

LAPORAN PRAKTIKUM KECERDASAN BUATAN
DATA GUGUR PRAKTIKUM MAHASISWA TI 2015



Disusun Oleh :

Rahmawati Rina H	1515015008
Sri Marsiah	1515015021
Edwin Ginanjar Firdaus	1515015022

Asisten Praktikum :

Niken Novirasari	M.Hilmy Ady S.	Anisa Nur Afiah
1415015064	1415055058	1415015068

PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI DAN KOMUNIKASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS MULAWARMAN

2017

KATA PENGANTAR

Dengan mengucapkan puji syukur kehadiran Tuhan yang Maha Esa, yang telah memberikan berkat, rahmat, serta karunia-Nya, sehingga kami dapat menyelesaikan laporan sederhana ini. Adapun maksud dan tujuan dari penulisan laporan ini untuk mengetahui seberapa besar kita mendalami materi praktikum yang telah diajarkan oleh ASLAB masing – masing dan merupakan tugas akhir dari praktikum kami di semester IV ini. Namun dalam penyusunannya, kami menyadari masih banyak kekurangan dan jauh dari taraf kesempurnaan. Selain itu banyak sekali kesulitan yang kami alami dalam membuat laporan ini terutama dalam membuat program dan penyatuan materi didalam modul. Oleh karena itu, dengan rendah hati kami membutuhkan saran dan kritik yang sifatnya membangun dari semua pembaca. Dalam kesempatan ini perkenankanlah kami menyampaikan rasa terima kasih kepada Dosen, Aslab Praktikum dan Aslab Penanggung Jawab Mata Kuliah Kecerdasan Buatan yaitu :

1. Masnawati, MT
2. Joan Angelina W, S.Kom, M.Kom
3. Niken Novirasari
4. Anisa N. Afiah
5. M.Hilmy Ady S.

dalam pelaksanaan pembuatan Program Final Project ini. Satu harapan yang kami inginkan semoga laporan ini dapat berguna bagi pembaca.

Samarinda, 1 Mei 2017.

Kelompok 6.

DAFTAR ISI

Halaman Judul	i
Kata Pengantar	ii
Daftar Isi	iii
Daftar Tabel	v
Daftar Gambar	vi
BAB I Pendahuluan	1
1.1. Latar belakang.....	1
1.2. Rumusan masalah	2
1.3. Batasan masalah.....	3
1.4. Tujuan dan Manfaat	3
BAB II Landasan Teori	4
2.1 Fungsi Visual Prolog.....	4
2.1.1. Kecerdasan Buatan.....	4
2.1.2. Visual Prolog.....	4
2.1.3. <i>Section</i> dalam Program	5
2.1.3.1. Lacakbalik (<i>Backtracking</i>)	5
2.1.3.2. Data Object Sederhana	5
2.1.3.3. <i>Section</i> Dasar Visual Prolog	6
2.1.3.4. <i>Section Facts</i>	7
2.1.3.5. List	9

	2.2. Teori Keilmuan Gugur Praktikum	9
BAB	III Metodologi	10
	3.1 Alur Pembuatan Sistem.....	10
BAB	IV Hasil dan Pembahasan	12
	4.1 Tabel Kebenaran	12
	4.2 Analisis Aplikasi.....	12
BAB	V Penutup	19
	5.1 Kesimpulan	19
	5.2 Saran	19
	Daftar Pustaka	20
	Lampiran	21
	1 Source Code	
	2 Kartu Konsul	

DAFTAR TABEL

Tabel 1.1.1. Tabel Kebenaran	12
------------------------------------	----

DAFTAR GAMBAR

Gambar 3.1.1. Alur Pembuatan Sistem.....	11
Gambar 1.2.1. Tampilan Awal Program	12
Gambar 1.2.2. Tampilan Masukan Pilihan yang program akan eksekusi.....	13
Gambar 1.2.3. Tampilan pengisian nama, nim, kelas dan matakuliah.....	13
Gambar 1.2.4. Tampilan Data Mahasiswa yang dimasukkan	14
Gambar 1.2.5. Tampilan Pertanyaan.....	14
Gambar 1.2.6. Tampilan Lanjutan Pertanyaan.....	14
Gambar 1.2.7. Tampilan Lanjutan Pertanyaan.....	15
Gambar 1.2.8. Tampilan Lanjutan Pertanyaan.....	15
Gambar 1.2.9. Tampilan Lanjutan Pertanyaan.....	15
Gambar 1.2.10. Tampilan seluruh data mahasiswa yang gugur	16
Gambar 1.2.11. Tampilan nama matakuliah praktikum.....	17
Gambar 1.2.12. Tampilan data yang gugur per praktikum	17
Gambar 1.2.13. Tampilan kembali ke menu	18
Gambar 1.2.14. Tampilan Info	18
Gambar 1.2.15. Tampilan Info	18

BAB I

PENDAHULUAN

1.1. Latar Belakang

Praktikum adalah subsistem dari perkuliahan yang merupakan kegiatan terstruktur dan terjadwal yang memberi kesempatan kepada mahasiswa untuk mendapatkan pengalaman yang nyata dalam rangka meningkatkan pemahaman mahasiswa tentang teori atau agar mahasiswa menguasai keterampilan tertentu yang berkaitan dengan suatu pengetahuan atau suatu mata kuliah.

Dalam merancang sistem pembelajaran, menyusun kurikulum atau dosen dapat memasukan kegiatan praktikum/latihan/responsi dalam perencanaannya jika ada dukungan kuat bahwa materi mata kuliah hanya dapat dipahami kalau disertai praktikum/responsi/latihan.

Suatu mata kuliah dapat semata-mata berupa praktikum. Hal ini dapat terjadi karena dua alasan. Pertama, mata kuliah itu merupakan mata kuliah yang selain diperlukan oleh jurusan yang bersangkutan, juga diperlukan oleh jurusan lain (mata kuliah layanan) yang tidak memerlukan praktikum. Kedua, praktikum itu sendiri mempunyai bobot kredit yang cukup banyak sehingga memerlukan penanganan tersendiri walaupun materinya tetap tidak terpisah dari mata kuliah yang menyelenggarakan tatap muka teorinya saja.

Kata praktikum berasal dari kata *practicu* / *pratique* (Prancis), *practicus* (Latin), atau *praktikos* (Yunani) yang secara harfiah berarti “aktif” atau *prattein* / *prassein* (Yunani) yang berarti “ mengerjakan”. Dalam bahasa Inggris, praktikum bermakna sama dengan *excercise* (*exercice* Prancis), *exercitium* / *execere* (Latin) yang secara harfiah berarti “tetap aktif/sibuk” yang juga bermakna sama dengan “latihan” atau “responsi”.

Responsi (*responsum* / *responsio* [Latin], jawaban) merupakan istilah untuk kegiatan tanya/jawab yang umumnya dipakai dalam bidang matematika dan statistika untuk meningkatkan pemahaman mahasiswa terhadap teori. Dalam sistem kredit semester (SKS) yang berlaku di

Indonesia, mata kuliah yang disertai kegiatan praktikum mempunyai simbol ABC IJK (X [Y-Z])).

1. ABC digunakan sebagai kode jurusan yang menawarkan mata kuliah itu.
2. X menyimbolkan tingkat keilmuan, Y dan Z adalah nomer mata kuliah itu di jurusan yang bersangkutan.
3. X merupakan bobot keseluruhan sks untuk teori.
4. Y merupakan bobot sks untuk teori.
5. Z adalah bobot sks untuk praktikum.

Satu sks praktek/respons/latihan memerlukan masing-masing 2--3 jam per minggu per semester untuk pelaksanaan tatap muka, perencanaan dan penilaian laporan praktikum oleh dosen, penyusunan laporan oleh mahasiswa, pengembangan materi oleh dosen, dan belajar mandiri yang dilakukan mahasiswa. Dengan demikian, jika setiap praktikum memerlukan waktu 4 jam, maka per semester hanya diperlukan 8 kali tatap muka. Jika setiap semester pelaksanaan tatap muka praktikum kurang dari 16 jam, mata kuliah tersebut tidak diselenggarakan secara baik dan mahasiswa tidak memenuhi kredit seperti yang seharusnya tertulis dalam kurikulum.

1.2. Rumusan Masalah

Berdasarkan latar belakang di atas maka diperoleh identifikasi masalah sebagai berikut:

- a. Program ini hanya dikhususkan untuk aslab angkatan TI 2015.
- b. Program ini hanya melihat data praktikan yang gugur praktikum untuk angkatan TI 2015.
- c. Program ini hanya melihat data praktikan inhal dan insus praktikan untuk angkatan TI 2015.

1.3. Batasan Masalah

Untuk menghindari agar tidak menyimpang dari tujuan utama, batasan masalah ini adalah sebagai berikut :

1. Dibuat menggunakan *software* Visual Prolog
2. Program ini hanya menentukana inhal,insus dan gugur berdasarkan kehadiran.
3. Program ini menampilkan nama-nama mahasiswa yang gugur praktikum TI angkatan 2015.
4. Program ini tidak bisa menghapus data yang sudah di masukan.
5. Program ini hanya bisa menginputkan nim dengan tipe data integer.

1.4. Tujuan Dan Manfaat

- a. Bisa mengetahui nama mahasiswa yang gugur praktikum angkatan 2015 dengan matakuliah yang diambil.
- b. Bisa mengetahui secara spesifik nama-nama mahasiswa tiap praktikum.
- c. Mempermudahkan aslab untuk melihat daftar yang gugur praktikum.

BAB II

LANDASAN TEORI

2.1. Fungsi Visual Prolog

2.1.1. Kecerdasan Buatan

Kecerdasan Buatan atau kecerdasan yang ditambahkan kepada suatu sistem yang bisa diatur dalam konteks ilmiah atau Intelegensi Artifisial (bahasa Inggris: *Artificial Intelligence* atau hanya disingkat *AI*) didefinisikan sebagai kecerdasan entitas ilmiah. Sistem seperti ini umumnya dianggap komputer. Kecerdasan diciptakan dan dimasukkan ke dalam suatu mesin (komputer) agar dapat melakukan pekerjaan seperti yang dapat dilakukan manusia. Beberapa macam bidang yang menggunakan kecerdasan buatan antara lain sistem pakar, permainan komputer (*games*), logika fuzzy, jaringan syaraf tiruan dan robotika.

2.1.2. Visual Prolog

Prolog menggunakan teknik pencarian yang disebut heuristik (heuristic) dengan menggunakan pohon logika. prolog banyak dikembangkan di *University of Melbourne* oleh *Lee Naish* dan *John Loyd*. Prolog dalam perkembangannya telah dikombinasikan dengan berbagai bahasa pemrograman terutama functional programming. Bahasa prolog ini secara intensif digunakan pada proyek komputer generasi ke-5 di Jepang. Walaupun demikian sebetulnya penggunaannya tidak terbatas untuk AI saja.

Prolog adalah bahasa pemrograman logika atau di sebut juga sebagai bahasa non-procedural. Namanya diambil dari bahasa Perancis *programmation en logique* (Pemrograman logika). Bahasa ini diciptakan oleh *Alain Colmerauer* dan *Robert Kowalski* sekitar tahun 1972 dalam upaya untuk menciptakan suatu bahasa pemrograman yang

memungkinkan pernyataan logika alih-alih rangkaian perintah untuk dijalankan komputer. Berbeda dengan bahasa pemrograman yang lain, yang menggunakan algoritma konvensional sebagai teknik pencariannya seperti pada *Delphi*, *Pascal*, *BASIC*, *COBOL* dan bahasa pemrograman yang sejenisnya, maka prolog menggunakan teknik pencarian yang disebut *heuristik* (*heuristic*) dengan menggunakan pohon logika.

2.1.3. Section dalam Program

Konsep yang digunakan untuk menyusun program dan penerapan visual prolog ini adalah Dasar-dasar Prolog, Data *Object* Sederhana, Lacakbalik, dan *Section Facts*.

2.1.3.1 Lacakbalik (*Backtracking*)

Pada waktu menyelesaikan masalah, seringkali, seseorang harus menelusuri suatu jalur untuk mendapatkan konklusi yang logis. Jika konklusi ini tidak memberikan jawaban yang dicari, orang tersebut harus memilih jalur yang lain. Perhatikan permainan *maze* berikut. Untuk mencari jalan keluar dari *maze*, seseorang harus selalu mencoba jalur sebelah kiri terlebih dahulu pada setiap percabangan hingga menemukan jalan buntu. Ketika menemukan jalan buntu maka orang tersebut harus kembali ke percabangan terakhir (*back-up*) untuk mencoba lagi (*try again*) ke jalur kanan dan jika menemukan percabangan lagi maka tetap harus mencoba jalur kiri terlebih dahulu. Jalur kanan hanya akan sekali-sekali dipilih. Dengan begitu orang tersebut akan bisa keluar dari *maze*, dan memenangkan permainan.

2.1.3.2 Data *Object* Sederhana

Data *object* sederhana terdiri dari 2 yaitu variabel atau konstanta. Konstanta yang dimaksud tidak sama dengan konstanta simbolis yang ditulis di *section constants* pada bagian program. Yang

dimaksud dengan konstanta di sini adalah apapun yang diidentifikasi sebagai sebuah *object* bukan *subject* yang nilainya bisa bervariasi, seperti sebuah karakter (*char*), angka (*integer* atau *real*) atau sebuah atom (simbol atau string).

2.1.3.3 Section Dasar Visual Prolog

Secara umum, program Visual Prolog terdiri dari empat *section* dasar, yaitu *section clauses*, *section predicates*, *section domains*, dan terakhir *section goal*. Berikut akan dijelaskan secara singkat masing-masing *section* tersebut.

a. Section Clauses

Section clauses merupakan *section* yang paling penting pada program Visual Prolog. Pada *section* inilah kita meletakkan fakta dan aturan. Ketika mencari jawaban, Visual Prolog akan mencari dari bagian paling atas dari *section clauses*, melihat setiap fakta dan aturan untuk mendapat jawaban benar, hingga ke bagian paling bawah dari *section* ini.

b. Section Predicates

Sebelum mendefinisikan predikat di *section clauses*, maka predikat tersebut harus dideklarasikan terlebih dahulu di *section predicates*. Kalau tidak, Visual Prolog tidak akan mengenal predikat yang kita tuliskan tersebut. Ketika mendeklarasikan suatu predikat, kita memberitahu Visual Prolog domain dari argumen yang dimiliki predikat tersebut.

c. Section Domains

Section domains mempunyai 2 manfaat utama, yaitu pertama, kita dapat memberikan nama yang berarti untuk domain, walaupun secara internal domain tersebut sama tipenya

dengan domain yang telah ada; yang kedua, kita dapat mendeklarasi domain khusus yang digunakan untuk mendeklarasikan struktur data yang tidak didefinisikan oleh standar domain. Dengan mendeklarasikan domain juga dapat mencegah kesalahan logika pada program.

d. *Section Facts*

Program Visual Prolog merupakan suatu koleksi dari sekumpulan fakta dan aturan. Kadang ketika program sedang berjalan, kita ingin meng-*update* (merubah, menambah, atau menghapus) beberapa fakta dari program. Pada kasus ini fakta menjadi suatu database yang dinamis atau database internal, dan fakta tersebut dapat berubah ketika program sedang berjalan. Visual Prolog menyediakan *section* khusus untuk mendeklarasikan fakta di program yang menjadi bagian dari database dinamis, yaitu *section facts*.

e. *Section Goal*

Secara esensial, *section goal* sama dengan *body* dari sebuah aturan (*rule*), yaitu sederetan sub-sub *goal*. Perbedaan antara *section goal* dengan suatu aturan adalah setelah kata kunci *goal* tidak diikuti tanda :- dan Visual Prolog secara otomatis mengeksekusi *goal* ketika program dijalankan.

2.1.3.4 *Section Facts*

Section facts terdiri dari fakta-fakta yang mana fakta-fakta tersebut dapat ditambah dan dihapus secara langsung dari sebuah program pada saat program sedang berjalan (*at run time*). Kita dapat mendeklarasikan sebuah predikat pada *section facts* dan predikat tersebut dapat digunakan sama halnya seperti kalau dideklarasikan pada *section predicates*.

2.1.3.4. Perulangan dan Rekursi

Komputer memiliki bermacam kemampuan yang berguna salah satunya adalah kemampuan melakukan sesuatu berulang-ulang. Prolog dapat melakukan perulangan dalam dua hal yaitu berupa prosedur dan struktur data. Ide dari struktur data repetitif (rekursif) adalah bagaimana menciptakan struktur data yang ukuran (*size*) akhirnya belum diketahui ketika struktur tersebut pertama kali dibuat (*create*).

a. Proses Perulangan

Prolog menyediakan dua jenis perulangan yaitu lacakbalik (mencari jawaban jamak dari satu pertanyaan) dan rekursi (prosedur pemanggilan dirinya sendiri).

b. Rekursi

Cara lain untuk melakukan perulangan adalah melalui rekursi. Prosedur rekursi adalah prosedur yang di dalamnya ada pemanggilan terhadap dirinya sendiri. Prosedur rekursi dapat merekam perkembangannya karena ia melewatkan (*passing*) pencacah, total, dan hasil sementara sebagai argumen dari satu iterasi ke iterasi berikutnya. Berikut ini merupakan contoh program untuk mencari faktorial dari suatu angka.

2.1.3.5 *List*

Pada Prolog, yang dimaksud dengan *list* adalah sebuah *object* yang didalamnya mengandung sejumlah *object* yang lain (jumlahnya dapat berubah-ubah). *List* dalam Bahasa pemrograman lain bisa disamakan dengan tipe data *pointer* (C dan Pascal). Berikut ini cara penulisan *list* pada Prolog.

```
[ 1, 2, 3 ] /* list yang mengandung integer 1, 2 dan 3 */
```

```
[ kucing, anjing, tikus ] /* list yang terdiri dari 3 buah symbol */
```

```
[ "Syarif Musadek", "Yusida Andriani", "Diana Putri" ]
```

```
/* list yang terdiri dari 3 buah string */
```

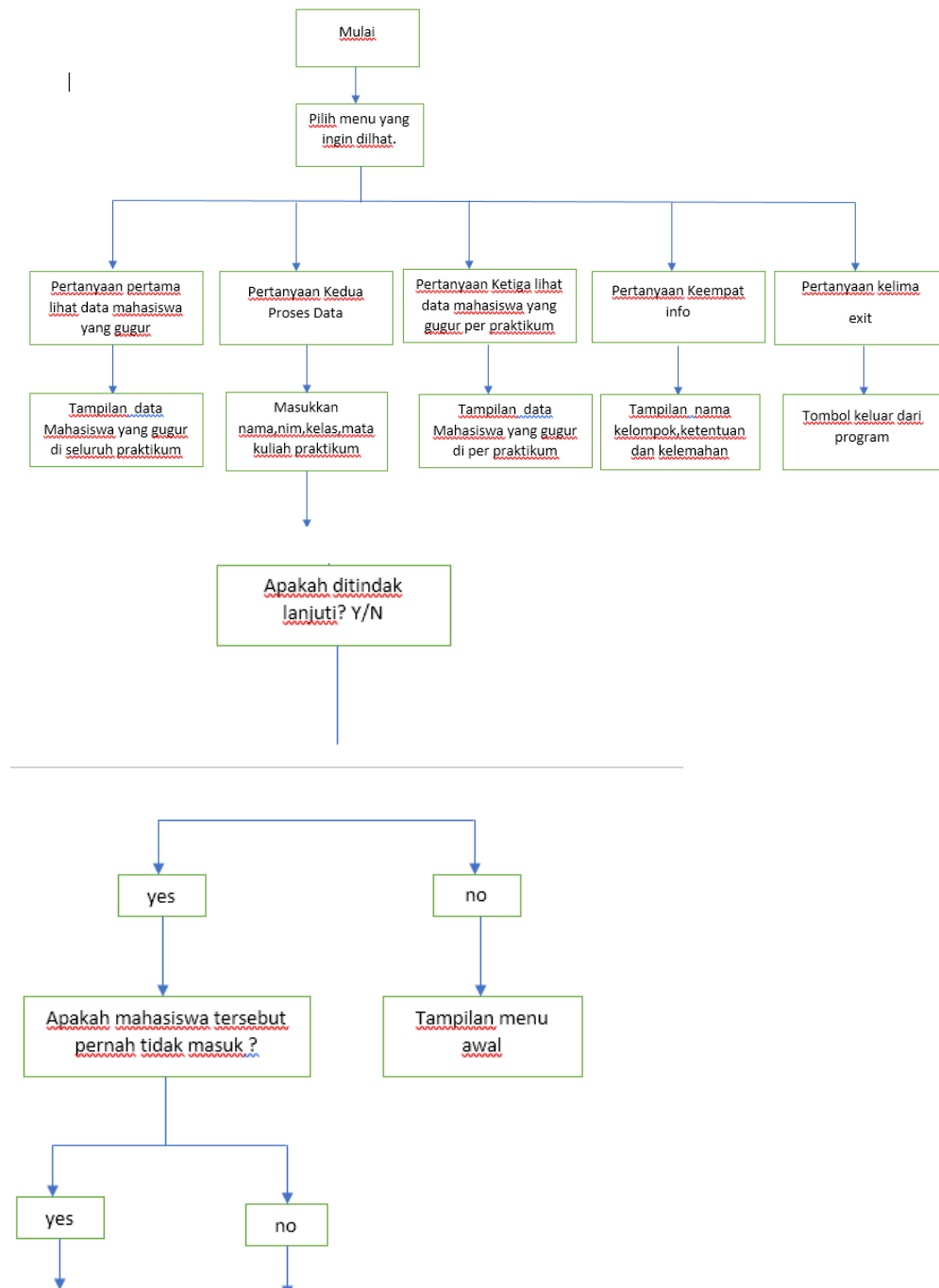
2.2. Teori Keilmuan Gugur Praktikum

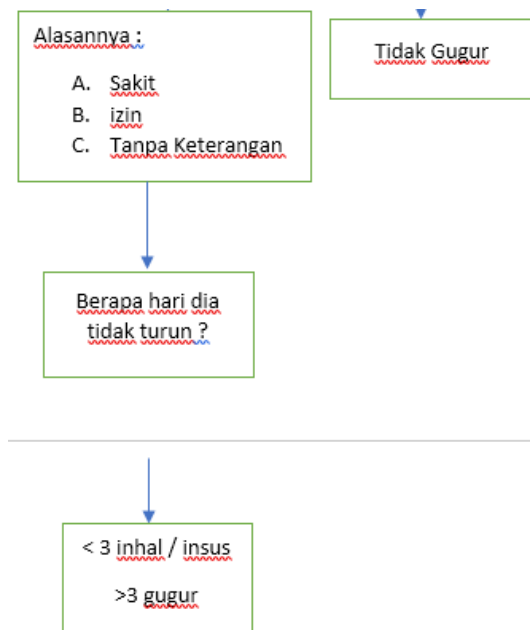
Keilmuan tentang gugur praktikum ini merupakan teori yang dilakukan dengan persyaratan tertentu sesuai dengan praktikum masing – masing yang telah ditentukan. Jika mahasiswa tersebut melanggar aturan yang telah ditetapkan maka mahasiswa tersebut dinyatakan gugur praktikum. Aturan tersebut berisi jika 3 kali tidak menghadiri praktikum tanpa keterangan maka mahasiswa tersebut akan gugur dalam praktikumnya. Jika mahasiswa tersebut tidak hadir tapi memiliki surat keterangan seperti ijin dan sakit maka mahasiswa tersebut tidak bisa dikatakan gugur.

BAB III

METODOLOGI

3.1. Alur Pembuatan Sistem





Gambar 3.1.1. Alur Pembuatan Sistem

Penjelasan : dalam proses pembuatan program data gugur praktikum mahasiswa TI 2015 , pertama kita menampilkan menu awal yang berisi lihat seluruh data mahasiswa yang gugur praktikum yang berisi data-data mahasiswa yang gugur praktikum angkatan 2015. Kedua proses data mahasiswa yang berisi pertanyaan yang menyatakan yang mana pertanyaan tersebut menyatakan mahasiswa inhal,insus atau gugur praktikum, yang ketiga melihat data mahasiswa gugur praktikum, keempat info itu berisi tentang ketentuan,kelemahan dalam program dan yang terakhir keluar dari program tanpa menggunakan ctrl pause break.

BAB IV

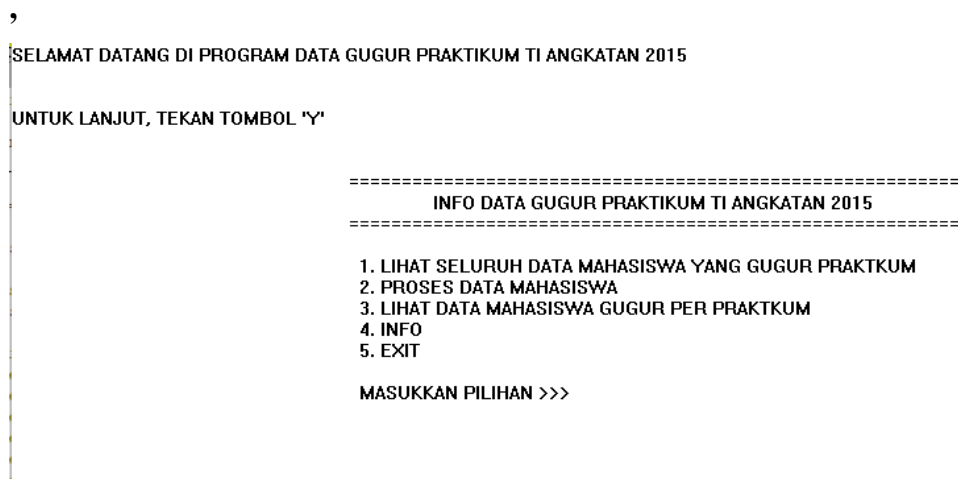
HASIL DAN PEMBAHASAN

1.1. Tabel Kebenaran

No	Apakah mahasiswa tersebut pernah masuk	Alasannya apa	izin	sakit	alpa	Berapa kali.	keterangan
1	✓	✓	✓			2	inhal
2	✓	✓		✓		3	insus
3	✓	✓			✓	3	gugur

Tabel 1.1.1. Tabel kebenaran

1.2. Analisis Aplikasi



Gambar 1.2.1. Tampilan awal program

Dimana saat program dijalankan maka akan muncul tampilan seperti diatas yang berisi menu yang ada diprogram.

MASUKKAN PILIHAN >>> 2

```
=====
Apakah Sebelumnya Mahasiswa Tersebut Pernah Tidak Masuk ? [Y/N]
>>>> |
```

Gambar 1.2.2. Tampilan Masukan pilihan yang program akan eksekusi

Jadi menampilkan menu maka akan memilih yang anda inginkan seperti contoh diatas kita memilih pilihan yang kedua yang berisi proses data mahasiswa.

```
=====
Apakah Sebelumnya Mahasiswa Tersebut Pernah Tidak Masuk ? [Y/N]
>>>> y

=====
+  PROSES DATA MAHASISWA  +
=====
```

Masukkan Data Mahasiswa

Nama : edwin
NIM : 1515015022
Kelas : A
Matakuliah Praktikum : kecerdasan buatan|

Gambar 1.2.3. Tampilan pengisian nama,nim,kelas dan matakuliah

Setelah memilih menu yang diinginkan maka kita akan mengisi data mahasiswa yang gugur di matakuliah kecerdasan buatan.

```

=====
Data Mahasiswa Yang Dimasukkan
=====

Nama           : edwin
NIM            : 1515015022
Kelas         : A
Matakuliah Praktikum : kecerdasan buatan

Tekan Y untuk melanjutkan
>>>> |

```

Gambar 1.2.4. Tampilan Data mahasiswa yang dimasukkan

Menampilkan ulang data mahasiswa yang sudah diinputkan dan memberikan pilihan untuk melanjutkan dengan menekan “Y”.

Alasannya Mahasiswa Tersebut Tidak Masuk :

```

A. Sakit
B. Izin
C. Tanpa Keterangan
>>> a

```

```

=====
Alasan yang di pilih adalah SAKIT
Berapa hari dia Sakit : |

```

Gambar 1.2.5. Tampilan Pertanyaan

Jika ingin melanjutkan maka akan memilih Y maka akan mengeluarkan pilihan sakit,izin dan tanpa keterangan, jika memilih sakit (a).tekan enter

```

=====
Alasan yang di pilih adalah SAKIT
Berapa hari dia Sakit : 1

=====
APABILA MAHASISWA TERSEBUT SAKIT / IZIN KURANG DARI 3 KALI MAKA DIA INHAL
—INHAL—
=====

```

Gambar 1.2.6. Tampilan Lanjutan Pertanyaan

Selanjutnya akan mengeluarkan pertanyaan alasan yang dipilih adalah sakit, berapa hari dia sakit : 1 hari, maka akan mengeluarkan apabila mahasiswa tersebut sakit / izin kurang dari 3 kali maka dia inhal.

```
=====
Alasan yang di pilih adalah IZIN
Berapa hari dia IZIN : 4

=====
APABILA MAHASISWA TERSEBUT SAKIT / IZIN LEBIH DARI 3 KALI MAKA DIA INSUS
—INSUS—
=====
```

Gambar 1.2.7. Tampilan Lanjutan Pertanyaan

Apabila mahasiswa tersebut sakit / izin lebih dari pada 3 kali maka mahasiswa tersebut insus.

```
=====
Alasan yang dipilih adalah TANPA KETERANGAN
Berapa hari dia TANPA KETERANGAN : 4

=====
APABILA MAHASISWA TERSEBUT TANPA KETERANGAN LEBIH DARI 3 KALI MAKA DIA GUGUR
—GUGUR—
=====
```

Gambar 1.2.8. Tampilan Lanjutan Pertanyaan

Apabila mahasiswa tersebut tanpa keterangan lebih dari 3 kali maka mahasiswa tersebut gugur.

```
=====
Alasan yang dipilih adalah TANPA KETERANGAN
Berapa hari dia TANPA KETERANGAN : 2

=====
APABILA MAHASISWA TERSEBUT TANPA KETERANGAN KURANG DARI 3 KALI MAKA DIA INSUS
—INSUS—
=====
```

Gambar 1.2.9. Tampilan Lanjutan Pertanyaan

Apabila mahasiswa tersebut tanpa keterangan kurang dari 3 kali maka mahasiswa tersebut di nyatakan insus.

MASUKKAN PILIHAN >>> 1

DATA SELURUH MAHASISWA TI 2015

Nama : edwin
NIM : 1515015022
Kelas : A
Matakuliah Praktikum : kecerdasan buatan

Nama : sri
NIM : 1515015021
Kelas : A
Matakuliah Praktikum : smbd

Nama : ani
NIM : 1515015008
Kelas : A
Matakuliah Praktikum : pemrograman visual

Nama : beti
NIM : 1515015004
Kelas : A
Matakuliah Praktikum : smbd

Gambar 1.2.10. Tampilan seluruh data mahasiswa yang gugur

Jika dipilih pilihan 1 maka akan menampilkan seluruh data mahasiswa yang gugur praktikum.

MASUKKAN PILIHAN >>> 3

```
=====
DATA MAHASISWA TI 2015 YANG INHAL/INSUS/GUGUR PER PRAKTIKUM
=====
```

1. Kecerdasan Buatan
2. Pemrograman Framework
3. Pemrograman Visual
4. Sistem Manajemen Basis Data
5. Kembali ke Menu

Gambar 1.2.11. Tampilan nama matakuliah praktikum.

Selanjutnya jika memilih pilihan yang ketiga untuk mengetahui nama-nama mahasiswa yang gugur per praktikum.

```
=====
DATA MAHASISWA TI 2015 YANG INHAL/INSUS/GUGUR PER PRAKTIKUM
=====
```

1. Kecerdasan Buatan
2. Pemrograman Framework
3. Pemrograman Visual
4. Sistem Manajemen Basis Data
5. Kembali ke Menu

SILAHKAN PILIH >>> 4

```
=====
DATA YANG GUGUR PRAKTIKUM SISTEM MANAJEMEN BASIS DATA
=====
```

Nama	: sri
NIM	: 1515015021
Kelas	: A
Matakuliah Praktikum	: smbd

Nama	: beti
NIM	: 1515015004
Kelas	: A
Matakuliah Praktikum	: smbd

Gambar 1.2.12. Tampilan data yang gugur per praktikum

Jika sudah selesai maka memilih matakuliah per praktikum contoh sistem manajemen basis data maka akan terlihat data mahasiswa yang gugur praktikum.

5. Kembali ke Menu

SILAHKAN PILIH >>> 5

Gambar 1.2.13. Tampilan kembali ke menu

Jika memilih pilihan ke 5 maka akan kembali ke menu awal.

```
MASUKKAN PILIHAN >>> 4

KETENTUAN DALAM PROGRAM
1. Apabila mahasiswa sakit / Izin lebih dari 3 kali maka dia Insus, jika kurang dari 3 kali maka dia kenakan inhal
2. Apabila mahasiswa tidak masuk dengan tanpa tanpa keterangan lebih dari 3 kali maka dia gugur, jika kurang dari 3 kali dia kenakan insus
3. Apabila mahasiswa selalu masuk maka dia tidak gugur

KELEMAHAN PROGRAM

Program ini masih banyak akan kelemahan di antaranya :
1. Jika Tipenya bernilai Integer semisal menginputkan nim, dan angkatan di inputkan selain integer maka program akan berulang ke menu awal
- kecerdasan buatan
- pemrograman framework
- pemrograman visual
- smbd
selain dari itu program tidak bisa terpanggil di pilihan 3. dan jika data yang di inputkan sembarang tetap akan tersimpan dan terpanggil di pilihan 1.
3. Data yang tersimpan tidak bisa menampilkan keterangan apakah dia Insus atau inhal
4. Data yang tersimpan tidak bisa di hapus
DAN MASIH BANYAK KELEMAHAN LAINNYA YANG SULIT DI UNGKAPKAN DENGAN KATA-KATA

KELOMPOK 6

Rahmawati Rina H      1515015008
Sri Marsiah           1515015021
Edwin Ginanjar Firdaus 1515015022
```

Gambar 1.2.14. Tampilan Info

Pilih menu yang ke 4 maka akan menampilkan informasi ketentuan dan kelemahan pada program.

```
=====
INFO DATA GUGUR PRAKTIKUM TI ANGKATAN 2015
=====

1. LIHAT SELURUH DATA MAHASISWA YANG GUGUR PRAKTIKUM
2. PROSES DATA MAHASISWA
3. LIHAT DATA MAHASISWA GUGUR PER PRAKTIKUM
4. INFO
5. EXIT

MASUKKAN PILIHAN >>> 5|
```

Gambar 1.2.15. Tampilan Info

Memilih pilihan 5 untuk keluar.

BAB V

PENUTUP

5.1. Kesimpulan

Jadi dalam program ini kita dapat mengetahui semua nama mahasiswa pada angkatan 2015 yang gugur dalam semua praktikum. Program ini memudahkan aslab untuk melihat nama mahasiswa angkatan 2015 secara spesifikasi pada tiap – tiap praktikum dan juga menghemat kertas.

5.2. Saran

Tak lupa juga penulis mencantumkan saran karena di dalam laporan ini masih terdapat banyak kekurangan oleh sebab itu penulis masih sangat membutuhkan saran bagi para pembaca. Diharapkan juga agar program ini dapat lebih dikembangkan untuk ke depannya dan bisa digunakan bagaimana semestinya.

DAFTAR PUSTAKA

<http://heru-id.blogspot.co.id/2010/01/arti-dan-tujuan-praktikum.html>

LAMPIRAN

1. Source Code

Domains

```
listnama_kel = nama_kel*  
nama_kel = symbol  
nama = symbol  
nim = integer  
mk = symbol  
kelas = symbol
```

facts

```
mahasiswa(string,integer,string,symbol)
```


predicates



```
nondeterm praktikan(nama_kel, listnama_kel)  
nondeterm menu  
nondeterm mulai  
nondeterm menux(char)  
nondeterm pilih(symbol)  
nondeterm hasil(symbol)  
nondeterm hasil_alpa(symbol)  
nondeterm hasil1  
nondeterm pilih1  
nondeterm pilih2  
nondeterm pilih3  
nondeterm pilih4  
nondeterm pilih5  
nondeterm gol(symbol)  
nondeterm go2(symbol)  
nondeterm go3(symbol)  
nondeterm go4(symbol)  
nondeterm go5(symbol)  
nondeterm go6(symbol)  
nondeterm go7(symbol)  
nondeterm lanjut1  
nondeterm lanjut2  
nondeterm lanjut3  
nondeterm lanjut4
```



```
go6(A):- A>"3",hasil_alpa(A).  
go6(A):- A<="3",hasil_alpa(A).  
go6(_):- lanjut5.  
  
pilih3:-  
write("\n\t\t\t\t=====\\n"),  
write("\\t\t\t\tDATA MAHASISWA II 2015 YANG INHAL/INSUS/GUGUR PER PRAKTIKUM"),  
write("\\n\t\t\t\t=====\\n\\n"),  
write("\\t\t\t\t1. Kecerdasan Buatan\\n"),  
write("\\t\t\t\t2. Pemrograman Framework\\n"),  
write("\\t\t\t\t3. Pemrograman Visual\\n"),  
write("\\t\t\t\t4. Sistem Manajemen Basis Data\\n"),  
write("\\t\t\t\t5. Kembali ke Menu"),nl,nl,  
write("\\t\t\t\tSILAHKAN PILIH >>> "),  
readln(A),go7(A).  
  
go7(A):- A="1",kb.  
go7(A):- A="2",fw.  
go7(A):- A="3",visual.  
go7(A):- A="4",smdb.  
go7(A):- A="5",menu.  
go7(_):- pilih3.  
  
kb:-  
write("\\n\t\t\t\t=====\\n"),  
write("\\t\t\t\tDATA YANG GUGUR PRAKTIKUM KECERDASAN BUATAN"),nl,  
write("\\n\t\t\t\t-----\\n"),  
mahasiswa>Nama,Nim,Kelas,"kecerdasan buatan"),  
write("\\t\t\t\tNama \\t\t : ",Nama),  
write("\\n\t\t\t\tNIM \\t\t : ",Nim),  
write("\\n\t\t\t\tKelas \\t\t : ",Kelas),  
write("\\n\t\t\t\tMatakuliah Praktikum : ","kecerdasan buatan"),nl,nl,  
fail.  
  
fw:-  
write("\\n\t\t\t\t=====\\n"),  
write("\\t\t\t\tDATA YANG GUGUR PRAKTIKUM PEMROGRAMAN FRAMEWORK"),nl,  
  
write("\\n\t\t\t\t-----\\n"),  
mahasiswa>Nama,Nim,Kelas,"pemrograman framework"),  
write("\\t\t\t\tNama \\t\t : ",Nama),  
write("\\n\t\t\t\tNIM \\t\t : ",Nim),  
write("\\n\t\t\t\tKelas \\t\t : ",Kelas),  
write("\\n\t\t\t\tMatakuliah Praktikum : ","pemrograman framework"),nl,nl,  
fail.  
  
visual:-  
write("\\n\t\t\t\t=====\\n"),  
write("\\t\t\t\tDATA YANG GUGUR PRAKTIKUM PEMROGRAMAN VISUAL"),nl,  
write("\\n\t\t\t\t-----\\n"),  
mahasiswa>Nama,Nim,Kelas,"pemrograman visual"),nl,  
write("\\t\t\t\tNama \\t\t : ",Nama),  
write("\\n\t\t\t\tNIM \\t\t : ",Nim),  
write("\\n\t\t\t\tKelas \\t\t : ",Kelas),  
write("\\n\t\t\t\tMatakuliah Praktikum : ","pemrograman visual"),nl,nl,  
fail.  
  
smdb:-  
write("\\n\t\t\t\t=====\\n"),  
write("\\t\t\t\tDATA YANG GUGUR PRAKTIKUM SISTEM MANAJEMEN BASIS DATA"),nl,  
write("\\n\t\t\t\t-----\\n"),  
mahasiswa>Nama,Nim,Kelas,"smdb"),nl,  
write("\\t\t\t\tNama \\t\t : ",Nama),  
write("\\n\t\t\t\tNIM \\t\t : ",Nim),  
write("\\n\t\t\t\tKelas \\t\t : ",Kelas),  
write("\\n\t\t\t\tMatakuliah Praktikum : ","smdb"),nl,nl,  
fail.  
  
pillih4:-  
write("\\nKETENTUAN DALAM PROGRAM"),  
write("\\n1. Apabila mahasiswa sakit / Izin lebih dari 3 kali maka dia Insus, jika kurang dari 3 kali maka dia kenakan inhal"),  
write("\\n2. Apabila mahasiswa tidak masuk dengan tanpa keterangan lebih dari 3 kali maka dia gugur, jika kurang dari 3 kali dia kenakan insus"),  
write("\\n3. Apabila mahasiswa selalu masuk maka dia tidak gugur").nl,nl.
```


2. Kartu Konsul

<p>PERATURAN</p> <ol style="list-style-type: none"> 1. Praktikan wajib konsultasi Programing Project dengan Asisten yang telah ditentukan. 2. Praktikan yang tidak konsultasi DILARANG mengikuti ujian. 3. Konsultasi I, membahas tentang konsep PP yang akan dibuat. 4. Konsultasi II, membahas tentang hal yang berkaitan dengan PP yang sedang dikerjakan. <p>NB. Kartu Konsultasi ini diprint menggunakan kertas Cover berwarna sesuai dengan warna pada cover modul praktikum</p>	<p style="text-align: center;">DATA GUGUR PRAKTIKUM MAHASISWA TI 2015 KECERDASAN BUATAN</p> <p style="text-align: center;">KELOMPOK : 06/A/2015</p> <p>Rahmawati Rina H / 1515015008 Sri Marsiah / 1515015021 Edwin Ginanjar Firdaus / 1515015022</p> <div style="text-align: center;">  </div> <p style="text-align: center;">LABORATORIUM BAHASA PEMROGRAMAN FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI 2017</p>
--	---

<p>Tanggal Konsultasi : 29-4-2017</p> <p>Uraian : "Data Gugur Praktikum Mahasiswa TI 2015" Berkas dalam program menampilkan persyaratan mahasiswa yang 2 kali tidak turun maka akan gugur praktikum. Fakta (tidak masuk), (tidak-tan), (tan), (gugur), (alpa), (inhal)</p> <p>- Pertanyaan</p> <ol style="list-style-type: none"> 1. Apakah pernah tidak masuk? (Y/N) jika Y 2. Apakah dia pernah tan? (Y/N) 3. Apakah dia masuk? (Y/N) 4. Apakah dia alpa? (Y/N) 5. Apakah dia pernah berapa kali dia tidak hadir? <p>- Jawaban</p> <ol style="list-style-type: none"> 1. Gugur 2. Inhal 3. Masuk <p>6. Apakah dia mengikuti inhal? (Y/N)</p> <p>Tanda Tangan Asisten : </p>	<p>Tanggal Konsultasi : 2-5-2017</p> <p>Uraian : Menampilkan fakta (tidak masuk), (tidak-tan), (tan) dan membuat pertanyaan untuk mencari goals dari program tersebut.</p> <p>Tanda Tangan Asisten : </p>
--	--