

**PENGUKURAN KINERJA AKADEMIK MAHASISWA TEKNIK
INFORMATIKA UIN MALANG MENGGUNAKAN
METODE RULE BASED PENDEKATAN
FORWARD CHAINING**

SKRIPSI

Oleh :
DEDY SATRIONO
NIM. 12650104



**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2019**

**PENGUKURAN KINERJA AKADEMIK MAHASISWA TEKNIK
INFORMATIKA UIN MALANG MENGGUNAKAN
METODE RULE BASED PENDEKATAN
FORWARD CHAINING**

SKRIPSI

**Diajukan kepada :
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)**

**Oleh :
DEDY SATRIONO
NIM. 12650104**

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2019**

HALAMAN PERSETUJUAN

**PENGUKURAN KINERJA AKADEMIK MAHASISWA TEKNIK
INFORMATIKA UIN MALANG MENGGUNAKAN
METODE RULE BASED PENDEKATAN
FORWARD CHAINING**

SKRIPSI

Oleh :
DEDY SATRIONO
NIM. 12650104

Telah diperiksa dan Disetujui untuk Diuji
Tanggal: 14 Juni 2019

Dosen Pembimbing I

Dosen Pembimbing II


Dr. Cahyo Crysdian, M.Cs
NIP. 19740424 200901 1 008


Fatchurrochman, M.Kom
NIP. 19700731 200501 1 002

Mengetahui,
Ketua Jurusan Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang




Dr. Cahyo Crysdian, M.Cs
NIP. 19740424 200901 1 008

HALAMAN PENGESAHAN

**PENGUKURAN KINERJA AKADEMIK MAHASISWA TEKNIK
INFORMATIKA UIN MALANG MENGGUNAKAN
METODE RULE BASED PENDEKATAN
FORWARD CHAINING**

SKRIPSI

Oleh :
DEDY SATRIONO
NIM. 12650104

Telah Dipertahankan di Depan Dewan Penguji Skripsi dan
Dinyatakan Diterima Sebagai Salah Satu Persyaratan untuk
Memperoleh Gelar Sarjana Komputer (S.Kom)
Tanggal : 17 Juni 2019

Susunan Dewan Penguji

Penguji Utama : **Dr. Muhammad Faisal**
NIP. 19740510 200501 1 007


Ketua Penguji : **Ainatul Mardhiyah, M.Cs**
NIDT. 19860330 20160801 2 075

Sekretaris Penguji : **Dr. Cahyo Crysdian, M.Cs**
NIP. 19740424 200901 1 008

Anggota Penguji : **Fatchurrochman, M.Kom**
NIP. 19700731 200501 1 002

Tanda Tangan

()

()

()

()

Mengetahui dan Mengesahkan,
Ketua Jurusan Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang



Dr. Cahyo Crysdian, M.Cs
NIP. 19740424 200901 1 008

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama : Dedy Satriono

NIM : 12650104

Jurusan : Teknik Informatika

Fakultas/ Jurusan : Sains dan Teknologi

Judul Skripsi : Pengukuran Kinerja Akademik Mahasiswa Teknik Informatika UIN Malang Menggunakan Metode Rule Based Pendekatan Forward Chaining

Menyatakan dengan sebenar-benarnya bahwa hasil penelitian saya ini tidak terdapat unsur-unsur penjiplakan karya penelitian atau karya ilmiah yang pernah dilakukan atau dibuat oleh orang lain, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila ternyata hasil penelitian ini terbukti terdapat unsur-unsur penjiplakan, maka saya bersedia untuk mempertanggungjawabkan, serta diproses sesuai peraturan yang berlaku.

Malang, 14 Juni 2019

Yang membuat pernyataan



Dedy Satriono
NIM 12650104

MOTTO

Jangan mengharapkan imbalan atas kebaikanmu.



HALAMAN PERSEMBAHAN

Alhamdulillah, segala puji bagi Allah SWT yang telah memberikan nikmat berlimpah kepada penulis. Shalawat serta salam semoga tetap tercurahkan kepada Nabi Besar Muhammad SAW, yang telah memberi suri tauladan yang luar biasa kepada umatnya.

Pertama, penulis ingin mengucapkan terima kasih yang tak terhingga, kepada kedua orang tua tercinta, Almarhum Suwignyo dan Ibu Ngatini. Terima kasih atas doa, dukungan, kasih sayang, kerja keras sehingga penulis bisa menyelesaikan segala kewajiban akademis ini.

Kedua, penulis ingin mengucapkan terima kasih kepada dosen-dosen yang telah membimbing penulis menempuh berbagai pelajaran, terutama kepada Bapak Dr. Cahyo Crysdiyan, M.Cs, dan Fatchurrochman, M.Kom, yang telah memberi masukan-masukan sehingga selesainya skripsi ini.

Ketiga, penulis ingin mengucapkan terima kasih kepada teman-teman Teknik Informatika angkatan 2012 yang telah berbagi informasi apapun atas selesainya skripsi ini.

Keempat, penulis ingin mengucapkan terima kasih kepada teman-teman Kunil Coffee atas segalanya mulai dari tempat singgah, tempat inspirasi, dan masih banyak lagi. Tidak lupa mengucapkan terima kasih kepada Cillu, Archimedio, Agil Diki, Dwiky, Aban, Yosa, Haqqi, Edowongki, Anita, Putri dan teman-teman lainnya yang sudah membantu, memfasilitasi, menemani dalam menyelesaikan skripsi ini.

KATA PENGANTAR

Assalamu'alaikum Wr. Wb.

Segala puji bagi Allah SWT, sholawat serta salam semoga senantiasa ditujukan bagi Nabi Besar Rasulullah SAW, keluarga, para sahabat, dan siapa saja yang meneladani mereka dengan baik hingga hari kiamat.

Dalam menyelesaikan skripsi ini, banyak pihak yang telah memberikan bantuan baik secara moril, nasihat, dan semangat maupun materil. Atas segala bantuan yang telah diberikan, penulis ingin menyampaikan doa dan ucapan terima kasih yang sedalam-dalamnya kepada:

1. Bapak Prof. Dr. Abdul Haris, M.Ag selaku rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Ibu Dr. Sri Harini, M.Si selaku dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Bapak Dr. Cahyo Crysdiyan, M.Cs selaku ketua jurusan teknik informatika sekaligus dosen pembimbing penulis yang telah membimbing dan memberikan motivasi untuk terus berjuang.
4. Bapak Fatchurrochman, M.Kom selaku dosen pembimbing penulis yang telah meluangkan waktu untuk membimbing dan memberikan motivasi untuk terus berjuang.
5. Seluruh Dosen, Laboran, dan Staff Administrasi Jurusan Teknik Informatika UIN Maliki Malang, terima kasih atas segala ilmu dan waktunya.
6. Segenap dosen teknik informatika yang telah mendidik penulis selama masa studi.
7. Seluruh rekan-rekan studi yang tidak dapat disebutkan satu-persatu, terima kasih atas segala kebaikan dan waktunya.

Penulis menyadari bahwa dalam penyusunan Skripsi ini masih terdapat kekurangan dan penulis berharap semoga Skripsi ini bisa memberikan manfaat kepada para pembaca khususnya bagi penulis secara pribadi. *Amin Ya Rabbal Alamin.*

Wassalamu'alaikum Wr. Wb.

Malang, Juni 2019

Penulis

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PERSETUJUAN.....	ii
HALAMAN PENGESAHAN	iii
PERNYATAAN KEASLIAN TULISAN	iv
MOTTO	v
HALAMAN PERSEMBAHAN.....	vi
KATA PENGANTAR.....	vii
DAFTAR ISI.....	viii
DAFTAR GAMBAR.....	xi
DAFTAR TABEL.....	xiii
DAFTAR KODE SUMBER	xiv
ABSTRAK	xv
ABSTRACT	xvi
ملخص.....	xvii
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Identifikasi Kinerja.....	4
1.3 Pernyataan Masalah.....	4
1.4 Tujuan Penelitian.....	4
1.5 Batasan Penelitian.....	4
1.6 Manfaat Penelitian.....	5
BAB II KAJIAN PUSTAKA.....	6
2.1 Kajian Teoritis	7
2.1.1 Expert System (Sistem Pakar)	7
2.1.1.1 Definisi Sistem Pakar	7
2.1.1.2 Proses Sistem Pakar.....	8
2.1.1.3 Arsitektur Sistem Pakar	9
2.1.2 Penalaran Berbasis Pengetahuan (Rule-based Reasoning) Forward Chaining	12
2.2 Skala Prioritas.....	16
BAB III DESAIN DAN PERANCANGAN SISTEM	18

3.1	Desain Sistem	18
3.1.1	Desain Input	18
3.1.2	Desain Proses	18
3.1.3	Desain Output	18
3.2	Rule Based Pendekatan Forward Chaining	18
3.2.1	Langkah Pertama Forward Chaining	20
3.2.2	Langkah Kedua Forward Chaining	20
3.2.3	Langkah Ketiga Forward Chaining	20
3.2.4	Langkah Keempat Forward Chaining	21
3.2.5	Langkah Kelima Forward Chaining	21
3.2.6	Langkah Keenam Forward Chaining	21
3.2.7	Langkah Ketujuh Forward Chaining	22
3.2.8	Langkah Kedelapan Forward Chaining	22
3.2.9	Langkah Kesembilan Forward Chaining	22
3.2.10	Langkah Kesepuluh Forward Chaining	23
3.2.11	Langkah Kesebelas Forward Chaining	23
3.2.12	Langkah Keduabelas Forward Chaining	23
3.3	Pengukuran	24
3.4	Flowchart Proses	25
3.5	Entity Relationship Diagram (ERD)	26
3.6	Desain Database	27
3.6.1	Tabel Mahasiswa	27
3.6.2	Tabel Khs	27
3.6.2.1	Tabel Khs Lulus	28
3.6.2.2	Tabel Khs Tidak Lulus	29
3.6.3	Tabel Pemasaran	29
3.6.3.1	Tabel Pemasaran Mata Kuliah Pilihan 1	30
3.6.3.2	Tabel Pemasaran Mata Kuliah Pilihan 2	31
3.6.3.3	Tabel Lewati Mata Kuliah Pilihan	31
3.6.3.4	Tabel Pemasaran Baru	32
3.6.4	Tabel Konversi	32
3.7	Desain Interface	33

3.7.1	Tampilan Menu Utama	33
3.7.2	Tampilan Dialog Browse	33
3.7.3	Pop-up Konversi Berhasil	34
3.7.4	Pop-up Berhasil Update	34
3.7.5	Pop-up Tidak Ada Update	35
BAB IV	HASIL DAN PEMBAHASAN.....	36
4.1	Pengukuran	36
4.2	User Interface	36
4.2.1	Halaman Utama	37
4.2.2	Halaman Browse Sebagai Input Data	37
4.2.3	Pop-up Konversi	38
4.2.4	Pop-up Cek Update Database	39
4.3	Rule Based Pendekatan Forward Chaining	40
4.3.1	Langkah Pertama Forward Chaining	40
4.3.2	Langkah Kedua Forward Chaining.....	41
4.3.3	Langkah Ketiga Forward Chaining.....	41
4.3.4	Langkah Keempat Forward Chaining.....	42
4.3.5	Langkah Kelima dan Keenam Forward Chaining	42
4.3.6	Langkah Ketujuh dan Kedelapan Forward Chaining	43
4.3.7	Langkah Kesembilan dan Kesepuluh Forward Chaining	43
4.3.8	Langkah Kesebelas Forward Chaining	43
4.3.9	Langkah Keduabelas Forward Chaining.....	44
4.4	Export Excel Sebagai Output	44
4.5	Hasil Uji Coba Aplikasi.....	45
4.6	Hasil Evaluasi dan Pembahasan	47
4.7	Integrasi	48
BAB V	PENUTUP	50
5.1	Kesimpulan.....	50
5.2	Saran	50
	DAFTAR PUSTAKA	52
	LAMPIRAN.....	54

DAFTAR GAMBAR

Gambar 2.1 Arsitektur Sistem Pakar.....	9
Gambar 2.2 Mesin Inferensi.....	11
Gambar 2.3 Forward Chaining.....	13
Gambar 2.4 Urutan pertama forward chaining	14
Gambar 2.5 Urutan kedua forward chaining.....	15
Gambar 2.6 Urutan ketiga forward chaining.....	15
Gambar 2.7 Urutan keempat forward chaining.....	16
Gambar 2.8 Alur penyelesaian Akademik Mahasiswa	16
Gambar 3.1 Langkah Pertama Forward Chaining.....	20
Gambar 3.2 Langkah Kedua Forward Chaining	20
Gambar 3.3 Langkah Ketiga Forward Chaining	20
Gambar 3.4 Langkah Keempat Forward Chaining	21
Gambar 3.5 Langkah Kelima Forward Chaining.....	21
Gambar 3.6 Langkah Keenam Forward Chaining	21
Gambar 3.7 Langkah Ketujuh Forward Chaining.....	22
Gambar 3.8 Langkah Kedelapan Forward Chaining.....	22
Gambar 3.9 Langkah Kesembilan Forward Chaining.....	22
Gambar 3.10 Langkah Kesepuluh Forward Chaining.....	23
Gambar 3.11 Langkah Kesebelas Forward Chaining	23
Gambar 3.12 Langkah Keduabelas Forward Chaining	23
Gambar 3.13 Flowchart proses aplikasi	25
Gambar 3.14 ERD Sistem Pengukuran Kinerja Akademik Mahasiswa UIN Maliki Malang.....	26
Gambar 3.15 Tampilan Menu Utama.....	33
Gambar 3.16 Tampilan pilih file Hasil Studi	34
Gambar 3.17 Tampilan pop-up berhasil konversi	34
Gambar 3.18 Tampilan pop-up berhasil update.....	35
Gambar 3.19 Tampilan Pop-up tidak ada update.....	35
Gambar 4.1 Tampilan halaman utama	37
Gambar 4.2 Tampilan halaman browse	37
Gambar 4.3 Tampilan pop-up konversi berhasil.....	38

Gambar 4.4 Tampilan pop-up konversi gagal karena SQL belum aktif	38
Gambar 4.5 Tampilan pop-up input belum dipilih.....	39
Gambar 4.6 Tampilan pop-up berhasil update database	39
Gambar 4.7 Tampilan pop-up tidak ada update terbaru.....	40



DAFTAR TABEL

Tabel 2.1 Tabel Kriteria Skala Prioritas.....	17
Tabel 3.1 Tabel mahasiswa.....	27
Tabel 3.2 Tabel khs.....	28
Tabel 3.3 Tabel khs lulus.....	28
Tabel 3.4 Tabel khs tidak lulus.....	29
Tabel 3.5 Tabel pemasaran.....	30
Tabel 3.6 Tabel pemasaran mata kuliah pilihan 1.....	30
Tabel 3.7 Tabel pemasaran mata kuliah pilihan 2.....	31
Tabel 3.8 Tabel lewati mata kuliah pilihan.....	32
Tabel 3.9 Tabel pemasaran baru.....	32
Tabel 3.10 Tabel konversi.....	33
Tabel 4.1 Hasil uji coba manual dan menggunakan aplikasi.....	46

DAFTAR KODE SUMBER

Kode Sumber 4.1 Langkah Pertama Forward Chaining	40
Kode Sumber 4.2 Langkah Kedua Forward Chaining	41
Kode Sumber 4.3 Langkah Ketiga Forward Chaining.....	41
Kode Sumber 4.4 Langkah Keempat Forward Chaining.....	42
Kode Sumber 4.5 Looping mata kuliah pilihan per kategori	42
Kode Sumber 4.6 Langkah Kelima dan Keenam forward chaining	42
Kode Sumber 4.7 Langkah Ketujuh dan Kedelapan Forward Chaining.....	43
Kode Sumber 4.8 Langkah Kesembilan dan Kesepuluh Forward Chaining	43
Kode Sumber 4.9 Langkah Kesebelas Forward Chaining	44
Kode Sumber 4.10 Langkah Keduabelas Forward Chaining.....	44
Kode Sumber 4.11 Pengecualian dari tabel skipmpilihan	45

ABSTRAK

Satriono, Dedy. 2019. **Pengukuran Kinerja Akademik Mahasiswa Teknik Informatika UIN Maliki Malang Menggunakan Metode Rule Based Pendekatan Forward Chaining**. Skripsi. Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
Pembimbing : (I) Dr. Cahyo Crysdiyan, M.Cs (II) Fatchurrochman, M.Kom

Kata Kunci : *Rule Based, Forward Chaining, Akademik, Teknik Informatika, Java.*

Banyaknya mata kuliah yang akan ditempuh mahasiswa, ada perubahan kode dan nama mata kuliah yang dipasarkan, penambahan kebijakan yang tidak ada sebelumnya di buku pedoman mahasiswa dapat menimbulkan suatu permasalahan oleh mahasiswa dalam pemrograman mata kuliah pada saat perancangan mata kuliah berlangsung. Di sini penulis bertujuan untuk mengembangkan sebuah aplikasi pengukuran kinerja akademik mahasiswa teknik informatika yang dapat membantu mahasiswa untuk memudahkan dalam perancangan studi berlangsung seperti menampilkan mata kuliah apa saja yang harus ditempuh yang menyesuaikan kode mata kuliah dan nama mata kuliah terbaru yang dipasarkan oleh akademik, menampilkan mata kuliah yang belum lulus, menentukan mahasiswa tersebut untuk menempuh mata kuliah pilihan atau tidak dengan menampilkan atau tidaknya mata kuliah pilihan menyesuaikan hasil studi dari mahasiswa. Pada penelitian ini, untuk pengukuran pada sistem digunakan metode *Forward Chaining*. *Forward Chaining* merupakan metode penalaran berdasarkan fakta yang ada, dimana metode ini dijalankan dengan mengumpulkan fakta-fakta yang ada untuk menarik kesimpulan. Dengan kata lain, prosesnya dimulai dari fakta-fakta yang ada melalui proses penalaran fakta-fakta menuju suatu tujuan. Metode ini menggunakan aturan IF-THEN dimana premis menuju kesimpulan.

Data yang diuji coba sebanyak 41 data mahasiswa jurusan teknik informatika UIN Maliki Malang, dari uji coba yang telah dilakukan menghasilkan tingkat akurasi sebesar 97,56%, dengan tingkat error 2,44%.

ABSTRACT

Satriono, Dedy. 2019. **Measurements of Informatics Engineering Students Academic Performance in UIN Maliki Malang Using Rule Based Methods with Forward Chaining Approach.** Essay. Informatics Engineering Faculty of Science and Technology Universitas Islam Negeri Maulana Malik Ibrahim Malang. Adviser : (I) Dr.Cahyo Crysdiان, M. Cs (II) Fatchurrochman, M. Kom

Keyword: *Rule Based, Forward Chaining, Academic, Informatics Engineering, Java.*

The amount of subjects that student must attend, changes of subjects' name and code available, additions of policy which is absent in student guide book caused an issue among students in the matter of programming subject in study planning period. The goals of this research is to develop an application to measure academic performance of informatics engineering students that will assist to plan their studies such as displaying which subjects they have to attend which also match the code and the name of the subjects recently added by the academic division, shows the subject they have yet to pass, determine if the students should take an optional subjects according to their results and displaying the optional subjects as well. In this research, the measurements is conducted using *Forward Chaining* method. *Forward Chaining* is a logical method using the facts presented, whereas this method is operated by gathering the facts to produce a conclusion. Hence, the process began from the facts through the logical cultivation towards an end goal. This method used the IF-THEN rules where the premise is building towards a conclusion.

Total amount of data tested is 41 data of Informatics Engineering students of UIN Maliki Malang. From the tests conducted, this application produced accuracy level of 97,56% with 2,44% error level.

ملخص

ساتريونو ، ديدي. ٢٠١٩. قياسات الأداء الأكاديمي لطلبة هندسة المعلوماتية في جامعة مالانج مالانج باستخدام الطرق المعتمدة على القواعد مع نهج التسلسل الأمامي. بحثٌ عمليٌّ. قسم هندسة المعلوماتية. كلية العلوم والتكنولوجيا. جامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج. المشرف الأول جحيو كريسيديان الماجستير و المشرف الثاني فتح الرحمن الماجستير.

الكلمات الرئيسية: تستند القاعدة، الأمام تسلسل، أكاديمي، هندسة المعلوماتية، جافا

هناك العديد من الدورات التي سيتم اتخاذها من قبل الطلاب ، وهناك تغيير في الكود واسم الدورات المقدمة ، وإضافة السياسات التي لم تكن موجودة من قبل في دليل الطالب يثير مشكلة للطلاب في ترتيب الدورات التي يتعين اتخاذها عندما يتم تصميم الدورة. يهدف المؤلف هنا إلى تطوير تطبيق لقياس الأداء الأكاديمي لطلاب هندسة المعلوماتية الذين يمكنهم مساعدة الطلاب على تبسيط تصميم الدراسات مثل عرض المواد التي يجب اتخاذها ، وتعديل رمز الدورة التدريبية واسم آخر الدورات التي يقدمها الأكاديميون ، وعرض الدورات بالنسبة لأولئك الذين لم يتخرجوا ، حدد ما إذا كان الطالب سيأخذ دورات اختيارية أم لا من خلال عرض ما إذا كانت المقررات الاختيارية أو عدمها تقوم بضبط نتائج الدراسة من الطلاب. في هذا البحث ، يتم استخدام طريقة التسلسل الأمامي لقياس النظام. التسلسل الأمامي هو طريقة للتفكير بناءً على الحقائق الموجودة ، حيث يتم تنفيذ هذه الطريقة من خلال جمع الحقائق الموجودة لاستخلاص النتائج. بمعنى آخر ، تبدأ العملية من الحقائق الموجودة من خلال عملية التفكير في الحقائق نحو الهدف. تستخدم هذه الطريقة القاعدة IF - THEN حيث تؤدي الفرضية إلى الاستنتاجات.

المبلغ الإجمالي للبيانات التي تم اختبارها هو ٤١ بيانات لطلاب هندسة المعلوماتية في جامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج. من الاختبارات التي أجريت ، أنتج هذا التطبيق مستوى دقة ٩١,٥٦ ٪ مع مستوى خطأ ٢,٤٤ ٪.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dalam menempuh jenjang pendidikan perguruan tinggi, setiap mahasiswa diwajibkan menempuh sejumlah mata kuliah sesuai dengan jurusan yang dipilihnya. Dalam menyelesaikan tiap mata kuliah, mahasiswa diwajibkan mendapatkan nilai minimal C agar dinyatakan lulus program Sarjana. Nilai akhir suatu mata kuliah menjadi penentu tingkat kinerja akademik mahasiswa pada rangkaian program sarjana yang ditempuhnya. Setiap mahasiswa baru mendapatkan Buku Pedoman Mahasiswa yang di dalamnya terdapat pemasaran mata kuliah yang akan ditempuh oleh mahasiswa per semesternya. Akademik menyediakan laman web Sistem Informasi Akademik (SIKAD) untuk membantu mahasiswa melakukan pemrograman mata kuliah, mengetahui transkrip nilai mahasiswa yang telah ditempuh, data diri mahasiswa, dan lain-lain.

Tidak semua mahasiswa dapat menempuh mata kuliah dengan nilai minimal C, yang menyebabkan mahasiswa diwajibkan mengulang kembali mata kuliah tersebut di semester selanjutnya sesuai pemasaran mata kuliah yang tersedia. Permasalahan ini bisa menyebabkan mahasiswa ketika melakukan pemrograman mata kuliah sesuai jadwal perancangan mata kuliah dari akademik perlu mengatur ulang mata kuliah apa saja yang perlu diprogramkan pada saat rancangan studi berlangsung.

Seiring berjalannya waktu, pada semester tertentu Akademik melakukan perubahan kode mata kuliah dan nama mata kuliah dari pemasaran mata kuliah

bahkan menambahkan kebijakan baru untuk menyesuaikan kinerja mahasiswa. Kebijakan ini menimbulkan suatu permasalahan kepada mahasiswa pada saat akan melakukan pemrograman mata kuliah seperti perbedaan kode mata kuliah dari buku pedoman mahasiswa dengan pemasaran yang disediakan oleh akademik, penambahan mata kuliah dari akademik yang membuat mahasiswa menentukan untuk mengambil mata kuliah tersebut atau tidak karena tidak samanya kode mata kuliah dari Buku Pedoman Mahasiswa.

Pada tahun 2015, akademik membuat kebijakan Mata Kuliah Pilihan dan dibagi menjadi 2 kategori, yaitu Mata Kuliah Pilihan 1 dan Mata Kuliah Pilihan 2. Setiap mahasiswa diwajibkan mengambil mata kuliah Pilihan per kategori setidaknya satu mata kuliah agar dapat memenuhi syarat mata kuliah pilihan. Karena kebijakan ini berlaku untuk mahasiswa sebelum tahun 2015 yang masih menempuh perkuliahan setelah tahun 2015, maka diadakan konversi mata kuliah yang dimana jika mahasiswa sudah lulus mata kuliah yang ditentukan, maka tidak perlu mengambil mata kuliah pilihan.

Banyaknya mata kuliah yang akan ditempuh mahasiswa, perubahan kode dan nama mata kuliah yang dipasarkan, penambahan kebijakan yang tidak ada sebelumnya di buku pedoman mahasiswa, dapat menimbulkan suatu permasalahan oleh mahasiswa dalam pemrograman mata kuliah pada saat perancangan mata kuliah berlangsung. Di sini penulis bertujuan untuk mengembangkan sebuah aplikasi pengukuran kinerja akademik mahasiswa teknik informatika yang dapat membantu mahasiswa memudahkan dalam perancangan studi berlangsung seperti menampilkan mata kuliah apa saja yang harus ditempuh menyesuaikan kode mata kuliah dan nama mata kuliah terbaru

yang dipasarkan oleh akademik, menampilkan mata kuliah yang belum lulus, menentukan mahasiswa tersebut untuk menempuh mata kuliah pilihan atau tidak dengan menampilkan atau tidaknya mata kuliah pilihan menyesuaikan hasil studi dari mahasiswa.

Dalam memudahkan urusan orang lain dijelaskan dipotongan surat Al-Maidah ayat 2, Allah SWT bersabda:

وَتَعَاوَنُوا عَلَى الْبِرِّ وَالتَّقْوَىٰ وَلَا تَعَاوَنُوا عَلَى الْإِثْمِ وَالْعُدْوَانِ

...Dan tolong-menolonglah kamu dalam (mengerjakan) kebajikan dan takwa, dan jangan tolong-menolong dalam berbuat dosa dan permusuhan...

Dalam melakukan pengembangan aplikasi ini, ide utama penulis adalah untuk melakukan perbandingan antara dua input pada laman web SIAKAD, yakni nilai mata kuliah dan daftar mata kuliah yang harus ditempuh untuk menghasilkan *output* berupa daftar mata kuliah yang belum ditempuh dan mata kuliah yang belum dinyatakan lulus.

Untuk menciptakan aplikasi pengukuran kinerja akademik ini peneliti menggunakan dasar sistem *rule based reasoning* dengan pendekatan *forward chaining* karena pendekatan ini dianggap paling sederhana untuk mengembangkan aplikasi yang diinginkan.

Metode *rule-based reasoning* menurut Lengyel (2015), adalah sekumpulan pernyataan IF-THEN yang memanfaatkan serangkaian deklarasi, dimana aturan dibuat untuk bagaimana bertindak atas deklarasi tersebut. Sistem *rule-based* seringkali membentuk dasar dari sebuah perangkat lunak yang dapat menyediakan jawaban dari sebuah permasalahan untuk menggantikan seorang ahli. Sistem ini disebut juga dengan sistem pakar.

1.2 Identifikasi Kinerja

Pada penelitian ini terdapat identifikasi kinerja pada sistem informasi yang akan dibangun, diantaranya adalah:

1. Penelitian ini hanya untuk Jurusan Teknik Informatika UIN Maliki Malang saja.
2. Data yang dipakai hanya Transkrip Nilai mahasiswa Teknik Informatika UIN Maliki Malang saja.
3. *File* input yang dapat digunakan hanya berjenis *Excel Worksheet* dengan ekstensi 'xlsx' dan *Excel 97-2003 Worksheet* dengan ekstensi 'xls' yang di dalamnya terdapat tabel transkrip nilai sesuai format laman SIAKAD Akademik UIN Maliki Malang.

1.3 Pernyataan Masalah

Berdasarkan latar belakang yang sudah dijelaskan, maka pernyataan masalah adalah bagaimana tingkat akurasi metode *Rule Based* dalam menentukan aplikasi pengukuran kinerja akademik yang memudahkan mahasiswa dalam mempersingkat waktu pada saat pemrograman mata kuliah.

1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah mengetahui tingkat akurasi, dan membuat aplikasi yang dapat memudahkan mahasiswa menampilkan mata kuliah apa yang belum ditempuh maupun mata kuliah apa yang belum lulus secara cepat.

1.5 Batasan Penelitian

Adapun batasan dalam penelitian ini adalah:

1. Data yang digunakan adalah data akademik mahasiswa Teknik Informatika UIN Maliki Malang.

2. *Platform* yang digunakan pada sistem lingkungan *Java*.
3. Sistem dijalankan pada *platform hardware* dengan spesifikasi minimal Intel Celeron RAM 2GB Storage free spaces 10GB dengan Operasi Sistem Windows 7.

1.6 Manfaat Penelitian

Berdasarkan rumusan masalah di atas maka manfaat dari penelitian ini adalah:

1. Menyaring mata kuliah yang belum diambil maupun belum lulus secara cepat dari SIAKAD
2. Menghasilkan output berupa *file excel* dengan daftar mata kuliah yang perlu diambil untuk menyelesaikan perkuliahan.

BAB II

KAJIAN PUSTAKA

Pemecahan suatu masalah adalah proses yang kompleks. Dalam pemecahan masalah membutuhkan banyak pengaturan. Proses ini mempunyai tahap-tahap dan yang utama adalah proses menentukan skala prioritas. Untuk menentukan skala prioritas suatu perkara dilakukan dengan mengurutkan skala prioritas mengikuti format *input* yang akan ditentukan, dalam menyelesaikan suatu permasalahan diperlukan suatu sistem informasi menggunakan teknologi informasi untuk mengatur sistem penyelesaian suatu perkara dengan menggunakan skala prioritas, pada penelitian ini akan membangun suatu sistem informasi yang berhubungan dengan Akademik Mahasiswa.

Penelitian ini menggunakan beberapa sumber sebagai rujukan penelitian diantaranya yaitu sistem pakar menggunakan metode *forward chaining* pada keuangan dan akutansi yang dapat menghubungkan pengetahuan dan menjalankan dengan sangat baik (Wagner et al, 2002).

Sumber lain yang berhubungan dengan metode penelitian ini yaitu simulasi sistem pakar menggunakan *forward chaining* dan basis aturan untuk mengetahui gangguan-gangguan pada tanaman bawang merah dan cabai seperti hama, gejala dan penyakit. Seperti sumber sebelumnya, metode yang digunakan di penelitian ini menggunakan *forward chaining* berbasis pengetahuan yang menghasilkan diagnosa gangguan-gangguan pada tanaman bawang merah dan cabai seperti hama, gejala dan penyakit (Sasmito dkk., 2011).

Adapun sumber lain yang digunakan pada penelitian kali ini adalah sistem pakar *rule-based* untuk membantu kinerja mahasiswa pada universitas seperti

penjadwalan kegiatan yang berhubungan dengan universitas maupun pengambilan suatu keputusan yang berhubungan dengan mahasiswa di universitas. Penelitian ini telah dibangun dan diuji menggunakan *software Oracle Policy Automation* (Engin dkk., 2014).

Menurut Al-Ajlan (2015), *Rule-based* pendekatan *forward chaining* merupakan metode eksekusi suatu fakta-fakta yang ada untuk menghasilkan suatu kesimpulan berdasarkan fakta-fakta yang dipilih.

2.1 Kajian Teoritis

Berdasarkan sumber-sumber yang digunakan pada penelitian ini, teori yang mendukung penelitian ini menggunakan teori berdasarkan keilmuan yang berhubungan dengan akademik. Teori ini tidak lepas dengan sistem pakar disesuaikan dengan pembahasan penelitian yaitu menentukan skala prioritas dalam suatu perkara. Pendukung penelitian ini yang digunakan sebagai dasar teori seperti definisi / tinjauan singkat, variabel yang sesuai, relasi antar variabel, langkah-langkah, dan sebagainya.

2.1.1 *Expert System* (Sistem Pakar)

Sistem pakar merupakan suatu bagian yang tidak bisa lepas dari penelitian ini, ada beberapa bagian yang menjelaskan sistem pakar seperti definisi sistem pakar, proses sistem pakar, dan arsitektur sistem pakar.

2.1.1.1 Definisi Sistem Pakar

Menurut Ignazio (1991), sistem pakar adalah model dan prosedur yang saling berhubungan, dimana tingkat keahliannya mampu dibandingkan dengan keahlian seorang pakar. Menurut

Durkin (1994), sistem pakar adalah suatu sistem informasi komputer yang dibentuk untuk menampilkan kemampuan menyelesaikan suatu masalah yang dilakukan oleh seorang pakar. Menurut Giarrantano dan Riley (2005), sistem pakar merupakan suatu sistem informasi komputer yang kemampuannya dapat menyamai seorang pakar.

Dapat diambil suatu kesimpulan dari beberapa sumber bahwa sistem pakar merupakan sistem yang di dalamnya terdapat suatu aturan-aturan dari seorang pakar ke dalam sistem informasi.

2.1.1.2 Proses Sistem Pakar

Proses sistem pakar ini seperti halnya konsultan menanyakan fakta-fakta yang dapat menghasilkan suatu kesimpulan, pada sistem pakar mengambil suatu fakta-fakta dari seorang pakar kemudian menjadikannya sebagai suatu aturan (*rule*), aturan-aturan ini dijadikan sebagai dasar spesifik yang digunakan sistem pakar dalam menanyakan fakta-fakta yang dapat menghasilkan suatu kesimpulan. Aturan bisa dari berbagai macam seperti jurnal, buku, aturan pemerintah, *website*, dan pakar yang mempunyai pengetahuan sesuai bidangnya. Sistem pakar yang baik dibuat agar dapat menyelesaikan suatu perkara tertentu dengan menyamakan kinerja dari suatu ahli (Kusumadewi, 2003).

Sistem pakar dapat membuat seseorang mengurangi ketergantungan kepada seorang pakar dan dapat meningkatkan produktivitas, memecahkan suatu masalah yang kompleks, dan memperbaiki kualitas suatu keputusan (Sasmito dkk., 2011).

2.1.1.3 Arsitektur Sistem Pakar

Adapun susunan dari sistem pakar dibagi menjadi dua bagian yaitu lingkungan pengembangan dan lingkungan konsultasi (Turban, 2001). Untuk lingkungan pengembangan pada *expert system* digunakan untuk *input rule* menuju ke lingkungan *expert system*, sedangkan untuk lingkungan konsultasi digunakan untuk *user* yang bukan pakar untuk memperoleh pengetahuan pakar. Bagian-bagian pada *expert system* dibagi menjadi dua bagian, tertera pada Gambar 2.1 sebagai berikut:



Gambar 2.1 Arsitektur Sistem Pakar (Turban dan Aronson, 2001).

Arsitektur sistem pakar dibagi menjadi bagian-bagian penting yaitu:

1. Antarmuka Pengguna (*User Interface*)

Antarmuka Pengguna memudahkan pengguna *expert system* untuk melakukan interaksi antara program dan *user* yang melakukan *input* berupa interaksi dan menghasilkan *output* kepada *user* juga (McLeod, 1995).

2. Basis Pengetahuan (*Knowledge Based*)

Basis Pengetahuan terdiri dari macam-macam pengetahuan seperti teori, fakta, prosedur, pemikiran, interaksi satu dengan lainnya, atau informasi tentang pengetahuan pendidikan maupun pengalaman dari pakar yang dimasukkan ke dalam sistem informasi (Sasmito dkk., 2011). *Rule-Based Reasoning* adalah salah satu pendekatan basis pengetahuan yang digunakan pada penelitian ini, *Rule-Based Reasoning* pada basis pengetahuan mengambil dari fakta dan aturan. Fakta dan aturan ini ditunjukkan sebagai IF-THEN pada bahasa pemrograman di sistem informasi yang akan dibentuk.

3. *Knowledge Acquisition*

Knowledge acquisition merupakan transformasi keahlian/pendapat, proses adopsi dan akumulasi dalam mengatasi masalah yang ada dari pengetahuan dimasukkan ke dalam sistem informasi yang akan

dibangun. *Knowledge acquisition* dibagi menjadi tiga metode utama yaitu: analisis protokol, observasi pada pekerjaan pakar, dan wawancara.

4. *Inference Engine*

Inference engine adalah sistem informasi yang memberikan metodologi penalaran tentang informasi yang ada di basis pengetahuan dan dari *workplace* untuk mendapatkan suatu kesimpulan. *Inference engine* merupakan inti dari *expert system* yang berfungsi sebagai pemandu prosesnya penalaran menggunakan sumber dari basis pengetahuan yang telah di-*input* sebelumnya. Adapun bentuk *Inference engine* dapat dilihat pada Gambar 2.2.



Gambar 2.2 Mesin Inferensi (Sasmito dkk., 2011).

5. *Workplace*

Workplace adalah memori kerja yang berfungsi sebagai penyimpan kondisi, hipotesa maupun kesimpulan sementara pada saat *user* melakukan interaksi terhadap sistem informasi.

6. Fasilitas Penjelasan

Adanya fasilitas penjelasan untuk mengetahui proses dari penalaran pakar karena *user* tidak semuanya adalah seorang ahli dalam bidang tersebut.

7. Perbaikan Pengetahuan

Tidak ada program yang sempurna, Perbaikan pengetahuan ada karena terjadinya kesuksesan maupun kegagalan. Masalah utama yang menjadi area luas aplikasi sistem pakar diantaranya yaitu: interpretasi, diagnosa, perencanaan, prediksi, perbaikan, kontrol, *monitoring*, perancangan, *debugging*, dan instruksi.

2.1.2 Penalaran Berbasis Pengetahuan (Rule-based Reasoning) Forward Chaining

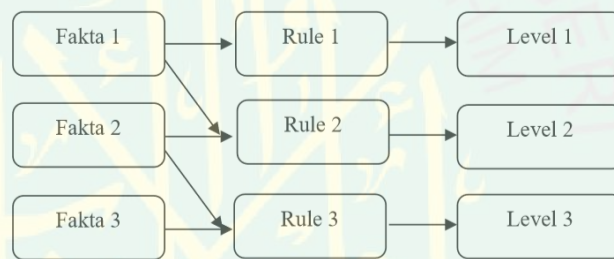
Forward Chaining merupakan penalaran yang berasal dari fakta-fakta untuk mendapatkan kesimpulan berdasarkan fakta-fakta sebelumnya (Giaratanno dan Riley, 2005).

Pada sistem informasi, penalaran berbasis aturan ditunjukkan sebagai IF-THEN pada bahasa pemrograman dan digunakan jika memiliki sejumlah pengetahuan pakar pada suatu permasalahan dan diselesaikan secara berurutan. Disamping itu digunakan jika penjelasan langkah-langkah pencapaian kesimpulan dibutuhkan (Eshete, 2009).

Basis pengetahuan yang digunakan untuk menyelesaikan suatu masalah, basis pengetahuan dibagi menjadi dua yaitu: fakta dan aturan khusus. Untuk fakta bisa diambil dari teori area persoalan dan situasi

persoalan, dan untuk aturan khusus digunakan untuk memecahkan suatu masalah khusus dalam *domain* tertentu. Aturan khusus menyatakan pengetahuan informasi dalam area aplikasi (Sharma dkk., 2012).

Mesin inferensi adalah bagian yang mengatur proses dari sistem informasi untuk mendapatkan suatu kesimpulan yang terbaik berdasarkan sumber dari aturan-aturan. Mesin inferensi ini menyediakan arahan bagaimana urutan yang mengatur langkah yang diambil untuk mengatasi suatu permasalahan pada saat konsultasi berlangsung (Al-ajlan, 2015).



Gambar 2.3 Forward Chaining.

Pada Gambar 2.3 terdapat 3 fakta yang digunakan sebagai dasar, 3 aturan dari pakar dengan kondisi tertentu, dan menghasilkan level skala prioritas satu sampai dengan tiga. Pada *forward chaining*, pendekatan berdasarkan data yaitu pelacakan berawal dari fakta-fakta yang dicocokkan pada bagian IF untuk menghasilkan suatu kesimpulan. Berikut contoh *forward chaining* menggunakan lima aturan sebagai berikut:

Aturan 1 : IF (Y dan D) THEN Z

Aturan 2 : IF (X dan B dan E) THEN Y

Aturan 3 : IF A THEN X

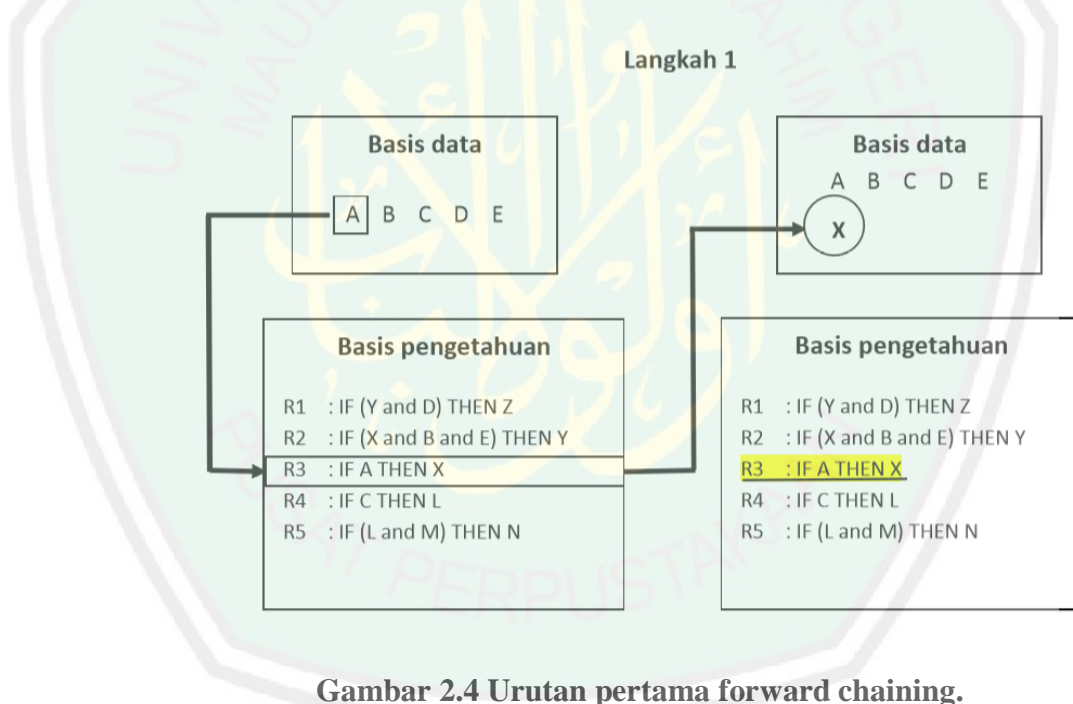
Aturan 4 : IF C THEN L

Aturan 5 : IF (L dan M) THEN N

Fakta : A, B, C, D, dan E adalah bernilai *true*.

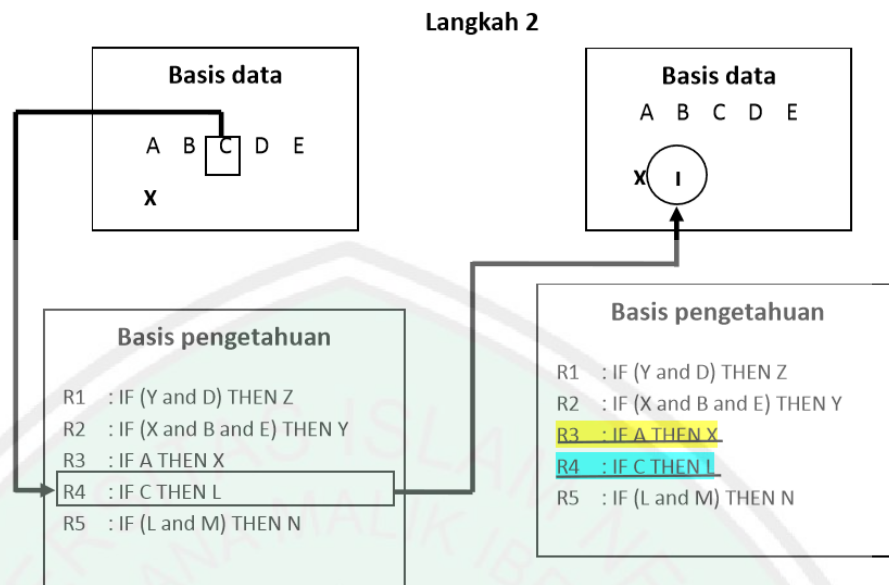
Kesimpulan: Menyimpulkan Z apakah *true* atau *false*.

Pada urutan pertama, fakta A sudah ditentukan di awal dan dimasukkan fakta A tersebut ke dalam aturan maka sistem akan memanggil *rule* R3 dan menghasilkan fakta baru atau kesimpulan sementara berupa fakta X dan dimasukkan fakta tersebut ke dalam basis data. Urutan pertama ditunjukkan pada Gambar 2.4.



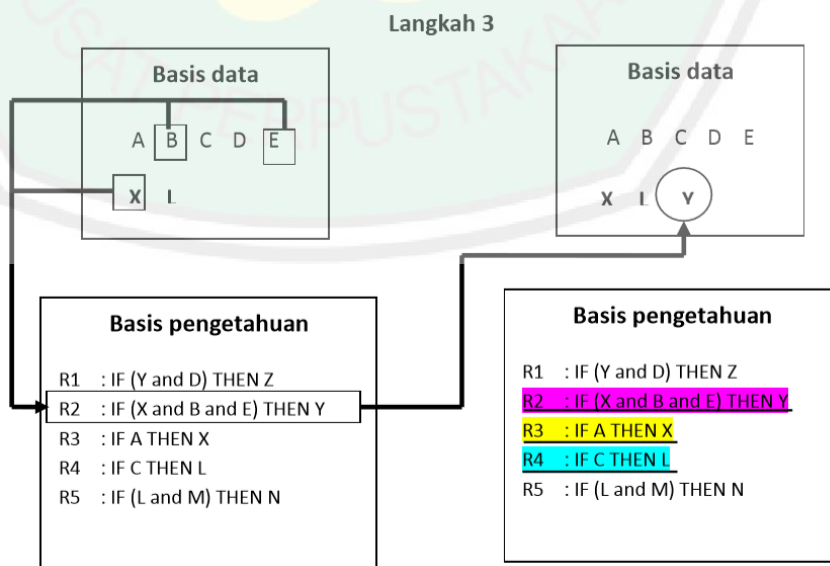
Gambar 2.4 Urutan pertama forward chaining.

Pada urutan kedua, fakta C sudah ditentukan di awal dan dimasukkan fakta C tersebut ke dalam aturan maka sistem akan memanggil *rule* R4 dan menghasilkan fakta baru atau kesimpulan sementara berupa fakta L dan dimasukkan fakta tersebut ke dalam basis data. Urutan kedua ditunjukkan pada Gambar 2.5.



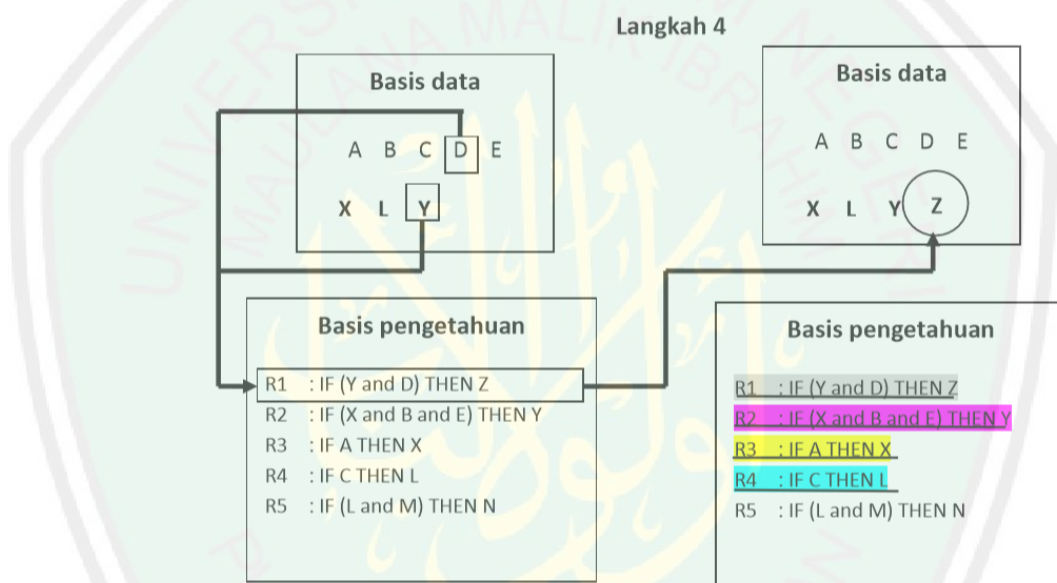
Gambar 2.5 Urutan kedua forward chaining.

Pada urutan ketiga, pada basis data terdapat fakta tambahan yaitu X dan L, kemudian dimasukkan fakta X, B dan E tersebut ke dalam aturan maka sistem akan memanggil *rule* R2 dan menghasilkan fakta baru atau kesimpulan sementara berupa fakta Y dan dimasukkan fakta tersebut ke dalam basis data. Urutan ketiga ditunjukkan pada Gambar 2.6.



Gambar 2.6 Urutan ketiga forward chaining.

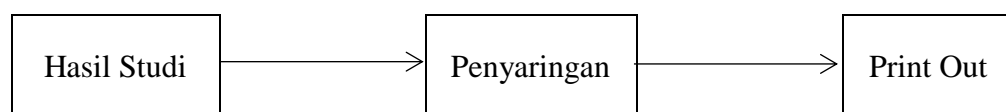
Pada urutan keempat, pada basis data terdapat fakta tambahan lagi yaitu fakta Y, kemudian dimasukkan fakta Y dan D tersebut ke dalam aturan maka sistem akan memanggil *rule* R1 dan menghasilkan fakta baru atau kesimpulan sementara berupa fakta Z dan dimasukkan fakta tersebut ke dalam basis data. Pada langkah ini terhenti karena sudah mendapatkan kesimpulan akhir yaitu Z. Urutan keempat ditunjukkan pada Gambar 2.7.



Gambar 2.7 Urutan keempat forward chaining.

2.2 Skala Prioritas

Dalam hal Akademik Mahasiswa, adanya persyaratan untuk dapat menghasilkan suatu *output* dan setiap persyaratan adanya prioritas tertinggi sampai terendah agar *output* dapat menghasilkan sesuai dengan kebutuhan sebagaimana ditunjukkan Gambar 2.8.



Gambar 2.8 Alur penyelesaian Akademik Mahasiswa

Adapun variabel-variabel yang dijadikan skala prioritas tersebut ditunjukkan dalam Tabel 2.1.

Tabel 2.1 Tabel Kriteria Skala Prioritas

No	Kriteria Skala Prioritas
1	Kode mata kuliah
2	Nilai mata kuliah di bawah C atau 2
3	Jumlah SKS
4	Mata kuliah

Berdasarkan pada Tabel 2.1 adapun kriteria adalah sebagai berikut:

1. Kode Mata Kuliah

Kode mata kuliah disini digunakan sebagai indikator suatu penyaringan di mana mata kuliah apa saja yang belum lulus maupun mata kuliah yang belum diambil, dan sebagai prioritas tertinggi suatu penyelesaian masalah.

2. Nilai Mata Kuliah

Nilai mata kuliah adalah salah satu penentu kelulusan mahasiswa, jika nilai mata kuliah mahasiswa di bawah C atau 2, maka mahasiswa diberi opsi untuk mengambil mata kuliah tersebut.

3. Jumlah SKS

Jumlah SKS adalah syarat kelulusan bagi setiap mahasiswa, jika jumlah SKS mahasiswa tidak memenuhi sesuai jumlah SKS kelulusan, maka akan ditampilkan sisa SKS yang belum diambil.

4. Mata kuliah

Opsi mata kuliah apa saja nanti yang akan ditampilkan sesuai kode mata kuliah, nilai mata kuliah yang belum lulus atau belum diambil.

BAB III

DESAIN DAN PERANCANGAN SISTEM

3.1 Desain Sistem

3.1.1 Desain Input

Pada desain *input*, sistem informasi akan menerima *input* berupa:

1. *User*, menginputkan berupa daftar mata kuliah yang telah ditempuh oleh mahasiswa berupa file excel yang sudah dikonversi dari web siacad bagian Transkrip Nilai.
2. *System*, input berupa pemasaran mata kuliah yang disediakan oleh *system*.

3.1.2 Desain Proses

Client memasukkan *file excel* yang berisi Hasil Studi selama perkuliahan yang sudah disediakan. Aplikasi memproses mata kuliah mana yang belum lulus maupun mata kuliah mana yang belum diambil.

3.1.3 Desain Output

Sistem informasi akan menghasilkan *output* berupa mata kuliah yang belum lulus maupun mata kuliah mana yang belum diambil dan kemudian *Client* bisa mempersiapkan pemrograman KRS untuk mempermudah pengambilan mata kuliah yang akan ditempuh.

3.2 Rule Based Pendekatan Forward Chaining

Aplikasi ini menggunakan *Rule Based* sebagai metodenya dan *Forward Chaining* sebagai eksekusinya. Berikut *forward chaining* pada aplikasi menggunakan 12 *rules* dan 10 fakta:

A: KHS Nilai Angka ≥ 2.0

B: KHS Nilai Angka < 2.0

C: Tabel Pemasaran

D: Mk1 List

E: Mk2 List

F: Tabel KHS

G: Tabel Mahasiswa

H: Pemasaran kode_mk_alias='0'

I: Pemasaran kode_mk_alias

J: NOT IN

R1: IF (F & A) THEN N (Tabel Lulus)

R2: IF (F & B) THEN O (Tabel Tidak Lulus)

R3: IF (N & O) THEN P (Update KHS)

R4: IF (P & B & I & J & H) THEN Q (Tabel Konversi)

R5: IF (C & D) THEN R (Tabel pemasaranmk1)

R6: IF (C & E) THEN S (Tabel pemasaranmk2)

R7: IF (R & A) THEN T (Tabel skipmkpilihan)

R8: IF (S & A) THEN T (Tabel skipmkpilihan)

R9: IF (R & J & F) THEN U (Tabel pemasarannew)

R10: IF (S & J & F) THEN U (Tabel pemasarannew)

R11: IF ((G & J & F) & (G & J & R) & (G & J & S) & (G & H)) THEN U
(Tabel pemasarannew)

R12: IF (U & G) THEN Q

Fakta : A, B, C, D, E, F, G, H, I, J adalah bernilai *true*.

Kesimpulan: Mengetahui apakah Q bernilai *true* atau *false*.

3.2.1 Langkah Pertama Forward Chaining

Langkah pertama forward chaining pada aplikasi ini ditunjukkan pada Gambar 3.1.

Basis Data									
A	B	C	D	E	F	G	H	I	J
N									

Gambar 3.1 Langkah Pertama Forward Chaining.

3.2.2 Langkah Kedua Forward Chaining

Langkah kedua forward chaining pada aplikasi ini ditunjukkan pada Gambar 3.2.

Basis Data									
A	B	C	D	E	F	G	H	I	J
N	O								

Gambar 3.2 Langkah Kedua Forward Chaining.

3.2.3 Langkah Ketiga Forward Chaining

Langkah ketiga forward chaining pada aplikasi ini ditunjukkan pada Gambar 3.3.

Basis Data									
A	B	C	D	E	F	G	H	I	J
N	O	P							

Gambar 3.3 Langkah Ketiga Forward Chaining.

3.2.4 Langkah Keempat Forward Chaining

Langkah keempat forward chaining pada aplikasi ini ditunjukkan pada Gambar 3.4.

Basis Data									
A	B	C	D	E	F	G	H	I	J
N	O	P	Q						

Gambar 3.4 Langkah Keempat Forward Chaining.

3.2.5 Langkah Kelima Forward Chaining

Langkah kelima forward chaining pada aplikasi ini ditunjukkan pada Gambar 3.5.

Basis Data									
A	B	C	D	E	F	G	H	I	J
N	O	P	Q	R					

Gambar 3.5 Langkah Kelima Forward Chaining.

3.2.6 Langkah Keenam Forward Chaining

Langkah keenam forward chaining pada aplikasi ini ditunjukkan pada Gambar 3.6.

Basis Data									
A	B	C	D	E	F	G	H	I	J
N	O	P	Q	R	S				

Gambar 3.6 Langkah Keenam Forward Chaining.

3.2.7 Langkah Ketujuh Forward Chaining

Langkah ketujuh forward chaining pada aplikasi ini ditunjukkan pada Gambar 3.7.

Basis Data									
A	B	C	D	E	F	G	H	I	J
N	O	P	Q	R	S	T			

Gambar 3.7 Langkah Ketujuh Forward Chaining.

3.2.8 Langkah Kedelapan Forward Chaining

Langkah kedelapan forward chaining pada aplikasi ini ditunjukkan pada Gambar 3.8.

Basis Data									
A	B	C	D	E	F	G	H	I	J
N	O	P	Q	R	S	T			

Gambar 3.8 Langkah Kedelapan Forward Chaining.

3.2.9 Langkah Kesembilan Forward Chaining

Langkah kesembilan forward chaining pada aplikasi ini ditunjukkan pada Gambar 3.9.

Basis Data									
A	B	C	D	E	F	G	H	I	J
N	O	P	Q	R	S	T	U		

Gambar 3.9 Langkah Kesembilan Forward Chaining.

3.2.10 Langkah Kesepuluh Forward Chaining

Langkah kesepuluh forward chaining pada aplikasi ini ditunjukkan pada Gambar 3.10.

Basis Data									
A	B	C	D	E	F	G	H	I	J
N	O	P	Q	R	S	T	U		

Gambar 3.10 Langkah Kesepuluh Forward Chaining.

3.2.11 Langkah Kesebelas Forward Chaining

Langkah kesebelas forward chaining pada aplikasi ini ditunjukkan pada Gambar 3.11.

Basis Data									
A	B	C	D	E	F	G	H	I	J
N	O	P	Q	R	S	T	U		

Gambar 3.11 Langkah Kesebelas Forward Chaining.

3.2.12 Langkah Keduabelas Forward Chaining

Langkah keduabelas forward chaining pada aplikasi ini ditunjukkan pada Gambar 3.12.

Basis Data									
A	B	C	D	E	F	G	H	I	J
N	O	P	Q	R	S	T	U		

Gambar 3.12 Langkah Keduabelas Forward Chaining.

3.3 Pengukuran

Dalam pembahasan ini, proses pengukuran dilakukan pada aplikasi Pengukuran Kinerja Akademik Mahasiswa Teknik Informatika UIN Maliki Malang menggunakan metode *Rule Based* pendekatan *Forward Chaining*. Kategori yang digunakan pada sistem informasi ini berupa:

1. Kode mata kuliah

Kode mata kuliah disini digunakan sebagai indikator suatu penyaringan di mana mata kuliah apa saja yang belum lulus maupun mata kuliah yang belum diambil, dan sebagai prioritas tertinggi suatu penyelesaian masalah.

- Belum diambil (true)
- Sudah diambil (false)

2. Nilai mata kuliah

Nilai mata kuliah adalah salah satu penentu kelulusan mahasiswa, jika nilai mata kuliah mahasiswa di bawah C atau 2, maka mahasiswa diberi opsi untuk mengambil mata kuliah tersebut.

- Lulus (≥ 2)
- Tidak lulus (< 2)

3. Jumlah SKS

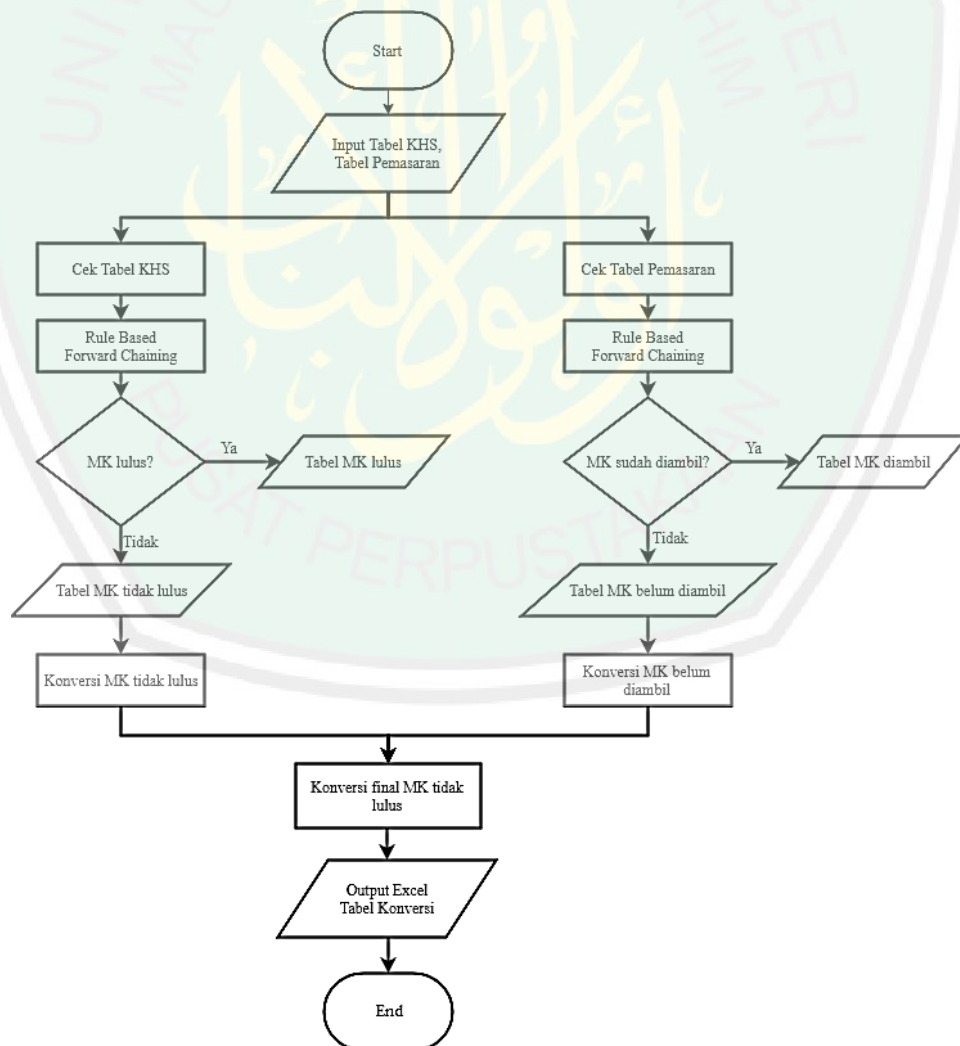
Jumlah SKS akan ditampilkan nanti di *output* berdasarkan mata kuliah yang belum lulus maupun mata kuliah yang belum ditempuh berdasarkan *input* maupun pemasaran.

4. Nama mata kuliah

Aplikasi ini menggunakan kode mata kuliah sebagai acuan dalam pengukuran, dan nama mata kuliah digunakan untuk memudahkan mahasiswa mengetahui nama mata kuliah dari kode mata kuliah yang akan ditempuh.

3.4 Flowchart Proses

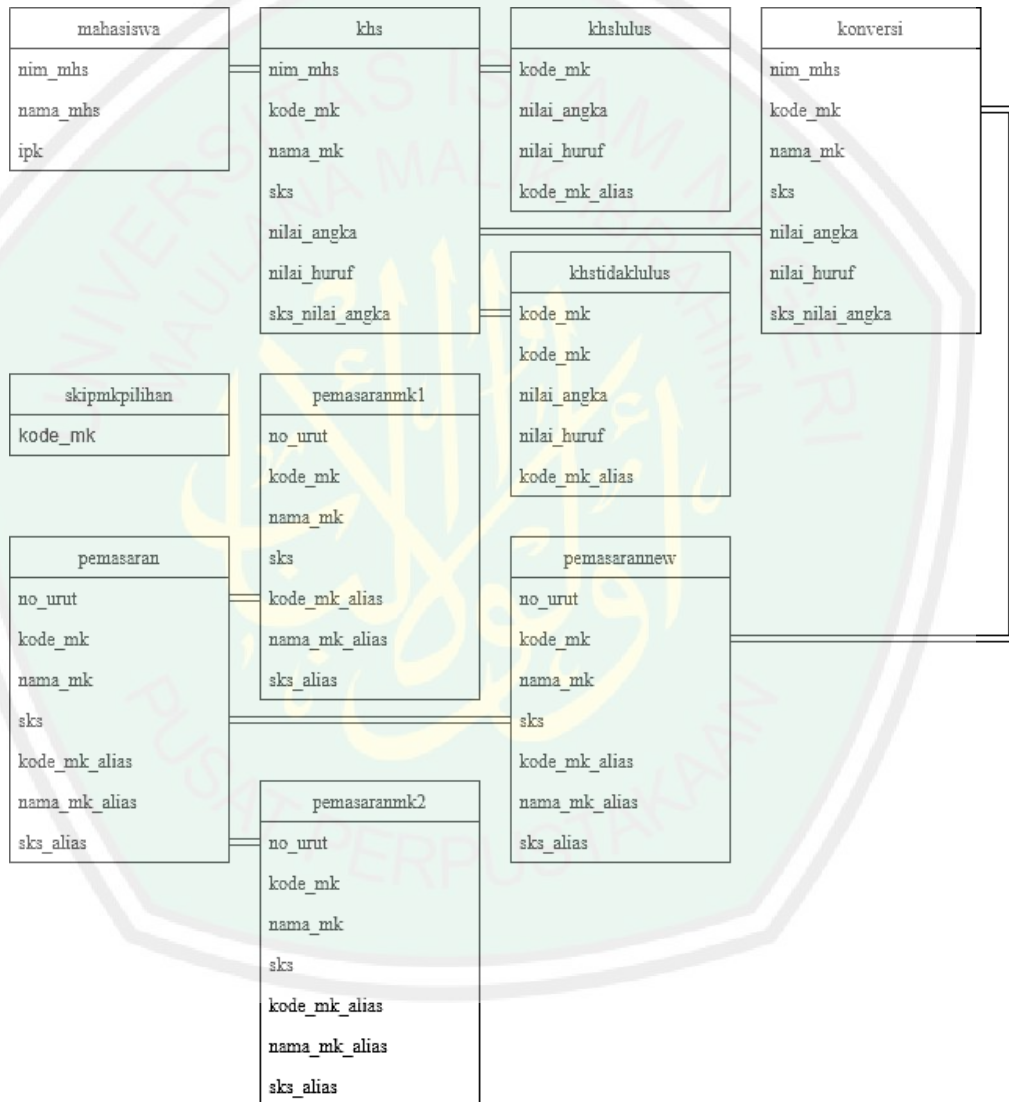
Proses ini berfungsi untuk kinerja aplikasi pada saat Mahasiswa menginputkan Hasil Studinya. Urutan *flowchart* proses dapat dilihat pada Gambar 3.13.



Gambar 3.13 Flowchart proses aplikasi.

3.5 Entity Relationship Diagram (ERD)

Pada penelitian ini dilampirkan ERD yang di dalamnya terdapat himpunan *entitas* dan himpunan relasi beserta atribut di dalamnya untuk menampilkan fakta-fakta berdasarkan dengan kenyataan dan berasal dari table-tabel yang dibuat. ERD dapat dilihat pada Gambar 3.14.



Gambar 3.14 ERD Sistem Pengukuran Kinerja Akademik Mahasiswa

UIN Maliki Malang.

Pembuatan sistem informasi ini terdapat desain database, digunakan untuk menyimpan data dalam menampung *input* dari user berupa tabel di dalamnya dan untuk proses Akademik Mahasiswa UIN Maliki Malang.

3.6 Desain Database

Dalam pembuatan program ini dibutuhkan desain database untuk menyimpan informasi user dalam menyimpan variabel tertentu sebagai indikator dalam proses aplikasi. Desain database ini menjelaskan tabel-tabel yang akan digunakan. Berikut adalah Tabel yang digunakan untuk proses seleksi mata kuliah yang belum lulus maupun belum diambil menggunakan metode Rule Based.

3.6.1 Tabel Mahasiswa

Tabel mahasiswa adalah tabel untuk menyimpan data diri mahasiswa. Pada SQL, tabel ini diberi nama 'mahasiswa'. Tabel ini terdapat bagian-bagian seperti pada Tabel 3.1.

Tabel 3.1 Tabel mahasiswa.

No	Field	Tipe Data	Key	Keterangan
1	nim_mhs	varchar (8)	PK	NIM Mahasiswa
2	nama_mhs	varchar (100)		Nama Mahasiswa
3	ipk	double		IPK Mahasiswa

3.6.2 Tabel Khs

Tabel khs adalah tabel yang diinput mahasiswa berisi tentang mata kuliah yang sudah lulus maupun mata kuliah yang belum lulus. Pada SQL, tabel ini diberi nama 'khs'. Tabel ini terdapat bagian-bagian seperti pada Tabel 3.2.

Tabel 3.2 Tabel khs.

No	Field	Tipe Data	Key	Keterangan
1	nim_mhs	varchar (8)		NIM Mahasiswa
2	kode_mk	int (7)	PK	Kode Mata Kuliah
3	nama_mk	varchar (100)		Nama Mata Kuliah
4	sks	int(1)		SKS Mata Kuliah
5	nilai_angka	double		Nilai Mata Kuliah dalam bentuk angka
6	nilai_huruf	varchar(2)		Nilai Mata Kuliah dalam bentuk huruf
7	sks_nilai_angka	double		SKS dikali Nilai Mata Kuliah

3.6.2.1 Tabel Khs Lulus

Tabel khs lulus adalah tabel yang struktur dan datanya sama dengan Tabel khs, tapi hanya menampung nilai mata kuliah yang sudah lulus saja. Pada SQL, tabel ini diberi nama 'khslulus'. Tabel ini terdapat bagian-bagian seperti pada Tabel 3.3.

Tabel 3.3 Tabel khs lulus.

No	Field	Tipe Data	Key	Keterangan
1	nim_mhs	varchar (8)		NIM Mahasiswa
2	kode_mk	int (7)	PK	Kode Mata Kuliah
3	nama_mk	varchar (100)		Nama Mata Kuliah
4	sks	int(1)		SKS Mata Kuliah
5	nilai_angka	double		Nilai Mata Kuliah dalam bentuk angka
6	nilai_huruf	varchar(2)		Nilai Mata Kuliah dalam bentuk huruf
7	sks_nilai_angka	double		SKS dikali Nilai Mata Kuliah

3.6.2.2 Tabel Khs Tidak Lulus

Tabel khs tidak lulus adalah tabel yang struktur dan datanya sama dengan Tabel khs, tapi hanya menampung nilai mata kuliah yang tidak lulus saja. Pada SQL, tabel ini diberi nama 'khstidaklulus'. Tabel ini terdapat bagian-bagian seperti pada Tabel 3.4.

Tabel 3.4 Tabel khs tidak lulus.

No	Field	Tipe Data	Key	Keterangan
1	nim_mhs	varchar (8)		NIM Mahasiswa
2	kode_mk	int (7)	PK	Kode Mata Kuliah
3	nama_mk	varchar (100)		Nama Mata Kuliah
4	sks	int(1)		SKS Mata Kuliah
5	nilai_angka	double		Nilai Mata Kuliah dalam bentuk angka
6	nilai_huruf	varchar(2)		Nilai Mata Kuliah dalam bentuk huruf
7	sks_nilai_angka	double		SKS dikali Nilai Mata Kuliah

3.6.3 Tabel Pemasaran

Tabel pemasaran adalah tabel yang digunakan aplikasi sebagai indikator perbandingan mata kuliah lama dan mata kuliah baru. Pada SQL, tabel ini diberi nama 'pemasaran'. Tabel ini terdapat bagian-bagian seperti pada Tabel 3.5.

Tabel 3.5 Tabel pemasaran.

No	Field	Type Data	Key	Keterangan
1	no_urut	int (4)	PK	Nomor Urut Tabel
2	kode_mk	int (7)		Kode Mata Kuliah
3	nama_mk	varchar (100)		Nama Mata Kuliah
4	sks	int(1)		SKS Mata Kuliah
5	kode_mk_alias	int (7)		Kode Mata Kuliah Baru
6	nama_mk_alias	varchar (100)		Nama Mata Kuliah Baru
7	sks_alias	int(1)		SKS Mata Kuliah Baru

3.6.3.1 Tabel Pemasaran Mata Kuliah Pilihan 1

Tabel pemasaran mata kuliah pilihan 1 adalah tabel yang struktur dan datanya sama, tetapi hanya terdiri dari mata kuliah pilihan 1 saja. Pada SQL, tabel ini diberi nama 'pemasaranmk1'.

Tabel ini terdapat bagian-bagian seperti pada Tabel 3.6.

Tabel 3.6 Tabel pemasaran mata kuliah pilihan 1.

No	Field	Type Data	Key	Keterangan
1	no_urut	int (4)	PK	Nomor Urut Tabel
2	kode_mk	int (7)		Kode Mata Kuliah
3	nama_mk	varchar (100)		Nama Mata Kuliah
4	sks	int(1)		SKS Mata Kuliah
5	kode_mk_alias	int (7)		Kode Mata Kuliah Baru
6	nama_mk_alias	varchar (100)		Nama Mata Kuliah Baru
7	sks_alias	int(1)		SKS Mata Kuliah Baru

3.6.3.2 Tabel Pemasaran Mata Kuliah Pilihan 2

Tabel pemasaran mata kuliah pilihan 2 adalah tabel yang struktur dan datanya sama, tetapi hanya terdiri dari mata kuliah pilihan 2 saja. Pada SQL, tabel ini diberi nama 'pemasaranmk2'.

Tabel ini terdapat bagian-bagian seperti pada Tabel 3.7.

Tabel 3.7 Tabel pemasaran mata kuliah pilihan 2.

No	Field	Tipe Data	Key	Keterangan
1	no_urut	int (4)	PK	Nomor Urut Tabel
2	kode_mk	int (7)		Kode Mata Kuliah
3	nama_mk	varchar (100)		Nama Mata Kuliah
4	sks	int(1)		SKS Mata Kuliah
5	kode_mk_alias	int (7)		Kode Mata Kuliah Baru
6	nama_mk_alias	varchar (100)		Nama Mata Kuliah Baru
7	sks_alias	int(1)		SKS Mata Kuliah Baru

3.6.3.3 Tabel Lewati Mata Kuliah Pilihan

Tabel lewati mata kuliah pilihan adalah tabel yang terdiri dari 1 kolom saja yaitu 'kode mata kuliah alias' dari tabel mata kuliah pilihan, yang digunakan untuk menampung kode mata kuliah pilihan per kategori. Contoh: Jika dalam tabel khs terdapat mata kuliah pilihan 1 yang sudah lulus, maka lewati kode mata kuliah pilihan per kategori 'mkpilihan1' tersebut pada hasil konversi nanti. Pada SQL, tabel ini diberi nama 'skipmkpilihan'. Tabel ini terdapat bagian-bagian seperti pada Tabel 3.8.

Tabel 3.8 Tabel lewati mata kuliah pilihan.

No	Field	Tipe Data	Key	Keterangan
1	kode_mk	int (7)	PK	Kode Mata Kuliah

3.6.3.4 Tabel Pemasaran Baru

Tabel pemasaran baru adalah tabel yang digunakan untuk menampung mata kuliah yang belum diambil oleh mahasiswa dari tabel pemasaran, tabel mata kuliah pilihan 1, dan tabel mata kuliah pilihan 2. Pada SQL, tabel ini diberi nama 'pemasarannew'. Tabel ini terdapat bagian-bagian seperti pada Tabel 3.9.

Tabel 3.9 Tabel pemasaran baru.

No	Field	Tipe Data	Key	Keterangan
1	no_urut	int (4)	PK	Nomor Urut Tabel
2	kode_mk	int (7)		Kode Mata Kuliah
3	nama_mk	varchar (100)		Nama Mata Kuliah
4	Sks	int(1)		SKS Mata Kuliah
5	kode_mk_alias	int (7)		Kode Mata Kuliah Baru
6	nama_mk_alias	varchar (100)		Nama Mata Kuliah Baru
7	sks_alias	int(1)		SKS Mata Kuliah Baru

3.6.4 Tabel Konversi

Tabel konversi adalah tabel output yang terdiri dari tabel 'khs' yang belum lulus, tabel 'pemasarannew' digabungkan dengan beberapa kolom dari tabel 'mahasiswa' yang terdiri dari mata kuliah mahasiswa yang belum diambil. Pada SQL, tabel ini diberi nama 'konversi'. Tabel ini terdapat bagian-bagian seperti pada Tabel 3.10.

Tabel 3.10 Tabel konversi.

No	Field	Type Data	Key	Keterangan
1	nim_mhs	varchar (8)		NIM Mahasiswa
2	kode_mk	int (7)	PK	Kode Mata Kuliah
3	nama_mk	varchar (100)		Nama Mata Kuliah
4	sks	int(1)		SKS Mata Kuliah
5	nilai_angka	double		Nilai Mata Kuliah dalam bentuk angka
6	nilai_huruf	varchar(2)		Nilai Mata Kuliah dalam bentuk huruf
7	sks_nilai_angka	double		SKS dikali Nilai Mata Kuliah

3.7 Desain Interface

3.7.1 Tampilan Menu Utama

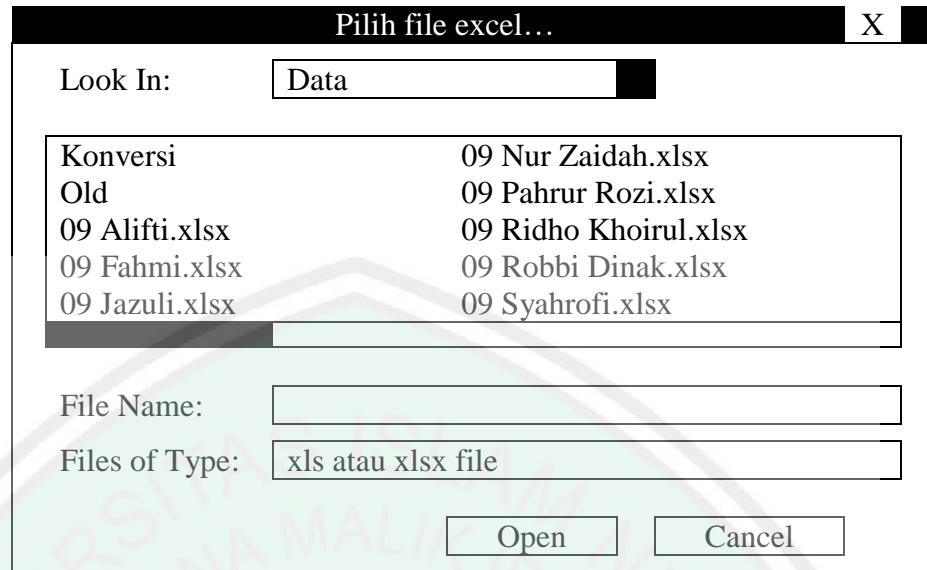
Activity ini mencakup keseluruhan aplikasi untuk mempermudah Mahasiswa menginput Hasil Studi ke Aplikasi. Tampilan menu utama dapat dilihat pada Gambar 3.15.



Gambar 3.15 Tampilan Menu Utama.

3.7.2 Tampilan Dialog Browse

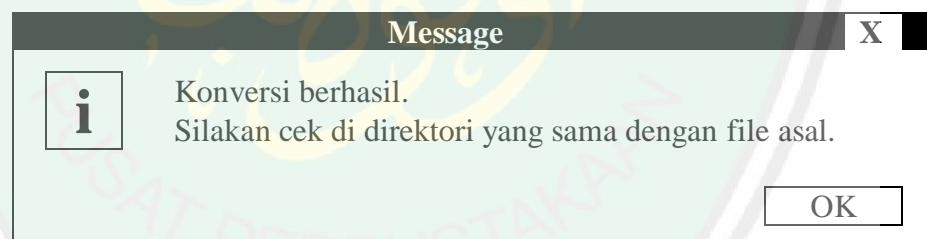
Activity ini mencakup mahasiswa dalam mencari *file* yang akan dimasukkan ke aplikasi. Tampilan *dialog* browse dapat dilihat pada Gambar 3.16.



Gambar 3.16 Tampilan pilih file Hasil Studi.

3.7.3 Pop-up Konversi Berhasil

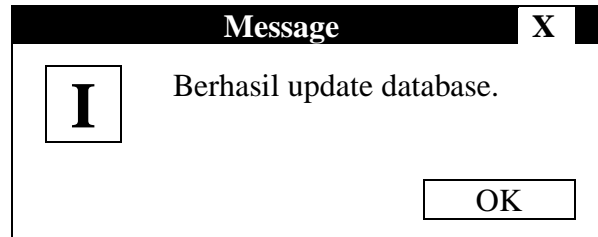
Activity ini menampilkan *pop-up* tombol konversi ketika *file* berhasil dikonversi. Tampilan *pop-up* konversi berhasil dapat dilihat pada Gambar 3.17.



Gambar 3.17 Tampilan pop-up berhasil konversi.

3.7.4 Pop-up Berhasil Update

Activity ini menampilkan *pop-up* tombol update ketika ada pembaruan *file* database mata kuliah dan otomatis ter-update. Tampilan *pop-up* berhasil update dapat dilihat pada Gambar 3.18.



Gambar 3.18 Tampilan *pop-up* berhasil update.

3.7.5 *Pop-up* Tidak Ada Update

Activity ini menampilkan *pop-up* tombol update ketika tidak ada pembaruan *file* database mata kuliah. Tampilan *pop-up* tidak ada update dapat dilihat pada Gambar 3.19.



Gambar 3.19 Tampilan *Pop-up* tidak ada update.

BAB IV

HASIL DAN PEMBAHASAN

Bab ini menjelaskan tentang alur uji coba dan evaluasi terhadap penelitian yang telah dilakukan. Diperlukan uji coba agar dapat mengetahui hasil akurasi dari sistem informasi yang telah dibuat dan evaluasi yang dilakukan untuk mengukur dari hasil uji coba dan untuk mendapatkan kesimpulan maupun saran agar dapat dikembangkan di penelitian selanjutnya.

4.1 Pengukuran

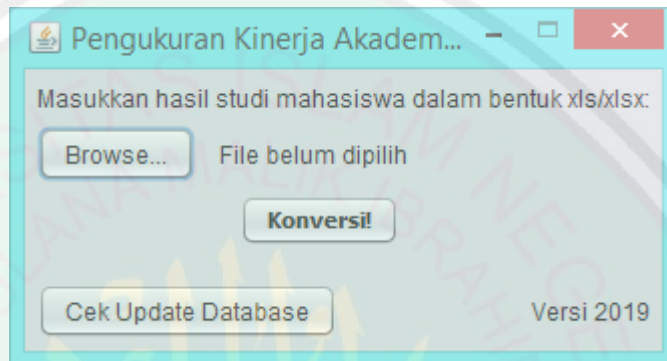
Sistem informasi ini dikembangkan menggunakan java, *excel* dan SQL di mana java bertindak sebagai client dan update data pemasaran sebagai server bila ada perubahan data pemasaran tiap semester, *excel* sebagai *input* data dan *output* data setelah proses aplikasi, dan SQL sebagai proses konversi. Sistem berjalan dalam keadaan *offline*, keadaan *online* hanya untuk *update* data pemasaran yang bilamana ada perubahan mata kuliah tiap semester pemrograman mata kuliah. Pengukuran kinerja akademik dilakukan pada proses konversi berlangsung, menampilkan mata kuliah yang belum ditempuh maupun yang belum diambil, dan client akan mendapatkan *file output* berupa *excel* di *path* yang sama dengan *input*.

4.2 User Interface

User Interface menampilkan aplikasi beserta proses sistem informasi yang terjadi di dalamnya saat aplikasi dijalankan. Terdapat antarmuka pengguna utama pada aplikasi ini dan dibagi menjadi beberapa bagian.

4.2.1 Halaman Utama

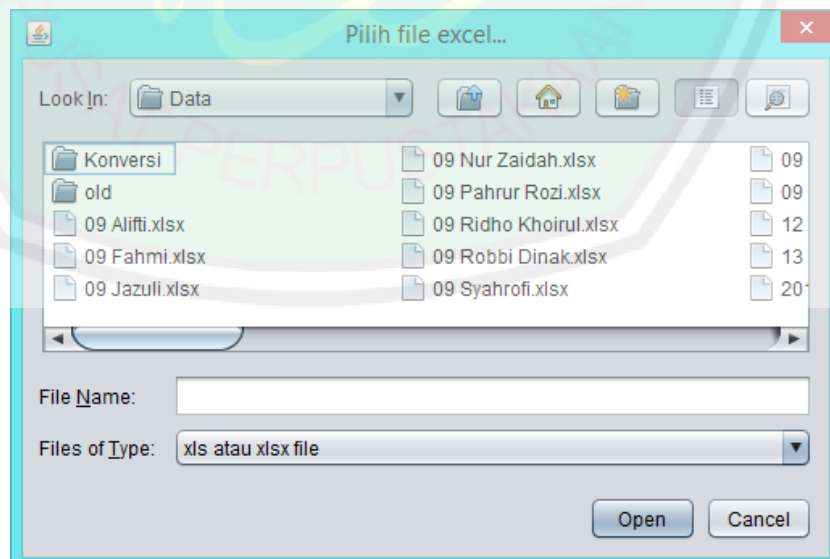
Halaman utama adalah halaman yang muncul pertama kali pada saat aplikasi dijalankan. Halaman utama menampilkan tombol yang dibutuhkan pada saat melakukan konversi mata kuliah. Halaman utama dapat dilihat pada Gambar 4.1.



Gambar 4.1 Tampilan halaman utama.

4.2.2 Halaman *Browse* Sebagai Input Data

Halaman *browse* akan muncul jika tombol 'Browse...' pada halaman utama dijalankan. Halaman *browse* dapat dilihat pada Gambar 4.2.

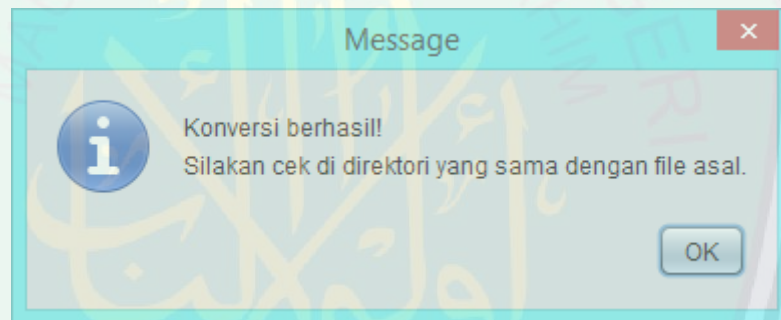


Gambar 4.2 Tampilan halaman *browse*.

Untuk kode sumber *browse* dapat dilihat di Lampiran Kode Sumber Browse.

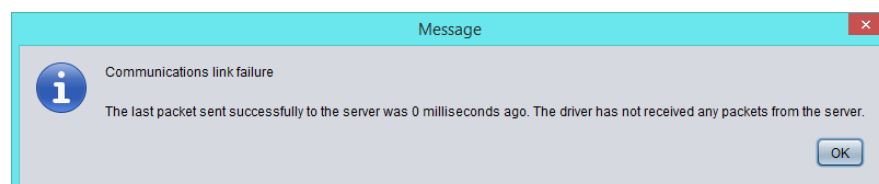
4.2.3 *Pop-up* Konversi

Pop-up konversi terbagi menjadi 3 yaitu: *pop-up* konversi berhasil, *pop-up* konversi gagal, dan *pop-up* input belum dipilih. *Pop-up* konversi berhasil akan muncul jika user sudah memasukkan *file* KHS melalui tombol *Browse*, menjalankan tombol Konversi dan eksekusi setiap *method* pada aplikasi yang sedang berjalan. *Pop-up* konversi berhasil dapat dilihat pada Gambar 4.3.



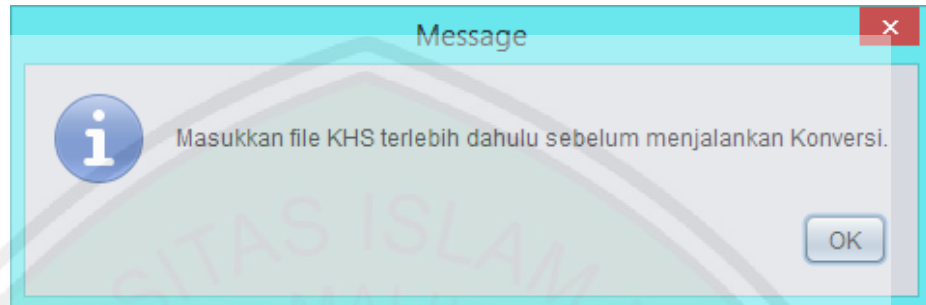
Gambar 4.3 Tampilan *pop-up* konversi berhasil.

Pop-up konversi gagal akan muncul jika user sudah memilih input tapi ada kesalahan pada saat eksekusi. *Pop-up* konversi gagal menampilkan pesan *error* sesuai kesalahan yang terjadi pada aplikasi. Pada Gambar 4.4 adalah salah satu *pop-up* konversi gagal ketika *user* mengeksekusi tombol Konversi dan SQL belum diaktifkan.



Gambar 4.4 Tampilan *pop-up* konversi gagal karena SQL belum aktif.

Pop-up input belum dipilih akan muncul jika *user* belum memilih *file KHS* sebagai *input* dan mengeksekusi tombol Konversi. *Pop-up input* belum dipilih dapat dilihat pada Gambar 4.5.

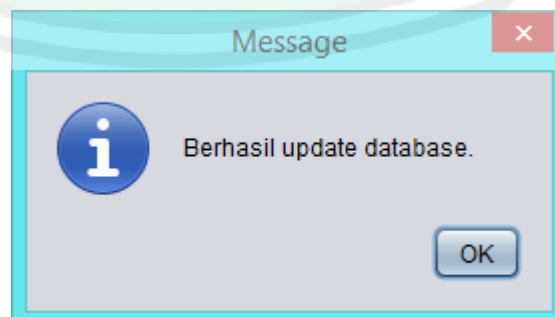


Gambar 4.5 Tampilan *pop-up* input belum dipilih.

Untuk kode sumber dari Konversi bisa dilihat di Lahiran Kode Sumber Konversi.

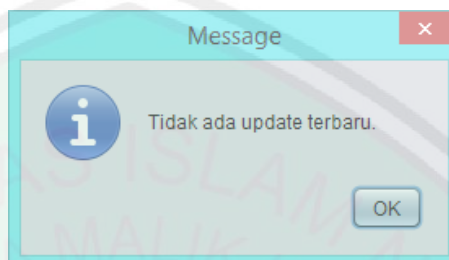
4.2.4 *Pop-up* Cek Update Database

Pop-up cek update database terbagi menjadi 2 yaitu: *pop-up* berhasil update database dan *pop-up* tidak ada update. *Pop-up* berhasil update database akan muncul jika *user* mengeksekusi tombol ‘Cek Update Database’ dan kondisi ada perubahan data *file* ‘Rule Konversi.xlsx’, ‘mkpilihan1.txt’ dan ‘mkpilihan2.txt’ dari server. *Pop-up* berhasil update database dapat dilihat pada Gambar 4.6.



Gambar 4.6 Tampilan *pop-up* berhasil update database.

Pop-up tidak ada update akan muncul jika *user* mengeksekusi tombol ‘Cek Update Database’ dan kondisi tidak perubahan data *file* ‘Rule Konversi.xlsx’, ‘mkpilihan1.txt’ dan ‘mkpilihan2.txt’ dari server. *Pop-up* tidak ada update dapat dilihat pada Gambar 4.7.



Gambar 4.7 Tampilan *pop-up* tidak ada update terbaru.

Untuk kode sumber dari ‘Cek Update Database’ bisa dilihat di Lampiran Kode Sumber Cek Update.

4.3 Rule Based Pendekatan *Forward Chaining*

Rule Based Forward Chaining pada aplikasi ini menggunakan SQL. Berikut adalah *query* SQL yang akan dijalankan di aplikasi ini.

4.3.1 Langkah Pertama *Forward Chaining*

Langkah pertama *forward chaining* pada aplikasi ini ditunjukkan pada Kode Sumber 4.1.

Kode Sumber 4.1 Langkah Pertama *Forward Chaining*.

```
INSERT INTO khslulus
SELECT * FROM (
  SELECT k.kode_mk, k.nilai_angka, k.nilai_huruf,
p.kode_mk_alias
  FROM khs k
  JOIN pemasaran p
  ON k.kode_mk = p.kode_mk
  WHERE k.nilai_angka >= '2.0'
  GROUP BY k.kode_mk
  ORDER BY p.kode_mk_alias ASC, k.nilai_angka DESC,
k.nilai_huruf ASC
) as lulus
GROUP BY kode_mk_alias
```

4.3.2 Langkah Kedua *Forward Chaining*

Langkah kedua *forward chaining* pada aplikasi ini ditunjukkan pada

Kode Sumber 4.2.

Kode Sumber 4.2 Langkah Kedua *Forward Chaining*.

```
INSERT INTO khstidaklulus
SELECT k.kode_mk, k.nilai_angka, k.nilai_huruf,
p.kode_mk_alias
FROM khs k
JOIN pemasaran p
ON k.kode_mk = p.kode_mk
WHERE k.nilai_angka < '2.0'
GROUP BY k.kode_mk
```

4.3.3 Langkah Ketiga *Forward Chaining*

Langkah ketiga *forward chaining* pada aplikasi ini ditunjukkan pada

Kode Sumber 4.3.

Kode Sumber 4.3 Langkah Ketiga *Forward Chaining*.

```
UPDATE khstidaklulus t
SET t.nilai_angka = (
SELECT l.nilai_angka
FROM khslulus l
WHERE t.kode_mk_alias = l.kode_mk_alias),
t.nilai_huruf = (
SELECT l.nilai_huruf
FROM khslulus l
WHERE t.kode_mk_alias = l.kode_mk_alias)
WHERE t.kode_mk_alias
IN (
SELECT kode_mk_alias
FROM khslulus);
UPDATE khs k
SET k.nilai_angka = (
SELECT t.nilai_angka
FROM khstidaklulus t
WHERE t.kode_mk = k.kode_mk),
k.nilai_huruf = (
SELECT t.nilai_huruf
FROM khstidaklulus t
WHERE t.kode_mk = k.kode_mk),
k.sks_nilai_angka = (k.sks * k.nilai_angka)
WHERE k.kode_mk
IN (
SELECT t.kode_mk
FROM khstidaklulus t);
```

4.3.4 Langkah Keempat *Forward Chaining*

Langkah keempat *forward chaining* pada aplikasi ini ditunjukkan pada Kode Sumber 4.4.

Kode Sumber 4.4 Langkah Keempat *Forward Chaining*.

```
INSERT INTO konversi
SELECT k.nim_mhs, p.kode_mk_alias, p.nama_mk_alias,
p.sks_alias, k.nilai_angka, k.nilai_huruf,
k.sks_nilai_angka
FROM khs k
JOIN pemasaran p
ON k.kode_mk = p.kode_mk
WHERE k.nilai_angka < '2.0'
AND p.kode_mk_alias != '0'
GROUP BY p.kode_mk alias
```

4.3.5 Langkah Kelima dan Keenam *Forward Chaining*

Langkah kelima dan keenam *forward chaining* dijadikan satu karena menggunakan *looping* per kategori, pada aplikasi ini proses *looping per kategori* ditunjukkan pada Kode Sumber 4.5.

Kode Sumber 4.5 *Looping* mata kuliah pilihan per kategori.

```
for (int i = 1; i <= 2; i++) // mk pilihan 1 ~ 2
```

Untuk langkah kelima dan keenam *forward chaining* ditunjukkan pada Kode Sumber 4.6 mengacu Kode Sumber 4.5 sebagai referensi *looping*.

Kode Sumber 4.6 Langkah Kelima dan Keenam *forward chaining*.

```
SELECT k.kode_mk, k.nilai_angka, p.kode_mk_alias
FROM khs k
JOIN pemasaranmk" + i + " p
ON k.kode_mk = p.kode_mk
WHERE k.nilai_angka >= '2.0'
GROUP BY k.kode_mk
```

4.3.6 Langkah Ketujuh dan Kedelapan *Forward Chaining*

Langkah ketujuh dan kedelapan *forward chaining* dijadikan satu karena menggunakan *looping* per kategori, pada aplikasi ini ditunjukkan pada Kode Sumber 4.7 mengacu Kode Sumber 4.6 sebagai referensi *looping* per kategori.

Kode Sumber 4.7 Langkah Ketujuh dan Kedelapan *Forward Chaining*.

```
for (int j = 0; j < pemasaranmk.length; j++)
INSERT INTO skipmcpilihan VALUES ('" + pemasaranmk[j]
+ "')
```

4.3.7 Langkah Kesembilan dan Kesepuluh *Forward Chaining*

Langkah kesembilan dan kesepuluh *forward chaining* dijadikan satu karena menggunakan *looping* per kategori, pada aplikasi ini ditunjukkan pada Kode Sumber 4.8 mengacu Kode Sumber 4.5 sebagai referensi *looping*.

Kode Sumber 4.8 Langkah Kesembilan dan Kesepuluh *Forward Chaining*.

```
INSERT INTO pemasarannew
SELECT * FROM pemasaranmk" + i + " pmk
WHERE pmk.kode_mk_alias
NOT IN (
SELECT k.kode_mk
FROM konversi k)
```

4.3.8 Langkah Kesebelas *Forward Chaining*

Langkah kesebelas *forward chaining* pada aplikasi ini ditunjukkan pada Kode Sumber 4.9.

Kode Sumber 4.9 Langkah Kesebelas *Forward Chaining*.

```

INSERT INTO pemasaranneu
SELECT * FROM pemasaran p
WHERE p.kode_mk_alias NOT IN (
  SELECT p.kode_mk_alias
  FROM khs k
  JOIN pemasaran p
  ON k.kode_mk = p.kode_mk)
AND p.kode_mk_alias NOT IN (
  SELECT p1.kode_mk_alias
  FROM pemasaranmk1 p1
  JOIN pemasaran p
  ON p1.kode_mk_alias = p.kode_mk_alias)
AND p.kode_mk_alias NOT IN (
  SELECT p2.kode_mk_alias
  FROM pemasaranmk2 p2
  JOIN pemasaran p
  ON p2.kode_mk_alias = p.kode_mk_alias)
AND p.kode_mk_alias != '0'
GROUP BY p.kode_mk_alias

```

4.3.9 Langkah Keduabelas *Forward Chaining*

Langkah keduabelas *forward chaining* pada aplikasi ini ditunjukkan pada Kode Sumber 4.10.

Kode Sumber 4.10 Langkah Keduabelas *Forward Chaining*.

```

INSERT INTO konversi
SELECT m.nim_mhs, p.kode_mk_alias, p.nama_mk_alias,
p.sks_alias, null as nilai_angka, null as nilai_huruf,
null as sks_nilai_angka
FROM pemasaranneu p
JOIN mahasiswa m
GROUP BY p.kode_mk_alias

```

4.4 Export Excel Sebagai Output

Sebelum dilakukan proses output, dilakukan proses pengecualian untuk mata kuliah pilihan per kategori yang tidak lulus dari table khs tetapi ada mata kuliah pilihan per kategori tersebut yang lulus agar tidak muncul mata kuliah pilihan per kategori yang tidak lulus tadi di *output* nanti. Proses ini ditunjukkan pada Kode Sumber 4.11.

Kode Sumber 4.11 Pengecualian dari tabel skipmcpilihan.

```
SELECT *
FROM konversi
WHERE kode_mk
NOT IN (
    SELECT kode_mk
    FROM skipmcpilihan)
```

4.5 Hasil Uji Coba Aplikasi

Uji coba aplikasi dilakukan dengan membandingkan hasil studi mahasiswa dengan rule konversi yang sudah tersedia oleh aplikasi. Rule Konversi adalah tabel *excel* yang berisi pemasaran mata kuliah yang disediakan oleh akademik mulai dari mata kuliah dengan kode mata kuliah lama hingga mata kuliah baru. Pada Rule Konversi, kode mata kuliah yang disediakan mulai dari tahun 2007 hingga tahun 2015. Adapun daftar mata kuliah lama maupun baru yang terdiri dari kode mata kuliah, nama mata kuliah, sks mata kuliah ditampilkan pada Tabel Pemasaran di Daftar Lampiran.

Hasil Studi adalah tabel *excel* yang berisi mata kuliah yang sudah ditempuh oleh mahasiswa yang digunakan sebagai *input* pada aplikasi ini. Format susunan hasil studi mengikuti format dari akademik atau transkrip nilai sementara di web akademik mahasiswa dalam format *file excel*, sehingga aplikasi dapat membaca *input* dan berjalan hingga menghasilkan *file output* sesuai *method-method* yang tersedia di aplikasi. Adapun daftar Hasil Studi ditampilkan pada tabel KHS di Daftar Lampiran.

Setelah aplikasi berjalan, akan menghasilkan *output* yang berupa *file excel* jua, agar dapat memudahkan mahasiswa melihat daftar mata kuliah apa saja yang belum ditempuh maupun mata kuliah yang belum lulus. Adapun daftar hasil konversi ditampilkan pada Tabel Konversi di Daftar Lampiran.

Pada uji coba ini, peneliti membandingkan *output* hasil konversi aplikasi dengan penghitungan manual dari hasil studi dengan referensi buku pedoman akademik dan pemasaran mata kuliah dari web akademik sebagai pemasaran mata kuliah. Peneliti menggunakan data sampel Hasil Studi mahasiswa sebanyak 41 untuk diuji. Adapun daftar pengujian ditampilkan pada Tabel 4.1.

Tabel 4.1 Hasil uji coba manual dan menggunakan aplikasi.

No	NIM	Manual		Forward Chaining		Keterangan
		Belum Lulus	Belum Diambil	Belum Lulus	Belum Diambil	
1	09650082	4	3	4	3	Sama
2	09650206	2	4	2	4	Sama
3	09650204	5	2	5	2	Sama
4	09650196	1	1	1	1	Sama
5	09650066	2	3	2	3	Sama
6	09650083	22	33	22	33	Sama
7	09650181	6	6	6	6	Sama
8	09650086	1	4	1	4	Sama
9	09650036	1	3	1	3	Sama
10	09650023	0	2	0	2	Sama
11	12650124	0	3	0	3	Sama
12	13650109	2	8	2	8	Sama
13	10650121	12	9	12	9	Sama
14	11650015	18	7	18	7	Sama
15	11650040	0	2	0	2	Sama
16	11650108	1	2	1	2	Sama
17	12650069	0	2	0	2	Sama
18	12650098	0	2	0	2	Sama
19	12650083	1	9	1	9	Sama
20	12650079	20	24	20	24	Sama
21	12650088	0	3	0	3	Sama
22	13650140	10	36	10	36	Sama
23	13650075	0	2	0	2	Sama
24	13650074	0	2	0	2	Sama
25	13650131	0	2	0	2	Sama
26	13650106	1	18	1	18	Sama
27	14650002	0	43	0	43	Sama
28	14650093	0	32	0	32	Sama
29	14650061	0	28	0	28	Sama
30	13650036	0	7	0	7	Sama
31	13650002	1	36	1	36	Sama
32	13650001	1	37	1	37	Sama

33	14650001	0	53	0	53	Sama
34	14650060	0	53	0	53	Sama
35	12650060	5	20	5	20	Sama
36	12650005	3	7	3	7	Sama
37	11650004	0	1	0	1	Sama
38	11650056	3	2	3	2	Sama
39	10650060	2	1	2	1	Sama
40	09650005	10	10	10	10	Sama
41	12650104	2	3	1	3	Tidak Sama

4.6 Hasil Evaluasi dan Pembahasan

Untuk pembahasa ini, uji coba dilakukan pada sistem informasi Pengukuran Kinerja Akademik Mahasiswa Teknik Informatika UIN Maliki Malang Menggunakan Metode Rule Based Pendekatan Forward Chaining. Uji coba dilakukan untuk mengetahui aplikasi yang dapat dibuat dapat menampilkan mata kuliah yang belum lulus maupun mata kuliah yang belum diambil secara tepat atau tidak.

Dari hasil uji coba pada tabel sebelumnya, dilakukan evaluasi dari seluruh data untuk mengetahui aplikasi yang dibuat sesuai dengan tujuan penelitian atau tidak. Uji coba dilakukan menggunakan 41 data sampel Hasil Studi, akurasi dinyatakan berdasarkan besarnya kesalahan relatif. Untuk menghitung berapa akurasi yang dihasilkan menggunakan rumus seperti di bawah ini (Nasution, 2017):

Uji coba data yang sesuai: 40

Uji coba data yang tidak sesuai: 1

$$\text{Akurasi} = \frac{\text{Jumlah data yang sesuai}}{\text{Jumlah data yang diuji}} \times 100\% = \frac{40}{41} \times 100\% = 97.56\%$$

$$\text{Error} = \frac{\text{Jumlah data yang tidak sesuai}}{\text{Jumlah data yang diuji}} \times 100\% = \frac{1}{41} \times 100\% = 2.43\%$$

4.7 Integrasi

Proses pengukuran kinerja akademik yang dilakukan merupakan suatu hal untuk memudahkan urusan orang lain. Seorang Muslim hendaknya berupaya untuk membantu Muslim lainnya. Membantu bisa dengan ilmu, harta, bimbingan, nasihat, saran yang baik, dengan tenaga dan lainnya.

Seorang muslim hendaknya berupaya menghilangkan kesulitan atau penderitaan Muslim lainnya. Bila seorang muslim Muslim membantu Muslim lainnya dengan ikhlas, maka Allah Azza wa Jalla akan memberikan balasan terbaik yaitu dilepaskan dari kesulitan terbesar dan terberat yaitu kesulitan pada hari Kiamat. Oleh karena itu, seorang Muslim mestinya tidak bosan membantu sesama Muslim. Semoga Allah Azza wa Jalla akan menghilangkan kesulitan kita pada hari kiamat.

Dalam hadits Ibnu ‘Umar Radhiyallahu anhumadisebutkan Nabi Shallallahu ‘alaihi wa sallam bersabda :

وَمَنْ كَانَ فِي حَاجَةٍ أَخِيهِ كَانَ اللَّهُ فِي حَاجَتِهِ

“...Dan barangsiapa menolong kebutuhan saudaranya, maka Allâh senantiasa menolong kebutuhannya.”

Dalam memudahkan urusan orang lain dijelaskan di potongan surat Al-Maidah ayat 2, Allah SWT bersabda:

وَتَعَاوَنُوا عَلَى الْبِرِّ وَالتَّقْوَىٰ وَلَا تَعَاوَنُوا عَلَى الْإِثْمِ وَالْعُدْوَانِ

“...Dan tolong-menolonglah kamu dalam (mengerjakan) kebajikan dan takwa, dan jangan tolong-menolong dalam berbuat dosa dan permusuhan...”

Dan untuk tafsir surat Al-Maidah ayat 2 yaitu:

Hai orang-orang yang beriman, janganlah kalian melanggar syiar-syiar Allah seperti manasik haji pada waktu ihram sebelum tahallul ('berhalal' dengan cara mencukur rambut) dan hukum-hukum syariat yang lainnya. Jangan melanggar kehormatan bulan-bulan haram dengan mengobarkan api peperangan, dan jangan pula menghalangi binatang yang dikhususkan untuk dibawa ke Bayt Allâh (Baitullah, Ka'bah) dengan merampas atau menghalanginya untuk sampai ke tempatnya. Jangan melepas kalung-kalung yang ada pada leher binatang sebagai tanda bahwa binatang itu akan dibawa ke Bayt Allâh untuk disembelih pada musim haji, dan jangan pula menghalangi orang-orang yang pergi ke Bayt Allâh dengan maksud mencari karunia dan keridaan-Nya. Jika kalian selesai melaksanakan ihram kemudian melakukan tahallul, maka kalian boleh berburu. Janganlah kebencian kalian kepada kaum yang menghalangi kalian pergi ke al-Masjid al-Haram, mendorong kalian untuk memusuhi mereka. Hendaknya kalian, wahai orang-orang Mukmin, saling menolong¹ alam berbuat baik dan dalam melaksanakan semua bentuk ketaatan dan jangan saling menolong dalam berbuat kemaksiatan dan melanggar ketentuan-ketentuan Allah. Takutlah hukuman dan siksa Allah, karena siksa-Nya amat kejam bagi orang-orang yang menentang-Nya. (Muhammad Ali Al-Sabuni, 1999)

¹Ayat ini menunjukkan bahwa al-Qur'ân telah terlebih dahulu beberapa ratus tahun menganjurkan konsep kerjasama dalam kebaikan, dibanding semua undang-undang positif yang ada.

BAB V

PENUTUP

5.1 Kesimpulan

Bab ini menjelaskan dari uji coba serta evaluasi terhadap penelitian yang telah dilakukan. Uji coba ditujukan untuk mengetahui sejauh mana pengukuran serta tingkat akurasi pada saat uji coba. Dari hasil uji coba didapatkan kesimpulan sebagai berikut:

1. Aplikasi yang dibuat dapat mengimplementasikan pengukuran kinerja akademik menggunakan metode *rule based* pendekatan *forward chaining* berbasis *java* berdasarkan hasil studi mahasiswa sesuai format *web* akademik bagian Transkrip Nilai.
2. Data yang diuji coba sebanyak 41 data mahasiswa jurusan teknik informatika UIN Maliki Malang, dari uji coba yang telah dilakukan menghasilkan tingkat akurasi sebesar 97,56%, dengan tingkat *error* 2,44%.

5.2 Saran

Penulis berharap penelitian selanjutnya dapat ditingkatkan dengan penambahan kriteria dan sub kriteria yang lebih spesifik. Pengukuran Kinerja Akademik Mahasiswa Teknik Informatika UIN Maliki Malang tidak lepas dari kekurangan yang dapat diperbaiki atau disempurnakan dalam penelitian selanjutnya. Ide dan saran yang dapat dilakukan yaitu:

1. Pencangkupan data yang digunakan diperluas agar tidak hanya mahasiswa teknik informatika saja yang dapat melakukan uji coba.

2. Penambahan metode yang digunakan untuk optimalisasi pengukuran kinerja akademik.



DAFTAR PUSTAKA

- Al-ajlan, A. (2015). *The comparison between forward and backward chaining*. International Journal of Machine Learning and Computing. 5. 2. 106-113.
- Al-Sabuni, Muhammad Ali. (1999). *Safwah al-Tafasir*. Jakarta: Daar al-Kutub al-Islamiyah. 326.
- Durkin, J., & Durkin, J. (1994). *Expert systems: design and development* (pp. 1-800). New York: Macmillan.
- Engin, G., Aksoyer, B., Avdagic, M., Bozanli, D., Hanay, U., Maden, D., dan Ertek, G. (2014). *Rule-based expert systems for supporting university students*. Procedia Computer Science. 31. 22-31.
- Eshete, A. B. (2009). *Integrated case based and rule based reasoning for decision support*. Norwegian University of Science and Technology – Department of Computer and Information Science. July. 1-100.
- Giaratanno, J., & Riley, G. (2005). *Expert systems principles and programming*. Boston: PWS-Kent Publishing Company.
- Ignazio, J.P. (1991). *Introduction to expert systems : the development and implementation of rule based expert system*. New York: McGraw-Hill International Editions.
- Kusumadewi, S. (2003). *Artificial Intelligence (Teknik dan aplikasinya)*. Yogyakarta: Graha Ilmu.
- Lengyel, L. (2015). *Validating Rule-based Algorithms*. Department of Automation and Applied Informatics – Budapest University of Technology and Economics.
- McLeod, R. J. (1995). *Management information system. Sixth Edition*. New Jersey: Prentice Hall Inc.
- Nasution, MD. (2017). *Pengembangan Bahan Ajar Metode Numerik Dengan Pendekatan Metakognitif Berbantuan Matlab* –Pendidikan Matematika, Universitas Muhammadiyah Sumatera Utara.
- Sasmito, G. W., Surarso, B., & Sugiharo, A. (2011). *Application expert system of forward chaining and the rule based reasoning for simulation diagnose pest and disease red onion and chili plant*. Proceedings of The 1st International Conference on Information Systems For Business Competitiveness (ICISBC). 392-398.
- Sharma, T., Tiwari, N., & Kelkar, D. (2012). *Study of difference between forward and backward reasoning*. International Journal of Emerging Technology and Advanced Endengineering. 2. 10. 271-273.

- Supartini, W. & Hindarto. (2016). *Sistem Pakar Berbasis Web Dengan Metode Forward Chaining Dalam Mendiagnosis Dini Penyakit Tuberkulosis di JawaTimur*. KINETIK. 1. 3. 148-149.
- Turban, E. & Aronson, E. (2001). *Decision support systems and intelligence system*. Sixth Edition. New Jersey: Pearson Education.
- Wagner, W.P., Otto, J., & Chung, Q. B. (2002). *Knowledge acquisition for expert systems in accounting and financial problem domains*. Knowledge-Based Systems. 15. 8. 439-447.



LAMPIRAN

Tabel Pemasaran

1	765101	Filsafat Pancasila	3	1400101	Pancasila dan Kewarganegaraan	2
2	865102	Bahasa Indonesia	2	1400102	Bahasa Indonesia	2
3	765103	Bahasa Inggris I	3	1400103	Bahasa Inggris I	3
4	765104	Bahasa Inggris II	3	1400104	Bahasa Inggris II	3
5	765105	Ilmu Sosial Dasar	1	1400105	Ilmu Sosial Budaya Dasar	2
6	765106	Ilmu Budaya Dasar	1	1400105	Ilmu Sosial Budaya Dasar	2
7	765107	Filsafat Ilmu	2	1400107	Filsafat Ilmu	2
8	765108	Studi al-Qur'an	2	1400108	Studi al-Qur'an dan al-Hadits	2
9	765109	Studi al-Hadits	2	1400108	Studi al-Qur'an dan al-Hadits	2
10	765110	Studi Fiqih	2	1400109	Studi Fiqh	2
11	765111	Tasawuf	2	1400110	Teosofi	2
12	765113	Teologi Islam	2	1400110	Teosofi	2
13	765112	Sejarah Peradaban Islam	2	1400111	Sejarah Peradaban Islam	2
14	765114	Bahasa Arab (Maharat al-Istima' I)	1	1400112	Bahasa Arab (Maharat al-Istima' I)	1
15	765115	Bahasa Arab (Maharat al-Kalam' I)	2	1400114	Bahasa Arab (Maharat al-Kalam' I)	2
16	765116	Bahasa Arab (Maharat al-Qira'ah I)	2	1400115	Bahasa Arab (Maharat al-Qira'ah I)	2
17	765117	Bahasa Arab (Maharat al-Kitabah I)	1	1400113	Bahasa Arab (Maharat al-Kitabah I)	1
18	765118	Bahasa Arab (Maharat al-Istima' II)	1	1400116	Bahasa Arab (Maharat al-Istima' II)	1
19	765119	Bahasa Arab (Maharat al-Kalam' II)	1	1400118	Bahasa Arab (Maharat al-Kalam' II)	2
20	765120	Bahasa Arab (Maharat al-Qira'ah II)	2	1400119	Bahasa Arab (Maharat al-Qira'ah II)	2
21	765121	Bahasa Arab (Maharat al-Kitabah II)	2	1400117	Bahasa Arab (Maharat al-Kitabah II)	1
22	865122	Tarbiyah Ulul Albab	1	0	0	0
23	1200101	FILSAFAT PANCASILA	3	1400101	Pancasila dan Kewarganegaraan	2
24	1200102	BAHASA INDONESIA	2	1400102	Bahasa Indonesia	2
25	1200103	BAHASA INGGRES I	3	1400103	Bahasa Inggris I	3
26	1200104	BAHASA INGGRES II	3	1400104	Bahasa Inggris II	3
27	1200106	ILMU SOSIAL BUDAYA DASAR	2	1400105	Ilmu Sosial Budaya Dasar	2
28	1200107	PENGANTAR FILSAFAT ILMU	2	1400107	Filsafat Ilmu	2
29	1200108	STUDI QURAN	2	1400108	Studi al-Qur'an dan al-Hadits	2
30	1200109	STUDI HADITS	2	1400108	Studi al-Qur'an dan al-Hadits	2
31	1200110	STUDI FIQH	2	1400109	Studi Fiqh	2
32	1200111	TASAWUF	2	1400110	Teosofi	2
33	1200113	TEOLOGI ISLAM	2	1400110	Teosofi	2
34	1200112	SEJARAH PERADABAN ISLAM	2	1400111	Sejarah Peradaban Islam	2
35	1200114	MAHARAT AL-ISTIMA I	1	1400112	Bahasa Arab (Maharat al-Istima' I)	1
36	1200115	MAHARAT AL-KALAM I	2	1400114	Bahasa Arab (Maharat al-Kalam' I)	2
37	1200116	MAHARAT AL-QIRAAH I	2	1400115	Bahasa Arab (Maharat al-Qira'ah I)	2
38	1200117	MAHARAT AL-KITABAH I	1	1400113	Bahasa Arab (Maharat al-Kitabah I)	1
39	1200118	MAHARAT AL-ISTIMA II	1	1400116	Bahasa Arab (Maharat al-Istima' II)	1
40	1200119	MAHARAT AL-KALAM II	1	1400118	Bahasa Arab (Maharat al-Kalam' II)	2
41	1200120	MAHARAT AL-QIRAAH II	2	1400119	Bahasa Arab (Maharat al-Qira'ah II)	2
42	1200121	MAHARAT AL-KITABAH II	2	1400117	Bahasa Arab (Maharat al-Kitabah II)	1
43	1200122	TARBIYAH ULUL ALBAB	1	0	0	0
44	1400101	Pancasila dan Kewarganegaraan	2	1400101	Pancasila dan Kewarganegaraan	2
45	1400102	Bahasa Indonesia	2	1400102	Bahasa Indonesia	2
46	1400103	Bahasa Inggris I	3	1400103	Bahasa Inggris I	3
47	1400104	Bahasa Inggris II	3	1400104	Bahasa Inggris II	3
48	1400105	Ilmu Sosial Budaya Dasar	2	1400105	Ilmu Sosial Budaya Dasar	2
49	1400107	Filsafat Ilmu	2	1400107	Filsafat Ilmu	2
50	1400108	Studi al-Qur'an dan al-Hadits	2	1400108	Studi al-Qur'an dan al-Hadits	2
51	1400109	Studi Fiqh	2	1400109	Studi Fiqh	2
52	1400110	Teosofi	2	1400110	Teosofi	2
53	1400111	Sejarah Peradaban Islam	2	1400111	Sejarah Peradaban Islam	2
54	1400112	Bahasa Arab (Maharat al-Istima' I)	1	1400112	Bahasa Arab (Maharat al-Istima' I)	1
55	1400114	Bahasa Arab (Maharat al-Kalam' I)	2	1400114	Bahasa Arab (Maharat al-Kalam' I)	2
56	1400115	Bahasa Arab (Maharat al-Qira'ah I)	2	1400115	Bahasa Arab (Maharat al-Qira'ah I)	2
57	1400113	Bahasa Arab (Maharat al-Kitabah I)	1	1400113	Bahasa Arab (Maharat al-Kitabah I)	1
58	1400116	Bahasa Arab (Maharat al-Istima' II)	1	1400116	Bahasa Arab (Maharat al-Istima' II)	1
59	1400118	Bahasa Arab (Maharat al-Kalam' II)	2	1400118	Bahasa Arab (Maharat al-Kalam' II)	2
60	1400119	Bahasa Arab (Maharat al-Qira'ah II)	2	1400119	Bahasa Arab (Maharat al-Qira'ah II)	2
61	1400117	Bahasa Arab (Maharat al-Kitabah II)	1	1400117	Bahasa Arab (Maharat al-Kitabah II)	1
62	765208	Pengantar Teknologi Informasi	2	1565001	Foundation of Computing	2
63	765201	Matematika Dasar	3	1565002	Calculus	3
64	765209	Logika dan Algoritma	2	1565003	Algorithm & Programming 1	3
65	765204	Matematika Diskrit	3	1565004	Discrete Mathematics	3
66	765203	Aljabar Linier dan Matriks	3	1565005	Linear Algebra	2
67	765210	Struktur Data	3	1565006	Data Structure	3
68	765211	Dasar-dasar Pemrograman	3	1565007	Algorithm & Programming 2	2
69	765216	Rangkaian Digital	2	1565008	Digital Electronic	3
70	765206	Statistika	2	1565009	Statistics	2
71	765212	Pemrograman Berorientasi Obyek	3	1565010	Object Oriented Programming	3
72	765214	Desain Basisdata	3	1565011	Database	3
73	765311	Manajemen Basisdata	2	1565011	Database	3
74	765219	Organisasi & Arsitektur Komputer	3	1565012	Computer System	3
75	765217	Komunikasi Data	3	1565012	Computer System	3
76	765205	Metode Numerik	2	1565013	Numerical Methods	2
77	765306	Pemrograman Web	3	1565014	Web Programming	3
78	765309	Rekayasa Perangkat Lunak	2	1565015	Software Engineering	3

79	765312	Analisis dan Perancangan Sistem Informasi	3	1565015	Software Engineering	3
80	765302	Grafika Komputer	3	1565016	Computer Graphic	2
81	765218	Jaringan Komputer	3	1565017	Computer Network	3
82	765301	Cerdasan Buatan	3	1565018	Artificial Intelligence	3
83	765303	Pengolahan Citra	3	1565019	Computer Vision	3
84	765313	Komputasi Bergerak	3	1565020	Mobile Programming	2
85	765213	Sistem Informasi	2	1565021	Information System	3
86	765305	Pemrograman Multimedia	3	1565022	Multimedia & Game Programming	3
87	765307	Sistem Terdistribusi	3	1565023	Distributed System & Security	3
88	765207	Metodologi Penelitian	2	1565024	Research Methodology	2
89	765314	Sistem Informasi Geografis	3	1565025	Geographical Information System	2
90	765202	Matematika Lanjut	3	1565026	Operation Research	2
91	765310	Interaksi Manusia dan Komputer	2	1565027	Human Computer Interaction	3
92	765215	Sistem Operasi	2	1565028	Operating System	3
93	765401	Kewirausahaan	2	1565029	Technopreneurship	2
94	765404	Manajemen Proyek Perangkat Lunak	2	1565030	Project Management	2
95	765405	Praktikum Dasar-dasar Pemrograman	1	1565031	Algorithm & Programming 1 Practicum	1
96	765405	Praktikum Dasar-dasar Pemrograman	1	1565032	Data Structure Practicum	1
97	765409	Praktikum Rangkaian Digital	1	1565033	Digital Electronic Practicum	1
98	765410	Praktikum Pemrog. Berorientasi Obyek	1	1565034	Object Oriented Programming Practicum	1
99	765410	Praktikum Pemrog. Berorientasi Obyek	1	1565035	Mobile Programming Practicum	1
100	765406	Praktikum Desain Basis Data	1	1565036	Database Practicum	1
101	765408	Praktikum Komunikasi Data	1	1565037	Computer System Practicum	1
102	765411	Praktikum Pemrograman Web	1	1565038	Web Programming Practicum	1
103	765415	Praktikum Analisis & Perancangan Sistem	1	1565039	Software Engineering Practicum	1
104	765415	Praktikum Analisis & Perancangan Sistem	1	1565040	Information System Practicum	1
105	765414	Praktikum Grafika Komputer	1	1565041	Computer Graphic Practicum	1
106	765407	Praktikum Jaringan Komputer	1	1565042	Computer Network Practicum	1
107	765413	Praktikum Pemrograman Multimedia	1	1565043	Multimedia & Game Programming Practicum	1
108	765412	Praktikum Sistem Terdistribusi	1	1565044	Distributed System Practicum	1
109	765416	Praktikum Sistem Informasi Geografis	1	1565045	Geographical Information System Practicum	1
110	765501	PKL	4	1565046	Internship	4
111	765502	Seminar	2	1565047	Research Proposal	2
112	765503	Skripsi	6	1565048	Undergraduate Thesis	6
113	765315	Sistem Informasi Cerdas	3	1565049	SOFT COMPUTING (OPTIONAL SUBJECT 1)	3
114	765402	Isu Profesional dan Proyek Group	2	1565057	Social Issues and Professional Practice (OPTIONAL SUBJECT 1)	3
115	765403	Etika Profesi	2	1565057	Social Issues and Professional Practice (OPTIONAL SUBJECT 1)	3
116	765304	Pengolahan Suara	3	1565058	Voice Processing (OPTIONAL SUBJECT 2)	3
117	765308	Keamanan Jaringan dan Sistem Komputer	3	1565059	Network Security (OPTIONAL SUBJECT 2)	3
118				1565050	VISUALIZATION & MODELING (OPTIONAL SUBJECT 2)	3
119				1565051	SOFTWARE QUALITY (OPTIONAL SUBJECT 1)	3
120				1565052	SOFTWARE MANAGEMENT (OPTIONAL SUBJECT 2)	3
121				1565053	ENTERPRISE ARCHITECTURE (OPTIONAL SUBJECT 1)	3
122				1565054	IT GOVERNANCE (OPTIONAL SUBJECT 2)	3
123				1565055	SCENARIO DESIGN (OPTIONAL SUBJECT 1)	3
124				1565056	CREATIVE GAME & ANIMATION (OPTIONAL SUBJECT 2)	3
125	1565001	Foundation of Computing	2	1565001	Foundation of Computing	2
126	1565002	Calculus	3	1565002	Calculus	3
127	1565003	Algorithm & Programming 1	3	1565003	Algorithm & Programming 1	3
128	1565004	Discrete Mathematics	3	1565004	Discrete Mathematics	3
129	1565005	Linear Algebra	2	1565005	Linear Algebra	2
130	1565006	Data Structure	3	1565006	Data Structure	3
131	1565007	Algorithm & Programming 2	2	1565007	Algorithm & Programming 2	2
132	1565008	Digital Electronic	3	1565008	Digital Electronic	3
133	1565009	Statistics	2	1565009	Statistics	2
134	1565010	Object Oriented Programming	3	1565010	Object Oriented Programming	3
135	1565011	Database	3	1565011	Database	3
136	1565011	Database	3	1565011	Database	3
137	1565012	Computer System	3	1565012	Computer System	3
138	1565012	Computer System	3	1565012	Computer System	3
139	1565013	Numerical Methods	2	1565013	Numerical Methods	2
140	1565014	Web Programming	3	1565014	Web Programming	3
141	1565015	Software Engineering	3	1565015	Software Engineering	3
142	1565015	Software Engineering	3	1565015	Software Engineering	3
143	1565016	Computer Graphic	2	1565016	Computer Graphic	2
144	1565017	Computer Network	3	1565017	Computer Network	3
145	1565018	Artificial Intelligence	3	1565018	Artificial Intelligence	3
146	1565019	Computer Vision	3	1565019	Computer Vision	3
147	1565020	Mobile Programming	2	1565020	Mobile Programming	2
148	1565021	Information System	3	1565021	Information System	3
149	1565022	Multimedia & Game Programming	3	1565022	Multimedia & Game Programming	3
150	1565023	Distributed System & Security	3	1565023	Distributed System & Security	3
151	1565024	Research Methodology	2	1565024	Research Methodology	2
152	1565025	Geographical Information System	2	1565025	Geographical Information System	2
153	1565026	Operation Research	2	1565026	Operation Research	2
154	1565027	Human Computer Interaction	3	1565027	Human Computer Interaction	3
155	1565028	Operating System	3	1565028	Operating System	3
156	1565029	Technopreneurship	2	1565029	Technopreneurship	2
157	1565030	Project Management	2	1565030	Project Management	2
158	1565031	Algorithm & Programming 1 Practicum	1	1565031	Algorithm & Programming 1 Practicum	1
159	1565032	Data Structure Practicum	1	1565032	Data Structure Practicum	1
160	1565033	Digital Electronic Practicum	1	1565033	Digital Electronic Practicum	1

161	1565034	Object Oriented Programming Practicum	1	1565034	Object Oriented Programming Practicum	1
162	1565035	Mobile Programming Practicum	1	1565035	Mobile Programming Practicum	1
163	1565036	Database Practicum	1	1565036	Database Practicum	1
164	1565037	Computer System Practicum	1	1565037	Computer System Practicum	1
165	1565038	Web Programming Practicum	1	1565038	Web Programming Practicum	1
166	1565039	Software Engineering Practicum	1	1565039	Software Engineering Practicum	1
167	1565040	Information System Practicum	1	1565040	Information System Practicum	1
168	1565041	Computer Graphic Practicum	1	1565041	Computer Graphic Practicum	1
169	1565042	Computer Network Practicum	1	1565042	Computer Network Practicum	1
170	1565043	Multimedia & Game Programming Practicum	1	1565043	Multimedia & Game Programming Practicum	1
171	1565044	Distributed System Practicum	1	1565044	Distributed System Practicum	1
172	1565045	Geographical Information System Practicum	1	1565045	Geographical Information System Practicum	1
173	1565046	Internship	4	1565046	Internship	4
174	1565047	Research Proposal	2	1565047	Research Proposal	2
175	1565048	Undergraduate Thesis	6	1565048	Undergraduate Thesis	6
176	1565049	SOFT COMPUTING (OPTIONAL SUBJECT 1)	3	1565049	SOFT COMPUTING (OPTIONAL SUBJECT 1)	3
177	1565057	Social Issues and Professional Practice (OPTIONAL SUBJECT 1)	3	1565057	Social Issues and Professional Practice (OPTIONAL SUBJECT 1)	3
178	1565057	Social Issues and Professional Practice (OPTIONAL SUBJECT 1)	3	1565057	Social Issues and Professional Practice (OPTIONAL SUBJECT 1)	3
179	1565058	Voice Processing (OPTIONAL SUBJECT 2)	3	1565058	Voice Processing (OPTIONAL SUBJECT 2)	3
180	1565059	Network Security (OPTIONAL SUBJECT 2)	3	1565059	Network Security (OPTIONAL SUBJECT 2)	3
181	1565050	VISUALIZATION & MODELING (OPTIONAL SUBJECT 2)	3	1565050	VISUALIZATION & MODELING (OPTIONAL SUBJECT 2)	3
182	1565051	SOFTWARE QUALITY (OPTIONAL SUBJECT 1)	3	1565051	SOFTWARE QUALITY (OPTIONAL SUBJECT 1)	3
183	1565052	SOFTWARE MANAGEMENT (OPTIONAL SUBJECT 2)	3	1565052	SOFTWARE MANAGEMENT (OPTIONAL SUBJECT 2)	3
184	1565053	ENTERPRISE ARCHITECTURE (OPTIONAL SUBJECT 1)	3	1565053	ENTERPRISE ARCHITECTURE (OPTIONAL SUBJECT 1)	3
185	1565054	IT GOVERNANCE (OPTIONAL SUBJECT 2)	3	1565054	IT GOVERNANCE (OPTIONAL SUBJECT 2)	3
186	1565055	SCENARIO DESIGN (OPTIONAL SUBJECT 1)	3	1565055	SCENARIO DESIGN (OPTIONAL SUBJECT 1)	3
187	1565056	CREATIVE GAME & ANIMATION (OPTIONAL SUBJECT 2)	3	1565056	CREATIVE GAME & ANIMATION (OPTIONAL SUBJECT 2)	3



Lampiran Kode Sumber Browse

```
JFileChooser jfc = new JFileChooser();
FileNameExtensionFilter filter = new FileNameExtensionFilter("xls atau xlsx
file", "xls", "xlsx");
private String pathFile;
private String pathFolder;

jfc.setDialogTitle("Pilih file excel...");
jfc.setAcceptAllFileFilterUsed(false);
jfc.addChoosableFileFilter(filter);
int returnValue = jfc.showOpenDialog(null);
if (returnValue == JFileChooser.APPROVE_OPTION) {
    File selectedFile = jfc.getSelectedFile();
    lblFile.setText(selectedFile.getName());
    pathFile = selectedFile.getPath();
    pathFolder = selectedFile.getParent();
}
```

Lampiran Kode Sumber Cek Update

```
try {
    File p = new File(pemasaran);
    File p1 = new File(pemasaran1);
    File p2 = new File(pemasaran2);
    URL o = new URL(online);
    URL o1 = new URL(online1);
    URL o2 = new URL(online2);
    if ((p.length() == getFileSize(o)) && (p1.length() == getFileSize(o1))
    && (p2.length() == getFileSize(o2))) {
        popup("Tidak ada update terbaru.");
    } else {
        try {
            InputStream in = new URL(online).openStream();
            Files.copy(in, Paths.get(pemasaran),
            StandardCopyOption.REPLACE_EXISTING);
            InputStream in1 = new URL(online1).openStream();
            Files.copy(in1, Paths.get(pemasaran1),
            StandardCopyOption.REPLACE_EXISTING);
            InputStream in2 = new URL(online2).openStream();
            Files.copy(in2, Paths.get(pemasaran2),
            StandardCopyOption.REPLACE_EXISTING);
            popup("Berhasil update database.");
        } catch (IOException e) {
            popup(e.getMessage());
        }
    }
} catch (MalformedURLException ex) {
    popup(ex.getLocalizedMessage());
}
```

Kode Sumber Mesin.java

```
package akademik;

import java.io.*;
import java.net.*;
import java.nio.file.*;
import java.sql.SQLException;
import java.text.*;
import java.util.*;
import javax.swing.JOptionPane;
import org.apache.poi.EncryptedDocumentException;
import org.apache.poi.hssf.usermodel.HSSFWorkbook;
import org.apache.poi.ss.usermodel.*;

public class Mesin {

    private final String pemasaran = "Rule Konversi.xlsx";
    private final String pemasaran1 = "mkpilihan1.txt";
    private final String pemasaran2 = "mkpilihan2.txt";
    private final String online =
"https://github.com/ddystrn/akademik/raw/master/Rule%20Konversi.xlsx";
    private final String online1 =
"https://raw.githubusercontent.com/ddystrn/akademik/master/mkpilihan1.txt";
    private final String online2 =
"https://raw.githubusercontent.com/ddystrn/akademik/master/mkpilihan2.txt";
    private String nim;
    private String pathFile;
    private String pathFolder;
    NumberFormat nf = NumberFormat.getInstance();

    //awal proses konversi
    public void jalankanMethod() {
        clearAll();
    }

    public void clearAll() {
        try {
            Koneksi k = new Koneksi();
            String[] clearAll = {
                "DELETE FROM khs",
                "DELETE FROM khslulus",
                "DELETE FROM khstidaklulus",
                "DELETE FROM konversi",
                "DELETE FROM mahasiswa",
                "DELETE FROM pemasaran",
                "DELETE FROM pemasaranmk1",
                "DELETE FROM pemasaranmk2",
                "DELETE FROM pemasarannew",
                "DELETE FROM skipmkpilihan"
            };
            for (int i = 0; i < clearAll.length; i++) {
                k.eksekusi(clearAll[i]);
            }
            k.tutup();
            inputExcelKeSQL();
        } catch (ClassNotFoundException | SQLException ex) {
            popup(ex.getMessage());
        }
    }

    public void inputExcelKeSQL() {
        try {
            System.out.println("==== Mulai input excel khs ke sql =====");
        }
    }
}
```

```

Sheet s = null;
Koneksi k = new Koneksi();
File file = new File(pathFile);
try (FileInputStream input = new FileInputStream(file)) {
    System.out.println("Path File: " + file);
    try {
        Workbook wb = WorkbookFactory.create(input);
        s = wb.getSheetAt(0);
    } catch (IOException | EncryptedDocumentException e) {
        popup(e.getMessage());
    }
}
Row row;
Row kosong;
Row nimrow = s.getRow(0);
Row namarow = s.getRow(1);

String nama = null;
double ipk = 0;
int last = -1;
int startFrom = 5;

System.out.println("Last Row: " + (s.getLastRowNum() + 1));
for (int i = 0; i <= s.getLastRowNum(); i++) {
    kosong = s.getRow(i);
    if (kosong == null || kosong.getCell(1) == null ||
getCellValueAsString(kosong.getCell(1)).isEmpty()) {
        last++;
    }
}
Row ipkrow = s.getRow((s.getLastRowNum() - last) + 2);
try {
    nim = nimrow.getCell(1).getStringCellValue().replace(":",
", ");
    nama =
((namarow.getCell(1).getStringCellValue()).replace(":", " ").replace("'",
"`"));
    ipk =
nf.parse(getCellValueAsString(ipkrow.getCell(3))).doubleValue();
} catch (ParseException e) {
    popup(e.getMessage());
}

String sqlmahasiswa = "INSERT INTO MAHASISWA VALUES('" + nim
+ "','" + nama + "','" + ipk + "')";
k.eksekusi(sqlmahasiswa);

for (int i = startFrom - 1; i <= s.getLastRowNum() - last;
i++) {
    row = s.getRow(i);

    int kode_mk = (int) row.getCell(1).getNumericCellValue();
    String nama_mk =
(row.getCell(2).getStringCellValue()).replace("'", "`");
    int sks = (int) row.getCell(3).getNumericCellValue();
    double nilai_angka =
nf.parse(getCellValueAsString(row.getCell(4))).doubleValue();
    String nilai_huruf =
row.getCell(5).getStringCellValue();
    double sks_nilai_angka =
nf.parse(getCellValueAsString(row.getCell(6))).doubleValue();

    String sql = "INSERT INTO KHS VALUES('" + nim + "','"
+ kode_mk + "','" + nama_mk + "','" + sks + "','"

```

```

        + nilai_angka + "','" + nilai_huruf + "','"
        + sks_nilai_angka + "')";
        k.eksekusi(sql);
        System.out.println("Import row: " + kode_mk);
    }
    k.tutup();
} catch (ParseException ex) {
    popup(ex.getMessage());
}
System.out.println("Berhasil impor excel khs ke tabel sql");
System.out.println("==== Selesai input excel ke sql =====");
inputPemasaranKeSQL();
} catch (IOException | ClassNotFoundException | SQLException e) {
    popup(e.getMessage());
}
}

public void inputPemasaranKeSQL() {
    try {
        System.out.println("==== Mulai input excel pemasaran ke sql
        =====");
        Sheet s = null;
        Koneksi k = new Koneksi();
        File file = new File(pemasaran);
        FileInputStream input = null;
        try {
            input = new FileInputStream(file);
        } catch (FileNotFoundException ex) {
            popup(ex.getMessage());
        }
        System.out.println("Path File: " + file);
        try {
            Workbook wb = WorkbookFactory.create(input);
            s = wb.getSheetAt(0);
        } catch (IOException | EncryptedDocumentException e) {
            popup(e.getMessage());
        }
        Row row;
        int startFrom = 1;
        for (int i = startFrom - 1; i <= s.getLastRowNum(); i++) {
            row = s.getRow(i);
            int no = (int) row.getCell(0).getNumericCellValue();
            int kode_mk = (int) row.getCell(1).getNumericCellValue();
            String nama_mk =
            (getCellValueAsString(row.getCell(2))).replace("'", "`");
            int sks = (int) row.getCell(3).getNumericCellValue();
            int kode_mk_alias =
            row.getCell(4).getNumericCellValue();
            String nama_mk_alias =
            (getCellValueAsString(row.getCell(5))).replace("'", "`");
            int sks_alias = (int) row.getCell(6).getNumericCellValue();

            String sql = "INSERT INTO PEMASARAN VALUES('" + no + "','" +
            kode_mk
            + "','" + nama_mk + "','" + sks + "','" +
            kode_mk_alias
            + "','" + nama_mk_alias + "','" + sks_alias + "')";
            k.eksekusi(sql);
            System.out.println("Import row: " + no);
        }
        k.tutup();
    }
}

```

```

        input.close();
    } catch (IOException ex) {
        popup(ex.getMessage());
    }
    System.out.println("Berhasil impor excel pemasaran ke tabel
sql");
    System.out.println("===== Selesai input excel pemasaran ke sql
=====");
    lulusKHS();
} catch (ClassNotFoundException | SQLException ex) {
    popup(ex.getMessage());
}
}

public void lulusKHS() {
    try {
        Koneksi k = new Koneksi();
        String insertLulus = "INSERT INTO khslulus SELECT * FROM ( SELECT"
+ " k.kode_mk, k.nilai_angka, k.nilai_huruf,
p.kode_mk_alias"
+ " FROM khs k JOIN pemasaran p ON k.kode_mk = p.kode_mk"
+ " WHERE k.nilai_angka >= '2.0' GROUP BY k.kode_mk ORDER
BY"
+ " p.kode_mk_alias ASC, k.nilai_angka DESC,
k.nilai_huruf"
+ " ASC ) as lulus GROUP BY kode_mk_alias";
        k.eksekusi(insertLulus);
        String insertTidakLulus = "INSERT INTO khstidaklulus SELECT"
+ " k.kode_mk, k.nilai_angka, k.nilai_huruf,"
+ " p.kode_mk_alias FROM khs k JOIN pemasaran p ON"
+ " k.kode_mk = p.kode_mk WHERE k.nilai_angka < '2.0'"
+ " GROUP BY k.kode_mk";
        k.eksekusi(insertTidakLulus);
        String updateNilaiTidakLulus = "UPDATE khstidaklulus t SET
t.nilai_angka"
+ " = (SELECT l.nilai_angka FROM khslulus l WHERE"
+ " t.kode_mk_alias = l.kode_mk_alias), t.nilai_huruf ="
+ " (SELECT l.nilai_huruf FROM khslulus l WHERE"
+ " t.kode_mk_alias = l.kode_mk_alias) WHERE"
+ " t.kode_mk_alias IN (SELECT kode_mk_alias FROM
khslulus)";
        k.eksekusi(updateNilaiTidakLulus);
        String updateNilaiKHS = "UPDATE khs k SET k.nilai_angka = (SELECT"
+ " t.nilai_angka FROM khstidaklulus t WHERE t.kode_mk ="
+ " k.kode_mk), k.nilai_huruf = (SELECT t.nilai_huruf
FROM"
+ " khstidaklulus t WHERE t.kode_mk = k.kode_mk),"
+ " k.sks_nilai_angka = (k.sks * k.nilai_angka)"
+ " WHERE k.kode_mk IN (SELECT t.kode_mk FROM
khstidaklulus"
+ " t)";
        k.eksekusi(updateNilaiKHS);
        k.tutup();
        khsKeKonversi();
    } catch (ClassNotFoundException | SQLException e) {
        popup(e.getMessage());
    }
}

public void khsKeKonversi() {
    try {
        Koneksi k = new Koneksi();
        String sql = "INSERT INTO konversi SELECT k.nim mhs,"

```

```

        + " p.kode_mk_alias, p.nama_mk_alias, p.sks_alias,"
        + " k.nilai_angka, k.nilai_huruf, k.sks_nilai_angka FROM"
        + " khs k JOIN pemasaran p ON k.kode_mk = p.kode_mk WHERE"
        + " k.nilai_angka < '2.0' AND p.kode_mk_alias != '0'"
        + " GROUP BY p.kode_mk_alias";
        k.eksekusi(sql);
        k.tutup();
        mkPilihan();
    } catch (ClassNotFoundException | SQLException ex) {
        popup(ex.getMessage());
    }
}

public void mkPilihan() {
    try {
        Koneksi k = new Koneksi();
        BufferedReader mkPilihan = null;
        for (int i = 1; i <= 2; i++) {
            try {
                mkPilihan = new BufferedReader(new FileReader("mkpilihan"
+ i + ".txt"));
                String str;
                List<String> l = new ArrayList<>();
                while ((str = mkPilihan.readLine()) != null) {
                    l.add(str);
                }
                String[] pemasaranmk = l.toArray(new String[0]);
                System.out.println("Kode MK Pilihan " + i + " :");
                for (int j = 0; j < pemasaranmk.length; j++) {
                    String inputmkpilihan = "INSERT INTO pemasaranmk" +
i + ""
+ " SELECT * FROM pemasaran WHERE
kode_mk_alias='"
+ pemasaranmk[j] + "'";
                    k.eksekusi(inputmkpilihan);
                    System.out.println(pemasaranmk[j]);
                }
                String selectnilaimkpilihan = "SELECT k.kode_mk,"
+ " k.nilai_angka, p.kode_mk_alias FROM khs k
JOIN"
+ " pemasaranmk" + i + " p ON k.kode_mk =
p.kode_mk WHERE"
+ " k.nilai_angka >= '2.0' GROUP BY k.kode_mk";
                k.select(selectnilaimkpilihan);
                if (!k.tampilData()) {
                    String mkPilihanKeNew = "INSERT INTO pemasarannew
SELECT"
+ " * FROM pemasaranmk" + i + " pmk WHERE"
+ " pmk.kode_mk_alias NOT IN (SELECT
k.kode_mk"
+ " FROM konversi k)";
                    k.eksekusi(mkPilihanKeNew);
                } else {
                    System.out.println("Kode MK Pilihan " + i + " Lulus");
                    for (int j = 0; j < pemasaranmk.length; j++) {
                        String kodeMkPilihan = "INSERT INTO
skipmkpilihan"
+ " VALUES ('" + pemasaranmk[j] + "')";
                        k.eksekusi(kodeMkPilihan);
                    }
                }
            } catch (IOException | SQLException e) {
            }
        }
    }
}

```



```

    }
    k.tutup();
    pemasaranToNew();
} catch (ClassNotFoundException | SQLException e) {
    popup(e.getMessage());
}
}

public void pemasaranToNew() {
    try {
        Koneksi k = new Koneksi();
        String sql = "INSERT INTO pemasarannew SELECT * FROM pemasaran p"
            + " WHERE p.kode_mk_alias NOT IN ( SELECT p.kode_mk_alias"
            + " FROM khs k JOIN pemasaran p ON k.kode_mk = p.kode_mk)"
            + " AND p.kode_mk_alias NOT IN ( SELECT p1.kode_mk_alias"
            + " FROM pemasaranmk1 p1 JOIN pemasaran p ON"
            + "     p1.kode_mk_alias = p.kode_mk_alias) AND
p.kode_mk_alias"
            + " NOT IN ( SELECT p2.kode_mk_alias FROM pemasaranmk2
p2"
            + "     JOIN     pemasaran     p     ON     p2.kode_mk_alias     =
p.kode_mk_alias)"
            + " AND p.kode_mk_alias != '0' GROUP BY p.kode_mk_alias";
        k.eksekusi(sql);
        String sqlKonversi = "INSERT INTO konversi SELECT m.nim_mhs,"
            + " p.kode_mk_alias, p.nama_mk_alias, p.sks_alias, null
as"
            + " nilai_angka, null as nilai_huruf, null as"
            + " sks_nilai_angka FROM pemasarannew p JOIN mahasiswa m"
            + " GROUP BY p.kode_mk_alias";
        k.eksekusi(sqlKonversi);
        k.tutup();
        exportSQLkeXLSX();
    } catch (ClassNotFoundException | SQLException ex) {
        popup(ex.getMessage());
    }
}

public void exportSQLkeXLSX() {
    try {
        Workbook wb = new HSSFWorkbook();
        Koneksi k = new Koneksi();
        Date date = new Date();
        SimpleDateFormat f = new SimpleDateFormat("yyyy-MM-dd HHmmss");
        Sheet s = wb.createSheet("Konversi KHS");

        Row n = s.createRow(0);
        Cell nimHeader = n.createCell(0);
        nimHeader.setCellValue("NIM");
        Cell nimCell = n.createCell(1);

        Row nm = s.createRow(1);
        Cell namaHeader = nm.createCell(0);
        namaHeader.setCellValue("Nama Mahasiswa");
        Cell namaCell = nm.createCell(1);

        Row r = s.createRow(3);
        Cell noHeader = r.createCell(0);
        noHeader.setCellValue("No");
        Cell kodeHeader = r.createCell(1);
        kodeHeader.setCellValue("Kode");
        Cell matkulHeader = r.createCell(2);
        matkulHeader.setCellValue("Nama Mata Kuliah");
    }
}

```

```

Cell sksHeader = r.createCell(3);
sksHeader.setCellValue("SKS");
Cell nilaiHeader = r.createCell(4);
nilaiHeader.setCellValue("Nilai Angka");
Cell hurufHeader = r.createCell(5);
hurufHeader.setCellValue("Nilai Huruf");
Cell sksnilaiHeader = r.createCell(6);
sksnilaiHeader.setCellValue("SKS x Nilai Angka");

String sqlkonversi = "SELECT * FROM konversi WHERE kode_mk NOT IN
("
    + " SELECT kode_mk FROM skipmpilihan )";
k.select(sqlkonversi);
int row = 4;
while (k.tampilData()) {
    int kodemk = k.rs().getInt("kode_mk");
    String namamk = k.rs().getString("nama_mk");
    int sks = k.rs().getInt("sks");
    Double nilaiangka = k.rs().getDouble("nilai_angka");
    String nilaihuruf = k.rs().getString("nilai_huruf");
    Double sksnilaiangka = k.rs().getDouble("sks_nilai_angka");

    Row dataRow = s.createRow(row);

    Cell dataNoCell = dataRow.createCell(0);
    dataNoCell.setCellValue(row - 3);

    Cell dataKodeMkCell = dataRow.createCell(1);
    dataKodeMkCell.setCellValue(kodemk);

    Cell dataNamaMkCell = dataRow.createCell(2);
    dataNamaMkCell.setCellValue(namamk);

    Cell dataSksCell = dataRow.createCell(3);
    dataSksCell.setCellValue(sks);

    Cell dataNilaiAngkaCell = dataRow.createCell(4);
    dataNilaiAngkaCell.setCellValue(nilaiangka);

    Cell dataNilaiHurufCell = dataRow.createCell(5);
    dataNilaiHurufCell.setCellValue(nilaihuruf);

    Cell dataSksNilaiAngkaCell = dataRow.createCell(6);
    dataSksNilaiAngkaCell.setCellValue(sksnilaiangka);

    row++;
}

Row t = s.createRow(row);
Cell skstotal = t.createCell(3);
Cell sksnilaiTotal = t.createCell(6);
String sqltotalsks = "SELECT SUM(sks) total FROM konversi WHERE"
    + " kode_mk NOT IN ( SELECT kode_mk FROM skipmpilihan )";
String sqltotalsksnilai = "SELECT SUM(sks_nilai_angka) total
FROM"
    + " konversi WHERE kode_mk NOT IN ( SELECT kode_mk FROM"
    + " skipmpilihan )";
k.select(sqltotalsks);
while (k.tampilData()) {
    int skst = k.rs().getInt("total");
    skstotal.setCellValue(skst);
    row++;
}

```

```

k.select(sqltotalsksnilai);
while (k.tampilData()) {
    int sksta = k.rs().getInt("total");
    sksnilaitotal.setCellValue(sksta);
    row++;
}

Row i = s.createRow(row);
Cell ipkHeader = i.createCell(0);
ipkHeader.setCellValue("IPK (Indeks Prestasi Kumulatif) =");
Cell ipkCell = i.createCell(3);

Row last = s.createRow(row + 1);
Cell lastHeader = last.createCell(0);
lastHeader.setCellValue("(SKS x Nilai Angka) / Jumlah SKS");

String sqlmahasiswa = "SELECT * FROM mahasiswa";
k.select(sqlmahasiswa);
while (k.tampilData()) {
    nim = k.rs().getString("nim_mhs");
    nimCell.setCellValue(": " + nim);
    String nama = k.rs().getString("nama_mhs");
    namaCell.setCellValue(": " + nama);
    Double ipk = k.rs().getDouble("ipk");
    ipkCell.setCellValue(ipk);
}

String outputDirPath = pathFolder + "\\\" + nim + " " +
f.format(date) + ".xls";
try (FileOutputStream fileOut = new
FileOutputStream(outputDirPath)) {
    wb.write(fileOut);
} catch (FileNotFoundException ex) {
    popup(ex.getMessage());
} catch (IOException ex) {
    popup(ex.getMessage());
}
popup("Konversi berhasil!\nSilakan cek di direktori yang sama
dengan file asal.");
} catch (ClassNotFoundException | SQLException ex) {
    popup(ex.getMessage());
}
}
//batas proses konversi

public void checkUpdate() {
    try {
        File p = new File(pemasaran);
        File p1 = new File(pemasaran1);
        File p2 = new File(pemasaran2);
        URL o = new URL(online);
        URL o1 = new URL(online1);
        URL o2 = new URL(online2);
        if ((p.length() == getFileSize(o) && (p1.length() ==
getFileSize(o1) && (p2.length() == getFileSize(o2)))) {
            popup("Tidak ada update terbaru.");
        } else {
            try {
                InputStream in = new URL(online).openStream();
                Files.copy(in, Paths.get(pemasaran),
StandardCopyOption.REPLACE_EXISTING);
                InputStream in1 = new URL(online1).openStream();

```

```

        Files.copy(in1, Paths.get(pemasaran1),
StandardCopyOption.REPLACE_EXISTING);
        InputStream in2 = new URL(online2).openStream();
        Files.copy(in2, Paths.get(pemasaran2),
StandardCopyOption.REPLACE_EXISTING);
        popup("Berhasil update database.");
    } catch (IOException e) {
        popup(e.getMessage());
    }
}
} catch (MalformedURLException ex) {
    popup(ex.getLocalizedMessage());
}
}

public void popup(String message) {
    JOptionPane.showMessageDialog(null, message);
}

public void setPathFile(String pathFile) {
    this.pathFile = pathFile;
}

public void setPathFolder(String pathFolder) {
    this.pathFolder = pathFolder;
}

private int getFileSize(URL url) {
    URLConnection con = null;
    try {
        con = url.openConnection();
        if (con instanceof HttpURLConnection) {
            ((HttpURLConnection) con).setRequestMethod("HEAD");
        }
        con.getInputStream();
        return con.getContentLength();
    } catch (IOException e) {
        popup("Tidak ada koneksi internet.");
    } finally {
        if (con instanceof HttpURLConnection) {
            ((HttpURLConnection) con).disconnect();
        }
    }
    return 0;
}

private String getCellValueAsString(Cell cell) {
    String strCellValue = null;
    if (cell != null) {
        switch (cell.getCellType()) {
            case STRING:
                strCellValue = cell.toString();
                break;
            case NUMERIC:
                Double value = cell.getNumericCellValue();
                strCellValue = value.toString();
                break;
            case BOOLEAN:
                strCellValue =
Boolean.toString(cell.getBooleanCellValue());
                break;
            case BLANK:
                strCellValue = "";

```

```
        break;
    }
}
return strCellValue;
}
```



Kode Sumber Koneksi.java

```
package akademik;

import java.sql.*;

public class Koneksi {

    private Connection con;
    private Statement stmt;
    private final String url = "jdbc:mysql://localhost:3306/";
    private final String db = "akademik";
    private final String user = "root";
    private final String pass = "";
    private PreparedStatement ps;
    private ResultSet rs;

    public Koneksi() throws ClassNotFoundException, SQLException {
        Class.forName("com.mysql.jdbc.Driver");
        con = DriverManager.getConnection(url + db, user, pass);
        stmt = con.createStatement();
        con.setAutoCommit(false);
    }

    public void eksekusi(String sql) throws SQLException {
        stmt.execute(sql);
    }

    public void select(String sql) throws SQLException{
        ps = con.prepareStatement(sql);
        rs = ps.executeQuery();
    }

    public boolean tampilData() throws SQLException{
        return rs.next();
    }

    public ResultSet rs(){
        return rs;
    }

    public void tutup() throws SQLException {
        con.commit();
        stmt.close();
        con.close();
    }
}
```