelas_ibfk_1` FOREIGN KEY (`KodeProdi`)
REFERENCES `prodi` (`KodeProdi`);
```

7. Tabel Kelas_perkuliahn_dosen

Query sql untuk tabel Kelas_perkuliahn_dosen adalah sebagai berikut:

```
CREATE TABLE `kelas_perkuliahan_dosen` (
  `kodekelasperkuliahan` int(11) NOT NULL default '0',
  `kodedosen` int(11) default NULL,
  `banyakpertemuan` int(11) default NULL,
  UNIQUE KEY `idx_kelasperkuliahan_dosen`
(`kodekelasperkuliahan`,`kodedosen`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
ALTER TABLE `kelas_perkuliahan_dosen`
  ADD CONSTRAINT `kelas_perkuliahan_dosen_ibfk_2` FOREIGN KEY
(`kodedosen`) REFERENCES `dosen` (`KodeDosen`) ON UPDATE
CASCADE,
  ADD CONSTRAINT `kelas_perkuliahan_dosen_ibfk_1` FOREIGN KEY
(`kodekelasperkuliahan`) REFERENCES `kelas_perkuliahan`
(`KodeKelasPerkuliahan`) ON UPDATE CASCADE;
```

8. Tabel Kelas_perkuliahan

Query sql untuk tabel Kelas_perkuliahan adalah sebagai berikut:

```
CREATE TABLE `kelas_perkuliahan` (
  `KodeKelasPerkuliahan` int(11) NOT NULL auto_increment,
  `Semester` varchar(5) NOT NULL,
  `KodeKelas` varchar(7) default '',
  `KodeMatakuliah` varchar(10) NOT NULL,
  `Kurikulum` varchar(4) default NULL,
  `KodeDosen` int(11) default NULL,
  `KodeHari` int(11) default NULL,
  `KodeWaktuAwal` int(11) default NULL,
  `KodeWaktuAkhir` int(11) default NULL,
  `KodeRuang` varchar(15) default NULL,
  `SubKelas` char(1) default 'A',
  `NamaKelasAlt` varchar(50) default NULL COMMENT 'Nama Kelas
alternatif',
  `KodeDosenLama` varchar(3) default NULL,
  `MaksimalMhs` int(11) default '0',
  PRIMARY KEY  (`KodeKelasPerkuliahan`),
  UNIQUE KEY `idx_kelas_perkuliahan`
(`Semester`,`KodeKelas`,`KodeMatakuliah`,`SubKelas`),
  KEY `IdxMKKur` (`KodeMatakuliah`,`Kurikulum`)
) ENGINE=InnoDB AUTO_INCREMENT=13351 DEFAULT
CHARSET=latin1;
ALTER TABLE `KELAS_PERKULIAHAN`
  ADD CONSTRAINT `KELAS_PERKULIAHAN_IBFK_6` FOREIGN KEY
(`KODEWAKTUAKHIR`) REFERENCES `TREF_WAKTU`
(`KODEWAKTU`),
  ADD CONSTRAINT `KELAS_PERKULIAHAN_IBFK_1` FOREIGN KEY
(`KODEMATAKULIAH`) REFERENCES `MATAKULIAH`
(`KODEMATAKULIAH`),
  ADD CONSTRAINT `KELAS_PERKULIAHAN_IBFK_2` FOREIGN KEY
(`KODERUANG`) REFERENCES `RUANG` (`KODERUANG`),
  ADD CONSTRAINT `KELAS_PERKULIAHAN_IBFK_3` FOREIGN KEY
(`SEMESTER`) REFERENCES `FRS` (`SEMESTER`),
  ADD CONSTRAINT `KELAS_PERKULIAHAN_IBFK_4` FOREIGN KEY
(`KODEHARI`) REFERENCES `TREF_HARI` (`KODEHARI`),
  ADD CONSTRAINT `KELAS_PERKULIAHAN_IBFK_5` FOREIGN KEY
(`KODEWAKTUAWAL`) REFERENCES `TREF_WAKTU`
(`KODEWAKTU`);
```

9. Tabel Tref_Waktu

Query sql untuk tabel Tref_Waktu adalah sebagai berikut:

```
CREATE TABLE `tref_waktu` (
  `KodeWaktu` smallint(6) NOT NULL default '0',
  `Awal` varchar(5) default NULL,
  `Akhir` varchar(5) default NULL,
  PRIMARY KEY  (`KodeWaktu`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

10. Tabel Tref_hari

Query sql untuk tabel Tref_hari adalah sebagai berikut:

```
CREATE TABLE `tref_hari` (
  `KodeHari` tinyint(1) NOT NULL default '0',
  `Nama` varchar(6) default NULL,
  PRIMARY KEY  (`KodeHari`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

11. Tabel Fakultas

Query sql untuk tabel Fakultas adalah sebagai berikut:

```
CREATE TABLE `fakultas` (
  `KodeFakultas` varchar(2) NOT NULL,
  `Nama` varchar(50) default NULL,
  `KodeDekan` varchar(3) default NULL,
  `KodeFak_EPSBED` varchar(3) default NULL,
  PRIMARY KEY  (`KodeFakultas`),
  KEY `K1` (`KodeFakultas`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
ALTER TABLE `fakultas`
  ADD CONSTRAINT `fakultas_ibfk_1` FOREIGN KEY (`KodeDekan`)
REFERENCES `dosen` (`KodeDosen`);
```

12. Tabel Prodi

Query sql untuk tabel Prodi adalah sebagai berikut:

```
CREATE TABLE `prodi` (
  `KodeProdi` varchar(3) NOT NULL,
  `KodeJurusan` varchar(3) default NULL,
  `KDPRODI_LAMA` varchar(2) default NULL,
  `KodeFakultas` varchar(2) default NULL,
  `KodeJenjang` varchar(2) default NULL,
  `KodeProdi_EPSBED` varchar(5) default NULL,
  `Nama` varchar(40) default NULL,
  `TahunKurikulum` varchar(4) default NULL,
  `Dibuka` varchar(1) default 'Y',
  `TGAWLMSPST` datetime default NULL,
  `SMAWLMSPST` varchar(5) default NULL,
  `STATUMSPST` varchar(1) default NULL,
  `MLSEMMSPST` varchar(5) default NULL,
  `SKSTTMSPST` smallint(5) default NULL,
  `EMAILMSPST` varchar(40) default NULL,
  `NOMSKMSPST` varchar(40) default NULL,
  `TGLSKMSPST` datetime default NULL,
  `TGLAKMSPST` datetime default NULL,
  `NOMBAMSPST` varchar(40) default NULL,
  `TGLBAMSPST` datetime default NULL,
  `TGLABMSPST` datetime default NULL,
  `KDSTAMSPST` varchar(1) default NULL,
  `KDFREMSPST` varchar(1) default NULL,
  `KDPELMSPST` varchar(1) default NULL,
  `NOKPSMSPST` varchar(10) default NULL,
  `TELPSMSPST` varchar(20) default NULL,
  `TELPOMSPST` varchar(20) default NULL,
  `FAKSIMSPST` varchar(20) default NULL,
  `NMOPRMSPST` varchar(40) default NULL,
  `TELPRMSPST` varchar(20) default NULL,
  `LISAHMSPST` varchar(1) default NULL,
  `KDPS1MSPST` varchar(5) default NULL,
  `Singkatan` varchar(3) default NULL,
  PRIMARY KEY  (`KodeProdi`),
  UNIQUE KEY `KDPRODI` (`KDPRODI_LAMA`),
  KEY `K1` (`KodeJenjang`,`KodeProdi_EPSBED`),
  KEY `K2` (`KodeFakultas`,`KodeJenjang`,`KodeProdi_EPSBED`),
```

```
  KEY `FK_prodi_jurusan` (`KodeJurusan`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
ALTER TABLE `prodi`
  ADD CONSTRAINT `prodi_ibfk_2` FOREIGN KEY (`KodeFakultas`)
REFERENCES `fakultas` (`KodeFakultas`),
  ADD CONSTRAINT `prodi_ibfk_1` FOREIGN KEY (`KodeJurusan`)
REFERENCES `jurusan` (`KodeJurusan`);
```

13. Tabel Jurusan

Query sql untuk tabel Jurusan adalah sebagai berikut:

```
CREATE TABLE `jurusan` (
  `KodeJurusan` varchar(3) NOT NULL,
  `Nama` varchar(47) default NULL,
  `KodeFakultas` varchar(2) default NULL,
  `KodeDosenKajur` int(11) default NULL,
  `KodeDosenSekjur` int(11) default NULL,
  `Pass` varchar(50) default NULL,
  `email` varchar(50) default NULL,
  PRIMARY KEY  (`KodeJurusan`),
  KEY `FK_jurusan_fakultas` (`KodeFakultas`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
ALTER TABLE `jurusan`
  ADD CONSTRAINT `jurusan_ibfk_3` FOREIGN KEY (`KodeDosenSekjur`)
REFERENCES `dosen` (`KodeDosen`),
  ADD CONSTRAINT `jurusan_ibfk_1` FOREIGN KEY (`KodeFakultas`)
REFERENCES `fakultas` (`KodeFakultas`),
  ADD CONSTRAINT `jurusan_ibfk_2` FOREIGN KEY (`KodeDosenKajur`)
REFERENCES `dosen` (`KodeDosen`)
```

14. Tabel Log_Ruang

Query sql untuk tabel Log_Ruang adalah sebagai berikut:

```
CREATE TABLE IF NOT EXISTS `log_ruang` (
  `KodeLog` bigint(20) unsigned NOT NULL AUTO_INCREMENT,
  `KodeRuang` varchar(20) NOT NULL,
  `tglAkses` date NOT NULL,
```

```
  `JamAksesMasuk` time NOT NULL,
  `JamAksesKeluar` time DEFAULT NULL,
  `KodePengguna` int(11) NOT NULL,
  `JenisPengguna` varchar(20) DEFAULT NULL,
  `KodeKelasPerkuliahan` int(11) DEFAULT NULL,
  `Keterangan` varchar(50) DEFAULT NULL,
  `StatusPintu` tinyint(1) NOT NULL DEFAULT '0',
  PRIMARY KEY (`KodeLog`),
  KEY `idx_kelas_perkuliahan` (`KodeKelasPerkuliahan`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=2;
ALTER TABLE `log_ruang`
  ADD CONSTRAINT `log_ruang_ibfk_1` FOREIGN KEY
(`KodeKelasPerkuliahan`) REFERENCES `kelas_perkuliahan`
(`KodeKelasPerkuliahan`) ON UPDATE CASCADE;
```

15. Tabel BAP

Query sql untuk tabel BAP adalah sebagai berikut:

```
CREATE TABLE IF NOT EXISTS `bap` (
  `KodeKelasPerkuliahan` int(11) NOT NULL,
  `PertemuanKe` tinyint(4) NOT NULL,
  `WaktuPengisian` datetime NOT NULL,
  `Materi` text NOT NULL COMMENT 'materi yang dibahas',
  `JenisPerkuliahan` enum('Perkuliahan Normal','Kuliah Pengganti') NOT NULL,
  `KodeLog` bigint(20) unsigned NOT NULL,
  PRIMARY KEY (`KodeKelasPerkuliahan`,`PertemuanKe`),
  UNIQUE KEY `WaktuPengisian` (`WaktuPengisian`),
  UNIQUE KEY `idxKodeLogRuang` (`KodeLog`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;ALTER TABLE `bap`
  ADD CONSTRAINT `bap_ibfk_2` FOREIGN KEY (`KodeLog`)
REFERENCES `log_ruang` (`KodeLog`) ON UPDATE CASCADE,
  ADD CONSTRAINT `bap_ibfk_1` FOREIGN KEY (`KodeKelasPerkuliahan`)
REFERENCES `kelas_perkuliahan` (`KodeKelasPerkuliahan`) ON UPDATE
CASCADE;
```

16. Tabel RFID_mahasiswa

Query sql untuk tabel RFID_mahasiswa adalah sebagai berikut:

```
CREATE TABLE `RFID_Mahasiswa`
 `NIM` varchar(8) UNIQUE,
  `KodeRFID` varchar(20) UNIQUE,
) ENGINE=InnoDB;
ALTER TABLE `rfid_mahasiswa`
 ADD CONSTRAINT `rfid_mahasiswa_ibfk_1` FOREIGN KEY (`NIM`)
REFERENCES `mahasiswa` (`NIM`) ON DELETE CASCADE ON UPDATE
CASCADE;
```

17. Tabel RFID_dosen

Query sql untuk tabel RFID_dosen adalah sebagai berikut:

```
CREATE TABLE `RFID_Dosen` (
 `KodeDosen` int(11) UNIQUE,
 `KodeRFID` varchar(20) UNIQUE,
) ENGINE=InnoDB;
ALTER TABLE `rfid_dosen`
 ADD CONSTRAINT `rfid_dosen_ibfk_1` FOREIGN KEY (`KodeDosen`)
REFERENCES `dosen` (`KodeDosen`) ON DELETE CASCADE ON UPDATE
CASCADE;
```

1.1.4 Implementasi Antarmuka

Implementasi antarmuka dilakukan dengan setiap tampilan program yang dibangun. Berikut ini adalah implementasi antarmuka dari perangkat lunak Aplikasi AKademik Mahasiswa Universitas Komputer Indonesia berbasis RFID.

No	Antarmuka	Usecase	Nama File
1.	Mengisi Absensi dengan RFID	Mengisi Absensi	UIabsensi.cs Cabsensi.cs Ckelasperkuliahan.cs Cruang.cs Cwaktu.cs Cmahasiswa.cs Mengelolaabsensi.cs
2.	Menampilkan Kelas Perkuliahan	Menampilkan Kelas Perkuliahan	UIKelasPerkuliahan.cs Cdosen.cs Cmatakuliah.cs Cdosen.cs CCkelasperkuliahan.cs
3.	Menampilkan Absensi Mahasiswa	Menampilkan Absensi	UCariAbsensi.cs Ckelas.cs Cmatakuliah.cs Cabsensi.cs Ckelasperkuliahan.cs Mengelolaabsensi.cs
4.	Pengisian absensi mahasisiwa yang mengikuti kelas lain	Kelas Pengganti	UIupdateAbsensi.cs Ckelas.cs Cabsensi.cs Cmahasiswa.cs Ckelaperkuliahan.cs Cdosen.cs
5.	Menampilkan laporan absensi Mahasiswa	Menampilkan Laporan	UIlaporan.cs Laporan_Absensi_Mahasiswa.rdl Akademikdataset.xsd

1. **Antarmuka Pengsian Absensi**

   Merupakan tampilan utama dari aplikasi akademik berbasis RFID untuk pengisian absensi oleh mahasiswa



Gambar 4.1 Antarmuka Pengsian Absensi

2. **Antarmuka Login**

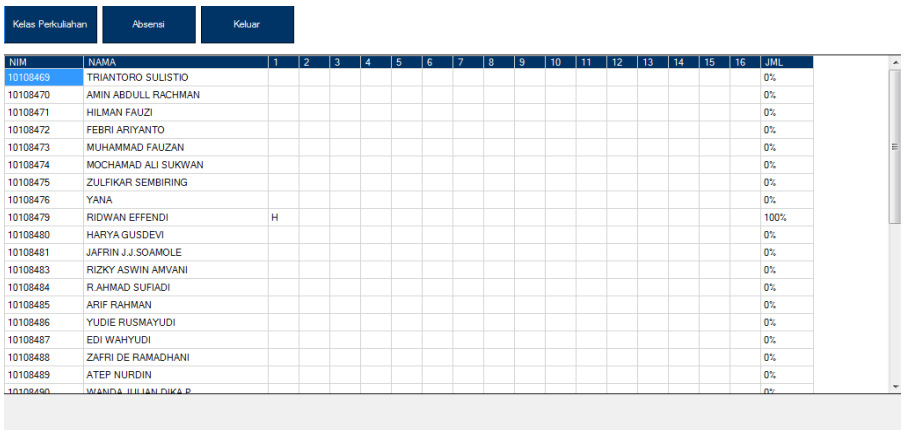
   Merupakan tampilan login dari aplikasi akademik berbasis RFID untuk verfikasi dosen dengan memasukan data nip dan *password.*



Gambar 4.2 Antarmuka Login

3. Antarmuka Menu Dosen

Merupakan tampilan menu Dosen dari aplikasi akademik berbasis RFID



Gambar 4.3 Antarmuka Menu Dosen

4. Antarmuka Kelas perkuliahan

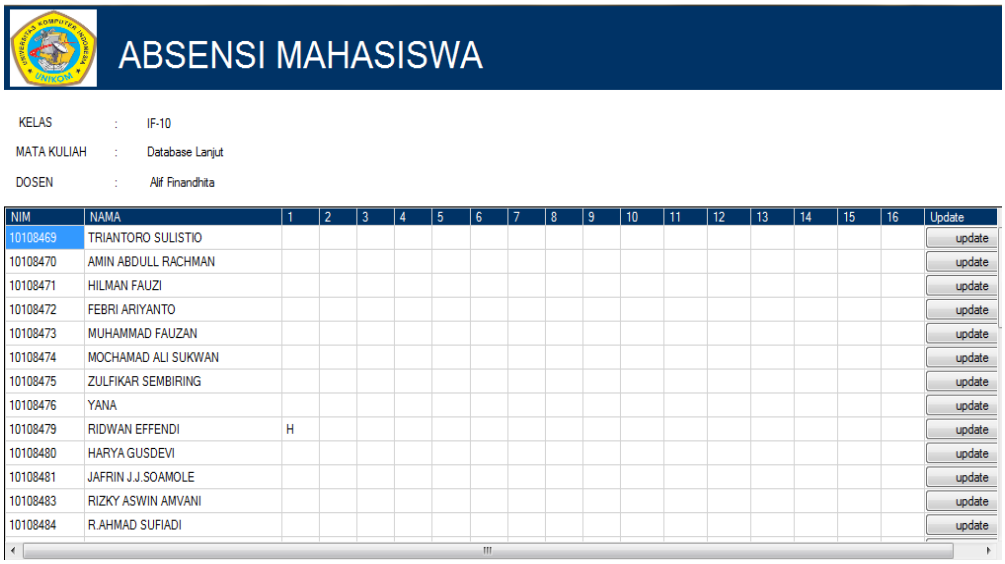
Merupakan tampilan kelas perkuliahan dari aplikasi akademik berbasis RFID dimana dosen dapat melakukan menampilkan seluruh kelas perkuliahan yang dimili oleh dosen bersangkutan.



Gambar 4.4 Antarmuka Kelas perkuliahan

5. Antarmuka Absensi Kehadiran

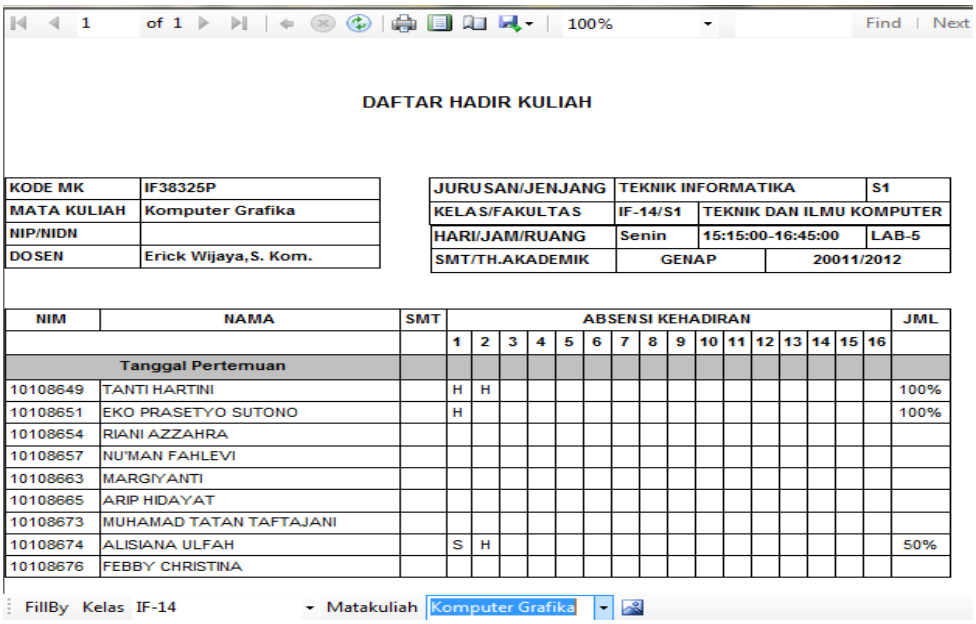
Merupakan tampilan absensi kehadiran dari aplikasi akademik berbasis RFID untuk dosen melakukan pengisian untuk keterangan mahasiswa izin, sakit dan alpha.



Gambar 4.5 Antarmuka Absensi Kehadiran

6. Antarmuka Laporan

Merupakan tampilan laporan dari aplikasi akademik berbasis RFID dimana dosen yang login dapat menampilkan laporan berdasarkan matakuliah dan laporan.



DAFTAR HADIR KULIAH

KODE MK	IF38325P
MATA KULIAH	Komputer Grafika
NIP/NIDN	
DOSEN	Erick Wijaya,S. Kom.

JURUSAN/JENJANG	TEKNIK INFORMATIKA		S1
KELAS/FAKULTAS	IF-14/S1	TEKNIK DAN ILMU KOMPUTER	
HARI/JAM/RUANG	Senin	15:15:00-16:45:00	LAB-5
SMT/TH.AKADEMIK	GENAP	20011/2012	

NIM	NAMA	SMT	ABSENSI KEHADIRAN															JML
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	
Tanggal Pertemuan																			
10108649	TANTI HARTINI		H	H															100%
10108651	EKO PRASETYO SUTONO		H																100%
10108654	RIANI AZZAHRA																		
10108657	NU'MAN FAHLEVI																		
10108663	MARGIYANTI																		
10108665	ARIP HIDAYAT																		
10108673	MUHAMAD TATAN TAFTAJANI																		
10108674	ALISIANA ULFAH		S	H															50%
10108676	FEBBY CHRISTINA																		

FillBy Kelas IF-14 Matakuliah Komputer Grafika

Gambar 4.5 Antarmuka Laporan

4.2 Pengujian

Pengujian merupakan metode yang dilakukan untuk menjelaskan mengenai pengoperasian perangkat lunak yang terdiri dari perangakat pengujian, metode pengujian dan pelaksana pengujian

4.2.1 Metode Pengujian

Dalam pengujian program ini menggunakan metode *Black Box*. Pengujian *Black Box* merupakan pengujian program berdasarkan fungsi dari program. Tujuan dari metode *Black Box* ini adalah untuk menemukan kesalahan fungsi pada program.

1.2.1.1 Rencana Pengujian

Pengujian perangkat lunak ini menggunakan data uji berdasarkan data dari masing-masing data. Rencana pengujian selengkapnya dapat dilihat pada tabel 4.1 berikut:

Tabel 4.1 Rencana Pengujian Perangkat Lunak Aplikasi Akademi

No.	Kelas Uji	Butir Uji	Jenis Pengujian
1.	Mengisi Absensi	Pengisian Absensi secara otomatis ke dalam *database*	*Black Box*
2.	Verifikasi Login	Verifikasi User	*Black Box*
		Verifikasi Password	
3.	Menampilkan Kelas Perkuliahan	Menampilkan Data kelas perkuliahan	*Black Box*
4.	Menampilkan Absensi	Merubah Keterangan Absensi mahasiswa selain keterangan hadir	*Black Box*
5.	Kelas pengganti	Memasukan NIM mahasiswa	*Black Box*
6.	Menampilkan Laporan	Memasukan Data kelas dan mtakuliah sebagai parameter untuk menyeleksi	*Black Box*
		Menampilkan Laporan	

1.2.1.2 Pengujian Alpha

Pengujian alpha merupakan pengujian yang menitik beratkan pada hasil output dari kendali input yang dimasukan pada tampilan kendali input tampilan. Pengujian dapat dikatakan berhasil apabila output sesuai dengan kendali input yang dimasukan pada setiap tampilan.

a. Pengujian Pengisian Absensi

Tabel 4.2 Pengujian Mengisi Absensi

Kasus dan Hasil Uji (Data Normal)				
No	**Data Masukan**	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
1.	Tag RFID mahasiswa Tag RFID mahasiswa : **78003E9D4992**	Dapat melakukan pengisian absensi kehadiran mahasiswa menampilkan pesan "Anda dengan NIM 10108674 telah berhasil mengisi absensi minggu ke-3 untuk matakuliah computer grafika"	Jika pintu telah terbuka baik dengan keterangan perkuliahan normal atau kuliah pengganti mahasiswa dapat mengisi absensi kehadiran	Diterima
Kasus dan Hasil Uji (Data Salah)				
No	**Data Masukan**	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
1.	Tag RFID mahasiswa Tag RFID mahasiswa : **78003E9D4992**	Menampilkan pesan "Pintu masih tertutup"	Pintu kelas masih tertutup sehingga mahasiswa tidak dapat melakukan pengisian absensi	Diterima
2.	Tag RFID mahasiswa Tag RFID mahasiswa : **78003E9D4992**	Menampilkan pesan " Anda tidak terdaftar di kelas perkuliahan ini"	Pintu kelas terbuka tetapi mahasisiwa bersangkutan tidak memiliki jadwal pada kelas ini	Diterima
3.	Tag RFID mahasiswa	Menampilkan pesan "Anda dengan NIM 10108674 telah	Pintu kelas terbuka tetapi mahasiswa	Diterima

	Tag RFID mahasiswa : **78003E9D4992**	terlambat mengisi absensi minggu ke-3 untuk matakuliah computer grafika”	terlambat dalam melakukan pengisian absensi mahasiswa	

b. Pengujian Login

Tabel 4.3 Pengujian Verifikasi Login

Kasus dan Hasil Uji (Data Normal)				
No	**Data Masukan**	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
1.	User dan Password User : 41273532002 Password : ADM	Dapat menampilkan Aplikasi Absensi dan tutup Form login	Dapat mengisi data login User dan Password sesuai yang diharapkan	Diterima
Kasus dan Hasil Uji (Data Salah)				
No	**Data Masukan**	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
1.	User dan Password User : 41273532002 Password : ADMM	Menampilkan Pesan “Password Salah”, Login Batal, Kembali ke Form Login	User tidak dapat login	Diterima
2.	User dan Password User : 41273532002111 Password : ADM	Menampilkan Pesan “User Salah”, Login Batal, Kembali ke Form Login	User tidak dapat login	Diterima

c. Menampilkan Kelas Perkuliahan

Tabel 4.4 Pengujian Menampilkan Kelas Perkuliahan

Kasus dan Hasil Uji (Data Normal)				
No	**Data Masukan**	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
1.	Semester Semester : Genap atau Ganjil	Menampilkan kelas perkuliahan yang dimiliki oleh dosen bersangkutan	Menampilkan kelas perkuliahan yang dimiliki oleh dosen bersangkutan	Diterima
Kasus dan Hasil Uji (Data Salah)				
No	**Data Masukan**	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
1.	Semester Semester : Genap atau Ganjil	Tidak menampilkan kelas perkuliahan yang dimiliki oleh dosen bersangkutan	Tidak menampilkan kelas perkuliahan yang dimiliki oleh dosen bersangkutan	Diterima

d. Menampilkan Absensi

Tabel 4.5 Pengujian Menampilkan Absensi

Kasus dan Hasil Uji (Data Normal)				
No	**Data Masukan**	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
1.	Data keterangan absensi M1=S(sakit)	Dapat mengubah absensi mahasiswa dengan keterangan yang diharapkan	Merubah absensi dengan keterangan yang diharpkan dan sesuai dengan minggu yang sedang diadakan	Diterima

Kasus dan Hasil Uji (Data Salah)				
No	**Data Masukan**	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
1.	Data keterangan absensi M1=S(sakit)	Menampilkan Pesan “absensi yang telah terisi tidak boleh dirubah”	Tidak dapat merubah absensi kehadiran yang telah diisi	Diterima
2.	Data keterangan absensi M1=S(sakit)	Tidak dapat merubah absensi	Tidak dapat melakukan pengisian absensi pada minggu berikut nya	Diterima

e. Kelas pengganti

Kasus dan Hasil Uji (Data Normal)				
No	**Data Masukan**	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
1.	NIM NIM:10108674	Mengisi absensi berdasrkan perkuliah yang sedang diikuti	Melakukan pengisian absensi kehadiran dan menampilkan pesan “anda dengan nim 10108674 telah terabsensi pada minggu ke 1 ”	Diterima

Kasus dan Hasil Uji (Data Salah)				
No	**Data Masukan**	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
1.	NIM NIM:10108674	Menampilkan pesan “ anda gagal melakukan pengisian absensi”	Tidak dapat melakukan pengisian absensi kehadiran mahasiswa	Diterima

Tabel 4.6 Pengujian Kelas pengganti

f. Menampilkan Laporan

Kasus dan Hasil Uji (Data Normal)				
No	**Data Masukan**	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
1.	Data matakuliah dan kelas Matakuliah : Komputer Grafika Kelas : IF-14	Menampilkan laporan sesuai yang diharapkan	Menampilkan laporan absensi berdasarkan matakuliah serta kelas yang telah ditentukan	Diterima
Kasus dan Hasil Uji (Data Salah)				
No	**Data Masukan**	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
1.	Data matakuliah dan kelas Matakuliah : “” Kelas :	Menampilkan pesan “Matakuliah tidak boleh kosong”	Tidak dapat menampilkan laporan karenan matakuliah kosong serta tidak dapat memilih kelas	Diterima

T4. Pengujian Menampilkan Laporan

1.2.1.3 Kesimpulan pengujian Alpha

Berdasarkan hasil pengujian kasus sample yang telah dilakukan, memberikan kesimpulan bahwa pada proses *Prototype* Aplikasi Akademik Mahasiswa Universitas Komputer Indonesia (UNIKOM) berbasis Teknologi *Radio Frequency Identification* (RFID) ini, kesalahan-kesalahan pada sintaks sudah melalui tahap perbaikan dan sudah di maksimalkan terhadap proses-proses tersebut dan cara fungsional sistem sudah dapat digunakan dan menghasilakn output yang diharap

1.2.1.4 Pengujian Betha

Pengujian betha merupakan pengujian yang dilakukan secara objektifdimana diuji secara langsung oleh pengguna yaitu dosen dan mahasiswa dengan menggunakan langsung aplikasi untuk mengetahui kepuasan user dan kandungan *point* syarat *user friendly*

a Kusioner untuk pengguna pengisian absensi dengan Tag RFID

1 Apakah anda setuju bahwa aplikasi ini memudahkan anda dalam melakukan pengisian absensi?

a. Sangat setuju
b. Setuju
c. Cukup setuju
d. Tidak setuju
e. Sangat tidak setuju

2 Apakah anda setuju bahwa aplikasi ini dapat mengurangi kesalahn anda dalam melakukan pengisian absensi?

a. Sangat setuju
b. Setuju
c. Cukup setuju
d. Tidak setuju
e. Sangat tidak setuju

3 Apakah anda setuju bahwa aplikasi ini dapat mempercepat dalam proses pengisian absensi ketika akan melakukan kelas perkuliahan?

a. Sangat setuju
b. Setuju
c. Cukup setuju
d. Tidak setuju
e. Sangat tidak setuju

4 Apakah anda setuju bahwa aplikasi ini sudah layak digunakan?

a. Sangat setuju
b. Setuju
c. Cukup setuju
d. Tidak setuju
e. Sangat tidak setuju

b Kuisioner untuk pengguna Dosen

1 Apakah anda setuju bahwa perintah atau intruksi yang ada aplikasi ini mudah di mengerti?

a. Sangat setuju
b. Setuju
c. Cukup setuju
d. Tidak setuju
e. Sangat tidak setuju

2 Apak anda setuju dengan aplikasi ini mempercepat proses pengisian absensi kehadiran mahasiswa??

a. Sangat setuju
b. Setuju
c. Cukup setuju
d. Tidak setuju
e. Sangat tidak setuju

3 Apakah anda setuju dengan aplikasi ini dapat mengurangi kesalahan dalam melakukan pengisian absensi mahasiswa??

a. Sangat setuju
b. Setuju
c. Cukup setuju
d. Tidak setuju
e. Sangat tidak setuju

4 Apakah anda setuju bahwa aplikasi ini layak digunakan?

a. Sangat setuju
b. Setuju
c. Cukup setuju
d. Tidak setuju
e. Sangat tidak setuju

Berdasarkan dari hasil kuisioner tersebut, discari prosentase masing-masing jawaban dengan menggunakan rumus : R=P/Q*100%

Keterangan : P=Banyaknya jawaban responden tiap soal

Q=Jumlah responden

R=Nilai Prosentase

Berikut ini hasil dari perhitungan prosentase tersebut:

a Kusioner untuk pengguna pengisian absensi dengan Tag RFID

1 Apakah anda setuju bahwa palikasi ini memudahkan anda dalam melakukan pengisian absensi?

Tabel 4. Hasil Jawaban Pertanyaan No 1 Kuisioner A

Kategori Jawaban	P	Q	R
Sangat setuju	3	17	17.65%
Setuju	11	17	64.7%
Cukup setuju	3	17	17.65%
Tidak setuju	0	17	0%
Sangat tidak setuju	0	17	0%

2 Apakah anda setuju bahwa aplikasi ini dapat mengurangi kesalahn anda dalam melakukan pengisian absensi?

Tabel 4. Hasil Jawaban Pertanyaan No 2 Kuisioner A

Kategori Jawaban	P	Q	R
Sangat setuju	4	17	23.53%
Setuju	10	17	62.5%
Cukup setuju	3	17	58.8%
Tidak setuju	0	17	0%
Sangat tidak setuju	0	17	0%

3 Apakah anda setuju bahwa aplikasi ini dapat mempercepat dalam proses pengisian absensi ketika akan melakukan kelas perkuliahan?

Tabel 4. Hasil Jawaban Pertanyaan No 3 Kuisioner A

Kategori Jawaban	P	Q	R
Sangat setuju	4	17	23.53%
Setuju	10	17	58,8%
Cukup setuju	4	17	23.53%
Tidak setuju	0	17	0%
Sangat tidak setuju	0	17	0%

4 Apakah anda setuju bahwa aplikasi ini sudah layak digunakan?

Tabel 4. Hasil Jawaban Pertanyaan No 4 Kuisioner A

Kategori Jawaban	P	Q	R
Sangat setuju	2	17	11.76%
Setuju	7	17	41.18%
Cukup setuju	7	17	41.18%
Tidak setuju	1	17	5.88%
Sangat tidak setuju	0	17	0%

b Kuisioner untuk pengguna Dosen

1 Apakah anda setuju bahwa perintah atau intruksi yang ada aplikasi ini mudah dimengerti?

Tabel 4. Hasil Jawaban Pertanyaan No 1 Kuisioner B

Kategori Jawaban	P	Q	R
Sangat setuju	1	5	20%
Setuju	3	5	60%
Cukup setuju	1	5	20%
Tidak setuju	0	5	0%
Sangat tidak setuju	0	5	0%

2 Apak anda setuju dengan aplikasi ini mempercepat proses pengisian absensi kehadiran mahasiswa??

Tabel 4. Hasil Jawaban Pertanyaan No 2 Kuisioner B

Kategori Jawaban	P	Q	R
Sangat setuju	4	5	80%
Setuju	1	5	20%
Cukup setuju	0	5	0%
Tidak setuju	0	5	0%
Sangat tidak setuju	0	5	0%

3 Apakah anda setuju dengan aplikasi ini dapat mengurangi kesalahan dalam melakukan pengisian absensi mahasiswa??

Tabel 4. Hasil Jawaban Pertanyaan No 3 Kuisioner B

Kategori Jawaban	P	Q	R
Sangat setuju	2	5	40%
Setuju	2	5	40%
Cukup setuju	1	5	20%
Tidak setuju	0	5	0%
Sangat tidak setuju	0	5	0%

4 Apakah anda setuju bahwa aplikasi ini layak digunakan?

Tabel 4. Hasil Jawaban Pertanyaan No 4 Kuisioner B

Kategori Jawaban	P	Q	R
Sangat setuju	1	5	20%
Setuju	3	5	60%
Cukup setuju	1	5	20%
Tidak setuju	0	5	0%
Sangat tidak setuju	0	5	0%

4.2.1.5 Kesimpulan pengujian Betha

Dari hasil pengujian betha yang telah dilakukan dapat ditarik kesimpulan bahwa *prototype* aplikasi akademik mahasiswa universitas komputer indonesia (unikom) berbasis teknologi *radio frequency identification* (rfid) dapat mengurangi kesalahan masiswa dalam melakukan pengisian absensi, intruksi yang digunakan nya baik, dalam pemrosesan data sangat cepat, memgelola informasi baik serta dapat membantu dalam mengelola data absensi mahasiswa dengan sesuai keinginan pengguna