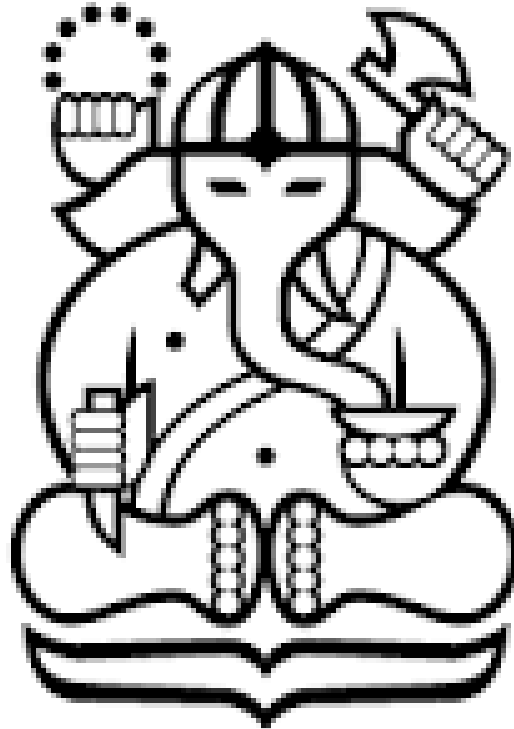


# **LAPORAN ONTOLOGI DAN SISTEM BERBASIS PENGETAHUAN**

**IF4070 - Representasi Pengetahuan dan Penalaran**



Nama Kelompok:

1. Adril Putra Merin (13522068)
2. Marvin Scifo Y. Hutahaeen (13522110)
3. Berto Richardo Togatorop (13522118)

**PROGRAM STUDI TEKNIK INFORMATIKA  
SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA  
INSTITUT TEKNOLOGI BANDUNG**

**2025**

# Daftar Isi

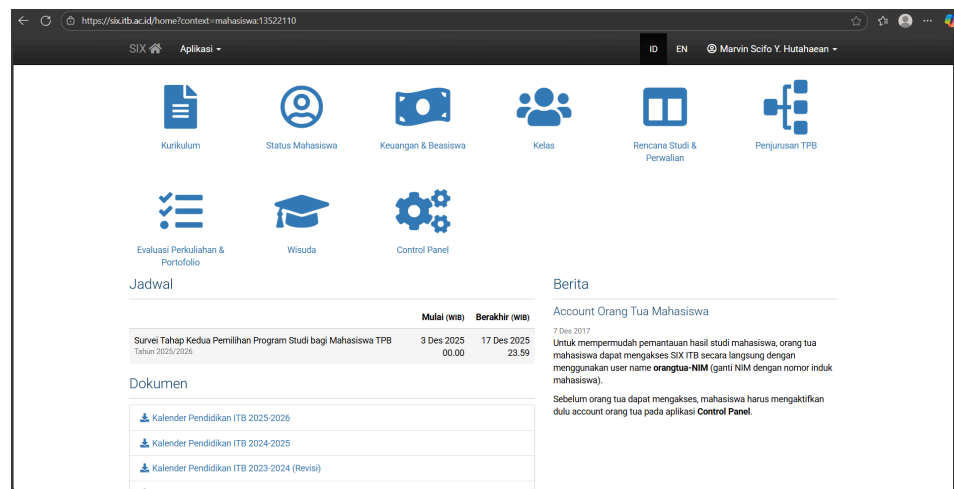
<b>BAB I</b>	
<b>DOMAIN.....</b>	<b>3</b>
<b>BAB II</b>	
<b>ONTOLOGI.....</b>	<b>4</b>
<b>BAB III</b>	
<b>SISTEM BERBASIS PENGETAHUAN.....</b>	<b>5</b>
<b>BAB IV</b>	
<b>PEMBAHASAN.....</b>	<b>6</b>
<b>Kesimpulan.....</b>	<b>7</b>
<b>Matrix Kontribusi.....</b>	<b>8</b>

# BAB I

## DOMAIN

### A. Penjelasan Domain dan Batasan

Sistem informasi akademik ITB (SIX ITB) adalah sebuah aplikasi sistem informasi berbasis web yang digunakan oleh seluruh civitas akademik ITB (mahasiswa, staf pengajar, ketua program studi, dll) untuk mengatur dan mengelola kebutuhan akademik mereka. Pada umumnya, sebuah universitas memiliki program studi yang merupakan kesatuan kegiatan pendidikan akademik yang memiliki kurikulum, tujuan, dan kompetensi tertentu untuk dipelajari oleh mahasiswa. Setiap mahasiswa memilih program studi dan berdasarkan kurikulum yang mengatur program studi tersebut mahasiswa memilih mata kuliah tertentu untuk syarat kelulusan.



**Gambar 1 - Halaman Utama SIX ITB**

Batasan domain yang dilakukan pada tugas besar IF4070 Representasi Pengetahuan dan Penalaran adalah sebagai berikut:

1. Program Studi yang akan digunakan dalam pengerjaan tugas besar adalah berasal dari fakultas STEI yang terdiri dari S1 Teknik Elektro, S1 Teknik Informatika, TPB STEI-R, S1 Teknik Tenaga Listrik, S1 Teknik Telekomunikasi, S1 Sistem dan Teknologi Informasi, S1 Teknik Biomedis, TPB STEI-K, S2 Teknik Elektro, S2 Informatika, S2 Sistem dan Teknologi Informasi, S3 Teknik Elektro dan Informatika
2. Mata kuliah yang akan digunakan dalam pengerjaan tugas besar adalah semua mata kuliah dari fakultas STEI
3. Gelar yang akan digunakan dalam pengerjaan tugas besar adalah gelar yang didapat ketika mahasiswa lulus dari program studi yang berasal dari fakultas STEI yaitu S.T., M.T., dan Dr.
4. Nama mata kuliah yang digunakan dalam instansi mata kuliah dalam pengerjaan tugas besar menggunakan bahasa Indonesia

## B. Kecocokan Domain dengan Pengembangan *Rule Based System*

*Rule Based System* yang dikembangkan bisa membantu banyak dalam domain SIX ITB dalam pencarian range tertentu seperti mata kuliah bila diberikan program studinya, tipe mata kuliahnya, tipe penilaiannya, dll. Terdapat banyak pemetaan yang perlu dilakukan pada domain SIX ITB. Salah satu contoh pemetaan dari domain ini adalah bagaimana cara memetakan mata kuliah ke program studi yang tepat dan bagaimana alat identifikasi tertentu sebagai kode mata kuliah bisa membantu mengetahui mata kuliah tersebut berasal dari program studi apa.

Tidak hanya itu *Rule Based System* juga bisa membantu dalam perihal mencari mata kuliah yang merupakan *prerequisite* dari mata kuliah lainnya yang artinya sebuah mata kuliah *prerequisite* tersebut perlu diambil atau diselesaikan terlebih dahulu sebelum mata kuliah yang diinginkan tersebut bisa diambil. Berdasarkan beberapa hal tersebut, bisa disimpulkan domain pada mata kuliah yang terdaftar di SIX ITB bisa digunakan untuk pengembangan *Rule Based System*.

## BAB II

# ONTOLOGI

### A. Sumber Pengetahuan Ontologi

Terdapat beberapa sumber yang diambil untuk mendapatkan pengetahuan-pengetahuan yang diperlukan dalam membangun ontologi. Berikut adalah sumber-sumber yang diambil

#### 1. SIX ITB

Di SIX ITB, terdapat sebuah halaman yang menunjukkan semua mata kuliah yang ada di ITB. Karena kurikulum ITB yang terbaru adalah kurikulum 2024, mata kuliah yang pertama kali muncul ketika membuka halaman tersebut adalah berasal dari kurikulum 2024.

SIX Aplikasi - Menu (2024) - Menu -				ID	EN	Marvin Sifo Y. Hutahaean -		
Kode	Tahun Kurikulum	No Prodi	Nama MK	Nama MK (Bahasa Inggris)	SKS	Kategori	Jenis Nilai	
1 AE2100	2024	136	Pengenalan Teknik Dirgantara	Introduction to Aerospace Engineering	2	Kuliah	ABCD	
2 AE2102	2024	136	Rekayasa Termal	Thermal Engineering	4	Kuliah	ABCD	
3 AE2104	2024	136	Kinematika dan Dinamika	Kinematics and Dynamics	4	Kuliah	ABCD	
4 AE2202	2024	136	Mekanika Fluida	Fluid Mechanics	4	Kuliah	ABCD	
5 AE2203	2024	136	Material dan Metode Manufaktur Dirgantara	Aerospace Materials and Manufacturing Methods	4	Kuliah	ABCD	
6 AE2204	2024	136	Sistem Dinamik	Dynamic System	3	Kuliah	ABCD	
7 AE3100	2024	136	Instrumentasi, Pengukuran, dan Eksperimen	Instrumentations, Measurements, and Experiments	3	Kuliah	ABCD	
8 AE3102	2024	136	Aerodinamika	Aerodynamics	4	Kuliah	ABCD	
9 AE3103	2024	136	Analisis dan Perancangan Struktur Ringan	Lightweight Structure Analysis and Design	4	Kuliah	ABCD	
10 AE3104	2024	136	Getaran Mekanik	Mechanical Vibration	3	Kuliah	ABCD	
11 AE3105	2024	136	Teknik Kendali	Control Engineering	3	Kuliah	ABCD	
12 AE3200	2024	136	Pewatatan Pesawat untuk Desain	Aircraft Performance for Design	3	Kuliah	ABCD	
13 AE3202	2024	136	Propulsi Pesawat Terbang	Aircraft Propulsion	4	Kuliah	ABCD	
14 AE3204	2024	136	Dinamika Terbang	Flight Dynamics	3	Kuliah	ABCD	
15 AE4000	2024	136	Desain Wahana Dirgantara	Aerospace Vehicle Design	4	Kuliah	ABCD	
16 AE4001	2024	136	Etika Profesi Insinyur	Engineering Professional Ethics	2	Kuliah	ABCD	
17 AE4002	2024	136	Sertifikasi Kelakudaraan	Airworthiness Certification	3	Kuliah	ABCD	
18 AE4010	2024	136	Pengenalan Dinamika Fluida Komputasional	Introduction to Computational Fluid Dynamics	3	Kuliah	ABCD	
19 AE4011	2024	136	Dinamika Fluida Eksperimental	Experimental Fluid Dynamics	3	Kuliah	ABCD	
20 AE4012	2024	136	Aerodinamika Kecepatan Tinggi	High Speed Aerodynamics	3	Kuliah	ABCD	
21 AE4013	2024	136	Aerodinamika Rotor	Rotor Aerodynamics	3	Kuliah	ABCD	
22 AE4020	2024	136	Metode Elemen Hingga	Finite Element Methods	3	Kuliah	ABCD	
23 AE4021	2024	136	Beban Pesawat Udara	Aircraft Loads	3	Kuliah	ABCD	
24 AE4022	2024	136	Mekanika Impak	Impact Mechanics	3	Kuliah	ABCD	
25 AE4030	2024	136	Pemodelan dan Simulasi Wahana Dirgantara	Flight Vehicles Modelling and Simulation	3	Kuliah	ABCD	
26 AE4031	2024	136	Operasi Pesawat Udara Ntr Awak (UAV)	Unmanned Aerial Vehicle (UAV) Operations	2	Kuliah	ABCD	
27 AE4032	2024	136	Operasi Wahana Antariksa	Spacecraft Operations	2	Kuliah	ABCD	

**Gambar 2 - Daftar Mata Kuliah ITB Kurikulum 2024**

Terdapat 7557 mata kuliah yang terdata pada halaman tersebut. Namun, hanya mata kuliah STEI yang akan digunakan dalam tugas besar ini sehingga jumlah mata kuliah yang akan dimasukkan dalam ontologi adalah 676 mata kuliah. Teknik pengambilan data ini adalah dengan menggunakan *scraping* lalu diproses berkali-kali sehingga menghasilkan file excel yang dengan konfigurasi sebagai berikut:

1. Mengandung kolom kode mata kuliah, nama mata kuliah, dan SKS
2. Mata kuliah yang tersedia di file excel berasal dari STEI

Setelah file excel didapat. Kakas Cellfie digunakan supaya Protege bisa membaca file excel dan diproses menjadi sebuah data instansi di dalam ontologi yang sudah dibuat dalam Protege.

#### 2. Penulisan Gelar ITB

ITB pernah mengeluarkan sebuah peraturan yang menentukan gelar-gelar apa saja yang didapat ketika mahasiswa lulus program studi yang bersangkutan. Berikut adalah detail dari gelar-gelar yang didapat dari mahasiswa yang lulus di STEI.

87	STEI	Sarjana	Teknik Elektro	S.T.	Sarjana Teknik	<i>Bachelor of Science in Electrical Engineering</i>
88	STEI	Sarjana	Teknik Informatika	S.T.	Sarjana Teknik	<i>Bachelor of Science in Informatics/ Computer Science</i>
89	STEI	Sarjana	Teknik Tenaga Listrik	S.T.	Sarjana Teknik	<i>Bachelor of Science in Electrical Power Engineering</i>
90	STEI	Sarjana	Teknik Telekomunikasi	S.T.	Sarjana Teknik	<i>Bachelor of Science in Telecommunication Engineering</i>
91	STEI	Sarjana	Sistem dan Teknologi Informasi	S.T.	Sarjana Teknik	<i>Bachelor of Science in Information System and Technology</i>
92	STEI	Sarjana	Teknik Biomedis	S.T.	Sarjana Teknik	<i>Bachelor of Science in Biomedical Engineering</i>
93	STEI	Magister	Teknik Elektro	M.T.	Magister Teknik	<i>Master of Science in Electrical Engineering</i>
94	STEI	Magister	Informatika	M.T.	Magister Teknik	<i>Master of Science in Informatics</i>
95	STEI	Doktor	Teknik Elektro dan Informatika	Dr.	Doktor	<i>Doctor of Philosophy</i>

**Gambar 3 - Gelar Program Studi STEI**

Pada ontologi yang dibuat, gelar S.T., M.T., dan Dr. ditambah sesuai dengan strata yang bersangkutan (S.T. artinya masuk ke ontologi gelar S1).

## B. Gambaran umum dari ontologi

Pada domain SIX, ontologi hanya memiliki 3 tingkatan hierarki. Angka tingkatan tersebut didapat karena beberapa konsep yang dirumuskan pada ontologi secara umum memiliki level yang sama. Contohnya jenis penilaian, mata kuliah, program studi, dan lainnya memiliki hierarki yang sama. Kelas seperti BachelorDegree, MasterDegree, dan DoctoralDegree berada di tingkatan hierarki yang paling spesifik karena ini membedakan gelar-gelar yang dimiliki oleh S1, S2, dan S3. Berikut adalah gambaran dari kelas ontologi yang dibuat.

```

|- owl:Thing
  |- AcademicStructure
    |- College
    |- CourseCategory
    |- Courses
    |- Degree
      |- BacheloralDegree
      |- MasterDegree
      |- DoctoralDegree
    |- GradingCategory
    |- Majors
  |- AcademicTimes
    |- Semester
    |- Year

```

Berikut adalah detail dari sebuah instansi sebuah kelas. Pada sebuah instansi biasanya memiliki sebuah properti data. Berikut adalah penjelasan kelas yang merupakan subkelas dari AcademicStructure

### **1. CourseCategory**

Merupakan kelas yang berisi kategori dari sebuah mata kuliah. Kategori tersebut terdiri dari Kerja\_Praktik, Kuliah, Penelitian, dan Tugas\_Akhir / \_\_Tesis / \_\_Disertasi.

### **2. Courses**

Merupakan kelas yang berisi mata kuliah yang merupakan bagian dari STEI ITB. Untuk setiap mata kuliah, terdapat properti data sebagai berikut:

- a. hasCourseCredit: Jumlah SKS mata kuliah
- b. hasCourseID: Kode mata kuliah
- c. hasCourseName: Nama mata kuliah

### **3. Degree**

Merupakan kelas yang berisi strata dari sebuah perkuliahan yaitu S1, S2, dan S3. Tidak hanya itu, kelas Degree memiliki subkelas yang berisi gelar-gelar yang didapat ketika mahasiswa lulus dari ITB

- a. BacheloralDegree  
Kelas yang berisi gelar sarjana. Di STEI ITB, gelar yang didapat hanyalah satu yaitu S.T.
- b. MasterDegree  
Kelas yang berisi gelar magister. Di STEI ITB, gelar yang didapat hanyalah satu yaitu M.T.
- c. DoctoralDegree  
Kelas yang berisi gelar doktor. Di STEI ITB, gelar yang didapat hanyalah satu yaitu Dr.

### **4. GradingCategory**

Merupakan kelas yang berisi kategori penilaian dari sebuah mata kuliah. Hanya ada dua instansi dari kelas ini yaitu ABCDE (penilaian memiliki bobot yang mempengaruhi IP) dan Pass/Fail (penilaian tidak mempengaruhi IP sama sekali tetapi menentukan lulus atau tidaknya mahasiswa tersebut pada mata kuliah yang bersangkutan)

### **5. Majors**

Merupakan kelas yang berisi program studi yang ditawarkan oleh STEI ITB. Program studi terdiri dari S1 Teknik Elektro, S1 Teknik Informatika, TPB STEI-R, S1 Teknik Tenaga Listrik, S1 Teknik Telekomunikasi, S1 Sistem dan

Teknologi Informasi, S1 Teknik Biomedis, TPB STEI-K, S2 Teknik Elektro, S2 Informatika, S2 Sistem dan Teknologi Informasi, S3 Teknik Elektro dan Informatika

Berikut adalah penjelasan kelas yang merupakan subkelas dari AcademicTimes

**1. Semester**

Merupakan jenis semester perkuliahan (bukan nominal semester) mata kuliah tertentu bisa diambil. Jenis semester yaitu BothSemester yang artinya mahasiswa boleh mengambil mata kuliah tersebut kapan saja, EvenSemester atau OddSemester yang artinya mahasiswa hanya boleh mengambil mata kuliah saat semester genap atau ganjil. SummerSemester yang artinya mata kuliah tersebut hanya bisa diambil pada saat semester pendek saja.

**2. Year**

Merupakan jenis tingkatan mahasiswa. Tingkatan mahasiswa terdiri dari FirstYear, SecondYear, ThirdYear, dan FourthYear yang artinya tingkatan pertama, kedua, ketiga, dan keempat.



## **BAB III**

### **SISTEM BERBASIS PENGETAHUAN**

Sistem berbasis pengetahuan (KBS) yang dikembangkan mengikuti arsitektur yang memisahkan antara Terminological Box (TBox) dan Assertion Box (ABox). Dalam implementasi ini, ABox direpresentasikan oleh berkas ontologi college-course.rdf, sedangkan TBox (logika dan aturan) diimplementasikan menggunakan modul-modul Prolog (course\_rules.pl dan queries.pl).

#### **A. Fungsionalitas penting dari sistem berbasis pengetahuan yang dikembangkan**

Fungsionalitas utama dari sistem yang dikembangkan berfokus pada klasifikasi dan querying data mata kuliah berdasarkan aturan yang telah didefinisikan.

##### **1. Pemuatan Knowledge Base (ABox)**

Fungsionalitas paling fundamental adalah kemampuan untuk mem-parsing ontologi RDF dan mengubahnya menjadi fakta Prolog. Hal ini dilakukan oleh modul `rdf_loader.pl` melalui predikat `load_rdf_facts` yang membaca file `college-course.rdf`. Sistem menghasilkan database fakta (fakta ABox) di dalam memori Prolog, terutama dalam bentuk

- `course(ID, Name, Credits)`
- `degree(ID, Name)`

##### **2. Klasifikasi Mata Kuliah (TBox)**

Ini adalah fungsionalitas inti dari sistem berbasis pengetahuan. Modul `course_rules.pl` menerapkan serangkaian aturan TBox untuk menyimpulkan (infer) pengetahuan baru dari fakta ABox yang ada. Fungsionalitas klasifikasi ini meliputi:

- Klasifikasi Departemen/Prodi: Menentukan program studi (misalnya, 'Informatics', 'Electrical Engineering') berdasarkan parsing kode awalan mata kuliah (misalnya, 'IF', 'EL').
- Klasifikasi Jenjang (Level): Menentukan jenjang mata kuliah (misalnya, 'First Year', 'Second Year', 'Masters') berdasarkan parsing digit ketiga pada kode mata kuliah.
- Klasifikasi Tipe Mata Kuliah: Menentukan tipe fungsional mata kuliah (misalnya, 'Practical Course', 'Thesis Course', 'Elective Course') berdasarkan parsing kata kunci dalam nama mata kuliah (misalnya, "Praktikum", "Tugas Akhir").

- Klasifikasi Bobot SKS: Mengklasifikasikan mata kuliah sebagai 'Heavy' ( $\geq 4$  SKS) atau 'Light' ( $\leq 2$  SKS)

### 3. Querying

Modul `queries.pl` menyediakan antarmuka tingkat tinggi bagi pengguna untuk berinteraksi dengan *knowledge base* yang telah diklasifikasi. Fungsionalitas ini meliputi:

- **Pencarian Lanjut:** Memungkinkan pengguna untuk mencari mata kuliah berdasarkan nama (`search_course_by_name/2`), jenjang (`courses_at_level/2`), departemen (`courses_in_department/2`), atau jumlah SKS (`courses_with_credits/2`).
- **Kalkulasi Kredit:** Menyediakan utilitas untuk menghitung total SKS dari daftar mata kuliah (`total_credits/2`).
- **Pengecekan Kelulusan:** Sebuah fungsionalitas klasifikasi turunan (`meets_credit_requirement/2`) yang menentukan apakah seorang mahasiswa telah memenuhi batas SKS minimum.
- **Penyajian Informasi:** Menggabungkan semua fakta ABox dan TBox yang diturunkan ke dalam satu format yang mudah dibaca (`course_info/1`)

## B. Pemanfaatan Ontologi

Pemanfaatan ontologi adalah inti dari arsitektur sistem ini, yang terbagi menjadi pemanfaatan ABox (fakta) dan implementasi TBox (logika)

### 1. Pemanfaatan ABox (Data Fakta)

ABox, atau *Assertion Box*, adalah kumpulan fakta atau asersi mengenai individu. Dalam proyek ini, `college-course.rdf` berperan sebagai ABox. Modul `rdf_loader.pl` secara spesifik memanfaatkan ontologi dengan cara:

- Mengidentifikasi Individu: Mencari semua `owl:NamedIndividual` yang memiliki `rdf:type` menunjuk ke kelas `college-course:Courses`
- Mengekstrak Properti Data: Untuk setiap individu mata kuliah, sistem mengekstrak *Data Properties* yang relevan:
  - `college-course:hasCourseID`
  - `college-course:hasCourseName`
  - `college-course:hasCourseCredit`
- Membangun Knowledge Base: Fakta-fakta yang diekstraksi ini di-assert ke dalam memori Prolog menggunakan predikat `course(ID, Name,`

Credits). *Database* fakta inilah yang menjadi dasar pengetahuan (ABox) untuk semua logika Prolog.

## 2. Pemanfaatan TBox (Logika Klasifikasi)

Dalam proyek ini, terdapat sebuah skenario desain yang unik. Ontologi RDF mendefinisikan kelas-kelas TBox (seperti Year, CourseCategory, Semester) , namun data ABox (individu mata kuliah) tidak memiliki *Object Property* yang menautkan individu mata kuliah ke individu kategori tersebut .

Sebagai contoh, individu IF2110 tidak memiliki tautan hasCourseLevel ke individu SecondYear.

Akibat keterbatasan pada ABox ini, TBox tidak dapat dieksploitasi secara langsung dari ontologi. Oleh karena itu, TBox diimplementasikan secara logis di dalam Prolog melalui modul `course_rules.pl`.

Contoh Pemanfaatan TBox (Prolog):

1. Fakta ABox (dari RDF): `course('IF2110', 'Algoritma dan Pemrograman 2', 3)`.
2. Aturan TBox (dari `course_rules.pl`): `course_level(CourseID, 'Second Year') :- atom_chars(CourseID, [_ , _ , '2'|_])`.

Ketika pengguna menjalankan kueri `course_level('IF2110', L)`, *reasoner* Prolog menggabungkan Aturan TBox dengan Fakta ABox untuk menyimpulkan pengetahuan baru: `L = 'Second Year'`.

Dengan demikian, sistem ini sepenuhnya memenuhi arsitektur KBS:

- ABox disediakan oleh `college-course.rdf`.
- TBox disediakan oleh `course_rules.pl`.
- Inference Engine disediakan oleh *reasoner* bawaan Prolog.

## BAB IV

### PEMBAHASAN

#### A. Hasil dari sistem berbasis pengetahuan (bukti bahwa sistemnya bekerja sesuai yang dirancang)

##### 1. Init

Langkah pertama adalah memuat sistem utama (main.pl) dan menginisialisasi knowledge base (ABox) dengan menjalankan init/0.

```
?- init.  
Initializing system...  
Loading facts from 'facts/college-course.rdf'...  
Loading RDF file...  
% Parsed "college-course.rdf" in 0.11 sec; 3,496 triples  
RDF loaded into triple store.  
Loaded 676 courses  
Loaded 1 Bachelor, 1 Master, 1 Doctoral degrees  
RDF facts extracted successfully!  
Initialization complete. System is ready.  
Please run menu. to see the command list.  
  
true.
```

##### 2. Verifikasi Fakta ABox dan Aturan TBox

Setelah inisialisasi, kita dapat memverifikasi bahwa fakta (ABox) dan aturan (TBox) bekerja bersama. Kita menggunakan kueri course\_info/1 untuk melihat gabungan data mentah dari RDF dan data yang diturunkan (derived) dari aturan.

```
?- course_info('IF4070').  
  
Course ID: IF4070  
Course Name: Representasi Pengetahuan dan Penalaran  
Credits: 3  
Level (derived): Fourth Year  
Department (derived): Informatics  
Type (derived): Mandatory Course  
  
true.
```

##### 3. Fungsionalitas Pencarian dan Statistik

Sistem juga menyediakan fungsionalitas TBox yang lebih kompleks, seperti print\_statistics/0, yang mengagregasi data dari ABox dan TBox.

```
?- print_statistics.  
  
=== Knowledge Base Statistics ===  
Total Courses: 676  
Total Degrees: 3  
Practical Courses (derived): 41  
  
true.
```

##### 4. Validasi Sistem (Test Suite)

Fungsionalitas sistem divalidasi menggunakan test.pl

```
?- test_all.

=== RUNNING ALL TESTS ===
Checking data status...
â|| Data is already initialized.

TEST 1: Basic Course Queries
-----
1.1 Testing course retrieval...
    â|| Found course IF2110: Algoritma dan Pemrograman 2 (3 credits)
1.2 Testing course count...
    â|| Total courses in database: 676

TEST 2: Course Categorization
-----
2.1 Testing department classification (from ID)...
    â|| IF2110 is in: Informatics
2.2 Testing level classification (from ID)...
    â|| IF2110 level: Second Year
2.3 Testing practical course detection (from Name)...
    â|| IF4090 correctly identified as practical course
2.4 Testing thesis course detection (from Name)...
    â|| EB4090 correctly identified as thesis course

TEST 3: Credit Calculations
-----
3.1 Testing total credits calculation...
    â|| Total credits for IF2110, IF2120, IF2130: 9
3.2 Testing heavy course detection...
    â|| IF2150 correctly identified as heavy course

TEST 5: Search Functions
-----
5.1 Testing search by course name (Bhs Indonesia)...
    â|| Found 4 courses containing "Basis Data"
5.2 Testing courses at level (from ID)...
    â|| Found 0 second year courses
5.3 Testing courses in department (from ID)...
    â|| Found 185 Informatics courses

=== ALL TESTS COMPLETED ===
true.
```

## B. Keterlibatan logika deskripsi

Tugas ini mensyaratkan pemanfaatan fitur-fitur dari Logika Deskripsi (LD). Dalam proyek ini, keterlibatan LD dapat dianalisis dalam dua tahap: desain ontologi (TBox) dan implementasi sistem (Prolog).

### 1. Desain TBox dalam Ontologi

Pada tahap desain di Protégé, Logika Deskripsi digunakan untuk mendefinisikan terminologi (TBox) dari domain. Ontologi college-course.rdf mendefinisikan kelas-kelas (konsep) secara hierarkis 8.

- owl:Thing
  - AcademicStructure
    - Courses
    - Majors
    - CourseCategory
  - AcademicTimes
    - Semester

■ Year

Idealnya, dalam LD, kita akan mendefinisikan konsep baru menggunakan Object Property. Sebagai contoh, untuk mendefinisikan "Mata Kuliah Praktikum", kita akan menulis aksioma:

$$\text{MataKuliahPraktikum} \equiv \exists \text{hasCourseCategory. Kerja_Praktik}$$

## 2. Kesenjangan ABox dan Implementasi TBox di Prolog

Setelah diinspeksi, ditemukan bahwa ABox (data individu) dalam college-course.rdf tidak lengkap. Individu mata kuliah (misalnya IF4090) tidak memiliki Object Property yang menautkannya ke individu kategori (misalnya Kerja\_Praktik).

Akibat kesenjangan data pada ABox ini, TBox berbasis Logika Deskripsi yang ideal tidak dapat dieksekusi oleh *reasoner*.

Untuk mengatasi hal ini, TBox diimplementasikan secara prosedural di dalam Prolog (dalam modul `course_rules.pl`). Logika deskripsi yang seharusnya ada di ontologi, disimulasikan menggunakan aturan Prolog yang melakukan parsing string.

- **Logika Deskripsi (Ideal):**

$$\text{MataKuliahTingkatDua} \equiv \exists \text{exists hasCourseLevel. SecondYear}$$

- **Implementasi Prolog (Aktual):**

Aturan TBox di Prolog untuk mensimulasikan LD

```
course_level(CourseID, 'Second Year') :-  
    course(CourseID, _, _),  
    atom_chars(CourseID, [_, _, '2'|_]).
```

Dengan demikian, sistem ini melibatkan Logika Deskripsi pada tahap desain konseptual (TBox ontologi), namun pada tahap implementasi, peran TBox dan *inference engine* diambil alih oleh Prolog untuk mengatasi keterbatasan data ABox.

## C. Keterbatasan ontologi dan sistem berbasis pengetahuan

Evaluasi terhadap sistem yang dibangun mengidentifikasi beberapa keterbatasan utama:

### 1. Keterbatasan ABox pada Ontologi

Keterbatasan paling signifikan adalah ABox dalam college-course.rdf tidak memanfaatkan TBox yang telah dirancang. Individu mata kuliah tidak memiliki tautan *Object Property* ke CourseCategory, Year, atau Semester. Hal ini memaksa sistem Prolog untuk "menebak" klasifikasi menggunakan parsing string, bukan menggunakan penalaran ontologis murni.

## 2. Aturan TBox yang Rapuh (*Brittle Rules*)

Sebagai akibat dari keterbatasan ABox, implementasi TBox di `course_rules.pl` menjadi rapuh. Aturan-aturan ini sangat bergantung pada format ID dan nama mata kuliah:

- Jika format kode mata kuliah berubah (misal, IF-3170 bukan IF3170), aturan `course_level/2` dan `course_department/2` akan gagal.
- Jika nama mata kuliah "Tugas Akhir" diubah menjadi "Proyek Final", aturan `is_thesis_course/1` akan gagal mengklasifikasikannya.
- Sistem ini tidak dapat membedakan antara mata kuliah wajib dan pilihan yang tidak memiliki kata kunci "Pilihan" atau "Topik Khusus" di namanya.

## Kesimpulan

Domain SIX ITB bisa digunakan sebagai data-data yang dibutuhkan untuk membuat *Rule Based System* berbasis ontologi. Pengetahuan-pengetahuan yang bisa diambil dari SIX ITB antara lain adalah mata kuliah, program studi, tipe penilaian mata kuliah, kategori mata kuliah, dll. Tidak hanya itu, pengetahuan terkait gelar yang akan didapatkan setelah mahasiswa lulus program studi tertentu bisa didapat dengan mencari sebuah dokumentasi aturan mengenai gelar yang dibuat oleh ITB itu sendiri.

Ontologi yang dibuat memiliki 3 hirarki kelas. Setiap kelas merepresentasikan percabangan dari kelas induk. Setiap kelas juga memiliki instansi yang bisa dibuat supaya suatu data bisa direpresentasikan pada kelas tersebut. Pada suatu instansi, terdapat properti data yang bisa mendefinisikan karakteristik terhadap instansi sebuah kelas.

Berdasarkan *Rule Based System* yang dibuat, arsitektur TBox dan ABox digunakan dalam pengembangan *Rule Based System* tersebut. Ontologi juga digunakan dengan melakukan *parsing* file rdf ke Prolog. RBS bisa digunakan dengan memberikan *query* yang akan dijadikan bahan pencarian akan informasi yang dibutuhkan.



## Matrix Kontribusi

NIM	Tugas
13522068	Rule Based System Prolog, Laporan
13522110	Ontologi Protege, Laporan
13522118	Rule Based System Prolog, Laporan