



## **Analisa Pemilihan Bahasa Pemrograman dalam Penyelesaian Tugas Akhir Mahasiswa**

Fitri Pratiwi  
Sistem Informasi  
STMIK Dumai  
[fitrimarten@gmail.com](mailto:fitrimarten@gmail.com)

Putri Yunita  
Sistem Informasi  
STMIK Dumai  
[yunita.santoso3710@gmail.com](mailto:yunita.santoso3710@gmail.com)

Sukri Adrianto  
Sistem Informasi  
STMIK Dumai  
[sukriadrianto@gmail.com](mailto:sukriadrianto@gmail.com)

### **Abstrak**

*Tugas Akhir adalah tugas yang harus dilakukan mahasiswa untuk menyelesaikan pendidikan sarjana mereka. Dalam penyelesaian Tugas Akhir mahasiswa harus mampu membuat suatu program aplikasi yang berhubungan dengan masalah yang diteliti. Namun permasalahan yang ada mahasiswa di STMIK Dumai memilih bahasa pemrograman yang tidak tepat dengan Skema Tugas Akhir sehingga mengakibatkan lamanya proses pembuatan tugas akhir. Dengan dilakukannya Analisa pemilihan bahasa pemrograman dalam penyelesaian Tugas Akhir mahasiswa dengan mengaplikasikan data mining yang menggunakan metode rough set, diharapkan bisa memberikan solusi yang sesuai dalam penyelesaian Tugas Akhir mahasiswa agar tepat waktu.*

*Kata Kunci: Tugas Akhir, Data mining, Rough Set*

### **1. Pendahuluan**

Tugas Akhir adalah istilah yang digunakan untuk mengilustrasikan suatu karya tulis ilmiah berupa paparan tulisan hasil penelitian sarjana S1 yang membahas suatu permasalahan dalam bidang ilmu tertentu dengan menggunakan kaidah-kaidah yang berlaku (Ni'mah, 2014).

Tugas Akhir (TA) adalah sebuah mata kuliah yang harus ditempuh oleh seorang mahasiswa menjelang akhir studinya (Dofi deliyanti, 2015). Tugas Akhir bertujuan agar mahasiswa mampu membuat suatu karya ilmiah yang memadukan ilmu dan keterampilan dalam menganalisa dan merancang dan menjelaskan

masalah yang berhubungan dengan bidang keilmuannya. Sekolah Tinggi Manajemen Informatika dan Komputer (STMIK) Dumai merupakan salah satu perguruan tinggi IT di kota Dumai yang menghasilkan sarjana komputer, dan mempunyai jurusan sistem Informasi dan Teknik Informatika. Di STMIK Dumai untuk menyelesaikan studi, mahasiswa dituntut untuk membuat sebuah karya ilmiah yaitu sebuah hasil dari penelitian dan sebuah aplikasi.

Didalam pembuatan tugas akhir di lingkungan STMIK Dumai ditemukan beberapa permasalahan seperti ketidaksesuaian bahasa pemrograman yang dipilih dengan skema tugas akhir atau penelitian yang akan diteliti oleh mahasiswa, serta ketidaksesuaian skema dengan jurusan sehingga mempengaruhi dan menyebabkan pengerjaan Tugas Akhir menjadi lama bahkan terdapat kecenderungan untuk menyewa orang lain untuk mengerjakan Tugas Akhir. Beberapa kendala tersebut, secara langsung maupun tidak langsung ikut berpengaruh terhadap keterlambatan penyelesaian Tugas Akhir mahasiswa.

Dikarenakan permasalahan diatas maka dilakukan penelitian dengan menganalisa pemilihan bahasa pemrograman dalam penyelesaian Tugas Akhir mahasiswa dengan mengaplikasikan data mining dengan menggunakan metode rough set. Analisa pemilihan bahasa pemrograman dengan metode rough set ini diharapkan bisa memberikan dalam penyelesaian Tugas Akhir mahasiswa agar tepat waktu.

### **2. Landasan Teori**

#### **2.1 Analisa Sistem**

Definisi Analisa Sistem “Analisa sistem adalah menguraikan dari suatu sistem informasi yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya (Lisnawati, 2011).

## 2.2 Tugas Akhir

Tugas Akhir (TA) adalah sebuah mata kuliah yang harus ditempuh oleh seorang mahasiswa menjelang akhir studinya. Mata kuliah ini berbentuk proyek mandiri yang dilakukan oleh mahasiswa di bawah bimbingan dosen pembimbing. Karya ilmiah yang dimaksud dapat berupa laporan ditulis sesuai dengan pedoman tugas akhir (Sofi Defiyanti, 2015), 2015).

Tugas Akhir merupakan karya tulis ilmiah akhir seorang mahasiswa dalam menyelesaikan program pendidikan S1 sebagai bukti kemampuan akademik yang dimiliki mahasiswa dalam melakukan penelitian yang sesuai dengan bidang studinya dan sebagai persyaratan untuk menyelesaikan tugas akhir atau program studinya (Ni'mah, 2014). Salah satu tujuan penulisan Tugas Akhir adalah agar mahasiswa mampu menyusun dan menulis suatu karya ilmiah, sesuai dengan bidang keilmuannya.

## 2.3 Knowledge Discovery in Database (KDD)

*Knowledge Discovery in Database (KDD)* telah didefinisikan sebagai: "Proses identifikasi yang valid, baru, berpotensi yang berguna, dan akhirnya dimengerti pola dalam data (Amihood Amir, dkk, 2005). *KDD* adalah proses untuk mencari dan mengidentifikasi pola (*pattern*) dalam database. Pola yang ditemukan bersifat *valid*, *potentially useful* (sangat nyata digunakan) dan *ultimately understandable* (bisa digunakan dengan segala aspek) (Andika Prajana, 2011).

*Knowledge Discovery* secara sederhana dapat dikatakan sebagai proses meng-ekstrak atau menggali (*mining*) pengetahuan informasi yang berharga (*interesting howledge*) dari sejumlah besar data baik yang disimpan di dalam *database*, *data warehouse* maupun media penyimpanan informasi lainnya. Sementara itu *data mining* merupakan salah satu tahap yang terdapat di dalam *Knowledge Discovery* (Gregorius Satia Budhi & Felicia Soedjianto, 2007)

*Knowledge Discovery in Database (KDD)* berasal dari bidang *Artificial Intelligence (AI)*, dan *DM* telah ditempatkan dalam konteks *KDD*. Sebagai sebuah proses, *KDD* melibatkan beberapa tahap. Proses dimulai dengan memilih set data target, mengubahnya ke format yang diperlukan dengan menggunakan

perangkat lunak data mining (tahap opsional), melakukan data mining untuk mengekstrak struktur, dan akhirnya, menafsirkan dan ditemukan nilai struktur (Imad Abugessaisa, 2008)

Data mining adalah sinonim untuk penemuan lain yang populer *Knowledge Discovery in Database (KDD)*, proses *KDD* yang ditunjukkan pada Gambar 2.1. Ada tiga proses dalam *KDD* (*preprocessing*, *data mining*, *post processing*), *preprocessing* dieksekusi sebelum teknik *data mining* diterapkan pada data. Proses *preprocessing* meliputi *data cleaning*, *data integration*, *data selection* dan *data transformation*. Proses utama dari *KDD* adalah proses *data mining*, dalam proses ini berbeda Algoritma yang diterapkan untuk menghasilkan pengetahuan yang tersembunyi. Setelah itu, proses lain disebut *post processing*, mengevaluasi hasil pertambahan sesuai dengan kebutuhan pengguna dan pengetahuan domain. Mengenai hasil evaluasi, pengetahuan dapat disajikan jika hasilnya memuaskan, sebaliknya kita harus menjalankan beberapa atau semua proses-proses lagi sampai kita mendapatkan hasil yang memuaskan (Jnanamurthy, dkk, 2013).

Menurut (Eko Nur Wahyudi, 2011) *Data mining* adalah bagian integral dari penemuan *Knowledge Discovery in Database (KDD)*, yang merupakan proses keseluruhan mengubah data mentah menjadi informasi yang bermanfaat. Seiring dengan perjalanan waktu dan berkembangnya pemahaman tentang *Data Mining*, maka persamaan pemahaman antara *Data Mining* dengan *KDD* berubah menjadi “*Data Mining* adalah bagian dari *KDD*”.

## 2.4 Definisi Data Mining

*Data mining* atau sering disebut sebagai *knowledge discovery in database (KDD)* adalah kegiatan yang meliputi pengumpulan, pemakaian data historis untuk menemukan keteraturan, pola atau hubungan dalam data berukuran besar. Keluaran *data mining* ini bisa dipakai untuk membantu pengambilan keputusan di masa depan (Subekti Mujiasih, 2011).

Menurut para ahli, data mining merupakan sebuah analisa dari observasi data dalam jumlah besar untuk menemukan hubungan yang tidak diketahui sebelumnya dan dua metode baru untuk meringkas data agar mudah dipahami serta kegunaannya untuk pemilihan data (Kamagi & Hansun, 2014).

*Data mining* adalah proses penggalian dan analisis pola, hubungan dan informasi yang berguna dari database besar. Ini biasanya melibatkan empat kelastugasyang merupakan klasifikasi, *clustering*, regresi dan aturan asosiasi belajar. Ada dua jenis utama *Data mining*: *-verification-oriented* (sistem memverifikasi hipotesis pengguna) dan *discovery-oriented* (sistem menemukan aturan dan pola-pola baru otonom) (Khalida, dkk, 2010).

Buku yang berjudul Algoritma Data Mining yang ditulis oleh Kusri dan Emha Taufiq Luthfi, juga

dikemukakan pendapat tentang pengertian *data mining*, yaitu :*Data Mining* adalah serangkaian proses untuk menggali nilai tambah dari suatu kumpulan data berupa pengetahuan yang selama ini tidak diketahui secara manual (Pramudiono,2006). *Data Mining* merupakan analisis dari peninjauan kumpulan data untuk menemukan hubungan yang tidak diduga dan meringkas data dengan cara yang berbeda dengan sebelumnya, yang dapat dipahami dan bermanfaat bagi pemilik data (Larose, 2005).(Emha Taufiq Luthfi, 2009)

*Data mining*, dikenal sebagai *Knowledge Discovery in Database (KDD)*, itu adalah ekstraksi dari implisit, informasi yang sebelumnya tidak diketahui dan berpotensi berguna dari data dalam *database*. Padahal, *data mining* dan *Knowledge Discovery in Database (KDD)* adalah sering diperlakukan sebagai sinonim, data mining sebenarnya adalah bagian dari *knowledge discovery process* (Deshpande dan Thakare, 2010). *Data mining* dan *knowledge discovery* adalah eksplorasi di alam, lebih induktif dari pada metode statistik tradisional (Diansheng Guo dan Jeremy Mennis, 2009). *Data mining* merupakan langkah besar dalam proses *Knowledge Discovery in Database (KDD)* yang terdiri dari penerapan teknik komputasi yang di bawah keterbatasan efisiensi komputasi dapat diterima dan menghasilkan beberapa model pecahan tertentu pada pola atau model data (V.Umarini, 2010).

Dalam bentuk yang paling sederhana, *data mining* mengotomatisasi deteksi pola yang relevan dalam *database*, menggunakan pendekatan yang ditetapkan dan algoritma untuk melihat ke dalam data saat ini dan sejarah yang kemudian dapat dianalisis untuk memprediksi *tren* masa depan. Karena alat *data mining* memprediksi *tren* dan perilaku masa depan dengan membaca melalui *database* untuk pola tersembunyi, mereka memungkinkan organisasi untuk membuat *proaktif*, keputusan berbasis pengetahuan dan menjawab pertanyaan-pertanyaan yang sebelumnya terlalu memakan waktu untuk menyelesaikan (Ramamohan, dkk, 2012).(fitri pratiwi, 2017)

## 2.5 Pengelompokan Data Mining

Ada beberapa teknik yang dimiliki *data mining* berdasarkan tugas yang bisa dilakukan, yaitu (Eko Prasetyo, 2012)):

1. Deskripsi  
Para peneliti biasanya mencoba menemukan cara untuk mendeskripsikan pola dan *trend* yang tersembunyi dalam data.
2. Estimasi  
Estimasi mirip dengan klasifikasi, kecuali variabel tujuan yang lebih ke arah numerik dari pada kategori.
3. Prediksi

Prediksi memiliki kemiripan dengan estimasi dan klasifikasi. Hanya saja, prediksi hasilnya menunjukkan sesuatu yang belum terjadi (mungkin terjadi di masa depan).

4. Klasifikasi  
Dalam klasifikasi variabel, tujuan bersifat kategorik. Misalnya, kita akan mengklasifikasikan pendapatan dalam tiga kelas, yaitu pendapatan tinggi, pendapatan sedang, dan pendapatan rendah.
5. *Clustering*  
*Clustering* lebih ke arah pengelompokan *record*, pengamatan, atau kasus dalam kelas yang memiliki kemiripan.
6. Asosiasi  
Mengidentifikasi hubungan antara berbagai peristiwa yang terjadi pada satu waktu.

## 2.6 Metode Rough Set

Teori *Rough set* telah dikembangkan oleh Zdzislaw Pawlak pada awal tahun 1980-an. Ini merupakan pengembangan dari teori *fuzzy set* (Lotfi A. Zadeh, 1965) yang mengatakan adanya daerah abu-abu (antara hitam dan putih) dimana suatu hal tidak hanya direpresentasikan sebagai *binary*, *boolean*, ataupun benar-salah saja. Dalam teori matematika, keanggotaan dalam suatu himpunan didefinisikan untuk setiap obyek dalam semesta yang dapat dikenali, dan sisanya merupakan milik komplemen dari himpunan yang dimaksud. Pada kenyataannya, informasi obyek yang tersedia sering tidak cukup untuk mendefinisikan obyek dan melakukan klasifikasi. Teori *rough set* membuat analisis formal agar situasi yang demikian dapat diatasi. Tujuan dari teori *Rough set* adalah untuk mengenali ketidakpastian dalam klasifikasi dari suatu obyek. Vitoria, Dam'asio dan Maluszy'nski (2004) mengembangkan bahasa untuk meng-ekspresikan data *Rough* dan dilengkapi dengan fitur *quantitative measure* seperti *support*, *strength* dan *accuracy* (Rully Soelaiman, Wiwik Anggraeni, 2008).

Menurut Ramadevi Yellasiri, dkk (2005), teori *rough set* yang dikembangkan oleh Z.Pawlak adalah alat matematika yang penting untuk teknologi komputer. Hal ini terlibat dalam beberapa pembuatan keputusan, *data mining*, pengetahuan representasi, akuisisi pengetahuan dan banyak aplikasi lainnya.

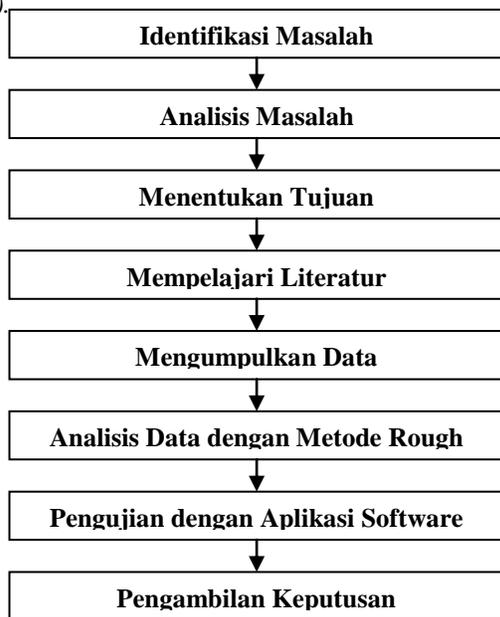
*Rough Set* adalah teori matematika baru-baru ini yang digunakan sebagai *data mining* dengan banyak keuntungan yang menguntungkan. Teori ini telah diterapkan untuk berbagai domain, yang sebagian besar aplikasi ini digunakan untuk memecahkan masalah klasifikasi, yang mengecualikan Faktor temporal dalam set data. *Rough Set* analisis disajikan sebagai suatu teknik untuk mengarahkan proses penemuan pengetahuan dari data (Winston, dkk, 2009).

Teori ini banyak digunakan untuk analisis klasifikasi yang melibatkan ketidakpastian, ketidakteraturan dan ketidaklengkapan informasi pada data. Tujuan analisis *Rough Set* adalah untuk mendapatkan perkiraan *rule* yang singkat dari suatu tabel atau sekumpulan data (Warhi Maharani, 2008). *Rough Set* menawarkan dua bentuk representasi data yaitu *Information Systems (IS)* dan *Decision Systems (DS)*. (Asriningtias & Mardhiyah, 2014)

### 3. Metode Penelitian

Metode yang digunakan pada penelitian ini adalah metode *Rough Set*. Dalam metode ini diharapkan dapat menyelesaikan permasalahan yang menentukan bahasa pemrograman yang sesuai dengan skema tugas akhir pada mahasiswa STMIK Dumai

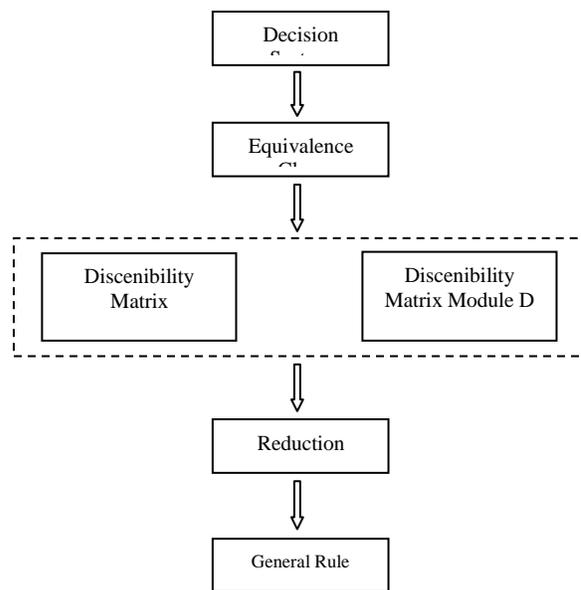
Tujuan dari analisis *Rough Set* adalah untuk mendapatkan perkiraan *rule* yang singkat dari suatu tabel. Hasil dari analisis *Rough Set* dapat digunakan dalam proses *data mining* dan *knowledge discovery*. *Rough Set* menawarkan dua bentuk representasi data yaitu *Information Systems (IS)* dan *Decision Systems (DS)*.



Gambar 1. Metode Penelitian

### 4. Hasil Dan Pembahasan

Dalam *data mining* terdapat beberapa teknik pengolahan data agar data yang diolah lebih bermanfaat dan bernilai ilmu. Salah satunya adalah Metode *Rough Set*. Berdasarkan teknik atau metode dari Analisa Metode *Rough Set* di atas dapat digambarkan sebagai berikut :



Gambar 2. Metode Rough Set

#### Sampel Data Yang Digunakan

Tahapan ini menjelaskan mengenai sampel data yang digunakan untuk proses analisa metode *Rough Set*. Sampel yang digunakan diambil berdasarkan data mahasiswa pada table data mahasiswa sebelumnya. Berikut ini adalah sampel data pendukung untuk proses penelitian

#### Penerapan Metode Rough Set

*Rough Set* menawarkan dua bentuk representasi data yaitu *Information Systems (IS)* dan *Decision Systems (DS)* yaitu sebuah pasangan  $IS = \{U, A\}$ , dimana  $U = \{e_1, e_2, e_n\}$  dan  $A = \{a_1, a_2, a_n\}$  yang merupakan sekumpulan contoh dan *attribute* kondisi secara berurutan. Definisi diatas menunjukkan bahwa sebuah *Information Systems* yang terdiri dari sekumpulan *example* seperti  $\{e_1, e_2, e_n\}$  dan *attribute* kondisi, seperti  $\{a_1, a_2, a_n\}$ . Berikut ini merupakan contoh analisa penerapan Metode *Rough Set*. Adapun studi kasus yang diangkat berdasarkan penelitian yaitu "Analisa Pemilihan Bahasa Pemrograman dalam Penyelesaian Tugas Akhir Mahasiswa". Adapun algoritma penyelesaian masalah dari metode *Rough Set* yaitu sebagai berikut :

##### 1. Decision Systems

*Decision Systems* merupakan *Information Systems* yang telah memiliki keputusan atau hasil berdasarkan hasil asumsi yang telah memenuhi syarat dan ketentuan berdasarkan atributnya. Berikut ini adalah table *Decision Systems* dari data pengajuan judul Tugas Akhir mahasiswa yang ada pada STMIK Dumai yang menjadi objek penelitian. Adapun nilai dari setiap atribut yang terdapat pada tabel 1 di bawah ini berasal

dari transformasi sampel data ke dalam bentuk kriteria kelayakan yang dibahas sebelumnya.

**Tabel 1. Decision System**

NO	PRODI	Skema TA	Bahasa Pemograman	Kecocokan
1	SI	WEB	PHP	YA
2	TI	WEB	PHP	YA
3	TI	Android	Java	YA
4	TI	WEB	PHP	YA
5	SI	Multimedia	Adobe Flash	YA
6	SI	Client Server	Visual Basic 6.0	YA
7	SI	Client Server	Visual Basic 6.0	YA
8	SI	WEB	PHP	YA
9	TI	Android	Java	YA
10	TI	Android	PHP	Tidak
11	TI	Client Server	PHP	YA
12	TI	Multimedia	Adobe Flash	YA
13	SI	Multimedia	Adobe Flash	YA
14	TI	Android	PHP	Tidak
15	SI	Android	PHP	Tidak

2. *Equivalence Class* merupakan pengelompokan objek-objek yang sama untuk *atribute* dari *decision system*. Dari table 1 untuk langkah selanjutnya mengelompokkan data yang sama tiap atribut kondisi maupun atribut keputusan. Terdapat beberapa kesamaan misalnya pada  $E1=E8$ ,  $E2=E4$ ,  $E6=E7$  maka hasil pengelompokan dapat dilihat pada tabel 2.

**Tabel 2 Equivalence Class**

NO	PRODI	Skema TA	Bahasa Pemograman	Kecocokan
EC1	SI	WEB	PHP	YA
EC2	TI	WEB	PHP	YA
EC3	TI	Android	Java	YA
EC4	SI	Multimedia	Java	YA
EC5	SI	Client Server	Visual Basic 6.0	YA
EC6	SI	Android	Java	YA
EC7	TI	Android	PHP	Tidak

3. Kemudian dilakukan proses *Discernibility Matrik* atau *Discernibility Matrik Module D*.

**Tabel 3 Acuan Discernibility Matrix dan Discernibility Matrix Module D**

NO	PRODI	Skema TA	Bahasa Pemograman	Kecocokan	Attribute Kondisi		Attribute Keputusan	
					A	B	C	D
EC1	SI	WEB	PHP	YA				
EC2	TI	WEB	PHP	YA				

EC3	TI	Android	Java	YA
EC4	SI	Multimedia	Adobe Flash	YA
EC5	SI	Client Server	Visual Basic 6.0	YA
EC6	TI	Android	PHP	Tidak
EC7	TI	Client Server	PHP	YA
EC8	TI	Multimedia	Adobe Flash	YA
EC9	SI	Android	PHP	Tidak

Berikut merupakan langkah-langkah dalam menyelesaikan *Discernibility Matrix* :

- Lihat perbandingan *attribute kondisi* yaitu A,B dan C dengan mengabaikan *attribute Hasil* yaitu D.
- Jika ada perbedaan nilai yang ada pada A,B dan C maka ditulis didalam kolom/baris hanya *attribute* yang berbeda saja.
- Jika nilai *attribute* A,B dan C sama maka ditulis dengan simbol (X) pada kolom atau baris yang menyatakan tidak ada nilai.

Adapun hasil *Discernibility Matrix* seperti pada tabel dibawah ini berasal dari tabel 3 Berdasarkan aturan yang ada dan data di atas maka berikut ini adalah *Discernibility Matrix* nya:

**Tabel 4 Tabel Hasil Discernibility Matrix**

Objek	EC1	EC2	EC3	EC4	EC5	EC6	EC7	EC 8	EC 9
EC1		A	ABC	BC	BC	ABC	AB	A BC	B
EC2	A		BC	ABC	ABC	B	B	BC	AB
EC3	ABC	BC		ABC	ABC	C	BC	BC	AC
EC4	BC	ABC	ABC		BC	ABC	ABC	A	BC
EC5	BC	ABC	ABC	BC		ABC	AC	A BC	BC
EC6	ABC	B	C	ABC	ABC		B	BC	A
EC7	AB	B	BC	ABC	AC	B		BC	AB
EC8	ABC	BC	BC	A	ABC	BC	BC		AB C
EC9	B	AB	AC	BC	BC	A	AB	A BC	

Langkah-langkah penyelesaian *Discernibility Matrix Module D* adalah :

- Yang dilihat hanya *attribute keputusan* yaitu C.
- Jika ada perbedaan nilai *attribute keputusan* yaitu C perbandingan antara nilai A dan B maka di tulis didalam kolom/baris hanya *attribute* yang berbeda saja.
- Tapi apabila *attribute keputusan* sama walaupun objek berbeda maka nilai yang diisi pada kolom/baris dengan simbol (X) yang menyatakan tidak ada nilai yang nantinya tidak ada nilai *reduction*-nya.

Adapun hasil *Discernibility Matrix Module D* seperti pada tabel dibawah ini berasal dari tabel

ACUAN 4 Berdasarkan aturan yang ada dan data di atas maka berikut ini adalah *Discernibility Matrix Module D* nya:

**Tabel 5 Tabel Hasil Discernibility Matrix Module D**

Objek	EC1	EC2	EC3	EC4	EC5	EC6	EC7
EC1						AB	
EC2						B	
EC3						C	
EC4						ABC	
EC5						ABC	
EC6	AB	B	C	ABC	ABC		B
EC7						B	
EC8						BC	
EC9	B	AB	AC	BC	BC		AB

Lakukan proses *Reduction* yang merupakan penyeleksian *attribute* minimal dari sekumpulan atribut yang menggunakan *prime implicant* fungsi boolean.

Teknik pencarian kombinasi atribut yang mungkin dikenal dengan *reduct*, yaitu dengan cara :

- Buat persamaan aljabar *boolean* berdasarkan *discernibility matrix* atau *modulo D*.
- Untuk nilai dalam satu kolom yang lebih dari satu atribut, misalnya nilai AB, maka dibeli simbol ( $\vee$ ) AVB yang dalam operasi aritmatikanya sama dengan (+).
- Dari baris pertama ke baris berikutnya, maka simbolnya ( $\wedge$ ) yang dalam operasi aritmatikanya sama dengan (\*).
- Jika nilai pada kolomnya X, maka tidak dibuat persamaan aljabar *boolean* karena tidak memiliki nilai.
- Sederhakan persamaan tersebut dengan prinsip atau konsep aljabar *boolean*.
- Jadikan hasil persamaan sebagai *reduct*.

Hasil *Discernibility Matrix Module D* mempunyai kesamaan misalnya EC4=EC5 maka proses *reduction* cukup dilakukan sekali saja. Berikut ini adalah *reduction* dari *data mining* "Judul Tugas Akhir" berdasarkan *discernibility matrix modulo D* :

$$\begin{aligned} EC1 &= (A \vee B) \wedge B \\ &= (A+B) * B \\ &= (AA+BB) \\ &= \{B\} \end{aligned}$$

$$\begin{aligned} EC3 &= C \wedge (A \vee C) \\ &= C*(A+C) \\ &= (AC+CC) \\ &= \{C\} \end{aligned}$$

$$\begin{aligned} EC4 &= (A \vee B \vee C) \wedge (B \vee C) \\ &= (A+B+C)*(B+C) \\ &= (AA+AB+AC+BB+BC+CB+CC) \\ &= AB+AC+B+C \end{aligned}$$

$$\begin{aligned} &= B+C \\ &= \{B\} \{C\} \end{aligned}$$

$$\begin{aligned} EC6 &= (A \vee B) \wedge B \wedge C \wedge (A \vee B \vee C) \wedge (A \vee B \vee C) \wedge B \wedge (B \vee C) \\ &= (A+B)*B*C*(A+B+C)*B*(B+C) \\ &= (AB+BB)*(AC+CB+CC)*(BB+BC) \\ &= B*C*B \\ &= \{B,C\} \end{aligned}$$

$$\begin{aligned} EC8 &= (B \vee C) \wedge (A \vee B \vee C) \\ &= (B+C) *(A+B+C) \\ &= BA+BB+BC+CA+CB+CC \\ &= B+C \\ &= \{B\} \{C\} \end{aligned}$$

$$\begin{aligned} EC9 &= B \wedge (A \vee B) \wedge (A \vee C) \wedge (B \vee C) \wedge (B \vee C) \wedge (A \vee B) \wedge (A \vee B \vee C) \\ &= B*(A+B)*(A+C)*(B+C)*(A+B+C) \\ &= B*(AA+AC+AB+BC)*(AB+BB+BC+AC+BC+CC) \\ &= B*(A+BC)*(B+C) \\ &= (AB+BB+BC)*(B+C) \\ &= (B+BC)*(B+C) \\ &= B*(B+C) = \{B\} \end{aligned}$$

Berdasarkan perhitungan di atas maka akan diambil salah satu hasilnya jika terdapat hasil *reduct* yang sama, berikut adalah tabel *reduct* yang telah dihasilkan dapat dilihat pada tabel 6

**Tabel 6. Hasil Proses Reduct**

Objek	Reduct Yang Dihasilkan
EC1	{B}
EC2	{C}
EC3	{B,C}

Untuk memperoleh hasil akhir dilakukan proses *Generate Rules*. Ketentuan dalam menentukan *generate rules* yang dihasilkan berdasarkan dari *reduction* adalah sebagai berikut:

- Jika atribut kondisi sama, atribut hasil berbeda, maka *rule* atribut kondisi hanya ditulis sekali saja dengan menggabungkan atribut hasilnya.
- Jika atribut kondisi berbeda, atribut hasil sama, maka *rule* ditulis semua atribut.

Setelah didapat hasil dari *reduction*, maka langkah akhir untuk mendapatkan *Generate rules*-nya dengan panduan melihat tabel 5.2. Adapun *Generate rules* dari hasil *reduction*-nya adalah sebagai berikut :

- B=Skema
  - If Skema = Web then Hasil = Ya.
  - If Skema = Android then Hasil = Ya or hasil = Tidak..
  - If Skema = Multimedia then Hasil = Ya.
  - If Skema = Client Server then Hasil = Ya..
- C = Bahasa Pemrograman
  - If Bahasa Pemrograman = PHP then Hasil = Ya or hasil = Tidak.
  - If Bahasa Pemrograman = Java then Hasil = Ya.

- If Bahasa Pemograman = VB then Hasil = Ya.  
 If Bahasa Pemograman = Adobe Flash then Hasil = Ya.
3. B = Skema dan C = Bahasa Pemograman  
 If Skema = Web and Bahasa Pemograman = PHP then Hasil = Ya.  
 If Skema = Android and Bahasa Pemograman = Java then Hasil = Ya.  
 If Skema = Multimedia and Bahasa Pemograman = Adobe Flash then Hasil = Ya  
 If Skema = Client Server and Bahasa Pemograman = VB or PHP then Hasil = Ya.  
 If Skema = Android and Bahasa Pemograman = PHP then Hasil = Tidak.

Jadi jumlah *General Rules* keseluruhan yang telah diproses adalah 13 keputusan atau pengetahuan baru. Setelah didapat *General Rule* maka proses pengolahan dari *data mining* telah selesai untuk mendapat keputusan dalam mengklasifikasi bahasa pemograman dan skema yang sesuai untuk menyelesaikan tugas akhir.

Setelah dihitung manual maka selanjutnya akan di uji implementasi sistem menggunakan *Rosetta 1.4.11* seperti pada gambar 1 dibawah ini:

	PRODI	Skema TA	Bahasa Pemograman	Kecocokan
1	SI	WEB	PHP	YA
2	TI	WEB	PHP	YA
3	TI	Android	Java	YA
4	TI	WEB	PHP	YA
5	SI	Multimedia	Adobe Flash	YA
6	SI	Client Server	Visual Basic	YA
7	SI	Client Server	Visual Basic	YA
8	SI	WEB	PHP	YA
9	TI	Android	Java	YA
10	TI	Android	PHP	Tidak
11	TI	Client Server	PHP	YA
12	TI	Multimedia	Adobe Flash	YA
13	SI	Multimedia	Adobe Flash	YA
14	TI	Android	PHP	Tidak
15	SI	Android	PHP	Tidak

**Gambar 1 Form Decision System dengan software Rosetta**

Lalu akan dicari Reduct nya dan reduct yang dihasilkan seperti yang ada pada gambar 2 dibawah ini :

	Reduct	Support	Length
1	{Skema TA, Bahasa Pemograman}	49	2
2	{PRODI, Skema TA}	3	2
3	{Skema TA}	6	1
4	{PRODI, Bahasa Pemograman}	4	2
5	{Bahasa Pemograman}	5	1

**Gambar .2 Reduct dengan software Rosetta**

## 5. Simpulan

Dalam penelitian ini dapat dianalisa dan disimpulkan beberapa hal sebagai berikut:

1. Dengan adanya sistem yang menggunakan metode rough set yang diimplementasikan dalam tool Rosetta dapat digunakan untuk membangun sebuah sistem yang memanfaatkan data pengajuan judul tugas akhir mahasiswa yang selama ini tidak digunakan dalam mengambil sebuah keputusan dalam hal kecocokan pemilihan bahasa pemograman dan skema dengan prodi mahasiswa tersebut untuk menyelesaikan tugas akhir.
2. Dengan adanya hasil yang disajikan dari metode Rough Set dan di uji menggunakan tools Rosetta 1.4.1.1 yang mendukung keputusan dalam pengendalian keputusan, maka proses pengambilan keputusan akan menjadi lebih optimal dan kesalahan dalam proses pengambilan keputusan dapat diminimalkan.
3. Setelah mendapatkan hasil pengujian berdasarkan data yang lalu, pihak prodi bisa memberikan arahan kepada mahasiswa dalam hal pemilihan bahasa pemograman dan skema yang sesuai agar penyelesaian tugas akhir mahasiswa tersebut tepat waktu.

## 6. Referensi

- Andika Prajana. (2011). Aplikasi Data Mining untuk Perbandingan Manajemen Laba terhadap Persistensi Laba Pada Perusahaan Perbankan yang Go Publik di Bursa Efek Indonesia. *STMIK Dharmasraya*.
- Asriningtias, Y., & Mardhiyah, R. (2014). APLIKASI DATA MINING UNTUK MENAMPILKAN INFORMASI, 8(1), 837–848.
- Eko Nur Wahyudi, D. (2011). Analisa Profil Data Mahasiswa Baru terhadap Program Studi yang dipilih di Perguruan Tinggi Swasta Jawa Tengah dengan Menggunakan Teknik Data Mining. *Universitas Stikubank*.
- Eko Prasetyo. (2012). *Data Mining Konsep dan*

- Aplikasi Menggunakan MATLAB* (ANDI Yogya). gresik.
- Emha Taufiq Luthfi. (2009). Penerapan Data Mining Algoritma Asosiasi Untuk Meningkatkan Penjualan. *STMIK AMIKOM Yogyakarta*.
- fitri pratiwi, sukri A. (2017). SATIN – Sains dan Teknologi Informasi Peningkatan Jumlah Mahasiswa Melalui Promosi Dengan Penerapan Analisa Data Mining, 3(2).
- Gregorius Satia Budhi, & Felicia Soedjianto. (2007). Aplikasi Data Mining Market Basket Analysis Pada Tabel Data Absensi Elektronik Untuk Mendeteksi Kecurangan Absensi (Check-Lock) Karyawan Di Perusahaan. *Jurnal Informatika*, 8(2), 119–129. Retrieved from <http://puslit2.petra.ac.id/ejournal/index.php/inf/article/view/16777>
- Kamagi, D. H., & Hansun, S. (2014). Implementasi Data Mining dengan Algoritma C4 . 5 untuk Memprediksi Tingkat Kelulusan Mahasiswa, VI(1), 15–20.
- Lisnawati, H. S. (2011). Aplikasi Pencatatan Persediaan Barang Jadi Menggunakan Visual Foxpro Pada CV. Maja Menjangan Kabupaten Majalengka. *Jurnal Kompak STMIK IKMI*, 4, 25–27.
- Ni'mah, A. (2014). Hubungan Antara Dukungan Sosial Dan Self Efficacy Dalam Menyelesaikan Skripsi. *Indonesian Journal of Guidance and Counseling: Theory and Application*, 3(1), 43–48. Retrieved from <https://journal.unnes.ac.id/sju/index.php/jbk/article/view/3752>
- Rully Soelaiman, Wiwik Anggraeni, E. S. (2008). Penerapan Rough Set Quantitative Measure Pada Aplikasi Pendukung keputusan. *Institut Teknologi Sepuluh November*.
- Sofi Defiyanti. (2015). Penentuan peminatan tugas akhir mahasiswa teknik informatika unsika. *Jurnal Ilmiah Solusi*, 2(5), 9–16.
- subekti mujiasih. (2011). pemanfaatan data mining untuk prakiraan cuaca. *Jurnal Meteorologi Dan Geofisika*, 12, 189–195.
- V.Umarini, P. (2010). A Study on Effective Mining of Association Rules from Huge Database. *Coimbatore*.