

## **LAPORAN TUGAS AKHIR**

### **PENJADWALAN OTOMATIS UJIAN TUGAS AKHIR JURUSAN TEKNIK INFORMATIKA UNIVERSITA PALANGKARAYA BERBASIS DESKTOP**



**Disusun Oleh :**

**MENTARI NOVIA SINTA**

**DBC 112 093**

**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS PALANGKA RAYA**

**2019**

**"PENJADWALAN OTOMATI UJIAN TUGAS AKHIR JURUSAN TEKNIK INFORMATIKA  
UNIVERSITAS PALANGKARAYA BERBASIS DESKTOP"**

**TUGAS AKHIR**

Sebagai salah satu syarat untuk menyelesaikan Program Strata-1 pada Jurusan Teknik  
Informatika Fakultas Teknik Universitas Palangka Raya

Oleh

**MENTARI NOVIA SINTA**

**DBC 112 093**

**Telah dipertahankan didepan tim penguji, pada :**

Hari/Tanggal : Senin, 8 Juli 2019

Waktu : 09.30 WIB

- |  |                 |
|--|-----------------|
| 1. Felicia Sylviana, ST., MM<br>NIP. 19760118 200312 2 003         | ..... (Ketua)   |
| 2. Viktor H. Pranatawijaya, ST., MT.<br>NIP. 19810606 200501 1 001 | ..... (Anggota) |
| 3. Sherly Christina, S.Kom., M.Kom<br>NIP. 19810929 200604 2 001   | ..... (Anggota) |
| 4. Enny D. Oktaviyani, ST., M.Kom<br>NIP. 19811003 200604 2 001    | ..... (Anggota) |
| 5. Putu Bagus A.A.P, ST., M.Kom<br>NIP. 19891022 201504 1 001      | ..... (Anggota) |

Mengetahui :

Fakultas Teknik  
Unveritas Palangka Raya  
Dekan,

Jurusan / Program Studi Teknik Informatika  
Fakultas Teknik Universitas Palangka Raya  
Ketua Jurusan,



**ABERTUN SAGIT SAHAY, ST., M.ENG**  
NIP. 19751212 200312 1 002

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Dalam sistem akademik perguruan tinggi, penjadwalan merupakan pekerjaan rutin yang dilakukan setiap semester. Ada dua penjadwalan yang sering dijumpai pada Jurusan Teknik Informatika UPR yaitu penjadwalan perkuliahan dan Ujian Tugas Akhir.

Jurusan Teknik Informatika UPR merupakan sebuah Program Strata 1 (S1) yang berada dalam lingkungan Fakultas Teknik UPR. Teknik Informatika merupakan disiplin ilmu pada ilmu computer dan Teknik ilmu pada masalah transformasi atau pengolahan data – data.

Proses penjadwalan adalah suatu proses untuk menerapkan event yang berisi komponen mata kuliah, dosen dan kelas pada time slot yang berisi komponen waktu dan ruang. Jika menggunakan sistem manual maka masalah ini membutuhkan waktu proses yang cukup lama untuk pencarian solusinya, terlebih lagi bila ukuran permasalahan semakin besar dengan bertambahnya jumlah komponen dan tetapan yang telah ditentukan oleh jurusan teknik informatika tempat jadwal tersebut di gunakan.

Selama proses, banyak aspek yang harus dipertimbangkan untuk memperoleh penjadwal ujian tugas akhir yang optimal. Oleh karena itu perlu ditetapkan suatu batasan yang menjadi acuan dalam proses penyusunan penjadwal ujian tugas akhir.

Berbagai aspek yang berkaitan dalam penjadwalan ujian tugas akhir tersebut dan harus dilibatkan dalam pertimbangan, Tidak boleh adanya penjadwal ujian tugas akhir yang saling bentrok antar dosen, kelas, ruang ataupun waktu perkuliahan .

Sistem penjadwal ujian tugas akhir di beberapa institusi perguruan tinggi sampai saat ini masih dilakukan secara manual, yaitu dengan pencarian jadwal dan waktu dosen yang kosong. Jadwal yang dihasilkan dengan cara seperti ini memerlukan waktu yang cukup lama dan cenderung mengabaikan berbagai aspek tersebut. Sehingga penjadwal ujian tugas akhir yang sudah dibuat seringkali perlu dilakukan perbaikan lagi. Oleh karena itu perlu dikembangkan suatu sistem penjadwal ujian tugas akhir yang dapat mengakomodasi berbagai aspek yang menjadi pertimbangan diatas.

Dengan ini diharapkan “Penjadwalan Otomatis Ujian Tugas Akhir Jurusan Teknik Informatika Universitas Palangkaraya Berbasis Desktop” ini dapat membantu Dosen Koordinator Ujian Tugas Akhir, Dosen dan Mahasiswa dalam berjalannya proses Ujian Tugas Akhir Jurusan Informatika UPR.

## **1.2 Rumusan Masalah**

Berdasarkan latar belakang yang diuraikan diatas, maka dirumuskan masalah sebagai berikut :

Bagaimana merancang dan membangun Penjadwalan Otomatis Ujian Tugas Akhir Jurusan Teknik Informatika Universitas Palangkaraya Berbasis Desktop yang dapat digunakan oleh koordintor TA?

### 1.3 Batasan Masalah

Batasan masalah yang dijadikan dasar perancangan dan pembuatan yaitu sebagai berikut :

1. Studi kasus Jurusan Teknik Informatika UPR.
2. Penjadalan otomatis ini digunakan untuk kepentingan menyampaikan data dan informasi kepada pengguna, seperti Koordinator TA dan Mahasiswa.
3. Fitur / fasilitas
4. Menejemen hak akses

### 1.4 Tujuan Penelitian

Adapun tujuan dari pembuatan Penjadwalan Otomatis ini, yaitu sebagai berikut:

1. Membangun system penjadwalan otomatis berbasis desktop pada Jurusan Teknik Informatika UPR.
2. Membangun fasilitas *Generate* (membuat otomatis) Jadwal Tugas Akhir dalam pembuatan Jadwal Otomatis Tugas Akhir.
3. Membangun fasilitas untuk membantu Koordinator TA dalam mengelola data Mahasiswa yang mendaftar TA.

### 1.5 Manfaat Penelitian

Adapun manfaat yang didapatkan dari Penjadwalan Otomatis Tugas Akhir Berbasis Desktop ini yaitu :

1. Tersedianya fasilitas untuk Koordinator TA dalam menentukan Jadwal Ujian Tugas Akhir.

## 1.6 Metodologi Penelitian

Adapun metodologi yang digunakan dalam membuat “Penjadwalan Otomatis Ujian Tugas Akhir Jurusan Teknik Informatika Universitas Palangka Raya Berbasis Desktop”, yaitu :

### 1. Metode Pengumpulan Data

Metode pengumpulan data merupakan tahap pengambilan data atau sampel yang berhubungan dengan permasalahan yang sedang dibahas. Pengumpulan data yang dilakukan pada proses ini, meliputi :

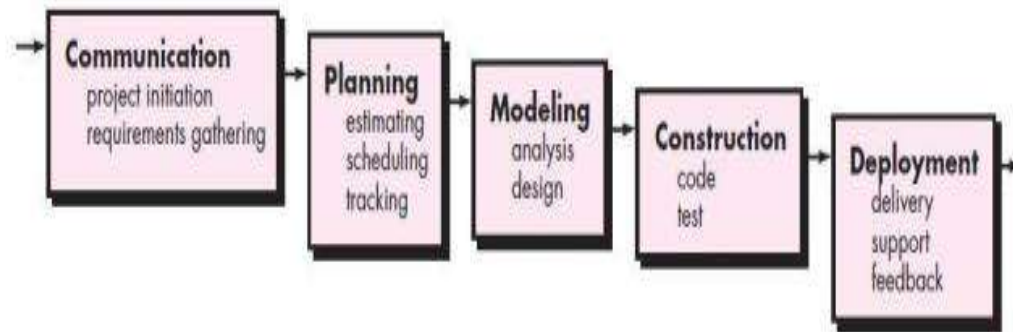
- a. Observasi, tahap ini dilakukan dengan cara mengumpulkan data mahasiswa, data dosen, dan jadwal mata kuliah. Jurusan Teknik Informatika Universitas Palangka Raya.
- b. Studi kepustakaan dan studi literatur, tahap ini dilakukan dengan cara mencari referensi seperti mempelajari teori-teori literatur dari buku dan informasi dari internet yang berhubungan dengan proses pembuatan *Penjadwalan Tugas Akhir*.

### 2. Metode Konsultasi

Mengkonsultasikan mengenai permasalahan yang akan dibahas kepada dosen pembimbing atau orang lain yang mengerti tentang objek program yang akan dibuat.

### 2. Metode Pengembangan system

Metode yang digunakan dalam pengembangan sistem adalah model *Waterfall* (Roger S. Pressman, 2010) dengan tahap-tahap sebagai berikut :



**Gambar 1.1 Waterfall Modelling**

*(sumber : Roger S. Pressman, 2010)*

a) Communication ( Komunikasi )

Pada tahap ini merupakan tahap analisis terhadap kebutuhan software, dan tahap untuk mengadakan pengumpulan data dengan melakukan pertemuan dengan customer, maupun mengumpulkan data - data yang diperlukan. Hasil pada tahap komunikasi yaitu data-data yang dibutuhkan dalam pembuatan penjadwalan ujian tugas akhir.

b) Planning ( Perencanaan )

Proses Proses planning merupakan lanjutan dari proses communication (komunikasi). Tahapan ini akan menghasilkan dokumen user requirement atau bisa dikatakan sebagai data yang berhubungan dengan keinginan user dalam pembuatan software, termasuk rencana yang akan dilakukan.

c) Modeling ( Permodelan )

Proses modeling ini akan menerjemahkan syarat kebutuhan sebuah perancangan software yang dapat diperkirakan sebelum dibuat coding. Proses

ini terbagi menjadi 2, yaitu analisis dan desain. Pada analisis menggunakan Data Flow Diagram (DFD) serta Entity Relationship Diagram. Pada desain akan mendesain tabel, desain navigasi menggunakan Sitemap dan desain antarmuka/representasi interface. Tahapan ini akan menghasilkan dokumen yang disebut software requirement.

d) Construction ( Konstruksi )

Construction merupakan proses membuat kode (coding). Coding atau pengkodean merupakan penerjemahan desain dalam bahasa yang bisa dikenali oleh komputer. Programmer akan menerjemahkan transaksi yang diminta oleh user. Tahapan inilah yang merupakan tahapan secara nyata dalam mengerjakan suatu software, artinya penggunaan komputer akan dimaksimalkan dalam tahapan ini. Setelah pengkodean selesai maka akan dilakukan testing terhadap sistem yang telah dibuat tadi. Tujuan testing adalah menemukan kesalahan - kesalahan terhadap sistem tersebut untuk kemudian bisa diperbaiki. Untuk menterjemahkan kode-kode program dengan menggunakan bahasa pemrograman VB & SQL dan pengujian dilakukan menggunakan Blackbox Testing.

e) Deployment ( Penyerahan )

Tahapan ini merupakan tahap akhir / final dalam pembuatan sebuah software atau sistem. Setelah melakukan analisis, desain dan pengkodean maka sistem yang sudah jadi akan digunakan oleh user. Kemudian software yang telah dibuat harus evaluasi jika ada kekurangan dan dilakukan pemeliharaan secara berkala.

## **1.7 SISTEMATIKA PENULISAN**

Penulisan laporan Tugas Akhir ini disusun dalam lima bab, dengan menggunakan sistematika sebagai berikut.

### **BAB I PENDAHULUAN**

Pada bab ini, diuraikan mengenai latar belakang masalah, rumusan masalah, batasan masalah, tujuan dan manfaat, metodologi penelitian dan sistematika penulisan.

### **BAB II LANDASAN TEORI.**

Pada bab ini, menjelaskan tentang teori yang digunakan untuk menguraikan mengenai suatu pedoman atau teori yang dikemukakan oleh pakar-pakar dalam suatu bidang tertentu untuk memecahkan masalah.

### **BAB III ANALISIS DAN DESAIN**

Pada bab ini menguraikan tentang perancangan proses arus sistem, perancangan basis data, perancangan antarmuka, yang meliputi perancangan input dan output, serta kebutuhan sistem.

### **BAB IV IMPLEMENTASI DAN PENGUJIAN**

Pada bab ini menguraikan tentang tahapan-tahapan dari implementasi sistem yang digunakan, yaitu penerapan rancangan antarmuka. Sedangkan pada tahap pengujian menggunakan *Blackbox Testing*.

### **BAB V PENUTUP**

Bagian ini berisi kesimpulan-kesimpulan yang merupakan rangkuman dari hasil analisis kinerja pada tahap-tahap sebelumnya, dan bagian ini berisi juga berisi saran yang berupa perbaikan/peningkatan yang diperlukan saat ini atau masa mendatang yang berhubungan dengan pelaksanaan pembuatan program ini.

## 1.8 JADWAL PENELITIAN

Adapun jadwal kegiatan Tugas Akhir ini adalah sebagai berikut :

**Tabel 1.1. Jadwal Kegiatan Tugas Akhir**

Rencana Kegiatan	Bulan / Tahun 2019																			
	Bulan I				Bulan II				Bulan III				Bulan IV				Bulan V			
	I	II	III	IV	I	II	III	IV	I	II	III	IV	I	II	III	IV	I	II	III	IV
Pengumpulan Data	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
Pengumpulan Proposal	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
Analisis Data	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
Sidang Proposal	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
Desain	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
Coding	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
Testing	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
Pembuatan Laporan Tugas Akhir	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
Seminar Tugas Akhir	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Pengertian Penjadwalan**

Penjadwalan adalah aktivitas perencanaan untuk menentukan kapan dan dimana setiap operasi sebagai bagian dari pekerjaan secara keseluruhan harus dilakukan pada sumber daya yang terbatas, serta pengalokasian sumber daya pada suatu waktu tertentu dengan memperhatikan kapasitas sumber daya yang ada. Penjadwalan dapat diartikan sebagai pengalokasian sejumlah sumber daya (resource) untuk melakukan sejumlah tugas atau operasi dalam jangka waktu tertentu dan merupakan proses pengambilan keputusan yang peranannya sangat penting dalam industry manufaktur dan jasa yaitu mengalokasikan sumber-sumber daya yang ada agar tujuan dan sasaran perusahaan lebih optimal (Baker & Trietsch, 2009).

Menurut Pinedo (2012), penjadwalan dapat didefinisikan sebagai proses pengalokasian sumber daya untuk mengerjakan sekumpulan tugas dalam jangka waktu tertentu dengan 2 arti penting sebagai berikut.

1. Penjadwalan merupakan suatu fungsi pengambilan keputusan untuk membuat atau menentukan jadwal.
2. Penjadwalan merupakan suatu teori yang berisi sekumpulan prinsip dasar, model, teknik, dan kesimpulan logis dalam proses pengambilan keputusan yang memberikan pengertian dalam fungsi penjadwalan.

Penjadwalan dibutuhkan untuk mengurangi alokasi tenaga operator, mesin dan peralatan produksi, dan dari aspek lainnya untuk lebih efisien. Hal ini sangat penting dalam pengambilan keputusan dalam proses kelangsungan produksi.

Tiga pembagian dari penjadwalan akademik (academic timetables), antara lain

#### 1. Penjadwalan Sekolah (School Timetabling)

Pada penjadwalan sekolah setiap kelas memiliki mata pelajaran tertentu serta memiliki ruangan tertentu dimana proses kegiatan belajar mengajar dilaksanakan. Pada dasarnya isi dari jadwal diatur oleh kurikulum dimana jumlah dari waktu tiap mata pelajaran yang diajar dalam seminggu sering ditetapkan secara nasional. Setiap kelas terdiri dari seorang pengajar, yang harus ditempati saat pelajar tiba di sekolah hingga meninggalkan sekolah dan memiliki seorang guru tertentu yang akan bertanggung jawab atas kelas tersebut dalam sebuah periode waktu tertentu. Pengajar biasanya dialokasikan di awal proses penjadwalan, yang menjadi masalah adalah menyesuaikan pertemuan dari pengajar dengan kelas untuk slot waktu tertentu sehingga setiap pengajar tertentu mengajar tiap kelas yang diwajibkan kepadanya. Setiap kelas atau pengajar tidak dapat terlibat lebih satu pertemuan pada saat waktu yang bersamaan.

#### 2. Penjadwalan Mata Kuliah (Course Timetabling)

Penjadwalan mata kuliah mencakup kumpulan scheduling dari perkuliahan dimana dalam setiap mata kuliah diberikan sejumlah ruangan dan periode waktu.

Karakteristik penjadwalan mata kuliah, antara lain:

- a. Setiap mahasiswa dapat memiliki jumlah mata kuliah yang berbeda.
- b. Ketersediaan ruangan berperan sangat penting.
- c. Jika dua ruangan memiliki mahasiswa yang sama, maka ruangan tidak dapat dijadwalkan pada waktu yang sama.

### 3. Penjadwalan Ujian (Exam Timetabling)

Karakteristik penjadwalan ujian, antara lain:

- a. Hanya ada satu ujian untuk tiap objek (mata kuliah).
- b. Ada banyak batasan yang berbeda, contohnya pada hari yang sama ada mahasiswa yang memiliki ujian yang sangat banyak dan berurutan waktunya tetapi ada juga yang tidak.
- c. Satu ujian dapat memiliki lebih dari satu ruangan.

#### 2.1.1 Tujuan Penjadwalan

Tujuan penjadwalan adalah untuk mengurangi waktu keterlambatan dari batas waktu yang ditentukan agar dapat memenuhi batas waktu yang telah disetujui dengan konsumen, penjadwalan juga dapat meningkatkan produktifitas mesin dan mengurangi waktu menganggur. Produktifitas mesin meningkat maka waktu menganggur berkurang, secara tidak langsung perusahaan dapat mengurangi biaya produksi. Semakin baik suatu penjadwalan semakin menguntungkan juga bagi perusahaan dan bisa menjadi acuan untuk meningkatkan keuntungan dan strategi bagi perusahaan dalam pemuasan pelanggan.

Beberapa tujuan yang ingin dicapai dengan dilaksanakannya penjadwalan adalah sebagai berikut (Baker & Trietsch, 2009).

1. Meningkatkan produktifitas mesin, yaitu dengan mengurangi waktu mesin menganggur.
2. Mengurangi persediaan barang setengah jadi dengan mengurangi jumlah rata-rata pekerjaan yang menunggu antrian suatu mesin karena mesin tersebut sibuk.
3. Mengurangi keterlambatan karena telah melampaui batas waktu dengan cara,
  - a. Mengurangi maksimum keterlambatan,
  - b. Mengurangi jumlah pekerjaan yang terlambat.
4. Meminimasi ongkos produksi.
5. Pemenuhan batas waktu yang telah ditetapkan (due date), karena dalam kenyataan apabila terjadi keterlambatan pemenuhan due date dapat dikenakan suatu denda (penalty). Teknik penjadwalan yang tepat bergantung pada volume pesanan, ciri operasi, keseluruhan kompleksitas pekerjaan, serta perhatian pada tujuan dari penjadwalan itu sendiri.

## **2.2 Konsep Dasar Sistem**

Pada konsep dasar sistem ini akan dijelaskan mengenai pengertian sistem dan karakteristik sistem.

### **2.2.1 Pengertian Sistem**

Menurut Tata Sutabri (2012:6) pada dasarnya sistem adalah sekelompok unsur yang erat hubungannya satu dengan yang lain, yang berfungsi bersama-sama untuk mencapai tujuan tertentu. Dalam Parno (2010:1) terdapat beberapa pengertian sistem menurut pendapat para ahli sebagai berikut :

1. Menurut Jerry FithGerald, “Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu”.
2. Menurut Ludwig Von Bartalanfy, “Sistem merupakan seperangkat unsur yang saling terikat dalam suatu antar relasi diantara unsur-unsur tersebut dengan lingkungan”.
3. Menurut Anatol Raporot, “Sistem adalah suatu kumpulan kesatuan dan perangkat hubungan satu sama lain”.

### **2.2.2 Karakteristik Sistem**

Menurut Tata Sutabri (2012:13) Model umum sebuah sistem terdiri dari input, proses, dan output. Hal ini merupakan konsep sebuah sistem yang sangat sederhana mengikuti sebuah sistem dapat mempunyai beberapa masukan dan keluaran sekaligus. Selain itu sebuah sistem juga memiliki karakteristik atau sifatsifat tertentu, yang mencirikan bahwa hal tersebut bisa dikatakan sebagai suatu sistem. Adapun karakteristik yang dimaksud adalah sebagai berikut :

1. **Komponen Sistem (Components)**

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang bekerja sama membentuk satu kesatuan. Komponen-komponen sistem tersebut dapat berupa suatu bentuk subsistem. Setiap subsistem memiliki sifat-sifat sistem yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan. Suatu sistem dapat mempunyai sistem yang lebih besar yang disebut dengan supra sistem.

## 2. Batasan Sistem (Boundary)

Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem lainnya atau sistem dengan lingkup luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan yang tidak dapat dipisah-pisahkan.

## 3. Lingkup Luar Sistem (Environment)

Bentuk apapun yang ada di luar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut dengan lingkungan luar sistem. Lingkungan luar sistem ini dapat menguntungkan dan dapat juga merugikan sistem tersebut. Lingkungan luar yang menguntungkan merupakan energi bagi sistem tersebut, yang dengan demikian lingkungan luar tersebut harus selalu dijaga dan dipelihara. Sedangkan lingkungan luar yang merugikan harus dikendalikan. Kalau tidak maka akan mengganggu kelangsungan hidup sistem tersebut.

## 4. Penghubung Sistem (Interface)

Media yang menghubungkan sistem dengan subsistem yang lain disebut dengan penghubung sistem atau interface. Penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lain. Keluaran suatu subsistem akan menjadi masukan untuk subsistem yang lain dengan melewati penghubung. Dengan demikian terjadi suatu integrasi sistem yang membentuk satu kesatuan.

#### 5. Masukan Sistem (Input)

Energi yang dimasukkan kedalam sistem disebut masukan sistem, yang dapat berupa pemeliharaan (maintenance input) dan sinyal (signal input). Sebagai contoh, didalam suatu unit sistem komputer, "program" adalah maintenance input yang digunakan untuk mengoperasikan komputer. Sementara "data" adalah signal input yang akan diolah menjadi informasi.

#### 6. Keluaran Sistem (Output)

Hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi subsistem yang lain. Seperti contoh sistem informasi, keluaran yang dihasilkan adalah informasi, di mana informasi ini dapat digunakan sebagai masukan untuk pengambilan keputusan atau hal-hal lain yang merupakan input bagi subsistem lainnya.

#### 7. Pengolahan Sistem (Proses)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran. Sebagai contoh, sistem akuntansi. Sistem ini akan mengolah data transaksi menjadi laporan-laporan yang dibutuhkan oleh pihak manajemen.

#### 8. Sasaran Sistem (Objective)

Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat deterministik. Kalau suatu sistem tidak memiliki sasaran, maka operasi sistem tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuan yang telah direncanakan.

## **2.3 Konsep Dasar Informasi**

Menurut Tata Sutabri (2012:21) Informasi merupakan proses lebih lanjut dari data yang sudah memiliki nilai tambah.

### **2.3.1 Pengertian Informasi**

Menurut Tata Sutabri (2012:22) Informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan. Sistem pengolahan informasi akan mengolah data menjadi informasi atau mengolah data dari bentuk tak berguna menjadi berguna bagi yang menerimanya. Nilai informasi berhubungan dengan keputusan. Bila tidak ada pilihan atau keputusan maka informasi tidak diperlukan. Sumber dari informasi adalah data. Data merupakan kenyataan yang menggambarkan suatu kejadian serta merupakan suatu kesatuan yang nyata, dan merupakan bentuk yang masih mentah sehingga perlu diolah lebih lanjut melalui suatu model untuk menghasilkan informasi.

### **2.3.2 Kualitas Informasi**

Menurut Tata Sutabri (2012:33) Kualitas dari suatu informasi tergantung dari 3 hal, yaitu informasi harus akurat (accurate), tepat waktu (timelines) dan relevan (relevance).

#### **a. Akurat (Accurate)**

Informasi harus bebas dari kesalahan dan tidak biasa atau menyesatkan. Akurat juga berarti bahwa informasi harus jelas mencerminkan maksudnya. Informasi harus akurat karena dari sumber informasi sampai ke penerima informasi

mungkin banyak mengalami gangguan (noise) yang dapat mengubah atau merusak informasi tersebut.

b. Tepat waktu (Timelines)

Informasi yang sampai kepada si penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan mempunyai nilai lagi, karena informasi merupakan landasan di dalam pengambilan keputusan. Bila pengambilan keputusan terlambat maka dapat berakibat fatal bagi organisasi. Dewasa ini informasi bernilai mahal karena harus cepat dikirim dan didapat sehingga memerlukan teknologi mutakhir untuk mendapatkannya, mengolah, dan mengirimkannya.

c. Relevan (Relevance)

Informasi tersebut memiliki manfaat untuk pemakainnya. Relevansi informasi untuk setiap orang berbeda. Menyampaikan informasi tentang penyebab kerusakan mesin produksi kepada akuntan perusahaan tentu kurang relevan. Akan lebih relevan bila ditujukan kepada ahli teknik perusahaan. Sebaliknya informasi mengenai harga pokok produksi disampaikan untuk ahli teknik merupakan informasi yang kurang relevan, tetapi akan sangat relevan untuk seorang akuntan perusahaan.

## **2.4 Perangkat Lunak (Software)**

### **2.4.1 Definisi Perangkat Lunak**

Komputer merupakan mesin yang memproses fakta atau data menjadi informasi.

Komputer di gunakan orang untuk meningkatkan hasil kerja dan memecahkan berbagai masalah. Yang menjadi pemroses data atau pemecah masalah itu adalah perangkat lunak.

Menurut Pressman (2010, p4), umumnya buku teks menjelaskan definisi perangkat lunak sebagai berikut:

1. Perintah (program computer) yang jika dijalankan akan menampilkan hasil sesuai dengan yang diinginkan.
2. Struktur data yang memungkinkan sebuah program untuk mengubah suatu informasi.
3. Informasi deskriptif dalam bentuk hardcopy atau softcopy yang menjelaskan cara kerja dan manfaat sebuah program.

#### **2.4.2 Karakteristik Perangkat Lunak**

Secara garis besar, karakteristik perangkat lunak berbeda dengan perangkat keras.

Menurut Pressman (2010, p4) karakteristik dari perangkat lunak adalah:

1. Perangkat lunak tidak usang, namun memburuk.
2. Perangkat lunak direkayasa, bukan dibuat dalam bentuk fisik.
3. Meski industri telah menuju perakitan berbasis komponen, tetapi sebagian besar perangkat lunak masih direkayasa secara unik.

#### **2.4.3 Rekayasa Perangkat Lunak**

Dalam buku yang ditulis Pressman (2010, p13), Fritz Bauer menjelaskan bahwa definisi dari rekayasa perangkat lunak adalah pembuatan perangkat lunak

dengan menggunakan prinsip rekayasa yang kuat untuk menghasilkan perangkat lunak yang ekonomis, handal, dan bekerja secara efisien. Pressman (2010, p13), mendefinisikan rekayasa perangkat lunak merupakan teknologi yang bertingkat atau berlapis, lapisanlapisan teknologi tersebut adalah sebagai berikut:



**Gambar 2.1 Lapisan Teknologi Rekayasa Perangkat Lunak**

(Sumber: Pressman, 2010, p14)

1. Berpusat pada kualitas (a quality focus)

Semua proses perancangan (termasuk rekayasa perangkat lunak) sangat dipengaruhi oleh komitmen organisasi terhadap kualitas. Filosofi – filosofi tentang kualitas akan terus memperbaiki proses perancangan dan akhirnya akan berpengaruh terhadap pendekatan rekayasa perangkat lunak yang lebih efektif.

2. Proses (process)

Lapisan ini merupakan lapisan yang menghubungkan teknologi – teknologi yang digunakan dalam perancangan program dan memungkinkan pembuatan program diselesaikan dengan tepat waktu. Proses ini mendefinisikan framework yang harus dibuat agar teknologi yang digunakan dalam pembuatan program dapat dimanfaatkan dengan efektif.

#### 4. Metode (Methods)

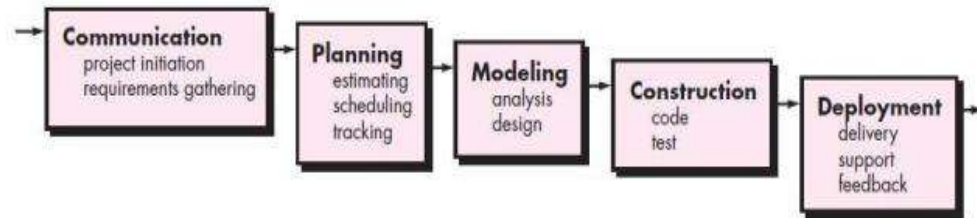
Lapisan metode dari rekayasa perangkat lunak menjelaskan secara teknis bagaimana cara membangun perangkat lunak. Lapisan ini terdiri dari tugas-tugas yang mencakup tentang: analisis kebutuhan (requirement analysis), model desain (design modelling), pembuatan program (program construction), pengujian (testing) dan pendukung (support).

#### 5. Alat (Tools)

Lapisan ini menyediakan bantuan secara otomatis dan semi-otomatis untuk Lapisan proses dan metode. Ketika alat sudah terintegrasi sehingga informasi yang dihasilkan oleh suatu alat dapat digunakan oleh yang lain, maka terbentuklah sebuah sistem untuk membantu proses perancangan perangkat lunak yang disebut Computer Aided Software Engineering.

### 2.5 Waterfall Modelling

Metode yang digunakan dalam mengembangkan sistem informasi menggunakan Metode pengembangan perangkat lunak Waterfall. Menurut Roger S. Pressman (2010:46) metode ini terdapat 5 (lima) tahap untuk mengembangkan suatu perangkat lunak yaitu Communication, Planning, Modeling, Construction, Deployment. Dimana konsep dari metode ini adalah melihat suatu masalah secara sistematis dan terstruktur dari atas ke bawah. Tahap-tahap pengembangan perangkat lunak metode Waterfall dapat dilihat sebagai berikut :



**Gambar 2.1 Waterfall Modelling**

(sumber : Roger S. Pressman, 2010)



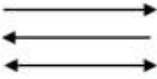

## 2.6 Data Flow Diagram (DFD)

Analisis perancangan Penjadwalan Otomatis Ujian Tugas Akhir Jurusan Teknik Informatika Universitas Palangkaraya Berbasis Desktop didefinisikan dengan menggunakan DFD (Data Flow Diagram). Semua aliran data yang masuk dan keluar melalui sistem digambarkan dengan DFD, sehingga dapat dilihat dengan jelas aliran data apa saja yang terlibat dalam proses di sistem ini. Saputra dan Agustin (2012:28) Data Flow Diagram atau yang disingkat DFD merupakan diagram yang menggambarkan alir data dalam suatu entitas ke sistem atau sistem ke entitas.

DFD juga dapat diartikan sebagai teknik grafis yang menggambarkan alir data dan transformasi yang digunakan sebagai perjalanan data dari input atau masukan menuju keluaran atau output.

DFD menurut Gane dan Sarson, digambarkan dengan menggunakan empat simbol seperti pada Tabel 2.1.

Tabel 2.1 Simbol – Simbol DFD

Simbol	Nama Simbol
	Entitas Eksternal
	Proses
	Aliran Data
	Media Penyimpanan Data

Rosa dan Shalahuddin (2011:118) menyatakan tahapan-tahapan perancangan dengan menggunakan DFD seperti berikut :

1. Membuat DFD Level 0 atau sering disebut juga Context Diagram

DFD level 0 menggambarkan sistem yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun sistem lain. DFD level 0 digunakan untuk menggambarkan interaksi antara sistem yang akan dikembangkan dengan entitas luar.

2. Membuat DFD Level 1

DFD level 1 digunakan untuk menggambarkan modul-modul yang ada dalam sistem yang akan dikembangkan. DFD level 1 merupakan hasil breakdown DFD level 0 yang sebelumnya sudah dibuat.

3. Membuat DFD Level 2

Modul-modul pada DFD level 1 dapat di-breakdown menjadi DFD level 2. Modul mana saja yang harus di-breakdown lebih detail tergantung pada

tingkat kedetailan modul tersebut. Apabila modul tersebut sudah cukup detail dan rinci maka modul tersebut sudah tidak perlu untuk dibreakdown lagi. Untuk sebuah sistem, jumlah DFD level 2 sama dengan jumlah modul pada DFD level 1 yang di-breakdown.





#### 4. Membuat DFD Level 3 dan seterusnya

DFD level 3, 4, 5, dan seterusnya merupakan breakdown dari modul pada DFD level di atasnya. Breakdown pada level 3, 4, 5, dan seterusnya aturannya sama persis dengan DFD level 1 atau level

### **2.7 Entity Relationship Diagram (ERD)**

Menurut Harsiti (2012:26) Entity Relationship Diagram (ERD) adalah diagram yang menghubungkan antar entitas di dalam penyusunan/perancangan basis data. Diagram Entitiy Relationship ini ditemukan oleh Chen tahun 1976. Tujuan dari Entity Relationship Diagram adalah untuk menunjukkan objek data dan relationship yang ada pada objek tersebut. Berikut komponen dan simbol ERD :

**Tabel 2.2 Tabel Komponen dan Simbol ERD**

No.	Komponen	Simbol
1.	<i>Entity</i> (entitas)	
2.	<i>Relationship</i> (keterhubungan)	
3.	<i>Atribut</i>	
4.	<i>Link</i> (penghubung antara himpunan relasi dengan himpunan entitas dan penghubung antara himpunan entitas dan atributnya)	

Berikut penjelasan dari komponen elemen-elemen ERD dalam tabel di atas :

#### 1. Entitas

Entitas merupakan individu yang mewakili sesuatu yang nyata dan dapat dibedakan dari sesuatu yang lain. Entitas bisa berupa orang, kejadian, atau benda dimana data akan dikumpulkan.

#### 2. Atribut

3. Setiap entitas pasti memiliki atribut yang mendeskripsikan karakteristik dari entitas tersebut. Berikut adalah syarat atribut suatu entitas :

- a. Informasi yang diambil tentang sebuah entitas
- b. Nama atribut harus merupakan kata benda
- c. Kadang nama entitas diletakkan di depan nama atribut untuk ketelitian.

#### 4. Relationship

Relasi menunjukkan adanya hubungan diantara sejumlah entitas yang berasal dari himpunan entitas yang berbeda. Entitas pertama dalam relationship disebut entitas induk, entitas kedua disebut sebagai entitas anak. Relationship harus mempunyai nama yang berupa kata kerja.

#### 5. Kardinalitas

Kardinalitas relasi menunjukkan jumlah maksimum entitas yang dapat berelasi dengan entitas yang lain.

Bentuk relasi antar file dapat dikategorikan menjadi tiga macam. Pengkategorian tersebut dikenal dengan nama cardinality ratio yang menjelaskan tentang batasan pada jumlah entity yang berhubungan melalui relationship. Kardinalitas relasi yang terjadi di antara dua himpunan entitas (misalnya A dan B) dapat berupa :

a. Satu ke Satu (one to one)

Yang berarti setiap entitas pada himpunan entitas A berhubungan paling banyak dengan satu entitas pada himpunan entitas B, dan begitu juga sebaliknya setiap entitas pada himpunan entitas B berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas A.

Contoh :

A : Jurusan , B : Dosen

Setiap Jurusan hanya bisa dikepalai oleh 1 Dosen.



**Gambar 2.3 Contoh Kardinalitas One To One**

b. Satu ke Banyak (one to many)

Yang berarti setiap entitas pada himpunan A dapat berhubungan banyak dengan entitas B, tetapi himpunan entitas B hanya dapat berhubungan dengan paling banyak satu entitas saja pada himpunan entitas A.

Contoh :

A : Dosen , B : Mahasiswa

Setiap dosen bisa membimbing banyak mahasiswa, tapi 1 mahasiswa hanya memiliki 1 dosen pembimbing akademik.



**Gambar 2.4 Contoh Kardinalitas One To Many**

c. Banyak ke Banyak (many to many)

Yang berarti setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, dan demikian juga sebaliknya, dimana setiap entitas pada himpunan entitas B dapat berhubungan dengan banyak entitas pada himpunan entitas A.

Contoh : A : Mahasiswa , B : Matakuliah

Setiap mahasiswa bisa mempelajari banyak banyak mata kuliah, sebaliknya setiap Mata Kuliah dapat dipelajari oleh banyak mahasiswa.



**Gambar 2.5** Contoh kardinalitas many to many

## 2.8 Depth First Search (DFS)

Aulia Husna (2013:33-35), Untuk menghindari kemungkinan adanya proses pelacakan suatu node secara berulang, maka digunakan struktur pohon. Struktur pohon digunakan untuk menggambarkan keadaan secara hirarkis. Pohon juga terdiri dari beberapa node. Node yang terletak pada level-0 disebut juga “akar”. Node akar menunjukkan keadaan awal yang biasanya merupakan topik atau objek. Node akar ini terletak pada level ke-0. Node akar mempunyai beberapa percabangan yang terdiri dari atas beberapa node successor yang sering disebut dengan nama “anak” dan merupakan node-node perantara. Namun jika dilakukan pencarian mundur, maka dapat dikatakan bahwa node tersebut memiliki predecessor.

Node node yang tidak memiliki anak sering disebut dengan nama node “daun” yang menunjukkan akhir dari suatu pencarian, dapat berupa tujuan yang diharapkan (goal) atau jalan buntu (dead end). Pada Depth First Search, proses pencarian akan dilaksanakan pada semua anaknya sebelum dilakukan pencarian ke node-node yang selevel. Pencarian dimulai dengan node akar ke level yang lebih tinggi. Pencarian dilakukan pada satu node dalam setiap level dari yang paling kiri. Jika pada level yang paling dalam solusi belum ditemukan, maka pencarian dilanjutkan pada node sebelah kanan. Node yang di kiri dapat dihapus dari memori. Jika pada level yang

paling dalam belum ditemukan solusi, maka pencarian dilanjutkan ke level selanjutnya. Proses ini diulangi terus hingga ditemukannya solusi.

## 2.9 Visual Basic

Visual BASIC (Beginners All-Purpose Symbolic Instruction Code) merupakan Bahasa pemrograman Integrated Development Environment (IDE), yaitu Bahasa pemrograman visual yang digunakan untuk membuat program aplikasi atau software berbasis sistem operasi Microsoft Windows, dengan menggunakan model pemrograman "Common Object Model (COM)". visual basic merupakan turunan bahasa pemrograman BASIC yang menawarkan pengembangan perangkat lunak computer berbasis grafik dengan cepat. Dengan menggunakan bahasa pemrograman VB, para programmer dapat membangun aplikasi dengan menggunakan komponen-komponen yang di sediakan VB.

Microsoft Visual Basic (sering disingkat sebagai VB saja) merupakan sebuah bahasa pemrograman yang menawarkan Integrated Development Environment (IDE) visual untuk membuat program perangkat lunak berbasis sistem operasi Microsoft Windows dengan menggunakan model pemrograman (COM).

Visual Basic merupakan turunan bahasa pemrograman BASIC dan menawarkan pengembangan perangkat lunak komputer berbasis grafik dengan cepat, Beberapa Bahasa skrip seperti Visual Basic for Applications (VBA) dan Visual Basic Scripting Edition (VBScript), mirip seperti halnya Visual Basic, tetapi cara kerjanya yang berbeda Para programmer dapat membangun aplikasi dengan menggunakan komponen-komponen yang disediakan oleh Microsoft Visual Basic Program-program

yang ditulis dengan Visual Basic juga dapat menggunakan Windows API, tapi membutuhkan deklarasi fungsi luar tambahan dalam pemrograman untuk bisnis.

Dalam sebuah survey yang dilakukan pada tahun 2005, 62% pengembang perangkat lunak dilaporkan menggunakan berbagai bentuk Visual Basic, yang diikuti oleh C++, JavaScript, C#, dan Java.

## 2.10 Microsoft Access

Pengertian Microsoft Access menurut Ahmad Iskandar dalam modulnya yang berjudul Microsoft Access (Bandung, UNIKOM, 2003) adalah “ Microsoft Access adalah salah satu software database yang berjalan dibawah sistem windows, dengan Microsoft Access kita dapat merancang, memuat dan mengelola database dengan cara mudah dan cepat ”.

Menurut Yahya Kurniawan dalam bukunya Microsoft .Access 2003, pengertian Microsoft Access adalah “Sebuah sistem manajemen database atau Database Management System (DBMS). Dengan Access 2003 anda dapat menyimpan berbagai macam informasi (selanjutnya disebut data), mengatur, dan megolahnya sedemikian rupa agar data tersebut mudah digunakan kembali pada saat diperlukan “.

Maka dari uraian pengertian diatas dapat penulis simpulkan bahwa Microsoft Access adalah salah satu software database yang dapat menyimpan berbagai informasi untuk dapat diolah sedemikian rupa dengan cara mudah dan cepat.

Dalam dunia teknologi informasi saat ini karena tersedia banyak perangkat lunak database, arti dari database bisa sedikit berbeda antara satu perangkat dengan

perangkat lain. Dengan demikian, bagian access database hanyalah merupakan container dari objek-objek yang meliputi :

1. Table adalah tempat dimana data itu sesungguhnya disimpan. Data disusun membentuk baris dan kolom dengan bagian baris disebut record dan bagian kolom disebut field.
2. Form adalah “formulir” yang memudahkan user untuk memasukkan atau menampilkan data , bahkan menganalisisnya.
3. Query adalah sebuah proses pemilihan atau penyaringan data sehingga hanya data yang diinginkan (memenuhi criteria tertentu) yang akan ditampilkan atau dicetak.
4. Report adalah pemaparan data bentuk tercetak atau tertulis.
5. Macro adalah kumpulan dari sebuah perintah atau lebih yang digunakan untuk mengotomatisasi tugas-tugas yang sering dilakukan.
6. Modul adalah suatu unit pemrograman berbasis Visual Basic yang membantu proses. Proses yang mungkin ada dalam pengolahan data.

### **2.11 Blackbox Testing**

Menurut Budiman (2012:4), pengujian Blackbox adalah pengujian aspek fundamental sistem tanpa memperhatikan struktur logika internal perangkat lunak. Metode ini digunakan untuk mengetahui apakah perangkat lunak berfungsi dengan benar. Pengujian blackbox merupakan metode perancangan data uji yang didasarkan pada spesifikasi perangkat lunak. Data uji dieksekusi pada perangkat lunak dan kemudian keluar dari perangkat lunak dicek apakah telah sesuai yang diharapkan.

Pengujian Blackbox berusaha menemukan kesalahan dalam kategori :

1. Fungsi-fungsi yang tidak benar atau hilang
2. Kesalahan interface
3. Kesalahan dalam struktur data atau akses database eksternal
4. Kesalahan kinerja
5. Inisialisasi dan kesalahan terminasi

## **2.12 DATABASE**

Menurut Ayu (2015), Database atau basis data terdiri dari 2 kata, yaitu basis dan data. Basis dapat diartikan sebagai markas atau gudang, tempat bersarang atau berkumpul. Sedangkan data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, siswa, pembeli, pelanggan), barang, hewan, peristiwa, konsep, keadaan, dan sebagainya, yang direkam dalam bentuk angka, huruf, symbol, teks, gambar, bunyi, atau kombinasinya . Basis data sendiri dapat didefinisikan dalam sejumlah sudut pandang seperti:

1. Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.
2. Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redundansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan.
3. Kumpulan file/tabel/arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronik.

Basis data dan lemari arsip sesungguhnya memiliki prinsip kerja dan tujuan yang sama. Prinsip utamanya adalah pengaturan data/arsip, dan tujuan utamanya adalah kemudahan dan kecepatan dalam pengambilan kembali data/arsip. Perbedaannya hanya terletak pada media penyimpanan yang digunakan. Jika lemari arsip menggunakan lemari besi atau kayu sebagai media penyimpanan, maka basis data menggunakan media penyimpanan elektronik seperti disk (disket atau hard disk). Hal ini merupakan konsekuensi logis, karena lemari arsip langsung ditangani oleh manusia, sementara basis data ditangani melalui perantara alat/mesin pintar elektronik (yang dikenal dengan nama komputer). Perbedaan media ini yang selanjutnya melahirkan perbedaan-perbedaan lain yang menyangkut jumlah dan jenis metoda yang dapat digunakan dalam upaya penyimpanan.

Satu hal yang juga harus diperhatikan, bahwa basis data bukan hanya sekedar penyimpanan data secara elektronik (dengan bantuan komputer). Artinya, tidak semua bentuk penyimpanan data secara elektronik adalah basis data. Penyimpanan dokumen berisi data dalam file teks (dengan program pengolah kata), file spread sheet, dan lain-lain tidak dapat disebut sebagai basis data. Hal ini dikarenakan di dalamnya tidak ada pemilihan dan pengelompokan data sesuai jenis atau fungsi data, sehingga akan menyulitkan pencarian data nantinya.

Pengaturan / pemilihan / pengelompokan / pengorganisasian data yang disimpan sesuai dengan fungsi dan jenisnya sangat ditonjolkan dalam basis data.

Pemilihan/pengelompokan/pengorganisasian tersebut dapat berbentuk sejumlah file/tabel terpisah atau dalam bentuk pendefinisian kolom-kolom / field-field data dalam setiap file atau tabel.

Di dalam sebuah disk, basis data dapat diciptakan dan dapat juga ditiadakan. Di dalam sebuah disk, dapat ditempatkan beberapa (lebih dari satu) basis data. Sementara dalam sebuah basis data, dapat ditempatkan satu atau lebih file/tabel. Pada file/tabel tersebut data disimpan. Setiap basis data umumnya dibuat untuk mewakili sebuah semesta data yang spesifik. Misalnya, ada basis data kepegawaian, basis data akademik, basis data inventori (pergudangan), dan sebagainya. Contohnya dalam basis data akademik dapat disimpan file mahasiswa, file mata kuliah, file dosen, file jadwal, file nilai, dan seterusnya. Sehingga dapat diketahui operasi-operasi dasar yang berkenaan dengan basis data adalah sebagai berikut:

1. Pembuatan basis data baru (create database).
2. Penghapusan basis data (drop database).
3. Pembuatan file/tabel baru ke suatu basis data (create table).
4. Penghapusan file/tabel dari suatu basis data (drop table).
5. Penambahan/pengisian data baru ke sebuah file/tabel di sebuah basis data (insert).
6. Pengambilan data dari sebuah file/tabel (retrieve/search).
7. Pengubahan data dari sebuah file/tabel (update).
8. Penghapusan data dari sebuah file/tabel (delete).

Operasi yang berkenaan dengan pembuatan objek (basis data dan tabel) merupakan operasi awal yang hanya dilakukan sekali dan berlaku seterusnya. Sedangkan operasi-operasi yang berkaitan dengan isi tabel (data) merupakan operasi rutin yang akan berlangsung berulang-ulang dan oleh karena itu operasi-operasi inilah

yang lebih tepat mewakili aktivitas pengelolaan (management) dan pengolahan (processing) data dalam basis data.

Seperti tertulis di atas bahwa tujuan awal dan utama dalam pengelolaan data dalam sebuah basis data adalah agar dapat memperoleh dan menemukan kembali data dengan mudah dan cepat. Selain pengelolaan data, basis data juga dimanfaatkan untuk tujuan-tujuan lain. Berikut ini adalah pemanfaatan basis data yang dilakukan untuk memenuhi sejumlah tujuan (objektif) :

1. Kecepatan dan kemudahan (speed), pemanfaatan basis data dapat memungkinkan untuk dapat menyimpan data atau melakukan perubahan/manipulasi terhadap data atau menampilkan kembali data tersebut dengan lebih cepat dan mudah, dibandingkan dengan menyimpan data secara manual (non elektronik) atau secara elektronik tetapi tidak dalam bentuk penerapan basis data, misalnya dalam bentuk spread sheet atau dokumen teks biasa.
2. Efisiensi ruang penyimpanan (space), dengan basis data, efisiensi/optimalisasi penggunaan ruang penyimpanan dapat dilakukan, karena dapat dilakukan penekanan jumlah redundansi data, baik dengan menerapkan sejumlah pengkodean atau dengan membuat relasi-relasi (dalam bentuk file) antar kelompok data yang saling berhubungan.
3. Keakuratan (accuracy), pemanfaatan pengkodean atau pembentukan relasi antar data bersama dengan penerapan aturan/batasan (constraint) tipe data, domain data, keunikan data, dan sebagainya, yang secara ketat dapat

diterapkan dalam sebuah basis data, sangat berguna untuk menekan ketidakakuratan pemasukan/penyimpanan data.

4. Ketersediaan (availability), karena kepentingan pemakaian data, sebuah basis data dapat memiliki data yang disebar di banyak lokasi geografis. Misalnya data nasabah sebuah bank dipisah-pisah dan disimpan di lokasi yang sesuai dengan keberadaan nasabah. Dengan pemanfaatan teknologi jaringan komputer, data yang berada di suatu lokasi atau cabang, dapat juga diakses (menjadi tersedia/available) bagi lokasi/cabang lain.
5. Kelengkapan (completeness), untuk mengakomodasi kebutuhan kelengkapan data yang semakin berkembang, maka tidak hanya dengan menambah record - record data, tetapi juga melakukan perubahan struktur dalam basis data, baik dalam bentuk penambahan objek baru (tabel) atau dengan penambahan field baru pada suatu tabel.
6. Keamanan (security), pengaksesan terhadap basis data dapat ditentukan mengenai siapa saja yang boleh menggunakan basis data beserta objek-objek di dalamnya dan penentuan terhadap jenis-jenis operasi apa saja yang boleh dilakukan.
7. Kebersamaan pemakaian (shareability), penggunaan basis data tidak terbatas oleh satu pengguna saja, atau di satu lokasi tertentu atau oleh satu aplikasi/sistem saja. Tetapi basis data dapat dimanfaatkan untuk mendukung lingkungan multiuser.

### 2.13 SQL

SQL Server menurut Ketut Darmayuda (2007 : 19), Microsoft SQL Server merupakan salah satu database relational yang banyak digunakan oleh dunia usaha.

Sedangkan definisi SQL Server menurut Feri Djuandi (2002:5), SQL Server adalah sebuah sistem berarsitektur terbuka yang memungkinkan para pengembang program memperluas dan menambahkan fungsi-fungsi ke dalam database tersebut.

Bunafit Nugroho dan Indah Indriyana (2007:2), SQL Server 2000 merupakan salah satu produk DBMS (*Database Management System*) yang dibuat oleh Microsoft. SQL Server 2000 menawarkan beberapa fitur didalam pengelolaan database, ada 2 fitur yang biasa digunakan untuk mengelola database di dalam SQL Server 2000, yaitu:

1. Menggunakan Enterprise Manager

Fitur ini relatif mudah digunakan karena mode pengelolaannya berbasis GUI (*Graphical User Interface*), oleh karena itu cukup dengan metode *click* dan *drag*. Membuat database dan tabel serta manajemen yang lain sangat mudah.

2. Menggunakan SQL Query Analyzer

Fitur ini menggunakan Transact SQL (perintah-perintah SQL) untuk mengelola database di dalam SQL Server 2000.

Perintah-perintah Transact SQL merupakan pengembangan dari perintah-perintah SQL standard yang disesuaikan dengan manajemen database pada SQL Server. Transact SQL memungkinkan untuk dapat membuat database,

membuat tabel, mengubah struktur tabel, menghapus database, menghapus tabel, menyisipkan data, mengubah data dan lain-lain.

Bunafit Nugroho dan Indah Indriyana (2007:8), perintah-perintah SQL dikelompokkan menjadi 5 macam yaitu:

### 1. *Data Definitions Language (DDL)*

Adalah perintah SQL yang digunakan untuk mendefinisikan kerangka basis data, perintah-perintahnya adalah:

- a. *Create* : untuk membuat atau menciptakan objek basis data
- b. *Alter* : untuk memodifikasi atau mengubah objek basis data
- c. *Drop* : untuk menghapus objek basisdata
- d. Objek *database* yang dimaksud adalah basis data, tabel, *index*

### 2. *Data Manipulations Language (DML)*

Adalah perintah yang digunakan untuk mengoperasikan atau memanipulasi isi basis data, SQL menyediakan 4 perintah DML yaitu:

- a. *Select* : digunakan untuk mengambil data dari basis data
- b. *Delete* : digunakan untuk menghapus data pada basis data
- c. *Insert* : digunakan untuk menambahkan data kedalam tabel
- d. *Update* : digunakan untuk memodifikasi data pada basis data

### 3. *Security*

Adalah perintah-perintah yang digunakan untuk menjamin keamanan data, perintahnya antara lain:

- a. *Grant* : digunakan untuk memberikan akses kepada *user* tertentu ke basis data

b. *Revoke* : digunakan untuk mencabut hak akses dari *user*

4. *Integrity*

Adalah perintah-perintah yang digunakan untuk menjaga kesatuan data, seperti: *recover table* digunakan untuk memperbaiki tabel pada basis data.

5. *Auxilliary*

Adalah perintah-perintah pelengkap atau tambahan, seperti: *unload* dan *rename*.



## **BAB III**

### **ANALISIS DAN DESAIN**

#### **3.1 *Communication* (Komunikasi)**

Komunikasi yang dilakukan berupa konsultasi dengan Koordinator TA Jurusan TI.

Adapun data yang dibutuhkan dalam pembuatan aplikasi ini, yaitu :

1. Data jadwal perkuliahan jurusan TI.
2. Data matakuliah jurusan TI.
3. Data mahasiswa jurusan TI.

#### **3.1.1 Teknologi**

Penggunaan teknologi sangat erat kaitannya dengan proses pembuatan Penjadwalan Otomatis Ujian Tugas Akhir Jurusan Teknik Informatika Universitas Palangkaraya Berbasis Desktop ini. Secara umum tahapan analisa teknologi ini yaitu analisa yang dipergunakan untuk membuat desain dan koding dalam merancang dan membangun aplikasi ini.

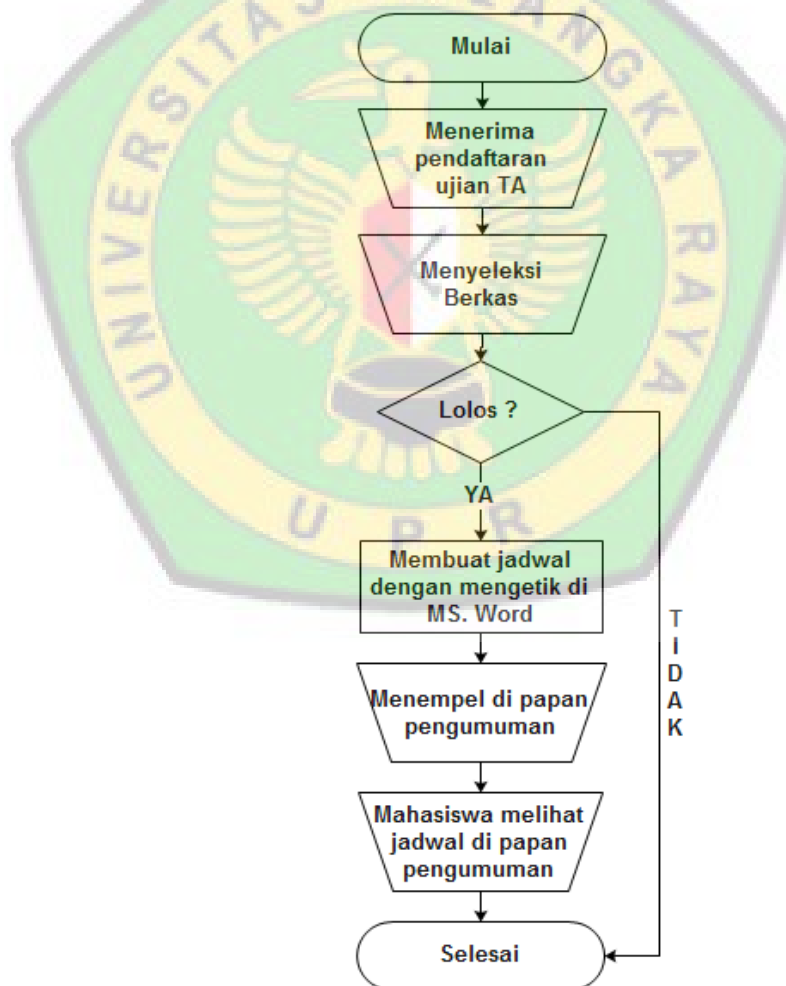
#### **3.2 *Planning* ( Perencanaan )**

Proses Proses *planning* merupakan lanjutan dari proses *communication* (komunikasi). Tahapan ini akan menghasilkan dokumen *user requirement* atau bisa dikatakan sebagai data yang berhubungan dengan keinginan user dalam pembuatan software, termasuk rencana yang akan dilakukan. Berikut merupakan proses bisnis sistem lama penjadwalan ujian tugas akhir dan rekomentasi sistem baru pembuatan Sistem Informasi Penjadwalan Otomatis Ujian Tugas Akhir Jurusan Teknik Informatika Universitas Palangka Raya Berbasis Desktop.

### 3.2.1 Analisis Sistem Lama

Bisnis proses sistem lama adalah alur kegiatan langkah per langkah dari sistem lama. Berdasarkan hasil konsultasi dan penelitian dilapangan pada Jurusan Teknik Informatika Universitas Palangka Raya dalam hal Sistem penjadwalah ujian Tugas Akhir masih menggunakan sistem manual yaitu dengan cara mengetiknya menggunakan MS. Word. Berikut merupakan bisnis proses sistem lama, flowchart sistem lama, dan kelemahan sistem lama :

#### A. Flowchart Sistem Lama



Gambar 3.1 Flowchart Sistem Lama

## **B. Bisnis Proses Sistem Lama**

1. Admin menerima pendaftaran ujian TA
2. Admin menyeleksi berkas mahasiswa yang mendaftar ujian TA.
3. Admin membuat Jadwal ujian TA dengan mengetiknya di MS. Word.
4. Admin menempel jadwal ujian TA di papan pengumuman.
5. Mahasiswa melihat jadwal ujian TA.

## **C. Kesimpulan**

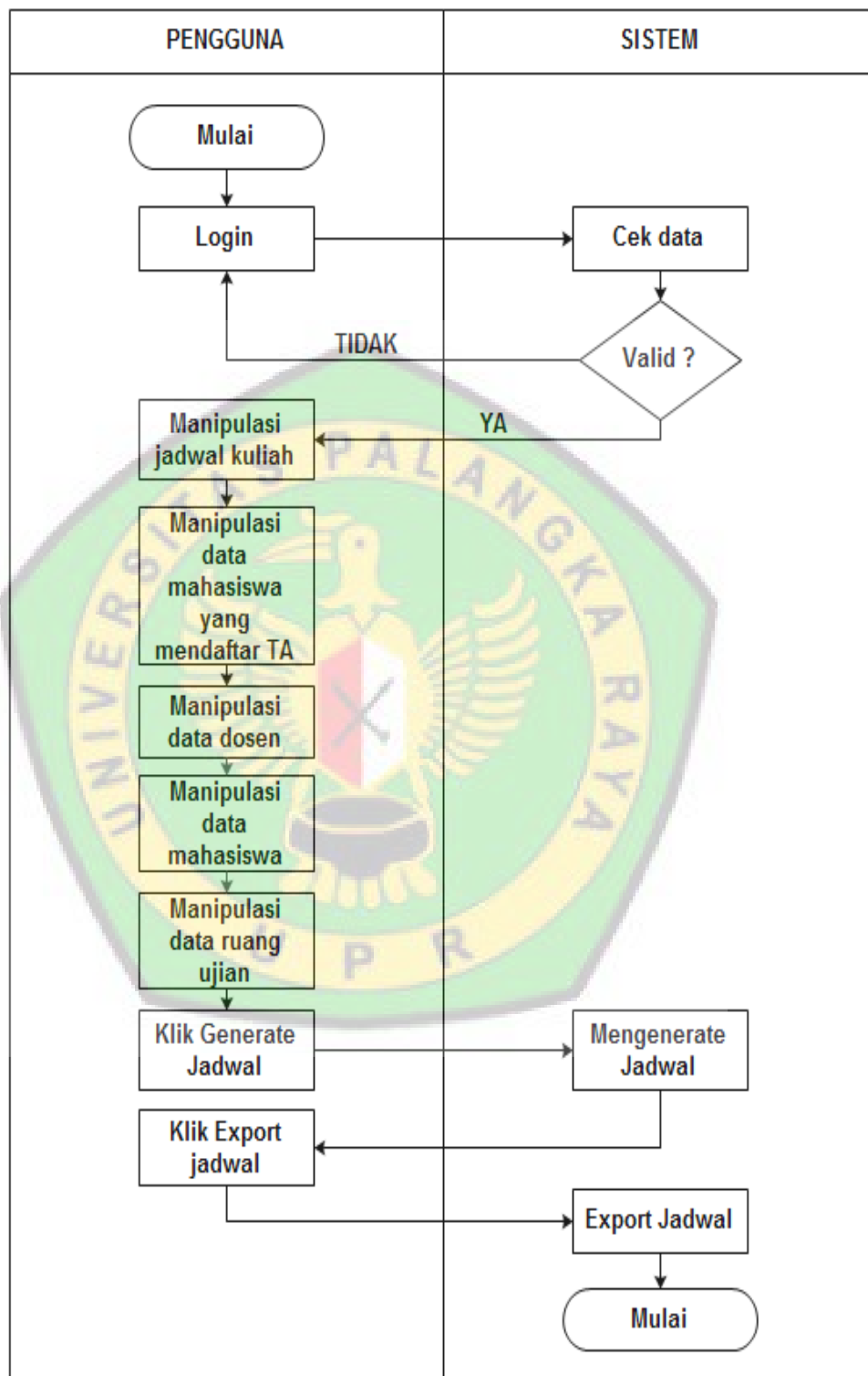
Jika dilihat dari bisnis proses maupun flowchart sistem lama memiliki beberapa kekurangan yaitu :

1. Admin harus mengetik jadwal ke dalam MS. Word sehingga memerlukan waktu yang lama untuk mengetiknya.
2. Seringkali terjadi bentrok jadwal Ujian Tugas akhir dengan jadwal matakuliah sehingga Dosen pengajar Tugas akhir harus meninggalkan kelas mengajar mata kuliah untuk menguji Tugas Akhir.

### **3.2.2 Analisis Sistem Baru**

Sistem baru yang ditawarkan adalah sistem Penjadwalan Otomatis Ujian Tugas Akhir Teknik Informatika Universitas Palangkaraya Berbasis Desktop. Berikut akan kami uraikan bisnis proses dari sistem Penjadwalan Tugas Akhir baru:

### A. Flowchart Sistem Baru



Gambar 3.2 Flowchar Sistem Baru

## B. Bisnis Proses Sistem Baru

### 1. Melakukan *Login*.

Proses awal dari sistem ini adalah pengguna dan *admin* melakukan login, kemudian sistem akan memeriksa username dan *password*, jika *password* benar maka sistem akan menampilkan halaman beranda sesuai dengan akses pengguna yang *login* .

### 2. Manipulasi data jadwal perkuliahan Teknik Informatika.

Pada proses ini pengguna memanipulasi data jadwal kuliah seperti Menambah, mengedit dan menghapus data.

### 3. Manipulasi data mahasiswa yang mendaftar Ujian Tugas Akhir.

Pada proses ini pengguna memanipulasi data mahasiswa yang mendaftar Ujian Tugas Akhir seperti Menambah, mengedit dan menghapus data.

### 4. Manipulasi data dosen Teknik Informatika.

Pada proses ini pengguna memanipulasi data dosen seperti Menambah, mengedit dan menghapus data.

### 5. Manipulasi data mahasiswa Teknik Informatika.

Pada proses ini pengguna memanipulasi data mahasiswa seperti Menambah, mengedit dan menghapus data.

### 6. Manipulasi data ruang ujian Tugas Akhir.

Pada proses ini pengguna memanipulasi data ruang ujian seperti Menambah, mengedit dan menghapus data.

### 7. Menggenerate jadwal ujian Tugas Akhir.

Pada tahap ini sistem akan membuat jadwal Ujian Tugas Akhir.

8. Export jadwal ujian Tugas Akhir.

C. Kesimpulan

Jika dilihat dari bisnis proses maupun flowchart sistem baru memiliki beberapa kelebihan yaitu :

1. Pengguna tidak perlu membuat jadwal ujian Tugas Akhir dengan MS. Word, pengguna hanya perlu menginputkan data, dan sistem akan membuat jadwal secara otomatis.
2. Meminimalisir terjadinya bentrok antara jadwal ujian Tugas Akhir dengan jadwal matakuliah.

**Tabel 3.1 Rencana Kegiatan**

No	Kegiatan	Estimasi Waktu Pelaksanaan
<b>1</b>	<b><i>Communication (Komunikasi)</i></b>	<b>3 Bulan</b>
	Observasi dan konsultasi untuk mengumpulkan kebutuhan	
<b>2</b>	<b><i>Planning (Perencanaan)</i></b>	
	Menganalisis Kebutuhan dan penyusunan rencana kerja	
<b>3</b>	<b><i>Modeling (Permodelan)</i></b>	
	Permodelan desain program dalam bentuk diagram	
	Perancangan desain antarmuka program	
<b>4</b>	<b><i>Construction (Konstruksi)</i></b>	
	Desain antarmuka program	
	Penulisan kode program ( <i>coding</i> )	
<b>5</b>	<b><i>Deployment (Penyerahan)</i></b>	
	Penyerahan Program dan evaluasi	

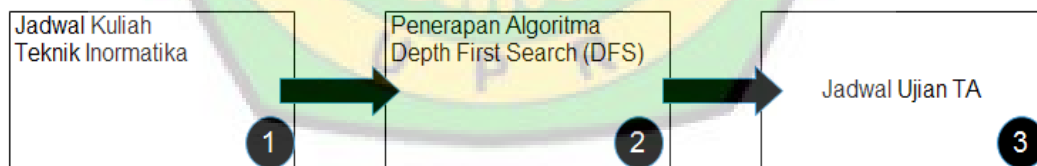
### 3.3 Modeling ( Permodelan )

Proses modeling ini akan menerjemahkan syarat kebutuhan ke sebuah perancangan software yang dapat diperkirakan sebelum dibuat coding. Proses ini terbagi menjadi 2, yaitu analisis dan desain. Pada analisis menggunakan Data Flow Diagram (DFD) serta Entity Relationship Diagram. Pada desain akan dilakukan desain tabel, desain navigasi menggunakan Sitemap dan desain antarmuka/representasi interface. Tahapan ini akan menghasilkan dokumen yang disebut software requirement.

#### 3.3.1 Analisis

##### 3.3.1.1 Algoritma *Generate Jadwal Praktikum*

Pada website ini terdapat fitur *generate* (membuat otomatis) jadwal praktikum. Algoritma yang digunakan yaitu algoritma *Depth First Search* (pencarian mendalam pertama) dimana algoritma ini digunakan untuk mendapatkan waktu kuliah yang kosong sehingga bisa diisi dengan jadwal praktikum. Berikut langkah-langkah generate jadwal praktikum pada website ini.



**Gambar 3.1 Alur Generate Jadwal Ujian TA**

Penjelasan dari gambar 3.1 alur generate jadwal Ujian Tugas Akhir, yaitu ;

Keterangan :

JKTI : Jadwal Kuliah Teknik Informatika

MKTI : Mata Kuliah Teknik Informatika

- MKTA : Mata Kuliah Tugas Akhir
- JUTA : Jadwal Ujian Tugas Akhir
- WU : Waktu Ujian (terdiri dari waktu ujian awal dan waktu akhir ujian)
- WK : Waktu Kuliah (terdiri dari waktu kuliah awal dan waktu kuliah akhir)
- SU : Sesi Ujian
- Node : Simpul

1. Jadwal Kuliah Teknik Informatika

JKTI berisi data seperti hari, waktu kuliah (waktu\_awal dan waktu\_akhir), matakuliah, kode matakuliah, jadwal.

2. Penerapan Algoritma DFS

Pada pembuatan generate jadwal ujian TA algoritma DFS digunakan untuk mencari jadwal ujian TA agar tidak bentrok dengan jadwal kuliah.

Input : JKTI

Proses :

a. Langkah 1

Membandingkan node jadwal, antara jadwal MKTI dengan jadwal MKTA. Node jadwal dimulai dari jadwal terendah ke tertinggi.

Catatan : Waktu MKTA **tidak boleh sama** dengan waktu MKTI pada jadwal yang sama.

b. Langkah 2

Membandingkan MKTI dengan MKTA pada JKTI.

Catatan : MKTI **tidak sama** dengan MKTA

c. Langkah 3

Membandingkan Node Hari. Node hari dimulai dari hari senin.

d. Langkah 4 :

Membandingkan Node WU dengan Node WK.

Jika node WU berada diluar node WK, maka input JUTA.

Jika JKTI kosong, maka input JUTA.

Menentukan node WU berada diluar node WK dengan

$$\begin{array}{c} ((\text{waktu awal ujian} \geq \text{waktu akhir kuliah}) \\ \text{OR} \\ (\text{waktu akhir ujian} \leq \text{waktu awal kuliah})) \end{array}$$

e. Langkah 5

Membandingkan Node Ruang Ujian(RU). Node RU JU dibandingkan dengan Node RU MKTA. Ruang Ujian bisa digunakan banyak MKTA.

Jika MKTA terah terisi pada Ruang Ujian tertentu dengan node hari dan node WU yang sama, maka MKTA **tidak boleh** diinput.

f. Langkah 6

Jumlah kelas ujian didapatkan berdasarkan jumlah mahasiswa yang mengambil MKTA tertentu dibagi kapasitas atau sarana yang digunakan.

Output : JUTA

3. Jadwal Ujian Tugas Akhir

Input : JUTA

Proses : export data JUTA ke MS. Exel

Output : file MS. Exel

➤ Jadwal Ujian Proposal

$\Sigma$  mahasiswa  $\times$  judul  $\times$  dosen pembimbing I  $\times$  dosen penguji  $\times$  waktu

$\Sigma$  looping mahasiswa

$\leftrightarrow$  mahasiswa  $\times$  judul  $\times$  dosen pembimbing I

$\leftrightarrow$  waktu  $\times$  MKTI

$\leftrightarrow$  insert jadwal

Tentukan dosen pembimbing II

$\leftrightarrow$  pengumuman hasil ujian proposal

Output : file Ms. exel

➤ Jadwal Ujian Hasil

$\Sigma$  looping mahasiswa  $\times$  jadwal proposal  $\times$  MKTI

$\leftrightarrow$  waktu

$\leftrightarrow$  sesi

$\leftrightarrow$  dosen pembimbing I  $\times$  dosen pembimbing II  $\times$  dosen penguji

$\leftrightarrow$  jadwal ujian hasil

$\leftrightarrow$  insert jadwal ujian hasil

Output : file Ms. Exel

$\leftrightarrow$  pengumuman hasil ujian hasil

➤ Jadwal Ujian Akhir

$\Sigma$  generate  $\times$  ujian akhir  $\times$  edit

$\Sigma$  looping

Mahasiswa  $\times$  jadwal hasil  $\times$  jadwal MK

↔ sesi

↔ waktu

↔ dosen pembimbing I × dosen pembimbing II × dosen penguji

↔ insert jadwal ujian akhir

Output : file Ms. Exel

Contoh Kasus :

1. JKTI, berisi data :

a. Hari : Senin, WK : 07 .00 – 08.00, MKTI : Bahasa Indonesia,  
Semester : 3

b. Hari : Selasa, WK : 08.30 – 09.30. MKTI : Basis Data, Semester : 5

2. Proses generate jadwal :

a. JKTI (1) Hari : Senin, WK : 07.00 – 08.00, MKTI : Bahasa  
Indonesia, Semester : 3

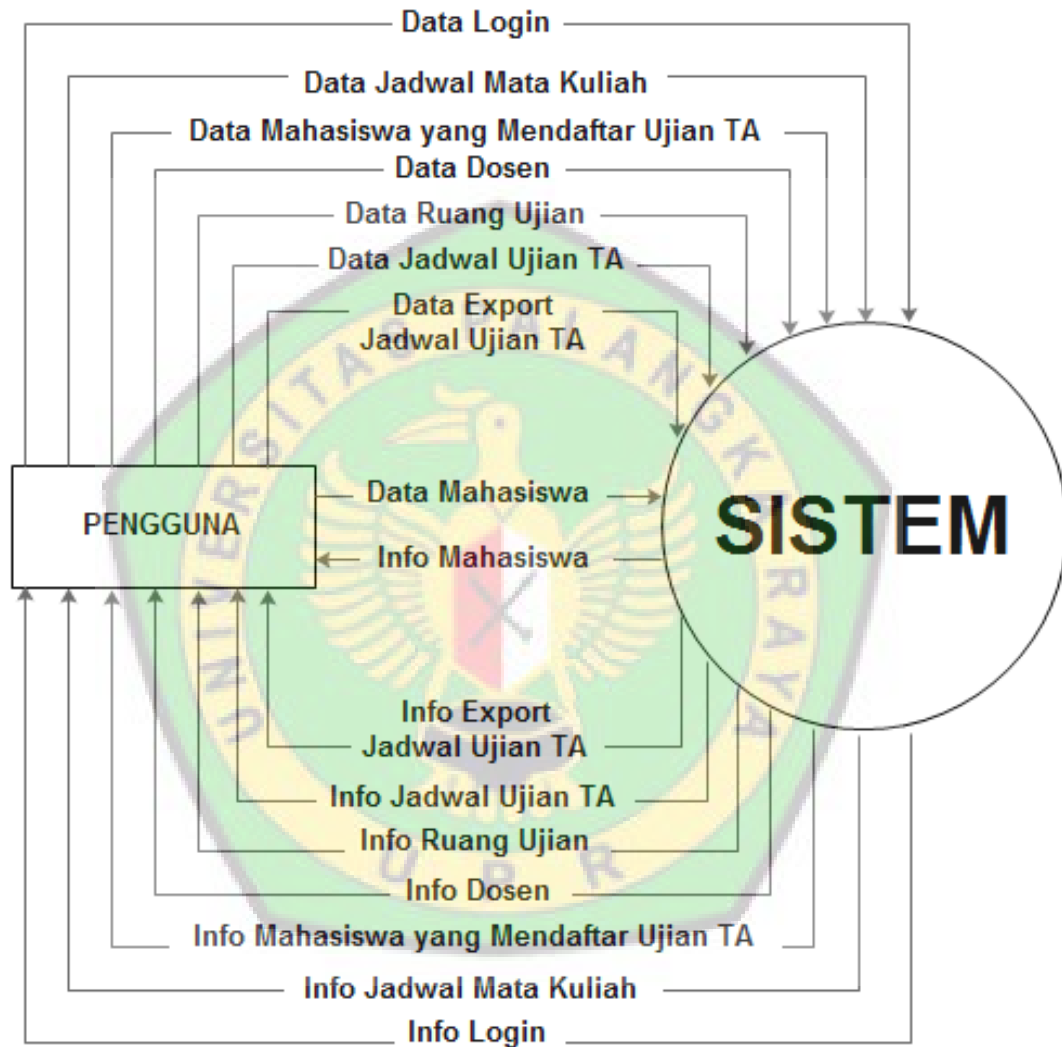
b. JKTI (2) Hari : Selasa WK : 08.30 – 10.30, MKTI : Basis Data,  
Semester : 5

### 3.3.1 Data Flow Diagram

Untuk memodelkan setiap proses yang ada pada sistem digunakan Data Flow Diagram (DFD) yang terbagi atas 3 diagram. Context diagram, *DFD( Data Flow Diagram )*.

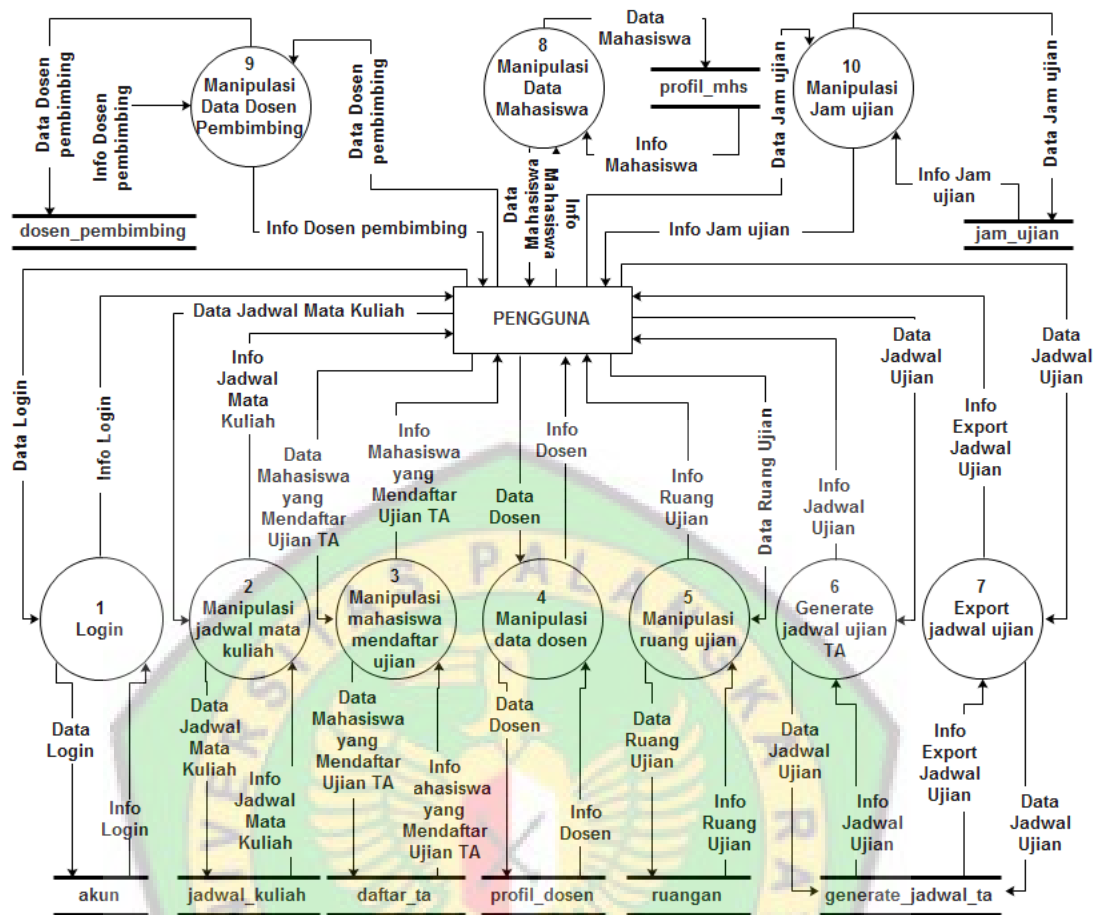
#### 3.3.1.1 Context Diagram

Context Diagram digunakan untuk menggambarkan proses kerja sistem secara umum. Pada sistem penjualan rumah ini yang menjadi entitas pada context diagram adalah pengguna.



Gambar 3.3 Diagram Konteks

### 3.3.1.2 Data Flow Diagram



**Gambar 3.4 Data Flow Diagram**

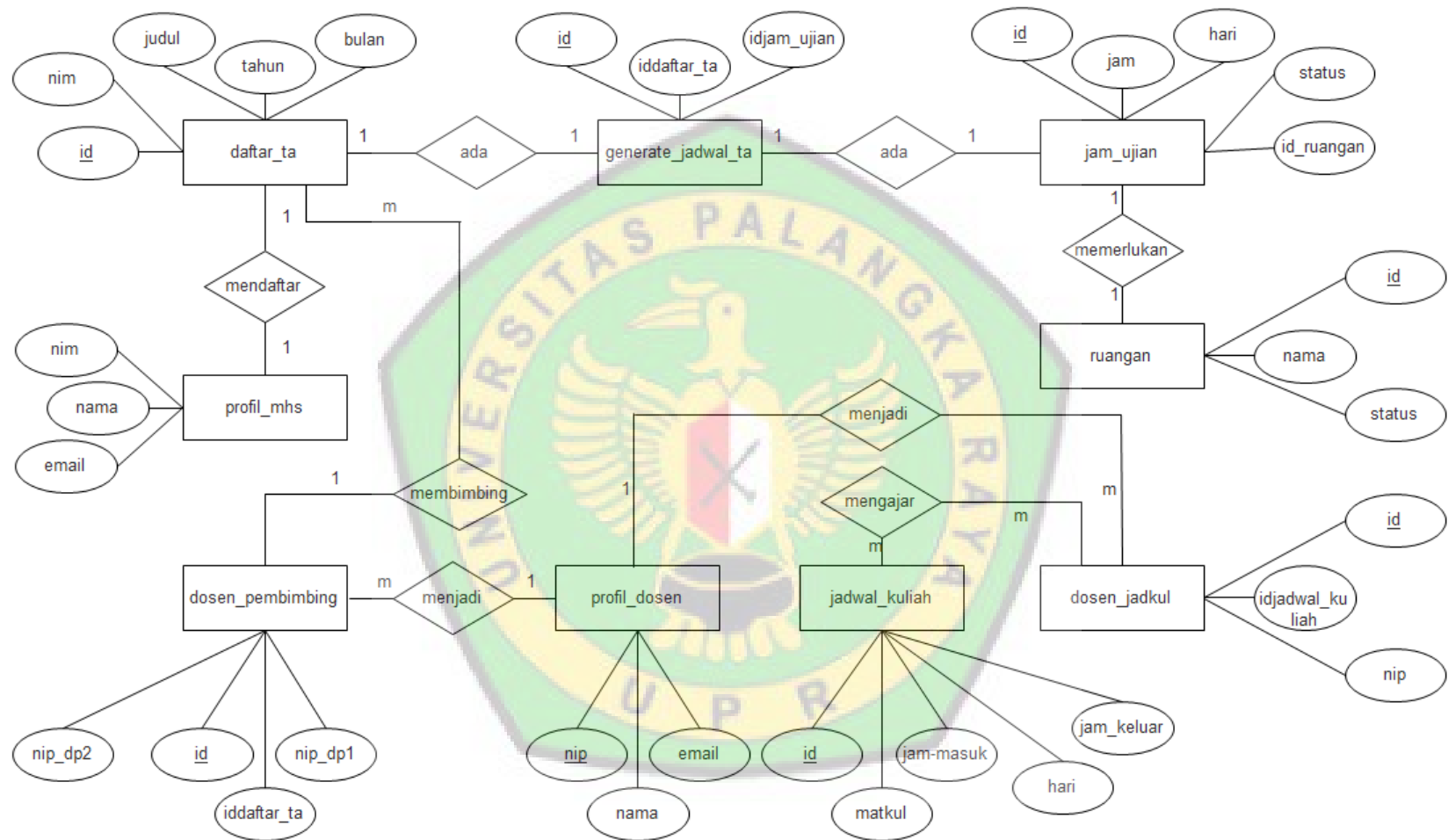
Dari Gambar 3.4 diatas dijelaskan proses DFD dan aliran datanya pada Tabel 3.1 di bawah ini.

**Tabel 3.1. Proses DFD**

No	Nama Proses	Deskripsi
1	Login	Proses login pengguna dimana pengguna menginputkan username dan password kemudian sistem akan mencari data di tabel akun berdasarkan username dan password
2	Manipulasi jadwal mata kuliah	Proses manipulasi data jadwal matakuliah dilakukan oleh pengguna. Manipulasi data jadwal mata kuliah meliputi Tambah , Edit dan Hapus Jadwal matakuliah

3	Manipulasi mahasiswa mendaftar ujian Tugas Akhir	Proses manipulasi mahasiswa yang mendaftar ujian tugas akhir dilakukan oleh admin yang mana datanya akan disimpan di table <u>daftar ta</u> . <u>Manipulasi data mahasiswa yang mendaftar ujian Tugas Akhir meliputi, Tambah, Edit dan Hapus.</u>
4	Manipulasi Dosen	Proses manipulasi dosen dilakukan oleh pengguna yang mana datanya akan disimpan di table profile_dosen. Manipulasi data dosen meliputi , Tambah, Edit dan Hapus.
5	Manipulasi ruang ujian	Proses manipulasi data ruang ujian dilakukan oleh pengguna yang mana datanya disimpan di tabel runga. Manipulasi data ruang ujian meliputi Tambah, Edit dan Hapus.
6	Generate jadwal ujian ta	Proses generate jadwal ujian dilakukan oleh sistem yang diberi perintah oleh admin, yang mana datanya disimpan di atbel generate_jadwal_ta
7	Export jadwal ujian	Proses export jadwal ujian dilakuka oleh sistem yang telah diberi perintah oleh admin.
8	Manipulasi Data mahasiswa	Proses manipulasi ahasiswa dilakukan oleh pengguna yang mana datanya akan disimpan di table profile_mhs. Manipulasi data dosen meliputi , Tambah, Edit dan Hapus.
9	Manipulasi data dosen pembimbing	Proses manipulasi data dosen pembimbing dilakukan oleh pengguna yang mana datanya akan disimpan di tabel dosen_pembimbing. Manipuasi data dosen pembimbing meliputi Tambah, Edit dan Hapus
10	Manipulasi jam ujian	Manipulasi jam ujian dilakukan oleh pengguna yang mana datanya disimpan di tabel jam_ujian. Manipulasi jam ujian meliputi Tambah, Edit dan Hapus

### 3.2.1.4 Entity Relationship Diagram



Gambar 3.5 Entity Relationship Diagram

Tabel 3.2 Keterangan Entitas pada ERD

No	Nama	Deskripsi	Atribut	Primary Key	Foreign Key
1	daftar_ta	Menyimpan data pendaftaran ujian	- Id - Nim - Judul - Tahun - bulan	id	-nim
2	Profil_mhs	Menyimpan data mahasiswa	- nim - nama - email	nim	-
3	Generate_jadwal_ta	Menyimpan jadwal ujian	- id - iddaftar_ta - idjam_ujian	id	- iddaftar_ta - idjam_ujian
4	Jam_ujian	Menyimpan data waktu ujian	- id - jam - hari - status - id_ruangan	id	Id_ruangan
5	ruangan	Menyimpan data ruangan	- id - nama - status	id	-
6	Dosen_jadkul	Menyimpan data jadwal mengajar dosen	- id - idjadwal_kuliah - nip	id	idjadwal_kuliah
7	Jadwal_kuliah	Menyimpan data adwal matakuliah	- id - matkul - jam_masuk - jam_keluar - hari	id	
8	Profil_dosen	Menyimpan data dosen	- nip - nama - email	nip	
9	Dosen-pembimbing	Menyimpan data dosen pembimbing	- id - iddaftar_ta - nip_dp1 - nip_dp2	id	Iddaftar_ta

### 3.3.1.5 Desain Tabel

Desain tabel merupakan rancangan isi dari database yang akan di gunakan dalam pembuatan aplikasi. Berikut tabel – tabel yang akan di gunakan.

#### 1. Tabel akun

Tabel akun digunakan untuk menyimpan data – data admin di dalam database pada Gambar 3.3 berikut.

**Tabel 3.3 Tabel akun**

No	Nama <i>Field</i>	Tipe Data	Lebar
1	Username	Varchar	5
2	Password	Varchar	5

#### 2. Tabel generate\_jadwal\_ta

Tabel generate\_jadwal\_t digunakan untuk menggerate jadwal TA di dalam database pada Tabel 3.4 berikut.

**Tabel 3.4 Tabel generate\_jadwal\_ta**

No	Nama <i>Field</i>	Tipe Data	Lebar
1	Id	Int	10
2	Iddaftar_ta	Int	10
3	Idjam_ujian	Int	10

#### 3. Tabel daftar\_ta

**Tabel 3.5 Tabel daftar\_ta**

No	Nama <i>Field</i>	Tipe Data	Lebar
1	Id	Int	10
2	Nim	Varchar	10
3	Judul	varchar	100
4	Tahun	int	5
5	Bulan	int	2

#### 4. Tabel jam\_ujian

Tabel jam\_ujian digunakan untuk menyimpan data jam\_ujian di dalam database pada Tabel 3.6 berikut.

**Tabel 3.6 Tabel jam\_ujian**

No	Nama <i>Field</i>	Tipe Data	Lebar
1	Id	Int	10
2	Jam	Varchar	10
3	Hari	Enum('senin', 'selasa', 'rabu', 'kamis', 'jumat')	
4	Status	Enum('Tersedia', 'Terpakai')	
5	Idruangan	Int	10

#### 5. Tabel ruangan

Tabel ruangan digunakan untuk menyimpan data ruangan di dalam database pada Tabel 3.7 berikut.

**Tabel 3.7 Tabel ruangan**

No	Nama <i>Field</i>	Tipe Data	Lebar
1	Id	Int	10
2	Nama	Varchar	30
3	Status	Enum('Tersedia', 'Terpakai')	

#### 6. Tabel profil\_mhs

Tabel profil\_mhs digunakan untuk menyimpan data profil\_mhs di dalam database pada Tabel 3.8 berikut.

**Tabel 3.8 Tabel profil\_mhs**

No	Nama <i>Field</i>	Tipe Data	Lebar
1	Id	Int	10
2	Iddaftar_ta	Int	10
3	Idjam_ujian	Int	10

### 7. Tabel *jadwal\_kuliah*

Tabel *jadwal\_kuliah* digunakan untuk menyimpan data *jadwal\_kuliah* di dalam database pada Tabel 3.9 berikut.

**Tabel 3.9 Tabel *jadwal\_kuliah***

No	Nama <i>Field</i>	Tipe Data	Lebar
1	Id	Int	10
2	Matkul	Varchar	20
3	Jam_masuk	Varchar	10
4	Jam_keluar	Varchar	10
5	hari	Enum('senin','selasa','rabu','kamis','jumat')	

### 8. Tabel *profil\_dosen*

Tabel *profil\_dosen* digunakan untuk menyimpan data *profil\_dosen* di dalam database pada Tabel 3.10 berikut.

**Tabel 3.10 Tabel *profil\_dosen***

No	Nama <i>Field</i>	Tipe Data	Lebar
1	Nip	Varchar	18
2	Nama	Varchar	30
3	Email	Varchar	30

### 9. Tabel *dosen\_pembimbing*

Tabel *dosen\_pembimbing* digunakan untuk menyimpan data *dosen\_pembimbing* di dalam database pada Tabel 3.11 berikut.

**Tabel 3.11 Tabel *dosen\_pembimbing***

No	Nama <i>Field</i>	Tipe Data	Lebar
1	Id	Int	10
2	Iddaftar	Int	10
3	Nip_dp1	Varchar	18
4	Nip_dp2	varchar	18

## 10. Tabel dosen\_jamkul

Tabel dosen\_jamkul digunakan untuk menyimpan data dosen\_jamkul di dalam database pada Tabel 3.12 berikut.

**Tabel 3.12 Tabel dosen\_jamkul**

No	Nama <i>Field</i>	Tipe Data	Lebar
1	Id	Int	10
2	Idjadwal_kuliah	Int	1
3	Nip	Varchar	18

### 3.4 Desain Antarmuka

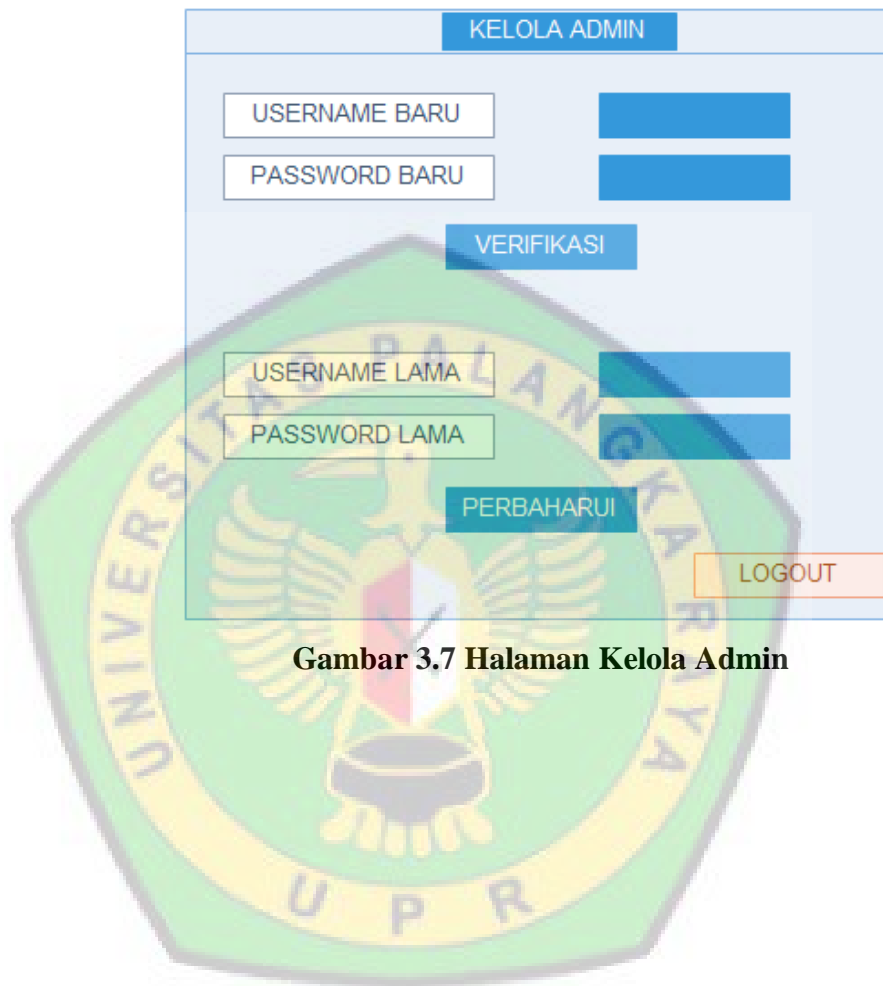
#### A. Login

Gambar 3.6 ini menggambarkan halaman login yang merupakan halaman pertama yang diakses oleh admin sebagai proses awal dalam melakukan pengolahan. Untuk melakukan login, admin harus memasukkan *username* dan *password*.

**Gambar 3.6 Halaman Login**

## 1. Halaman Kelola Admin

Gambar 3.7 ini menggambarkan halaman kelola admin yang merupakan pembaharuan *username* dan *password*.



The image shows a web interface for 'KELOLA ADMIN' (Admin Management). The page features a light blue header with the title 'KELOLA ADMIN'. Below the header, there are two sets of input fields. The first set includes 'USERNAME BARU' (New Username) and 'PASSWORD BARU' (New Password), each followed by a blue input box. Below these is a blue button labeled 'VERIFIKASI' (Verify). The second set includes 'USERNAME LAMA' (Old Username) and 'PASSWORD LAMA' (Old Password), each followed by a blue input box. Below these is a blue button labeled 'PERBAHARUI' (Update). In the bottom right corner, there is an orange button labeled 'LOGOUT'. A large, semi-transparent watermark of the University of Pahlawan Rakyat (UPR) logo is overlaid on the page. The logo is a green shield with a yellow border, featuring a yellow eagle with a red and white shield on its chest, and the text 'UNIVERSITAS PAHLAWAN RAKYAT' and 'U P R'.

**Gambar 3.7 Halaman Kelola Admin**

## **BAB IV**

### **IMPLEMETASI DAN PENGUJIAN**

#### **4.1 Implementasi Antarmuka**

Untuk perancangan antarmuka digunakan dua metode pemrograman, aplikasi server menggunakan Database MySQL dan aplikasi menggunakan PHP. Pada implementasi antarmuka Penjadwalan Otomatis Ujian Tugas Akhir Jurusan Teknik Informatika Universitas Palangkaraya Berbasis Desktop ini dibagi menjadi delapan, yaitu implementasi antarmuka halaman admin, halaman profil mahasiswa, halaman profil dosen, halaman tugas akhir, halaman jadwal ujian, halaman ruangan, halaman jadwal kuliah dan halaman hasil generate.

##### **4.1.1 Implementasi antarmuka Halaman Admin**

Pada implementasi antarmuka halaman admin ini akan menjelaskan halaman login, dashboard dan fitur-fitur yang dapat diakses admin.

Control Panel

Akun

Profil Mahasiswa   Profil Dosen   Tugas Akhir   Jadwal Kuliah   Jadwal Ujian   Ruangan   Hasil Generate

Profil Mahasiswa

Nama

NIM

Email

NIM	Nama	Email
db 114 085	agustriani	agustriani@gmail.com
dbc 111 081	alan alfredo sitanggang	alan@gmail.com
dbc 111 089	yenny astion rompas	yenny@gmail.com
dbc 111 114	akhmad noor fajar	akhmad@gmail.com
dbc 112 034	mia anjani sambolinggi	mia@gmail.com
dbc 112 067	m. farits dwi kusuma	farits@gmail.com
dbc 112 070	yus winaryo	yus@gmail.com
dbc 112 080	mukin daritoyo	mukin@gmail.com
dbc 112 084	kevin odriel panenga	kevin@gmail.com
dbc 112 086	pandu galih	pandu@gmail.com
dbc 112 093	mentari novia sinta	mentari@gmail.com
dbc 112 099	ardiansun	ardiansun@gmail.com
dbc 112 117	daniel marcho eli katparu	daniel@gmail.com
dbc 112 131	mario frasetyo	mario@gmail.com
dbc 112 157	fupon mandin	fupon@gmail.com
dbc 112 161	deby chrisman	chrisman@gmail.com
dbc 112 171	andika bayu wicahyono	andika@gmail.com
dbc 113 005	mustafa arifin	mustafa@gmail.com
dbc 113 023	clara agnes t	clara@gmail.com
dbc 113 069	oktapianus parulianda sirat	oktapianus@gmail.com
dbc 113 098	muhamad an hidayat	hidayat@gmail.com
dbc 113 135	thio samhong	thio@gmail.com
dbc 113 140	daniel pratama	pratama@gmail.com
dbc 114 012	sonny grimaldi	sonny@gmail.com
dbc 114 014	muhammad azis rizqi anugerah	azisrizqi@gmail.com
dbc 114 027	septyoga	septyoga@gmail.com
dbc 114 042	maria kristin efria dita	maridita@gmail.com
dbc 114 048	kasih karunia	kasih@gmail.com

Gambar 4.1 fitur Profil Mahasiswa

Akun

Profil Mahasiswa Profil Dosen Tugas Akhir Jadwal Kuliah Jadwal Ujian Ruangan Hasil Generate

Profil Dosen

Nama

NIP

Email

NIP	Nama	Email
196304231985021001	Drs. Jadianan Parhusip, M.Kom	Drs.JadiananParhusip@gmail.com
197505052008011032	V. Abdi Gunawan, ST., MT.	VAbdiGunawan@gmail.com
197512122003121002	Abertun Sagit Sahay, ST., M.Eng	Abertunsagitsahay@gmail.com
197601182003122003	Felicia Sylviana, ST., MM	FeliciaSylviana@gmail.com
197605092008122001	Licantik, S.Kom., M.Kom	Licantik@gmail.com
197606242005011015	Rony Teguh, S.Kom., MT.Ph.D	Ronyteguh@gmail.com
197910092008012016	Nahumi Nugrahaningsih, Ph.D	NahumiNugrahaningsih@gmail.com
198003222005012004	Ariesta Lestari, S.Kom., M.Cs	AriestaLestari@gmail.com
198012262008121002	Deddy Ronaldo, ST., MT.	DeddyRonaldo@gmail.com
198106062005011001	Viktor H. Pranatawijaya, ST., MT.	ViktorHPranatawijaya@gmail.com
198109292006042001	Sherly Christina, S.Kom., M.Kom	SherlyChristina@gmail.com
198110032006042001	Enny D. Oktaviani, ST., M.Kom	EnnyDOktaviani@gmail.com
198207172003122002	Widiatry, ST., MT.	Widiatry@gmail.com
198212062006042001	Devi Karolita, S.Kom., M.Kom	DeviKarolita@gmail.com
198508182012121003	Agus S. Saragih, ST., M.Eng	AgusSSaragih@gmail.com
198702032014041001	Ade Candra Saputra, S.Kom., M.Cs	AdeCandraSaputra@gmail.com
198904072015042004	Nova Noor Kamala Sari, ST., M.Kom	NovaNoorKamalaSari@gmail.com
198910222015041001	Putu Bagus A.A.P, ST., M.Kom	PutuBagusAAP@gmail.com

Gambar 4.2 Fitur Profil Dosen

Akun

Profil Mahasiswa Profil Dosen Tugas Akhir Jadwal Kuliah Jadwal Ujian Ruangan Hasil Generate

Informasi Tugas Akhir

NIM

Nama

NIP Dosen Pemb. 1

Nama

NIP Dosen Pemb. 2

Nama

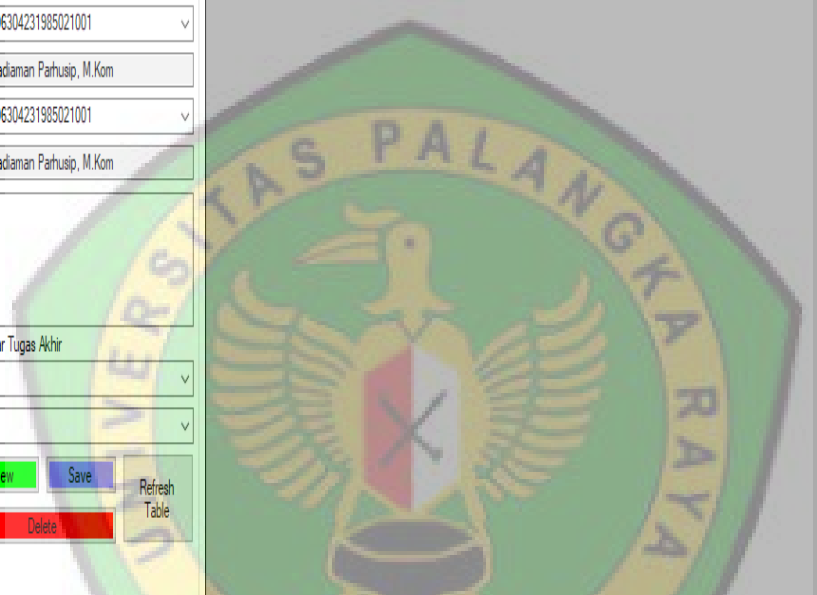
Judul TA

Seminar Tugas Akhir

Tahun

Bulan

NIM	Nama Mahasiswa	Dosen Pembimbing 1	Dosen Pembimbing 2	Tahun	Bulan
112 093	mentari novia sinta	Viktor H. Pranatawijaya, ST., MT.	Sherly Christina, S.Kom., M.Kom	2019	7



**Gambar 4.3 Fitur Tugas Akhir**

Akun

Profil Mahasiswa Profil Dosen Tugas Akhir Jadwal Kuliah **Jadwal Ujian** Ruangan Hasil Generate

Jadwal

Jam: 7:00

Status: Tersedia

Hari: Senin

Ruangan: Ruang A

New Save Refresh Table Delete

Jam	Hari	Status	Ruangan
10:00	Selasa	Tersedia	Ruang D
10:00	Selasa	Tersedia	Ruang B
10:00	Senin	Tersedia	Ruang B
10:00	Kamis	Tersedia	Ruang A
10:00	Rabu	Tersedia	Ruang D
11:30	Kamis	Tersedia	Ruang A
12:30	Senin	Tersedia	Ruang D
12:30	Selasa	Tersedia	Ruang D
12:30	Senin	Tersedia	Ruang A
13:00	Selasa	Tersedia	Ruang B
13:30	Kamis	Tersedia	Ruang B
13:30	Kamis	Tersedia	Ruang D
14:00	Selasa	Tersedia	Ruang A
14:00	Senin	Tersedia	Ruang D
14:00	Selasa	Tersedia	Ruang B
8:00	Selasa	Tersedia	Ruang D
8:00	Senin	Tersedia	Ruang A
8:00	Rabu	Tersedia	Ruang B
8:00	Kamis	Tersedia	Ruang B
9:30	Selasa	Tersedia	Ruang A
9:30	Selasa	Tersedia	Ruang B
9:30	Senin	Tersedia	Ruang A
9:30	Senin	Tersedia	Ruang B
9:30	Rabu	Tersedia	Ruang A

Gambar 4.4 Fitur Jadwal Ujian

Akun

Profil Mahasiswa Profil Dosen Tugas Akhir Jadwal Kuliah Jadwal Ujian Ruangan Hasil Generate

Informasi Tugas Akhir

NIM: dbc 111 081

Nama: alan alfredo sitanggang

NIP Dosen Pemb. 1: 196304231985021001

Nama: Drs. Jadianan Parhusip, M.Kom

NIP Dosen Pemb. 2: 196304231985021001

Nama: Drs. Jadianan Parhusip, M.Kom

Judul TA:

Seminar Tugas Akhir

Tahun: 2019

Bulan: 7

New Save Refresh Table Delete

NIM	Nama Mahasiswa	Dosen Pembimbing 1	Dosen Pembimbing 2	Tahun	Bulan
dbc 111 081	alan alfredo sitanggang	Abertun Sagit Sahay, ST., M.Eng	Felicia Sylviana, ST., MM	2019	7
dbc 112 007	yus winaryo	Abertun Sagit Sahay, ST., M.Eng	Ariesta Lestari, S.Kom., M.Cs	2019	7
dbc 112 080	mukin dantiyono	Abertun Sagit Sahay, ST., M.Eng	Drs. Jadianan Parhusip, M.Kom	2019	7
dbc 112 086	pandu gaih	Abertun Sagit Sahay, ST., M.Eng	Widiatry, ST., MT.	2019	7
dbc 112 093	mentari novia sinta	Wktor H. Pranatawijaya, ST., MT.	Sheryl Christina, S.Kom., M.Kom	2019	7
dbc 112 099	ardiansun	Wktor H. Pranatawijaya, ST., MT.	Agus S. Saragih, ST., M.Eng	2019	7
dbc 112 157	fupon mandiri	Rony Teguh, S.Kom., MT.Ph.D	Enny D. Oktavijani, ST., M.Kom	2019	7

Gambar 4.5 Fitur Tugas Akhir

Akun

Profil Mahasiswa Profil Dosen Tugas Akhir **Jadwal Kuliah** Jadwal Ujian Ruangan Hasil Generate

Jadwal

Mata Kuliah

Jam Masuk

Jam Keluar

Hari

Dosen Kuliah

Mata Kuliah	Jam Masuk	Jam Keluar	Hari
Basis Data I	12:30	14:00	Senin
Metodologi Penelitian	7:00	8:30	Senin
Multimedia	13:30	14:30	Selasa
Pengantar Arsitektur Komputer	8:00	10:00	Selasa
Sistem Infotmasi Manajemen	7:00	8:30	Rabu

Gambar 4.6 Fitur Jadwal Kuliah

Akun

Profil Mahasiswa Profil Dosen Tugas Akhir Jadwal Kuliah Jadwal Ujian Ruangan Hasil Generate

Generate

Bulan

Tahun

Refresh Table

Export ke Excel

Nim	Nama Mahasiswa	Jam	Hari	Ruangan
dbc 112 099	ardiansun	8:00	Senin	Ruang A
dbc 112 157	fupon mandiri	13:00	Selasa	Ruang B
dbc 112 007	yus winaryo	14:00	Selasa	Ruang B
dbc 112 093	mentari novia sinta	10:00	Selasa	Ruang B
dbc 111 081	alan alfredo sitanggang	10:00	Selasa	Ruang B
dbc 112 086	pandu galih	9:30	Selasa	Ruang B
dbc 112 080	mukin daityono	9:30	Rabu	Ruang A

Gambar 4.7 Fitur Hasil Generate

Akun

Profil Mahasiswa Profil Dosen Tugas Akhir Jadwal Kuliah Jadwal Ujian **Ruangan** Hasil Generate

Jadwal

Ruangan

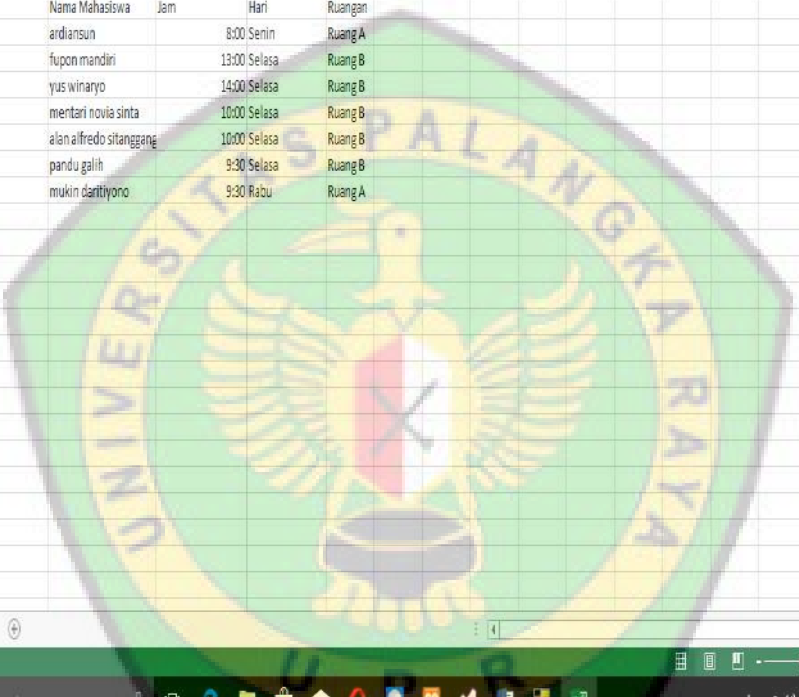
Status

Nama	Status
Ruang A	Tersedia
Ruang B	Tersedia
Ruang C	Terpakai
Ruang D	Tersedia

**Gambar 4.8 Fitur Ruangan**

Jadwal Ujian Bulan 7 Tahun 2019 - Microsoft Excel

id	Nim	Nama Mahasiswa	Jam	Hari	Ruangan
5	dbc112099	ardiansun	8:00	Senin	Ruang A
6	dbc112157	fupon mandiri	13:00	Selasa	Ruang B
1	dbc112007	yus winaryo	14:00	Selasa	Ruang B
3	dbc112093	mentari novia sinta	10:00	Selasa	Ruang B
7	dbc111081	alan alfredo sitanggang	10:00	Selasa	Ruang B
2	dbc112086	pandu galih	9:30	Selasa	Ruang B
4	dbc112080	mukin darithyono	9:30	Rabu	Ruang A



**Gambar 4.9 Hasil Generate Penjadwalan Otomatis**

## DAFTAR PUSTAKA

- Agustiar, Budiman. 2012. *Pengujian Perangkat Lunak dengan Metode Black Box Pada Proses Pra Registrasi User Via Website*. Makalah, halaman : 4.
- Aktama, Gede Erik. 2014. *Cascading style Sheet (CSS)*. (<http://aktama.com/wp-content/uploads/2014/02/Pertemuan-5-CascadingStyle-Sheet-CSS.pdf>) Diakses 15 April 2016.
- Budiyarto, Anwar. 2015. *Pengertian PHP*. ( <http://budiyarto.net/pengenalan-php/>) Diakses 15 April 2016.
- Ceri, Stefano dan Piero Fraternali.2000. *Computer Networks: The International Journal of Computer and Telecommunications Networking*. New York. North-Holland Publishing Co. (<http://books.google.co.id>). Diakses Tanggal 15 April 2016.
- Hariyanto, Rakhmad Satriya. 2012. *Sistem Informasi Pendaftaran dan Penilaian Praktikum Di Laboratorium Teknik Informatika UPN "VETERAN" JATIM*. Universitas Pembangunan Nasional "VETERAN". Surabaya.
- Harsiti. 2012. *Sistem Basis Data*. (<https://www.academia.edu/5528409/Sistembasis-data>) Diakses 15 April 2016.
- Husna, Aulia. 2013. *Algoritma dan Pemrograman*. ([https://www.academia.edu/6228630/Bab\\_4\\_Algoritma\\_Pencarian](https://www.academia.edu/6228630/Bab_4_Algoritma_Pencarian)) Diakses 18 April 2016.
- Jannah, Nurul. (2012). *Implementasi Kecerdasan Buatan Untuk Sistem Informasi Penjadwalan Ruang Dan Kelas Pada Universitas Almuslim berbasis web*. Universitas Almuslim. Bireuen.
- Lestaringtyas, Ayu. 2015. *Basis Data dan Pengembangan Sistem*. ([http://ayulestaringtyas.widyatama.ac.id/2016/03/11/basis-data dan-pengembangan-sistem/](http://ayulestaringtyas.widyatama.ac.id/2016/03/11/basis-data-dan-pengembangan-sistem/)) Diakses 15 April 2016.

- Lutfi Indrawan, Muhammad. 2013. Pengenal dan Instalasi XAMPP. V. Institut Teknologi Bandung. Bandung.
- Pressman, Roger S. 2010. Pendekatan Praktisi Rekayasa Perangkat Lunak Edisi 7. Andi. Yogyakarta.
- Rahman, Syakir. 2014. Mengenal Apa Itu PHP (Hypertext PreProcessor). (<http://www.syakirrohman.net/2014/06/mengenal-apa-itu-php-hypertextpreprocessor.html>) Diakses 15 April 2016.
- Rosa A. S dan M. Shalahuddin. 2011. Modul Pembelajaran Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek). Modula. Bandung.
- Saputra, Agus dan Feni Agustin. 2012. Membangun Sistem Aplikasi E-Commerce dan SMS. PT Alex Media Komputindo. Jakarta.
- Suprianoor. 2012. Rancang Bangun Sistem Informasi Manajemen Praktikum Menggunakan Website Studi Kasus : Jurusan Teknik Informatika Fakultas Teknik Universitas Palangka Raya. Universitas Palangka Raya. Palangka Raya.
- Sutabri, Tata. 2012. Analisis Sistem Informasi. Andi. Yogyakarta.
- Sutojo, T., Mulyanto Edy., dan Suhartono Vincent., Kecerdasan Buatan. Yogyakarta: Andi Offset, 2011.
- Tsauri, Asep Sufyan. 2008. Metode Asynchronous and XML (AJAX) (<http://dokumen.tips/documents/ajax-makalah-seminar-senin-050109.html>) Diakses 26 Mei 2016.