

**PERBAIKAN *BUGS* PADA *UTILITY*  
PENCETAKAN DAFTAR HADIR KULIAH  
JURUSAN ILMU KOMPUTER  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS LAMPUNG**

(Skripsi)

Oleh:  
GABRIEL ERICK RASMANA



JURUSAN ILMU KOMPUTER  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS LAMPUNG  
2018

**ABSTRAK**  
**PERBAIKAN *BUGS* PADA *UTILITY***  
**PENCETAKAN DAFTAR HADIR KULIAH**  
**JURUSAN ILMU KOMPUTER**  
**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**  
**UNIVERSITAS LAMPUNG**  
  
**OLEH**  
**GABRIEL ERICK RASMANA**

Daftar hadir kuliah merupakan satu unsur yang dibutuhkan di dalam kegiatan perkuliahan. Beberapa kegunaannya yaitu, sebagai bukti pertanggungjawaban pelaksanaan kegiatan belajar mengajar bagi dosen, juga sebagai salah satu aspek penilaian bagi mahasiswa berdasarkan kehadirannya. Pada sistem *utility* ini dalam mencetak daftar hadir, data yang digunakan diambil dari SIAKAD (Sistem Informasi Akademik) UNILA (Universitas Lampung). SIAKAD UNILA adalah satu *software* yang digunakan di UNILA dalam mengurus kegiatan administrasi khususnya mahasiswa di UNILA. Setiap memulai semester baru di dalam perkuliahan setiap mahasiswa akan mengisi KRS (Kartu Rencana Studi), di mana KRS ini berisi daftar mata kuliah yang akan diambil oleh setiap mahasiswa. Semua daftar mahasiswa yang sudah memilih mata kuliah di KRS mereka, akan masuk ke dalam DNK (Daftar Nama Kelas). Data DNK ini yang akan digunakan sebagai sumber data untuk mencetak daftar hadir kuliah di dalam *utility* ini. Pada penelitian yang terdahulu sistem *utility* ini yaitu di Jurusan Ilmu Komputer Universitas Lampung, tetapi masih ditemukan beberapa *bugs* (masalah). *Bugs* tersebut sangat berpengaruh di dalam pencetakan daftar hadir kuliah di sistem *utility* ini. Dalam proses menyelesaikan *bugs* yang ditemukan di awal penelitian, ditemukan kembali beberapa *bugs* yang masih mengganggu berjalannya sistem dengan baik. Adapun sistem *utility* ini mencetak daftar hadir kuliah yang sesuai dengan kebutuhan administrasi di Jurusan Ilmu Komputer. *Cover*, pemberitahuan, matrikulasi teori, matrikulasi praktikum, daftar hadir teori, dan daftar hadir praktikum yang akan dicetak. Hasil cetakan bisa dicetak dalam dua jenis ukuran kertas yaitu ukuran A4 dan F4.

Kata kunci: *bugs* , *utility*, daftar hadir kuliah

**PERBAIKAN *BUGS* PADA *UTILITY*  
PENCETAKAN DAFTAR HADIR KULIAH  
JURUSAN ILMU KOMPUTER  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS LAMPUNG**

Oleh

*Gabriel Erick Rasmana*

**Skripsi**

Sebagai Salah Satu Syarat untuk Mencapai Gelar  
**SARJANA KOMPUTER**

Pada

**Jurusan Ilmu Komputer**

**Fakultas Matematika dan Ilmu Pengetahuan Alam**

**Universitas Lampung**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS LAMPUNG  
2018**

Judul Skripsi : **PERBAIKAN BUGS PADA UTILITY  
PENCETAKAN DAFTAR HADIR KULIAH  
JURUSAN ILMU KOMPUTER FAKULTAS  
MATEMATIKA DAN ILMU PENGETAHUAN  
ALAM UNIVERSITAS LAMPUNG**

Nama Mahasiswa : **Gabriel Erick Rasmana**

No. Pokok Mahasiswa : 1317032028

Jurusan : Ilmu Komputer

Fakultas : Matematika dan Ilmu Pengetahuan Alam

**MENYETUJUI**

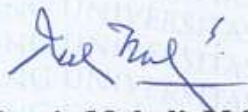
1. Komisi Pembimbing



**Dwi Sakethi, S.Si, M.Kom.**  
NIP. 19680611 199802 1 001

2. Mengetahui

Ketua Jurusan Ilmu Komputer  
FMIPA Universitas Lampung



**Dr. Ir. Kurnia Muludi, M.S.Sc.**  
NIP. 19640616 198902 1 001

## MENGESAHKAN

### 1. Tim Penguji

Ketua : **Dwi Sakethi, S.Si., M.Kom.**

Penguji I

Bukan Pembimbing : **Anie Rose Irawati, S.T., M.Cs.**

Penguji II

Bukan Pembimbing : **Drs. Rd. Irwan Adi Pribadi, M.Kom.**

### 2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam



**Drs. Suratman, M.Sc.**

NIP. 19640604 199003 1 002

Tanggal Lulus Ujian Skripsi : **14 Desember 2018**



## PERNYATAAN SKRIPSI MAHASISWA

Yang bertanda tangan di bawah ini:

Nama : Gabriel Erick Rasmana

NPM : 1317051028

Jurusan : Ilmu Komputer

Fakultas : Matematika dan Ilmu Pengetahuan Alam

Dengan ini menyatakan bahwa skripsi ini dibuat sendiri oleh penulis dan bukan merupakan hasil plagiat sebagaimana diatur dalam Pasal 36 Peraturan Akademik Universitas Lampung dengan Surat Keputusan Rektor Nomor: 06 Tahun 2016.

Bandar Lampung, 15 Mei 2019

Yang Menyatakan



Gabriel Erick Rasmana

NPM 1317051028

## **RIWAYAT HIDUP**



Penulis dilahirkan di Jakarta pada 25 Mei 1995, merupakan anak dari Bapak Jusup Tarigan dan Ibu Ukur Malem Br Sembiring. Tetapi dalam masa proses pendidikan penulis tinggal bersama dengan nenek bayang di Tanah Karo, Kabanjahe Sumatera Utara. Penulis menyelesaikan pendidikan di TK Kemala Bhayangkari 2

Kabanjahe, SD Negeri 4 Kabanjahe, SMP Negeri 2 Kabanjahe, dan SMA Swasta GBKP Kabanjahe pada tahun 2013.

Pada tahun 2013 penulis terdaftar sebagai mahasiswa Jurusan Ilmu Komputer di Universitas Lampung dan lolos melalui jalur Ujian SBMPTN.

Selama menjadi mahasiswa di Jurusan Ilmu Komputer, penulis juga aktif dalam organisasi di dalam dan diluar kampus. Di dalam kampus penulis ikut aktif menjadi anggota bidang eksternal kepengurusan HIMAKOM (Himpunan Mahasiswa Jurusan Ilmu Komputer) periode 2015/2016. Selain aktif di dalam organisasi kemahasiswaan di dalam kampus penulis juga aktif dalam organisasi diluar kampus antara lain:

- IMKA Rudang Mayang Lampung (Ikatan Mahasiswa Karo Rudang Mayang Lampung) periode 2016/2017 sebagai anggota bidang pendidikan dan periode 2017/2019 sebagai coordinator bidang HUMAS (Hubungan Masyarakat).

- Organisasi gereja di Gereja Batak Karo Protestan (GBKP) Bandar Lampung sebagai Guru Sekolah Minggu, dan juga terpilih sebagai Wakil Ketua di Badan Pengurus Harian (BPH) Sekolah Minggu periode (2014/2019).

Pada bulan februari 2017 melaksanakan Kerja Praktek di BPOM (Badan Pengawasan Obat dan Makanan) Provinsi Lampung. Pada bulan agustus tahun 2017 penulis melaksanakan KKN (Kuliah Kerja Nyata) di desa Gadingrejo, Pringsewu. Pada tahun 2017 penulis melakukan penelitian di Jurusan Ilmu Komputer sebagai syarat untuk mendapatkan gelar sarjana di Jurusan Ilmu Komputer di bawah bimbingan Bapak Dwi Sakethi, S.Si., M.Kom.



**MOTTO**

**“Be The Best In Your Life”**

***“Tetaplah Berdoa”***

**1 Tesalonika 5:17**

## **PERSEMBAHAN**

Puji dan Syukur Saya Ucapkan Kepada Tuhan Yesus Kristus

*Kupersembahkan karya ini untuk  
Jurusan Ilmu Komputer Universitas Lampung, Kepada Orang tua tercinta  
Juga kepada keluarga besar Ilmu Komputer 2013*

*Almamater Tercinta  
Universitas Lampung*

## SANWACANA

Syalom dan Salam Sejahtera

Puji syukur atas kehadiran Tuhan Yang Maha Esa, atas kasihNya yang setia dan juga berkatNya yang melimpah sehingga penulis dapat menyelesaikan penelitian dan menuangkannya di dalam karya tulis ilmiah di Jurusan Ilmu Komputer Universitas Lampung ini.

Skripsi ini ditulis oleh penulis dengan judul “PERBAIKAN *BUGS* PADA *UTILITY* PENCETAKAN DAFTAR HADIR KULIAH JURUSAN ILMU KOMPUTER FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM UNIVERSITAS LAMPUNG”.

Dalam penyusunan skripsi ini penulis meyakini walaupun kerap menghadapi kendala baik kendala oleh diri penulis itu sendiri ataupun dari pihak yang lain tetapi puji syukur kepada Tuhan akhirnya kewajiban ini dapat diselesaikan. Ini semua juga berkat bantuan dan dukungan juga semangat oleh berbagai pihak. Untuk itu melalui kesempatan ini, penulis ingin mengucapkan terima kasih kepada:

1. Tuhan Yang Maha Esa atas berkat dan kasih penyertaan-Nya.
2. Bapak Dwi Sakethi, S.Si., M.Kom selaku pembimbing utama dalam skripsi ini, yang sudah sangat baik dan sabar juga kerap memberikan ide masukan dalam kelancaran penulisan skripsi ini. Terima kasih Pak, Tuhan memberkati Bapak.

3. Ibu Anie Rose Irawati, S.T., M.Cs. selaku pembahas dalam skripsi ini, yang juga memberikan saran dan masukan dalam penulisan skripsi ini. Terima kasih Bu. Tuhan memberkati Ibu.
4. Bapak Drs. Rd. Irwan Adi Pribadi, M.Kom selaku pembahas dalam penulisan skripsi ini, yang juga banyak memberikan saran dan masukan. Terima kasih Pak. Tuhan memberkati Bapak.
5. Bapak Ir. Machudor Yusman, M.Kom selaku dosen pembimbing akademik penulis, yang juga sering memberikan masukan dan semangat kepada penulis.
6. Dr. Ir. Kurnia Muludi, M.Sc selaku ketua Jurusan Ilmu Komputer Universitas Lampung.
7. Bapak Drs. Suratman, M. Sc. selaku Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
8. Bapak dan Ibu Dosen beserta seluruh staff yang ada di lingkungan Jurusan Ilmu Komputer Universitas Lampung, yang sudah membantu, membimbing, menasihati dan juga mau memberikan ilmu terhadap penulis sehingga sampai di tahap ini.
9. Kepada Orang Tua tercinta, yang selalu memberi semangat dan tidak jemu-jemuanya bertanya apa kesulitan kendala dan tetap memberikan *support* kepada penulis untuk bisa menyelesaikan skripsi ini.
10. Kepada nenek Bayang dan Bik Uda serta semua Mama dan Mami yang selalu memberikan semangat untuk bisa menyelesaikan skripsi ini.
11. Kepada paman dan tante, Mama Bp. Abram Ginting dan Mami Nd. Abram Ginting yang selalu memberikan semangat, dan tidak lelah mengingatkan penulis

supaya bisa menyelesaikan setiap tanggung jawab. Dan juga terima kasih sudah mau menjadi orang tua yang baik bagi penulis selama ada di tanah perantauan ini.

12. Para sahabat, rekan sepelayanan, juga Komisi KAKR abang Pt. Gelora Sinuhaji di KAKR GBKP Bandar Lampung, yang juga selalu menjadi tempat berbagi dan sumber motivasi bagi penulis.
  13. Para sahabatku Roma Rio Sitepu, Romario Sihombing yang sudah terlebih dahulu lulus dan tetap memberikan semangat. Dan tidak lupa kepada sahabatku Rido Afriza yang akhirnya berdua bersama saling menyemangati untuk mendapatkan gelar sarjana.
  14. Saudara-saudariku di PERMATA GBKP Bandar Lampung dan juga IMKA Rudang Mayang Lampung, tempat penulis bisa mengembangkan diri dan juga tempat penulis bisa mendapat begitu banyak pengalaman.
  15. Teman-teman di Jurusan Ilmu Komputer angkatan 2013 terkhusus yang bersama menjalani masa-masa penulisan skripsi yang walaupun sudah banyak ditinggal oleh teman-teman yang sudah lulus tetapi bisa saling memberikan semangat.
- Penulis menyadari bahwa dalam penulisan skripsi ini masih jauh dari kata sempurna. Karena itu, penulis sangat berterima kasih apabila diberikan saran dan kritik yang dapat membangun. Dan juga penulis berharap semoga melalui tulisan dan karya dalam penelitian ini bisa bermanfaat terkhususnya bagi Jurusan Ilmu Komputer Universitas Lampung.

Bandar Lampung, 15 Mei 2019

Gabriel Erick Rasmana

NPM 1317051028



## DAFTAR ISI

	Halaman
<b>ABSTRAK</b> .....	i
<b>HALAMAN JUDUL</b> .....	ii
<b>LEMBAR PERSETUJUAN</b> .....	iii
<b>LEMBAR PENGESAHAN</b> .....	iv
<b>LEMBAR PERNYATAAN</b> .....	v
<b>RIWAYAT HIDUP</b> .....	vi
<b>MOTTO</b> .....	viii
<b>PERSEMBAHAN</b> .....	ix
<b>SANWACANA</b> .....	x
<b>DAFTAR ISI</b> .....	xiv
<b>DAFTAR TABEL</b> .....	xvii
<b>DAFTAR GAMBAR</b> .....	xviii
<b>I. PENDAHULUAN</b> .....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	4
1.3 Batasan Masalah.....	4
1.4 Tujuan.....	5

<b>II. TINJAUAN PUSTAKA.....</b>	<b>7</b>
2.1 Penelitian Terdahulu.....	7
2.2 <i>Software Utility</i> .....	10
2.3 SIAKAD.....	11
2.4 HTML.....	11
2.5 PHP.....	12
2.6 PHP <i>Excel Reader</i> .....	13
2.7 FPDF Library.....	13
2.8 Entity Relationship Diagram(ERD).....	15
2.9 Data Flow Diagram (DFD).....	16
2.10 Metode <i>Waterfall</i> .....	17
2.11 <i>Blacbox Testing</i> .....	18
2.12 <i>Equivalence Partitioning (EP)</i> .....	18
2.13 Skala Guttman.....	19
<b>III. METODE PENELITIAN.....</b>	<b>20</b>
3.1 Tempat Pengembangan.....	20
3.2 Tahapan Pengembangan.....	20
<b>IV. HASIL DAN PEMBAHASAN.....</b>	<b>34</b>
4.1 Hasil.....	34
4.2 Pembahasan.....	44
<b>V. SIMPULAN DAN SARAN.....</b>	<b>70</b>
5.1 Simpulan.....	70
5.2 Saran.....	71

DAFTAR PUSTAKA

LAMPIRAN

## DAFTAR TABEL

Tabel	Halaman
3.1 Rancangan Pengujian Sistem.....	25
4.1 Hasil Pengujian Equivalence Partitioning Fungsi Masukkan DNK..	45
4.2 Hasil Pengujian Equivalence Partitioning Fungsi Cetak Daftar Hadir Kuliah.....	47
4.3 Hasil Pengujian Equivalence Partitioning Fungsi Tambah, Ubah Hapus Data Dosen.....	50
4.4 Hasil Pengujian Equivalence Partitioning Fungsi Tambah dan Ubah Data Verifikator.....	52
4.5 Pengujian Equivalence Partitioning Fungsi Tambah dan Ubah Data Jurusan.....	53
4.6 Hasil Pengujian Equivalence Partitioning Fungsi Tambah dan Ubah Data Program Studi.....	55
4.7 Hasil Pengujian Equivalence Partitioning Fungsi Kelola Data Detail Mata Kuliah.....	57
4.8 Hasil Pengujian.....	60
4.9 Presentase Pengujian Fungsional.....	65

## DAFTAR GAMBAR

Gambar	Halaman
1.1 Cetak <i>Cover</i> Presensi.....	2
1.2 Menambahkan Data Dosen Pada Mata Kuliah.....	3
2.1 Tampilan Halaman Utama <i>Utility</i> Pecetakan Daftar Hadir Kuliah...	7
2.2 Tampilan Halaman Menu Masukkan DNK.....	7
2.3 Tampilan Halaman Menu Cetak Daftar Hadir Kuliah.....	8
2.4 Tampilan Halaman Daftar Mata Kuliah Sesuai Pilihan Admin.....	9
2.5 Tampilan Halaman Pilihan Cetak Daftar Hadir.....	10
2.6 Simbol- simbol ERD.....	15
2.7 Metode <i>Waterfall</i> .....	18
3.1 Tahapan Pengembangan.....	20
3.2 <i>Entity Relationship Diagram</i> (ERD).....	23
3.3 <i>Data Flow Diagram</i> (DFD) Level 0 ( <i>Context Diagram</i> ).....	24
4.1 Tampilan Beranda (Halaman Utama).....	35
4.2 Tampilan Kelola Data Master.....	36
4.3 Tampilan Kelola Data Master (Dosen).....	37
4.4 Tampilan Kelola Data Master (Verifikator).....	38
4.5 Tampilan Kelola Data Master (Jurusan).....	38

4.6 Tampilan Kelola Data Master (Program Studi).....	39
4.7 Tampilan Kelola Data Master (Mata Kuliah).....	40
4.8 Tampilan Masukkan DNK.....	41
4.9 Tampilan Proses Cetak Daftar Hadir.....	43



## **I. PENDAHULUAN**

### **1.1. Latar Belakang**

Kegiatan perkuliahan merupakan salah satu kegiatan akademik utama yang dilaksanakan oleh setiap perguruan tinggi. Selama pelaksanaan perkuliahan dibutuhkan beberapa dokumen pendukung aktifitas, seperti daftar kehadiran. Daftar kehadiran ini dibutuhkan oleh setiap Jurusan sebagai salah satu bahan pertanggungjawaban pelaksanaan kegiatan belajar mengajar setiap dosen dan mahasiswa. Daftar hadir juga berfungsi untuk mencatat kehadiran peserta belajar selama perkuliahan dan beberapa informasi penting terkait kegiatan pembelajaran.

Di Universitas Lampung, pembuatan daftar hadir dapat memanfaatkan data peserta kuliah di SIAKAD (Sistem Informasi Akademik) Universitas Lampung. Pada dasarnya kegunaan dari SIAKAD adalah suatu *software* yang berguna untuk mengurus kegiatan administrasi di Universitas Lampung. Oleh sebab itu data yang diambil dari SIAKAD pasti lebih valid. Banyak bagian-bagian yang ada di SIAKAD, salah satunya yaitu DNK.

DNK (Daftar Nama Kelas) merupakan catatan atau daftar nama mahasiswa yang mengambil suatu mata kuliah tertentu. Data DNK sudah terisi apabila

mahasiswa sudah mengisi KRS (Kartu Rencana Studi). Oleh karena itu DNK bisa dijadikan suatu media yang lebih valid dalam pencetakan daftar hadir.

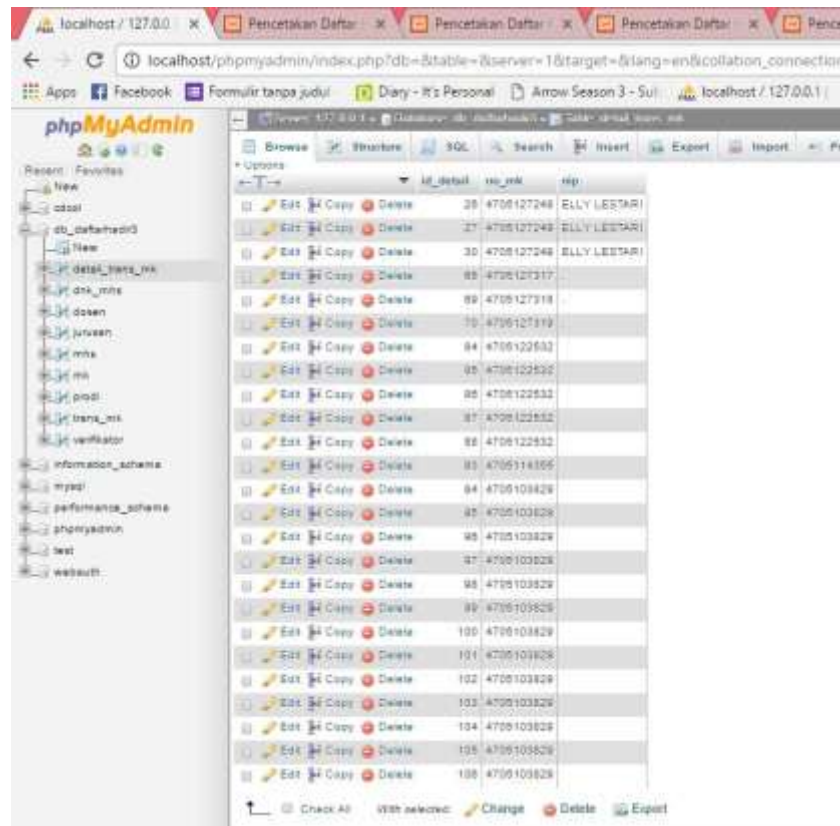
Di Jurusan Ilmu Komputer FMIPA Universitas Lampung, sebenarnya sudah ada *utility* untuk pencetakan daftar hadir (Kadir, 2015) tetapi dalam hal ini masih ada terdapat beberapa *bugs* yang ada. Beberapa *bugs* yang dapat ditemukan pada *utility* tersebut antara lain:

1. Pada saat pencetakan *cover* dari presensi, pada bagian kelas dan angkatan tidak dapat dibaca secara otomatis oleh sistem. Hal ini bisa diperhatikan pada Gambar 1.1.



Gambar 1.1 Cetak *Cover* Presensi.

2. Dalam menambahkan data dosen pada mata kuliah yang ada juga harus dilakukan secara manual melalui *localhost*. Seperti yang bisa diperhatikan pada Gambar 1.2.



Gambar 1.2 Menambahkan Data Dosen Pada Mata Kuliah.

Oleh sebab itulah penelitian ini berfokus pada perbaikan *bugs* dan penambahan beberapa fungsi yang dibutuhkan. Karena dengan adanya *utility* ini dapat mempermudah administrasi dalam masalah pencetakan daftar hadir.

3. Tambahan informasi bahwa pada sistem operasi Linux, upload DNK akan bermasalah jika hak akses pada folder tidak dibuat rw- jadi waktu *upload* mesti diberi keterangan.

## **1.2. Rumusan Masalah**

Permasalahan yang dikaji dalam penelitian ini yaitu bagaimana memperbaiki *bugs* yang ada dan meningkatkan fungsi yang lebih banyak untuk kemudahan dalam pengoperasian sistem.

## **1.3. Batasan Masalah**

Adapun batasan masalah dalam penelitian ini adalah:

1. Menambahkan fungsi untuk membaca jenis kelas dari DNK yang ada.
2. Menambahkan fungsi untuk membaca angkatan yang ada di dalam DNK berdasarkan data yang ada
3. Menambahkan fungsi untuk menambah nama dosen berdasarkan NIP di setiap mata kuliah yang diampu.
4. Menambahkan fungsi untuk menambah jadwal mata kuliah.
5. Memperbaiki *error/bugs* yang ditemukan apabila mengganggu proses untuk pencetakan daftar hadir kuliah, tetapi tidak sampai melewati batasan masalah yang ada.
6. Studi kasus pengembangan sistem dilakukan pada Jurusan Ilmu Komputer.

#### **1.4. Tujuan**

Tujuan dari penelitian ini yaitu untuk memperbaiki masalah/*bugs* yang ada pada sistem *utility* yang telah ada. Dengan memperbaiki masalah/*bugs* pada sistem *utility* ini diharapkan akan dapat memberikan manfaat sesuai kebutuhan yang ada.

## **II. TINJAUAN PUSTAKA**

### **2.1 Penelitian Terdahulu**

Sebelumnya telah dilakukan penelitian yang sama oleh Aqillah Aliffah Kadir yang merupakan mahasiswa angkatan 2011 dan alumni dari Jurusan Ilmu Komputer Universitas Lampung pada tahun 2015.

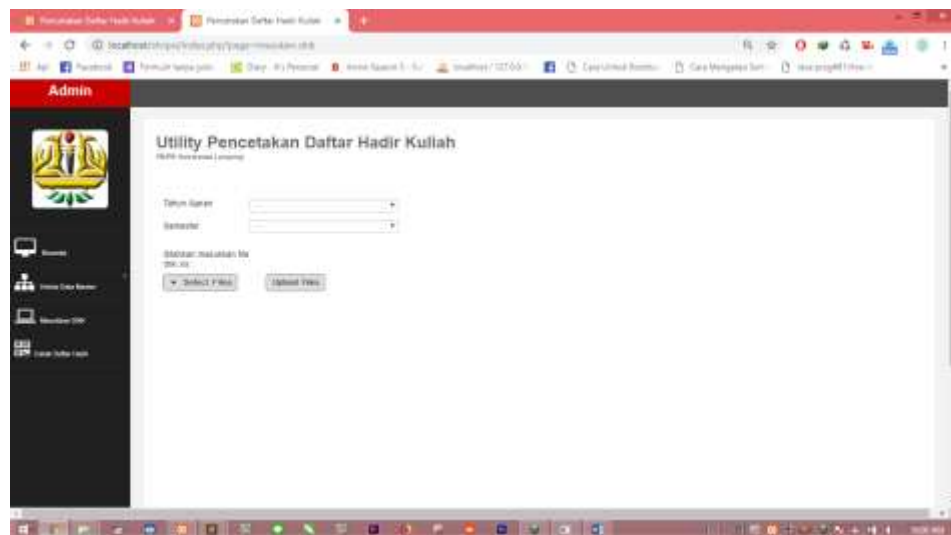
*Utility* Pencetakan Daftar Hadir Kuliah ini terdiri dari 4 menu utama, yaitu: menu beranda, menu kelola data master, menu masukkan DNK, dan menu cetak daftar hadir. Pada menu kelola data master, terdiri dari sub menu kelola data dosen, kelola data verifikator, kelola data Jurusan, kelola data program studi, dan kelola data detail mata kuliah. Gambar 2.1 merupakan tampilan halaman utama dari *utility* pencetakan daftar hadir kuliah ini.





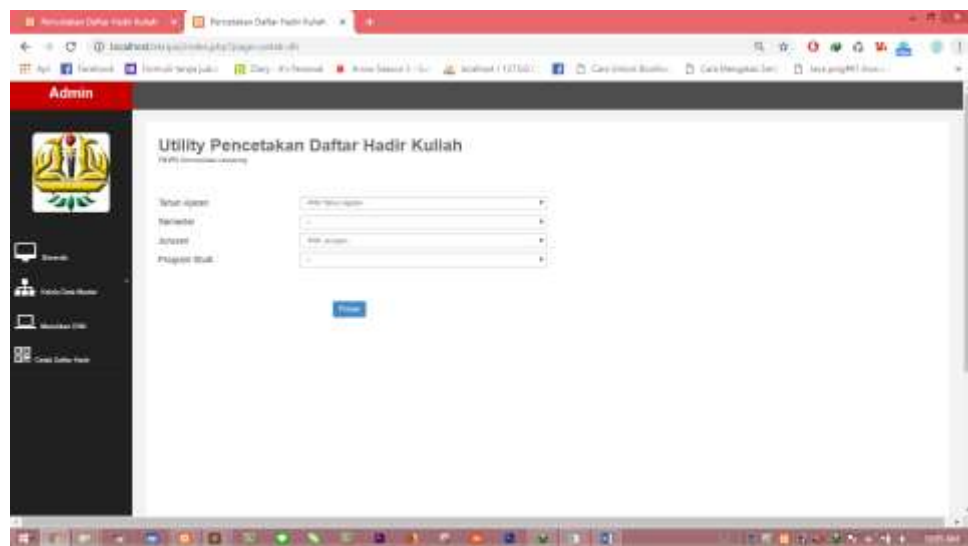
Gambar 2.1 Tampilan Halaman Utama *Utility* Pencetakan Daftar Hadir Kuliah.

Untuk fungsi utama sistem adalah memasukkan DNK dan melakukan pencetakan daftar hadir. Gambar 2.2 merupakan tampilan halaman menu masukkan DNK.



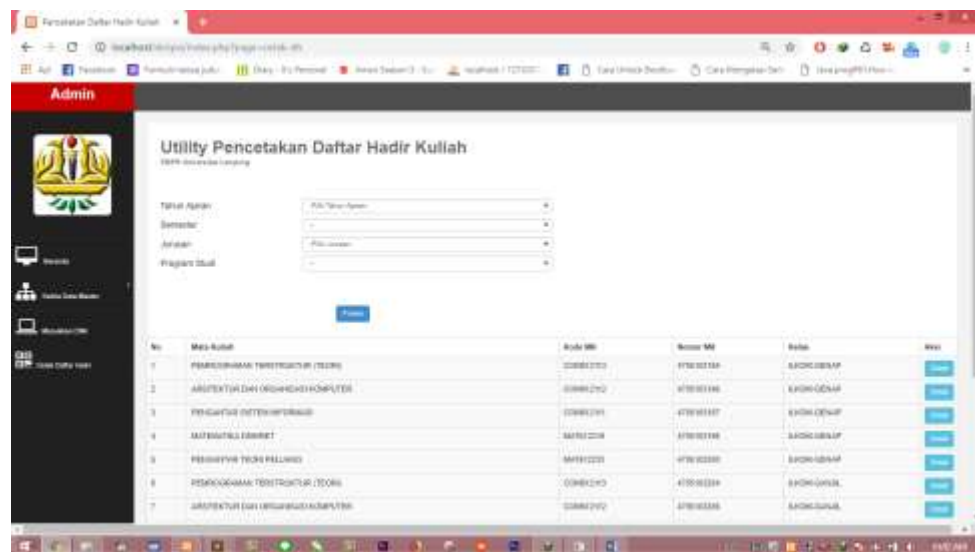
Gambar 2.2 Tampilan Halaman Menu Masukkan DNK.

Pada Gambar 2.2 menunjukkan tampilan halaman menu masukkan DNK. Sebelum memasukkan dnk.xls, admin harus memasukkan pilihan tahun ajaran dan pilihan semester terlebih dahulu. Pilihan tahun ajaran dan semester yang dimasukkan harus sesuai dengan DNK yang dimasukkan dan kemudian diproses untuk disimpan ke dalam database. Pada proses ini data dari dnk.xls di-import sebagai sumber data untuk membuat daftar hadir yang dibutuhkan. Setelah memasukkan DNK, admin dapat melakukan pengolahan data mengenai nama dosen penanggung jawab dan detail dari mata kuliah melalui menu kelola data mata kuliah. Apabila semua data untuk membuat daftar hadir yang dibutuhkan telah lengkap, maka pencetakan daftar hadir dapat dilakukan. Tampilan halaman dari menu cetak daftar hadir dapat dilihat pada Gambar 2.3.



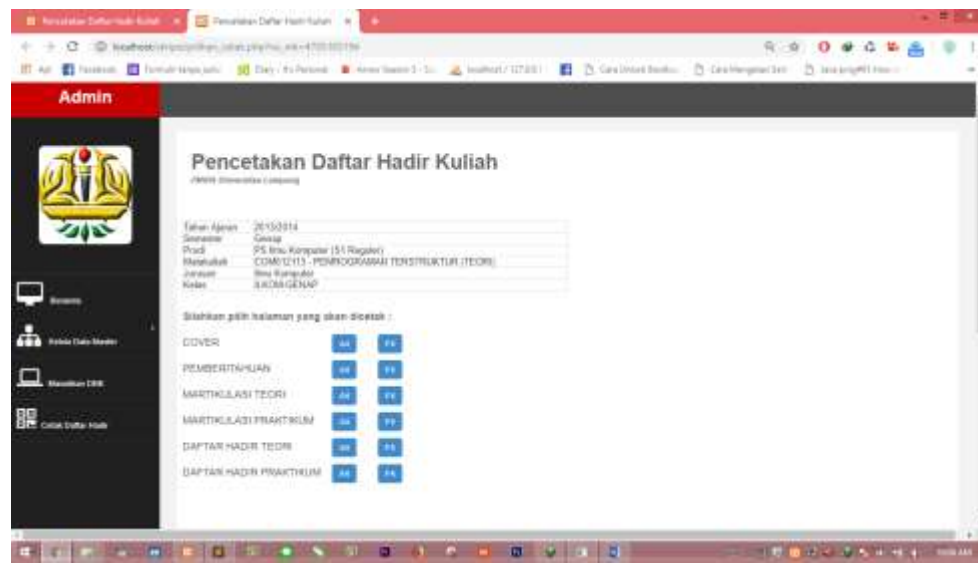
Gambar 2.3 Tampilan Halaman Menu Cetak Daftar Hadir Kuliah.

Pada Gambar 2.3, menunjukkan tampilan halaman menu cetak daftar hadir kuliah. Pada halaman ini, admin diminta untuk memasukkan pilihan tahun ajaran, semester, jurusan, dan program studi. Pada saat pilihan telah dimasukkan dan diproses, maka sistem akan merespond dengan memberikan daftar nama mata kuliah yang sesuai dengan tahun ajaran, semester, jurusan, dan program studi yang telah dipilih oleh admin. Gambar 2.4, menunjukkan tampilan halaman daftar mata kuliah.



Gambar 2.4 Tampilan Halaman Daftar Mata Kuliah Sesuai Pilihan Admin.

Gambar 2.4, menunjukkan tampilan halaman daftar mata kuliah yang disesuaikan dengan pilihan admin. Pada halaman ini, admin dapat memilih mata kuliah yang akan dicetak daftar hadirnya. Langkah selanjutnya, admin masuk ke halaman pilihan pencetakan. Gambar 2.5, merupakan tampilan halaman pilihan cetak daftar hadir.



Gambar 2.5 Tampilan Halaman Pilihan Cetak Daftar Hadir.

Pada Gambar 2.5, menunjukkan tampilan halaman pilihan cetak. Admin dapat memilih pilihan ukuran kertas dari daftar hadir yang akan dicetak yaitu dengan ukuran A4 atau F4 sesuai dengan kebutuhan. Dari proses ini, ditampilkan halaman PDF dari halaman-halaman daftar hadir tersebut. Admin dapat melakukan proses cetak melalui halaman PDF ini.

Telah dilakukan pengujian pada hasil dari penelitian tersebut. Pengujian yang dilakukan yaitu pengujian *black-box testing* dengan metode *Equivalence Partitioning (EP)*. Dan hasil yang didapatkan dari pengujian tersebut adalah sistem *utility* ini dapat berjalan dengan baik sesuai dengan kriteria yang dibutuhkan.

## 2.2 Software Utility

*Software utility* adalah sebuah *software* yang dirancang untuk melakukan tugas tertentu. *Sotware utility* hanya dapat membantu user untuk melakukan

sesuai dengan tujuan perancangan *software utility* tersebut. Adapun contoh dari *software utility* adalah *diagnostic tools and maintenance tools, setup wizard, communications program, security software , device drivers, and apps* (Parsons dkk, 2016).

### 2.3 **SIKAD (Sistem Informasi Akademik)**

Sistem informasi akademik merupakan akses utama untuk mengatur segala urusan perkuliahan dan hal-hal lainnya yang berkaitan dengan akademik. Sistem informasi akademik ini merupakan salah satu pelayanan publik bagi dosen, mahasiswa, dan karyawan guna mempertingkat kinerja mereka. Sistem informasi akademik mempunyai komponen yang sama dengan sistem informasi lainnya. Komponen sistem informasi yaitu *hardware, software , data, prosedur, dan manusia* (Danella, 2014).

### 2.4 **HTML**

HTML adalah kumpulan dari simbol *markup* atau kode yang disisipkan dalam suatu *file* untuk menampilkan isi *file* tersebut pada *browser*. Adapun penulisan kode-kode tersebut ditandai dengan *tag*. *Tag* ini akan selalu berpasangan yang menandakan kapan efek tampilan mulai dan berakhir. Adapun contoh dari *tag* itu adalah,

```
<html>
  <head>
    <title>

    </title>
```

```
        </head>
    <body>
        .....
    </body>
</html>
```

## 2.5 PHP

PHP singkatan dari *PHP Hypertext Preprocessor* yaitu bahasa pemrograman *web server-side* yang bersifat *open source*. PHP merupakan *script* yang menyatu dengan HTML dan berada pada *server* (*server side HTML embedded scripting*). PHP adalah *script* yang digunakan untuk membuat halaman web yang dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat saat halaman itu diminta oleh *client*. Mekanisme ini menyebabkan informasi yang diterima *client* selalu yang terbaru/ *up to date*. Semua *script* PHP dieksekusi pada *server* dimana *script* tersebut dijalankan (Anhar, 2010). Kode PHP diawali dengan tanda lebih kecil (<) dan diakhiri dengan tanda lebih besar (>). Ada empat cara untuk menuliskan skrip PHP, yaitu:

- a. <? Hallo ?>
- b. <? php Hallo ?>
- c. <% Hallo %>
- d. <SCRIPT LANGUAGE="php">Hallo</script>

Pemisah antara instruksi adalah titik koma (;) dan untuk membuat atau menambahkan komentar/standar penulisan adalah,

- 1. /\*komentar\*/
- 2. // komentar



3. # komentar.

Cara penulisan dibedakan menjadi *Embedded* dan *Non-Embedded script* (Nazrul, 2013).

## **2.6 PHP Excel Reader**

PHP *Excel Reader* merupakan salah satu *library* yang ada di PHP yang berguna untuk membaca *file* yang bertipe xls atau xlsx.

## **2.7 FPDF Library**

Menurut (Erawan, 2007) *software* bantu FPDF merupakan pustaka (*library*) yang dibuat oleh Oliver Plathey untuk membuat dokumen PDF melalui pemrograman PHP dengan mudah, lengkap dengan pengaturan *header*, *footer*, gambar, dan penomoran halaman. Pustaka FPDF berbentuk *open source*.

### **2.7.1 Metode- metode dalam penggunaan FPDF Library**

Perintah- perintah pustaka FPDF didefinisikan oleh metod-metode kelas PDF yang berada pada *file* utama fpdf.php. Metode-metode ini diperlukan dalam program. *File* ini akan dipanggil pada setiap program yang menggunakan pustaka FPDF. Metode-metode tersebut adalah sebagai berikut:

#### **1. Open**

Metode ini akan membangkitkan dokumen PDF. Pengguna metode ini bersifat optional sebab metode *AddPage* secara otomatis melakukan tugas ini. Metode ini tanpa parameter.

## 2. *AddPage*

Metode *AddPage* akan membuat sebuah halaman baru dalam dokumen. Jika metode *Footer* dan *Header* sudah didefinisikan, metode–metode ini akan dipanggil ketika halaman baru sudah terbentuk. Sintaks metode:

```
AddPage ( [string orientation] )
```

## 3. **FPDF**

Merupakan kelas konstruktor. Dalam kelas ini, dapat diatur format, orientasi, dan satuan ukuran yang digunakan dalam seluruh metode (kecuali untuk ukuran *font* ). Sintaks konstruktor ini:

```
FPDF ( [string orientation [, string unit [, mixed format]]] )
```

## 4. *SetFont*

Menentukan *font* yang digunakan untuk mencetak. Metode ini harus digunakan paling tidak sekali sebelum mencetak atau hasil cetakan tidak akan valid. Jenis *font* standar atau jenis *font* yang ditambahkan melalui metode *AddFont* . Sintaks metode ini:

```
SetFont (string family [, string style [, float size] ] )
```

## 5. *Cell*

Mencetak suatu *cell* (area berbentuk empat persegi panjang) dengan optional *border*, warna *background*, dan *string* karakter. Sintaks metode ini:

```
Cell (float w[, float h[, string txt[, mixed border[, int ln[, string align[, int fill[, mixed link] ] ] ] ] ] ] )
```

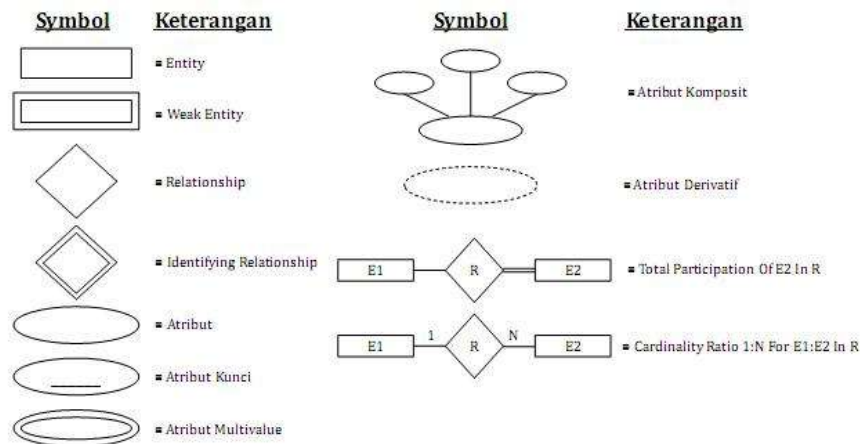
## 6. Output

Metode ini akan mengirim dokumen ke tujuan yang ditentukan berupa *string*, *file* lokal atau *browser*. Sintaks metode ini:

```
String Output ( [ string name [, string dest] ] ).
```

### 2.8 Entity Relationship Diagram (ERD)

*Entity Relationship Diagram* (ERD) adalah bagian yang menunjukkan hubungan antara *entity* yang ada dalam sistem. Ataupun dapat dikatakan sebagai suatu model jaringan kerja (*network*) yang menguraikan susunan data yang disimpan dari sistem secara abstrak (Yulianto dkk, 2015). Adapun simbol-simbol yang digunakan dalam ERD bisa dilihat di Gambar 2.6.



Gambar 2.6 Simbol-simbol ERD.

## 2.9 *Data Flow Diagram (DFD)*

*Data Flow Diagram (DFD)* adalah alat pembuatan model yang memungkinkan profesional sistem untuk menggambarkan sistem sebagai suatu jaringan proses fungsional yang dihubungkan satu sama lain dengan alur data, baik secara manual maupun komputerisasi. DFD ini sering disebut juga dengan nama *Bubble chart*, *Bubble diagram*, model proses, diagram alur kerja, atau model fungsi.

DFD ini adalah salah satu alat pembuatan model yang sering digunakan, khususnya bila fungsi-fungsi sistem merupakan bagian yang lebih penting dan kompleks dari pada data yang dimanipulasi oleh sistem. Dengan kata lain, DFD adalah alat pembuatan model yang memberikan penekanan hanya pada fungsi sistem.

DFD ini merupakan alat perancangan sistem yang berorientasi pada alur data dengan konsep dekomposisi dapat digunakan untuk penggambaran analisa maupun rancangan sistem yang mudah dikomunikasikan oleh profesional sistem kepada pemakai maupun pembuat program.

Di dalam DFD terdapat 3 level, yaitu :

### 1. Diagram Konteks :

menggambarkan satu lingkaran besar yang dapat mewakili seluruh proses yang terdapat di dalam suatu sistem. Merupakan tingkatan tertinggi dalam DFD dan biasanya diberi nomor 0 (nol). Semua entitas eksternal

yang ditunjukkan pada diagram konteks berikut aliran-aliran data utama menuju dan dari sistem. Diagram ini sama sekali tidak memuat penyimpanan data dan tampak sederhana untuk diciptakan.

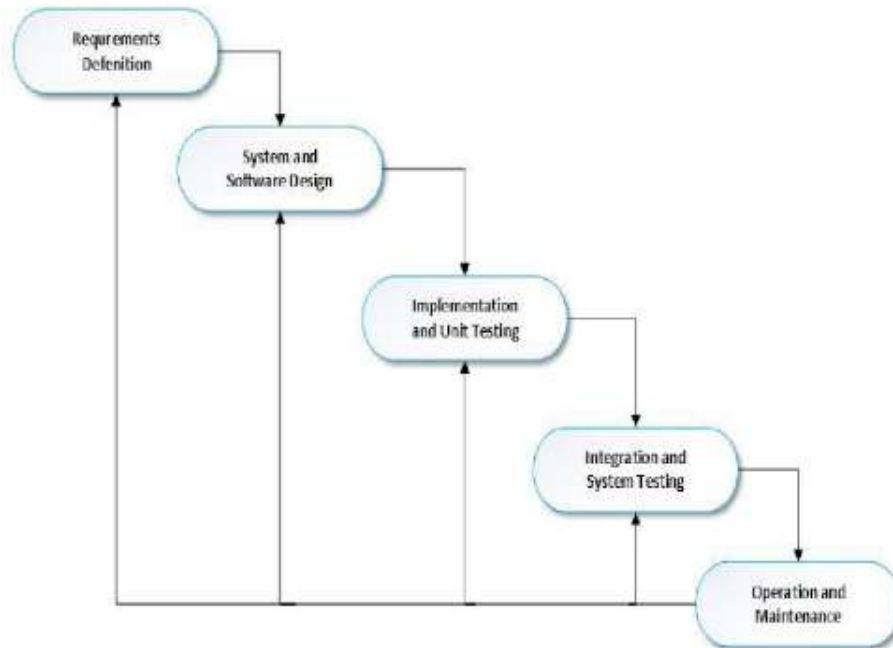
2. Diagram Nol (diagram level-1) : merupakan satu lingkaran besar yang mewakili lingkaran-lingkaran kecil yang ada di dalamnya. Merupakan pemecahan dari Diagram Konteks ke Diagram Nol. di dalam diagram ini memuat penyimpanan data.
3. Diagram Rinci : merupakan diagram yang menguraikan proses apa yang ada dalam Diagram Nol.

#### **2.10 Metode *Waterfall***

Menurut (Arbani, 2011) metode *Waterfall* adalah suatu metode dalam pengembangan sistem informasi yang dilakukan dalam beberapa tahap dan dilakukan secara berurutan. Tahapan-tahapan yang ada dalam Metode *Waterfall* adalah:

1. Analisis Kebutuhan
2. Desain Sistem
3. Penulisan Kode Program
4. Pengujian Program
5. Penerapan Program dan Pemeliharaan

Ataupun untuk lebih jelas dapat dilihat pada Gambar 2.7 (Muharto, 2016).



Gambar 2.7. Metode *Waterfall*.

### 2.11 *Blackbox Testing*

*Blackbox Testing* merupakan metode pengujian yang dilakukan dengan mengetahui fungsi yang seharusnya didapatkan ketika sistem atau produk dirancang. Pengujian dilakukan dengan memperlihatkan apakah setiap fungsi yang ada berjalan dengan sepenuhnya dan pada waktu yang sama juga mencari kesalahan-kesalahan pada setiap fungsi.

### 2.12 *Equivalence Partitioning (EP)*

*Equivalence Partitioning (EP)* adalah teknik pengujian *blackbox testing* yang berkaitan dengan mengidentifikasi kelas- kelas input dimana perilaku

perangkat lunakya serupa. Secara sederhana *Equivalence Partitioning* yaitu ketika dimasukkan pengujian yang sama seharusnya akan menampilkan hasil yang sama juga. Jikalau pada bagian ini, hasil dari inputan berbeda maka masih ada masalah pada bagian yang diuji. (Ould, 1986)

### **2.13 Skala Guttman**

Skala Guttman adalah skala kumulatif yang menggambarkan sikap seorang pada suatu hal, kondisi atau situasi melalui pilihan jawaban tegas, biasanya terdiri dari dua pilihan jawaban seperti Ya-Tidak, Setuju-Tidak Setuju, Pernah-Tidak Pernah, dan lain sebagainya. (Suryani, 2015)

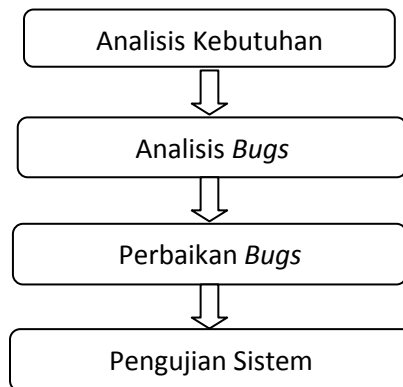
### III. METODE PENELITIAN

#### 3.1 Tempat Pengembangan

Pengembangan sistem *utility* ini dilakukan di Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.

#### 3.2 Tahapan Pengembangan

Penelitian ini dilakukan dengan melakukan beberapa tahapan yaitu pengumpulan data, analisis *bugs* dan melakukan perbaikan pada *bugs* tersebut, ataupun dapat dilihat pada Gambar 3.1.



Gambar 3.1 Tahapan Pengembangan.

##### 3.2.1 Analisis Kebutuhan

Adapun analisis kebutuhan dari *utility* ini terdiri atas kebutuhan fungsional dan non fungsional.



#### A. Kebutuhan Fungsional

- a. Sistem digunakan oleh Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
- b. Sistem dapat menerima data yang bersumber dari DNK (Daftar Nilai Kelas) Siakad Universitas Lampung berupa *file* *dnk.xls* (*Excel*).
- c. Sistem dapat melakukan *import* data dari format *file Excel* untuk mendapatkan informasi nama mahasiswa beserta NPM (Nomor Pokok Mahasiswa), tahun ajaran, dan semester.
- d. Daftar hadir yang tercetak dilengkapi dengan data nama mata kuliah yang dipilih beserta kode mata kuliah, SKS dari mata kuliah, tahun ajaran, semester, nama dosen penanggung jawab, nama-nama mahasiswa peserta perkuliahan, nama ketua Jurusan, nama TPMPS beserta dengan NIP, nama prodi, dan nama Jurusan.
- e. Sistem dapat menghasilkan cetakan daftar hadir sesuai ukuran kertas yaitu ukuran kertas A4 dan F4.

#### B. Kebutuhan Non Fungsional

Adapun kebutuhan non fungsional yang akan dibutuhkan untuk melakukan pengembangan *utility* ini adalah sebagai berikut:

- a. Kebutuhan Kinerja :

1. Pada daftar hadir yang dicetak telah dilengkapi dengan data kode mata kuliah beserta nama mata kuliah, dosen penanggung jawab, tahun ajaran, semester, nama prodi, nama Jurusan, nama kelas, nama angkatan nama ketua Jurusan, nama ketua TPMPS beserta NIP akan ditampilkan dengan jelas dan lengkap pada setiap halaman *cover*, matrikulasi, dan daftar hadir.
2. Memudahkan user untuk mencetak daftar hadir mata kuliah, tanpa perlu lagi khawatir apabila adalah formata yang berbeda dari hasil cetakan sebelumnya.

b. Kebutuhan Keamanan :

1. Pengguna yang dapat mengakses sistem ini hanya Sekretaris Jurusan atau Staff Jurusan. Karena sistem *utility* ini akan di-*install* secara *stand alone* dan tidak akan di-*hosting*.

c. Kebutuhan Informasi

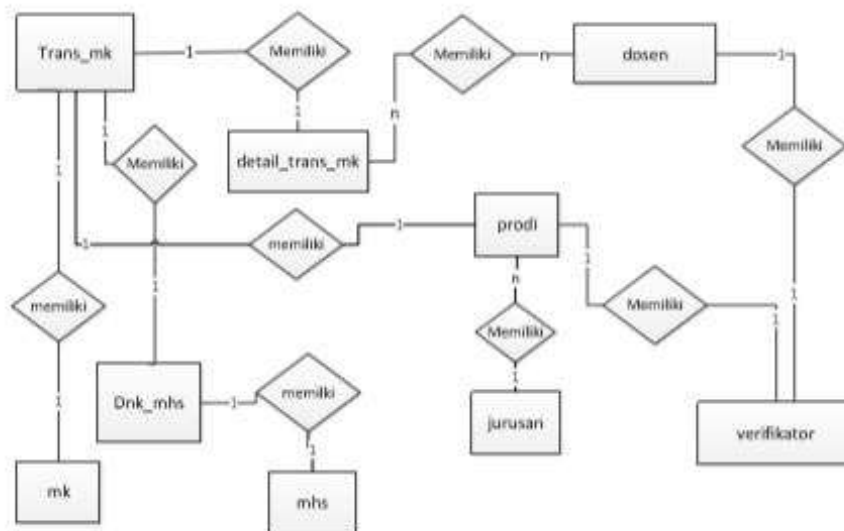
1. Informasi yang dibutuhkan untuk dapat melakukan proses cetak daftar hadir kuliah adalah informasi yang terdapat pada DNK (Daftar Nilai Kelas), yang terdiri dari: informasi mengenai tahun ajaran, semester, nama prodi, nama mata kuliah, kode mata kuliah, dosen penanggung jawab, nomor mata kuliah, keterangan kelas, dan nama mahasiswa serta NPM.

2. Keluaran (*output*) dari hasil proses ini adalah cetakan daftar hadir yang terdiri dari halaman *cover*, halaman pemberitahuan, halaman matrikulasi, dan halaman daftar hadir kuliah.

### 3.2.2 Analisis Bugs

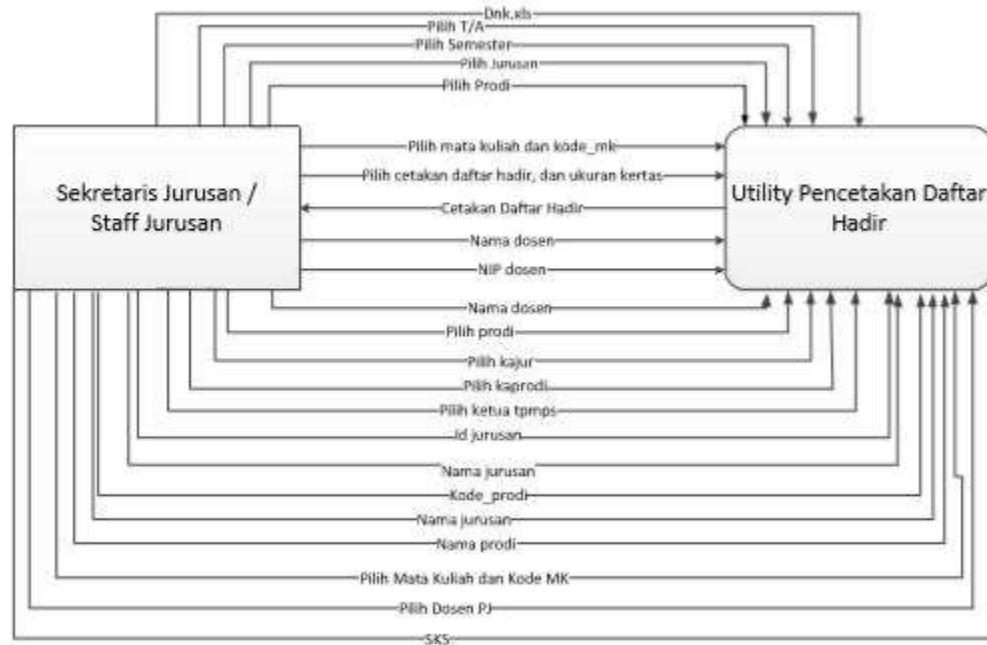
Sebelum menganalisis *bugs* yang ada pada sistem, maka perlu terlebih dahulu melihat desain *database* dan desain proses dari sistem ini. Adapun desain *database* dan desain proses dari sistem yang ada yaitu dapat dilihat pada Gambar 3.2.

#### Desain Database



Gambar 3.2 Entity Relationship Diagram (ERD).

## Desain Proses



Gambar 3.3 Data Flow Diagram (DFD) Level 0 (*Context Diagram*).

Setelah mengetahui desain *database* dan desain proses dari sistem maka selanjutnya dilakukan pengujian untuk melihat *bugs* yang ada pada sistem. Dalam tahapan ini dilakukan pengujian terhadap sistem dengan metode *black-box testing*, dimana sistem dilihat kinerjanya berdasarkan fungsi-fungsi yang harusnya bisa dilakukan oleh sistem.

### 3.2.3 Perbaikan *Bugs*

Setelah dilakukan analisis terhadap kinerja sistem melalui segi fungsinya, maka diharapkan mendapatkan hasil berupa kegagalan fungsi yang kemudian akan diperbaiki di dalam tahapan ini. Adapun beberapa *bugs* yang telah ditemukan yaitu:

- Pada saat pencetakan *cover*, pada bagian angkatan dan kelas tidak terisi.
- Ketika memasukkan keterangan dosen mata kuliah yang diampu harus dilakukan secara manual melalui *localhost*.

### 3.2.4 Pengujian Sistem

Setelah dilakukan perbaikan terhadap *bugs* yang telah ditemukan, maka selanjutnya akan dilakukan pengujian terhadap sistem yang ada. Adapun yang akan melakukan pengujian adalah bagian administrasi ataupun sekretaris Jurusan Ilmu Komputer. Metode dalam pengujian sistem ini menggunakan metode *black-box testing*. Rancangan pengujian yang akan dilakukan dapat dilihat di Tabel 3.1

**Tabel 3.1 Rancangan Pengujian Sistem**

Tabel Pengujian *Equivalence Partitioning* Fungsi Masukkan DNK

No.	Skenario Uji	Data yang dimasukkan	Hasil yang diharapkan	Status Uji
1.	Memasukkan pilihan tahun ajaran dan semester yang sesuai dengan dnk yang dimasukkan	Pilihan tahun ajaran dan pilihan semester	Data isian tahun ajaran dan semester tersimpan ke dalam <i>database</i> dan ditampilkan pada halaman daftar hadir benar dan sesuai dengan dnk.	V

2.	Memasukkan <i>file Excel</i> dnk (dnk.xls)	<i>File Excel</i> dnk yang telah dipilih	Data dapat diproses tanpa ada pesan <i>error</i>	V
3.	Memasukkan <i>file.xls</i> (dnk.xls) dengan format <i>file</i> yang tidak sesuai.	<i>File dnk.xls</i> dengan format yang tidak sesuai	<i>File</i> tidak dapat diproses dan terdapat pemberitahuan bahwa format data tidak sesuai.	V

Tabel Pengujian *Equivalence Partitioning* Fungsi Cetak Daftar Hadir Kuliah

No.	Skenario Uji	Data yang dimasukkan	Hasil yang diharapkan	Status Uji
1.	Memasukkan pilihan tahun ajaran, semester, Jurusan, prodi, mata kuliah, dan kode mata kuliah.	Pilihan tahun ajaran, pilihan semester, pilihan Jurusan, pilihan prodi, pilihan mata kuliah beserta kode mata kuliah	Pilihan tahun ajaran, semester, Jurusan prodi, mata kuliah dan kode mata kuliah muncul/terisi pada halaman- halaman daftar hadir yang dicetak.	V

2.	Memilih halaman yang ingin dicetak sesuai dengan ukuran kertas yang dibutuhkan.	Pilihan halaman beserta pilihan ukuran kertas	Terdapat <i>output</i> dari halaman dengan ukuran kertas sesuai yang telah dipilih dalam bentuk PDF.	V
3.	Tidak memasukkan/ mengosongkan pilihan tahun ajaran, semester, Jurusan, prodi, mata kuliah, dan kode mata kuliah pada proses awal cetak.	Tidak memilih tahun ajaran, semester, Jurusan, prodi, mata kuliah, dan kode mata kuliah.	Data tahun ajaran, semester, Jurusan, prodi, mata kuliah dan kode mata kuliah tidak terisi/kosong pada halaman-halaman daftar hadir yang dicetak.	V
4.	Tidak memasukkan pilihan halaman yang ingin dicetak beserta pilihan ukuran kertasnya.	Tidak memasukkan pilihan halaman dan ukuran kertas	<i>Output</i> yang dibutuhkan tidak didapatkan	V

Tabel Pengujian *Equivalence Partitioning* Fungsi Tambah, Ubah, Hapus Data Dosen

No.	Skenario Uji	Data yang dimasukkan	Hasil yang diharapkan	Status Uji
1.	Mengisi data nama dosen dan NIP pada proses tambah data dosen.	Data nama dosen dan NIP	Data nama dosen dan NIP tersimpan ke dalam <i>database</i> saat diproses	V
2.	Mengisi perubahan nama dosen pada proses ubah data dosen.	Ubah nama dosen.	Data nama dosen yang telah diubah tersimpan ke dalam <i>database</i> saat diproses.	V
3.	Menghapus data dosen tertentu.	Hapus data dosen.	Data berhasil dihapus.	V
4.	Tidak memasukkan data nama dosen dan NIP dosen.	Tidak memasukkan data nama dosen dan NIP	Terjadi <i>error</i> pada saat data diproses.	V
5.	Mengisi data nama dosen namun tidak mengisi data NIP	Memasukkan nama dosen, tidak memasukkan NIP	Terjadi pesan <i>error</i> pada saat data diproses.	V



6.	Mengisi data NIP dosen namun tidak mengisi data nama dosen.	Memasukkan NIP, tidak memasukkan nama dosen	Terjadi <i>error</i> pada saat data diproses.	V
7.	Mengosongkan data nama dosen pada proses ubah data dosen.	Tidak memasukkan nama dosen dan NIP	Terjadi <i>error</i> pada saat data diproses.	V

Tabel Pengujian *Equivalence Partitioning* Fungsi Tambah dan Ubah Data Verifikator

No.	Skenario Uji	Data yang dimasukkan	Hasil yang diharapkan	Status Uji
1.	Memasukkan pilihan prodi, ketua Jurusan, ketua prodi, ketua TPMPS proses tambah data	Pilihan prodi, ketua Jurusan, pilihan ketua prodi, pilihan ketua TPMPS	Data dapat diproses dan disimpan dalam <i>database</i> .	V
2.	Memasukkan pilihan prodi, ketua Jurusan, ketua prodi, ketua TPMPS pada proses ubah data verifikator sesuai yang dibutuhkan.	Ubah pilihan prodi atau pilihan ketua Jurusan, atau pilihan ketua prodi, atau pilihan ketua TPMPS sesuai kebutuhan.	Data dapat diproses dan data yang ada sebelumnya diperbaharui sesuai dengan perubahan data yang dilakukan.	V

3.	Memasukkan pilihan prodi yang sama pada proses tambah data verifikator yang telah dilakukan sebelumnya.	Pilihan prodi	Data tidak dapat diproses/terdapat pesan <i>error</i> .	V
----	---	---------------	---	---

Tabel Pengujian *Equivalence Partitioning* Fungsi Tambah dan Ubah Data Jurusan

No.	Skenario Uji	Data yang dimasukkan	Hasil yang diharapkan	Status Uji
1.	Memasukkan data id Jurusan dan nama Jurusan.	Id Jurusan dan nama Jurusan	Data dapat diproses dan disimpan dalam <i>database</i> .	V
2.	Mengubah data nama Jurusan.	Ubah nama Jurusan	Data dapat diproses dan diperbaharui.	V
3.	Memasukkan id Jurusan tapi tidak memasukkan nama Jurusan pada proses tambah data Jurusan dan sebaliknya.	Memasukkan id Jurusan, tidak memasukkan nama Jurusan atau sebaliknya.	Data tidak dapat diproses, terjadi <i>error</i> pada saat data disimpan.	V

Tabel Pengujian *Equivalence Partitioning* Fungsi Tambah dan Ubah Data Program Studi

No.	Skenario Uji	Data yang dimasukkan	Hasil yang diharapkan	Status Uji
1.	Memasukkan pilihan Jurusan, kode prodi & nama program studi	Pilihan Jurusan, memasukkan kode prodi dan nama prodi	Data dapat diproses dan disimpan dalam <i>database</i> .	V
2.	Mengubah data nama program studi pada proses ubah data prodi.	Ubah nama prodi sesuai kebutuhan	Data dapat diproses dan diperbaharui.	V
3.	Memasukkan kode prodi dan pilihan Jurusan tapi tidak memasukkan nama prodi pada proses tambah data prodi.	Memasukkan kode prodi dan pilihan Jurusan, tidak memasukkan nama prodi	Data tidak dapat diproses, terjadi <i>error</i> pada saat data disimpan.	V
4.	Mengosongkan data nama prodi saat proses ubah data prodi	Tidak memasukkan perubahan nama prodi	Data tidak dapat diproses, terjadi <i>error</i> pada saat data disimpan.	V

Tabel Pengujian *Equivalence Partitioning* Fungsi Kelola Data Detail Mata Kuliah

No.	Skenario Uji	Data yang dimasukkan	Hasil yang diharapkan	Status Uji
1.	Memasukkan pilihan mata kuliah beserta kode mk dan dosen beserta NIP.	Pilihan mata kuliah beserta kode mk dan pilhan dosen beserta NIP.	Data tersimpan ke dalam <i>database</i> .	V
2.	Menghapus data detail mata kuliah tertentu.	Hapus data detail mata kuliah tertentu	Data terhapus dari <i>database</i> .	V
3.	Memasukkan pilihan mata kuliah beserta kode mk namun tidak mengisi nama dosen beserta NIP dan sebaliknya.	Memasukkan pilihan mata kuliah beserta kode mk, tidak memasukkan pilihan dosen beserta NIP dan sebaliknya.	Terjadi <i>error</i> data belum lengkap, data tidak dapat disimpan ke dalam <i>database</i> .	V

4.	Mengosongkan <i>form</i> /tidak memasukkan pilihan mata kuliah beserta kode mk dan dosen beserta NIP pada proses tambah data kelola mata kuliah.	Tidak memasukkan pilihan mata kuliah beserta kode mk dan tidak memasukkan pilihan dosen dan NIP.	Data tidak dapat diproses, terjadi <i>error</i> pada saat data disimpan.	V
5.	Tidak mengisi data Dosen Penanggung Jawab, Jadwal Hari dan Ruangan pada menu kelola data master, submenu Mata Kuliah	Tidak memasukkan pilihan	Pada saat pencetakan <i>cover</i> daftar hadir maka pada bagian Dosen, Jadwal, Waktu dan Ruangan akan kosong	X

## **V. SIMPULAN DAN SARAN**

### **5.1. Simpulan**

Berdasarkan tahapan penelitian yang telah dilakukan maka didapatkan kesimpulan sebagai berikut:

1. Telah dilakukan perbaikan terhadap *bugs* yang terdapat di dalam sistem *utility* ini. Adapun *bugs* yang telah diperbaiki yaitu:
  - Pada saat menambahkan data dosen pada detail transkrip mata kuliah dimana sebelumnya proses ini tidak bisa dijalankan.
  - Pada saat proses pencetakan daftar hadir terkhusus pada bagian *cover* dimana pada sistem ini telah ditambahkan keterangan semester.
2. Dalam proses pencetakan sistem ini dapat mencetak pada ukuran kertas A4 dan F4.

### **5.2. Saran**

Berdasarkan penelitian yang telah dilakukan maka didapatkan saran- saran dalam pengembangan kedepannya:

1. Sistem ini dapat diaplikasikan pada semua instansi/jurusan yang ada di UNILA.

Perlu untuk melakukan intregrasi data dengan SIAKAD UNILA, sehingga dalam prosen pengambilan data DNK peluang terjadi kesalahan menjadi kecil.

## DAFTAR PUSTAKA

- Arbani, M. (2011). *Pengembangan Sistem Informasi Sekolah Berbasis Web*. Universitas Islam Negeri Syarif Hidayatullah Jakarta.
- Anhar. (2010). *Panduan Menguasai PHP & MySQL Secara Otodidak*. Jakarta Selatan: mediakita.
- Danella, R. (2014). *Analisis Efektivitas Sistem Informasi Akademik Universitas Negeri Surabaya*. Universitas Negeri Surabaya: tidak diterbitkan..
- Erawan, L. (2007). *Program Laporan Kehadiran FPDF Library Support (Studi Kasus Aplikasi Kehadiran Pegawai Udinus)*. Techno.Com, Vol. 7 No.2, Mei 2007.
- Kadir, A. A. (2015). *Pengembangan Utility Pencetakan Daftar Hadir Kuliah Jurusan Ilmu Komputer Fakultas Matematika Dan Ilmu Pengetahuan Alam Universitas Lampung*. Skripsi pada Jurusan Ilmu Komputer Universitass Lampung ; Digilib.unila.ac.id.
- Muharto, & Ambarita, A. (2016). *Metode Penelitian Sistem Informasi: Mengatasi Kesulitan Mahasiswa dalam Menyusun Proposal Penelitian*. Deepublish, Yogyakarta.
- Nazrul, A. (2013). *Rancangan Website dan Profil Usaha Advertising Menggunakan PHP dan MySQL*. AMIK SIGMA, Palembang.

Ould, Martyn A., & Unwin, C. (1986). *Testing in Software Development*. Cambridge University Press, New York.

Parsons, Jamrich, J., Beskeen, D. W., Cram, C. M., Duffy, J., & Friedrichsen, L. (2016). *Illustrated Computer Concepts and Microsoft Office 365 & Office 2016*. Orlando: Course Technology.

Suryani, H. (2015). *Metode Riset Kuantitatif: Teori dan Aplikasi Pada Penelitian Bidang Manajemen dan Ekonomi Islam*. Jakarta: Prenadamedia Group.

Yulianto, Eko, R., Yuhendra. (2015). *Rekayasa Perangkat Lunak Pengolahan Data Distribusi Obat- Obatan Di PT. Anugrah Pharmindo Lestari Berbasis Web*.