

PENERAPAN OPERASI HIMPUNAN DAN FUNGSI AGREGASI PADA PERANCANGAN BASIS DATA ALUMNI UNIVERSITAS HALU OLEO

Aswani^{*1}, Natalis Ransi², Rahmat Ramadhan³

^{*1}Jurusan Matematika, FMIPA, Universitas Halu Oleo

^{2,3}Jurusan Teknik Informatika, Fakultas Teknik, Universitas Halu Oleo, Kendari

e-mail : ^{*1}aswani.ugm09@gmail.com, ²natalis.ransi@uho.ac.id, ³rahmat.ramadhan@uho.ac.id

Abstrak

Universitas Halu Oleo sedang membangun basis data universitas yang terintegrasi dengan sistem basis data terpusat. Salah satu data yang diintegrasikan adalah basis data alumni. Basis data alumni dapat dirancang dengan menggunakan sistem basis data relasional. Menurut [1] basis data relasional bisa menjadi alternatif karena memiliki keunggulan segi kapasitas penyimpanan. Beberapa teori dasar matematika misalnya operasi himpunan dapat digunakan untuk merancang maupun mengembangkan sebuah basis data. Selanjutnya fungsi agregasi dapat digunakan untuk melakukan *retrieve* informasi dari pada sebuah basis data. Pada penelitian ini akan dilakukan perancangan pada basis data alumni Universitas Halu Oleo dengan menerapkan pendekatan operasi himpunan dan fungsi agregasi

Kata kunci— Operasi Himpunan, Fungsi Agregasi, Basis Data Alumni

Abstract

Halu Oleo University is building an integrated university database with a centralized database system. One of the data that is integrated is the alumni database. The alumni database can be designed using a relational database system. According to [1] the relational database can be an alternative because it has the advantage of storage capacity. Some basic mathematical theories such as set operations can be used to design and develop a database. Furthermore, the agreement function can be used to retrieve information from a database. In this research will be done on the database design alumni of Halu Oleo University by applying the approach of set operation and aggregation function.

Keywords— Set Operation, Aggregation Function, Alumni Database

1. PENDAHULUAN

Saat ini Universitas Halu Oleo sedang membangun basis data universitas yang terintegrasi yang dikelola dengan sistem basis data terpusat. Salah satu data yang diintegrasikan adalah basis data alumni. Basis data alumni dapat dirancang dengan menggunakan sistem basis data relasional. Menurut [1] basis data relasional bisa menjadi alternatif karena memiliki keunggulan segi kapasitas penyimpanan.

Beberapa kasus pengembangan aplikasi yang menggunakan basis data sebagai penyimpanan data juga menghasilkan

pengembangan algoritma. Oleh [2] menyatakan bahwa pengembangan fungsi agregasi Fuzzy dapat digunakan sebagai pemeriksaan lengkap database relasional dan bahasa SQL. Selanjutnya [3] menyatakan bahwa strategi pembagian untuk *query points* dan perkiraan ANN *query* untuk kasus di mana jumlah *query* sangat besar. Eksperimen yang mereka lakukan dengan menggunakan dataset menunjukkan bahwa pendekatan yang diajukan memiliki tingkat kinerjanya yang baik.

Penelitian-penelitian tersebut menggunakan teori dasar matematika dalam pengembangan basis data dan implementasi

query. Beberapa teori dasar matematika misalnya operasi himpunan dapat digunakan untuk merancang maupun mengembangkan sebuah basis data. Selanjutnya fungsi agregasi dapat digunakan untuk melakukan *retrieve* informasi dari pada sebuah basis data. Pada penelitian ini akan dilakukan perancangan pada basis data alumni Universitas Halu Oleo dengan menerapkan pendekatan operasi himpunan dan fungsi agregasi.

Beberapa penelitian tentang fungsi agregasi maupun pengembanya telah dilakukan. Oleh [4] menyatakan bahwa fungsi agregasi dan operator GROUP BY pada SQL menghasilkan agregasi *zero-dimensional* atau *one-dimensional* atau lebih dikenal dengan *data cube* atau *simply cube*. Penelitiannya melaporkan tentang (1) penjelasan tentang *cube* dan operator *roll-up*, (2) implementasinya SQL, (3) menjelaskan bagaimana pengguna dapat mendefinisikan fungsi agregasi baru, dan (4) teknik yang efisien untuk perhitungan pada *data cube*. Selanjutnya [5] melaporkan hasil penelitian mereka tentang *query* pada data base dengan pertama-tama memberikan gambaran tentang ketidakpastian data, termasuk tipe ketidakpastian, model representasi probabilitas, dan sumber probabilitas. Hasil penelitian mereka juga menyajikan dan menganalisis beberapa *query* yang tidak pasti secara detail, seperti *skylinequery*, *top-k queries*, *nearest-neighbor queries*, *aggregate queries*, *join queries*, *range queries*.

Rumusan masalah pada penelitian ini adalah merancang basis data alumni Universitas Halu Oleo menggunakan operasi himpunan dan fungsi agregasi.

Tujuan penelitian ini adalah merancang basis data alumni Universitas Halu Oleo menggunakan operasi himpunan dan fungsi agregasi. Tahapan penelitian akan dimulai dari tahapan pengumpulan data, mendefinisikan struktur data pada basis data alumni, implementasi SQL menggunakan driver MySQL. Tujuan jangka panjang penelitian ini terintegrasinya sistem pengelolaan data alumni Universitas Halu Oleo. Sehingga dapat mendukung terwujudnya sebuah sistem pendukung pengambilan keputusan berbasis teknologi informasi yang handal.

2. LANDASAN TEORI

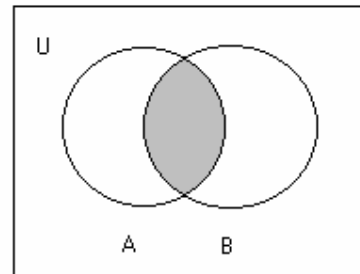
2.1 Operasi Himpunan

Bagian ini akan memberikan dasar teori tentang operasi pada himpunan. Himpunan itu sendiri didefinisikan sebagai data (objek) yang dikumpulkan berdasarkan kriteria tertentu.

Oleh [6] diberikan beberapa operasi himpunan antara lain irisan (*intersection*), gabungan (*union*), komplemen (*complement*). Operasi-operasi ini sering ditemukan pada perancangan sebuah basis data.

2.2 Irisan (*Intersection*)

Irisan antara dua buah himpunan dinotasikan oleh tanda ' \cap '. Misalkan A dan B adalah himpunan yang tidak saling lepas, maka $A \cap B = \{x \mid x \in A \text{ dan } x \in B\}$. Jika dinyatakan dalam bentuk diagram Venn dapat dilihat pada Gambar 1.



Gambar 1 Diagram Venn operasi Irisan (*Intersection*)

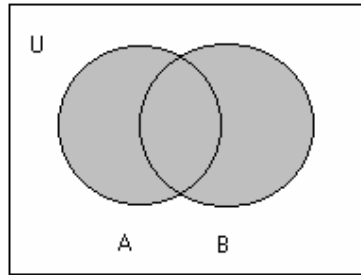
Contoh irisan :

Misalkan $A = \{2, 3, 5, 7, 11\}$ dan $B = \{3, 6, 9, 12\}$, maka $A \cap B = \{3\}$

Misalkan A adalah himpunan mahasiswa TI STT Telkom dan B merupakan himpunan wanita lanjut usia (50 tahun ke atas), maka $A \cap B = \emptyset$. Hal ini berarti A dan B adalah saling lepas atau $A // B$.

2.3 Gabungan (*Union*)

Gabungan antara dua buah himpunan dinotasikan oleh tanda ' \cup '. Misalkan A dan B adalah himpunan, maka $A \cup B = \{x \mid x \in A \text{ atau } x \in B\}$. Gambar 2 memperlihatkan Diagram Venn operasi Gabungan (*Union*).



Gambar 2 Diagram Venn operasi Gabungan (*Union*)

Contoh union :

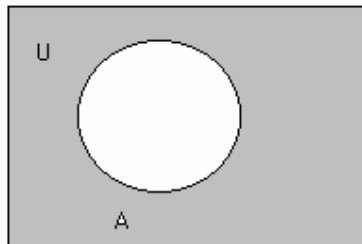
Jika $A = \{ 2, 3, 5, 7 \}$ dan $B = \{ 1, 2, 3, 4, 5 \}$, maka $A \cup B = \{ 1, 2, 3, 4, 5, 7 \}$, $A \cup \emptyset = A$.

2.4 Komplemen (*Complement*)

Komplemen dari suatu himpunan merupakan unsur-unsur yang ada pada himpunan universal (semesta pembicaraan) kecuali anggota himpunan tersebut. Misalkan A merupakan himpunan yang berada pada semesta pembicaraan U, maka komplemen dari himpunan A dinotasikan oleh:

$$\bar{A} = \{ x \mid x \in U \text{ dan } x \notin A \}$$

Gambar 3 memperlihatkan Diagram Venn operasi Komplemen (*Complement*).



Gambar 3 Diagram Venn operasi Komplemen (*Complement*)

Contoh komplemen :

Misalkan $U = \{ 1, 2, 3, \dots, 9 \}$,

jika $A = \{ 1, 3, 7, 9 \}$, maka $\bar{A} = \{ 2, 4, 5, 6, 8 \}$

jika $A = \{ x \in U \mid x \text{ habis dibagi dua} \}$, maka $\bar{A} = \{ 1, 3, 5, 7, 9 \}$.

2.5 Fungsi Agregasi

Dalam bidang basis data terdapat beberapa fungsi yang dipakai. Salah satunya adalah fungsi agregasi, yang biasanya diterapkan pada pemrograman *Structured Query Language* (SQL). Fungsi ini merupakan proses untuk mendapatkan nilai dari sekumpulan data yang telah dikelompokkan. Pengelompokan data didasarkan pada kolom

atau kombinasi kolom yang dipilih. Beberapa fungsi untuk agregasi dapat dilihat pada Tabel 1.

Tabel 1 Fungsi Agregasi [7]

No.	Nama Fungsi	Kegunaan
1.	SUM()	Untuk mendapatkan nilai total suatu atribut numerik hasil query
2.	AVG()	Untuk mendapatkan nilai rata-rata dari suatu atribut numerik hasil query
3.	MAX()	Untuk mendapatkan nilai terbesar dari suatu atribut hasil query
4.	MIN()	Untuk mendapatkan nilai terkecil dari suatu atribut hasil query
5.	COUNT()	Untuk mendapatkan nilai banyaknya record hasil query

Sebagai contoh Gambar 4 memperlihatkan fungsi agregasi yang digunakan untuk mencari nilai gaji tertinggi berdasarkan departemen.

```
SELECT MAX(gaji), departemen
FROM karyawan
GROUP BY departemen;
```

Gambar 4 Contoh Penggunaan Fungsi Agregasi

3. METODE PENELITIAN

3.1 Tahapan Penelitian

Tahapan penelitian akan dimulai dari tahapan pengumpulan data, mendefinisikan struktur data yang dibutuhkan pada basis data alumni dan implementasi SQL menggunakan driver MySQL.

3.2 Pengumpulan Data

Pengumpulan data mengenai hal-hal yang berkaitan dengan topik penelitian meliputi:

1. Observasi dilakukan langsung pada basis data Universitas Halu Oleo
2. Studi pustaka yaitu mempelajari tentang teori dasar operasi himpunan dan fungsi agregasi serta metode pada sistem basis data yang mendukung penelitian ini.

3.3 Pengembangan Sistem

Pada tahap ini dilakukan analisis sistem, analisa kebutuhan-kebutuhan perangkat lunak dari sistem yang akan dikembangkan. Setelah tahap analisis dikerjakan dan menghasilkan spesifikasi kebutuhan perangkat lunak kemudian dilakukan pemodelan sistem. Pemodelan ini dilakukan dengan menggunakan tool *Unified Modeling Language* (UML) yang terdiri dari pembuatan *use case*, *activity diagram*, *class diagram* dan *Entity Relationship Diagram* (ERD) untuk pemodelan database, DFD untuk pemodelan proses dan rancangan antar muka pengguna.

3.4. Lokasi Penelitian

Penelitian ini direncanakan berlokasi di Laboratorium Komputasi Matematika FMIPA UHO, dan Laboratorium Rekayasa Perangkat Lunak Program Studi Teknik Informatika Fakultas Teknik serta Laboratorium Komputer UPT. Teknologi Informasi dan Komunikasi, Universitas Halu Oleo.

3.5 Jenis Data penelitian

Data yang akan digunakan pada penelitian ini adalah data akademik Universitas Halu Oleo. Data tersebut berasal dari basis data Sistem Informasi Akademik

(SIKAD) Universitas Halu Oleo maupun Sistem informasi pendataan wisuda yang dikelola oleh UPT. Teknologi Informasi dan Komunikasi

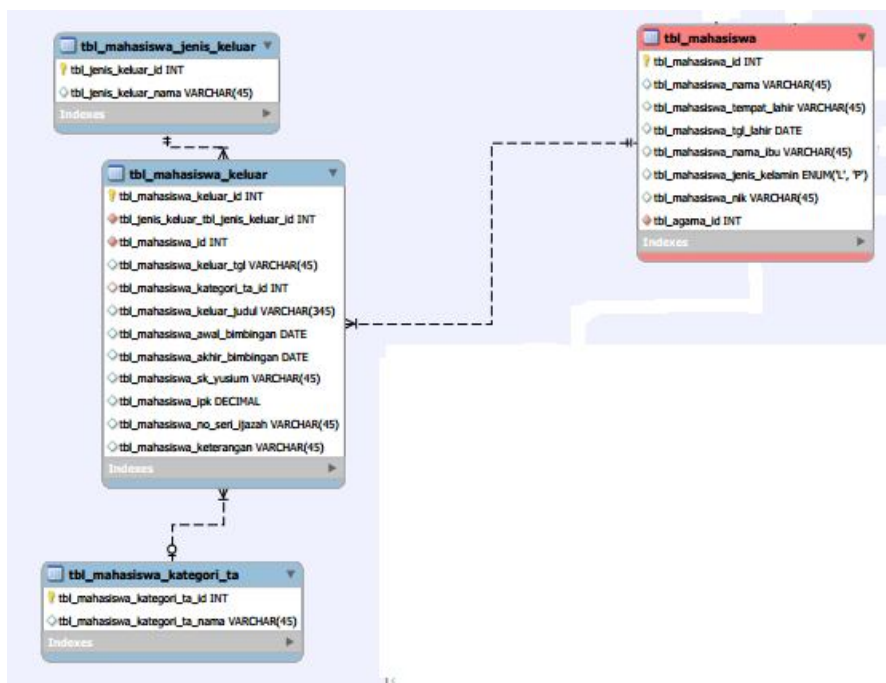
3.6 Model Data Warehouse

Rancang bangun *data warehouse* Universitas Halu Oleo akan menggunakan *model stars schemas*. Skema yang digunakan untuk pemodelan data adalah *starschema* dimana terdapat satu tabel fakta dan beberapa tabel dimensi. Penggunaan *star schema* memungkinkan proses *query* yang lebih ringan dan memudahkan penjelajahan terhadap data dimensinya. Tabel fakta yang terbentuk dari perancangan *data warehouse* ini merupakan tabel yang berhubungan dengan evaluasi kegiatan akademik.

4. HASIL DAN PEMBAHASAN

4.1 Rancangan ERD (*Entity Relational Diagram*) Data Alumni

Rancangan ERD data Alumni dibuat untuk menggambarkan bagaimana data antara table saling berelasi. Gambar 5 menunjukkan rancangan ERD data Alumni.



Gambar 5 Rancangan ERD Data Alumni

4.2 Implementasi Query

Informasi tentang Alumni dapat

diperoleh dari database dengan membuat query. Query merupakan perintah dalam

database untuk menampilkan hasil table atau relasi antar table. Berikut adalah beberapa query untuk mendapatkan informasi tentang Alumni :

- a. Query SQL menampilkan Data Alumni
Query SQL untuk menampilkan data Alumni ditunjukkan oleh Gambar 6.

```
SELECT m.tbl_mahasiswa_nama,
mk.tbl_mahasiswa_keluar_tgl,
mk.tbl_mahasiswa_judul,
mk.tbl_mahasiswa_ipk,
kta.tbl_mahasiswa_kategori_ta_nama,
jk.tbl_jenis_keluar_nama
FROM tbl_mahasiswa_jenis_keluar jk,
tbl_mahasiswa_kategori_ta kta,
tbl_mahasiswa_keluar mk,
tbl_mahasiswa m
WHERE mk.tbl_mahasiswa_id =
m.tbl_mahasiswa_id
AND
mk.tbl_jenis_keluar_tbl_jenis_keluar_id =
jk.tbl_jenis_keluar_id
AND mk.tbl_mahasiswa_kategori_ta_id =
kta.tbl_mahasiswa_kategori_ta_id
```

Gambar 6 Query SQL menampilkan Data Alumni

Query SQL untuk menampilkan data Alumni tersebut dibentuk berdasarkan aljabar relasional. Bentuk aljabar relasional untuk query tersebut yaitu

$$\prod_{a,b,c,d,e,f} (\sigma_{g=h \wedge i=j \wedge k=l} (PxQxRxS))$$

dimana

a = m.tbl_mahasiswa_nama
 b = mk.tbl_mahasiswa_keluar_tgl
 c = mk.tbl_mahasiswa_judul
 d = mk.tbl_mahasiswa_ipk
 e = kta.tbl_mahasiswa_kategori_ta_nama
 f = jk.tbl_jenis_keluar_nama
 g = mk.tbl_mahasiswa_id
 h = m.tbl_mahasiswa_id
 i = mk.tbl_jenis_keluar_tbl_jenis_keluar_id
 j = jk.tbl_jenis_keluar_id
 k = mk.tbl_mahasiswa_kategori_ta_id
 l = kta.tbl_mahasiswa_kategori_ta_id
 P = tbl_mahasiswa_jenis_keluar **jk**
 Q = tbl_mahasiswa_kategori_ta **kta**
 R = tbl_mahasiswa_keluar **mk**
 S = tbl_mahasiswa **m**

- b. Query SQL menampilkan Jumlah Alumni berdasarkan Tingkat Pendidikan (D3,S1, S2 dan S3)

Query SQL menampilkan Jumlah Alumni berdasarkan Tingkat Pendidikan ditunjukkan oleh Gambar 7.

```
SELECT
kta.tbl_mahasiswa_kategori_ta_id,
kta.tbl_mahasiswa_kategori_ta_nama,
COUNT(mk.tbl_jenis_keluar_tbl_jenis_keluar_id) AS Jumlah_Alumni
FROM tbl_mahasiswa_jenis_keluar jk,
tbl_mahasiswa_kategori_ta kta,
tbl_mahasiswa_keluar mk
WHERE
mk.tbl_jenis_keluar_tbl_jenis_keluar_id =
jk.tbl_jenis_keluar_id
AND mk.tbl_mahasiswa_kategori_ta_id =
kta.tbl_mahasiswa_kategori_ta_id

GROUP BY
kta.tbl_mahasiswa_kategori_ta_id,
kta.tbl_mahasiswa_kategori_ta_nama
```

Gambar 7 Query SQL menampilkan Jumlah Alumni berdasarkan Tingkat Pendidikan

Query SQL untuk menampilkan data Alumni tersebut dibentuk berdasarkan aljabar relasional. Bentuk aljabar relasional untuk query tersebut yaitu

$$a.b\mathcal{G}_c \left(\sigma_{i=j \wedge k=l} (PxQxR) \right)$$

dimana

a = kta.tbl_mahasiswa_kategori_ta_id
 b = kta.tbl_mahasiswa_kategori_ta_nama
 c = COUNT(mk.tbl_jenis_keluar_tbl_jenis_keluar_id)
AS Jumlah_Alumni
 i = mk.tbl_jenis_keluar_tbl_jenis_keluar_id
 j = jk.tbl_jenis_keluar_id
 k = mk.tbl_mahasiswa_kategori_ta_id
 l = kta.tbl_mahasiswa_kategori_ta_id
 P = tbl_mahasiswa_jenis_keluar **jk**
 Q = tbl_mahasiswa_kategori_ta **kta**
 R = tbl_mahasiswa_keluar **mk**

Dengan menggunakan query tersebut dapat diketahui secara bersamaan banyaknya Alumni D3, S1, S2 dan S3 pada Universitas Halu Oleo. Untuk mengetahui banyaknya Alumni D3, S1, S2 dan S3 pada Universitas Halu Oleo pada periode Wisuda tertentu dapat diperoleh berdasarkan query tersebut dengan menambahkan syarat tanggal Wisuda. Adapun querynya ditunjukkan oleh Gambar 8.

```
SELECT
kta.tbl_mahasiswa_kategori_ta_id,
kta.tbl_mahasiswa_kategori_ta_nama,
COUNT(mk.tbl_jenis_keluar_tbl_jenis_keluar_id) AS Jumlah_Alumni
FROM tbl_mahasiswa_jenis_keluar jk,
tbl_mahasiswa_kategori_ta kta,
tbl_mahasiswa_keluar mk
WHERE
mk.tbl_jenis_keluar_tbl_jenis_keluar_id =
jk.tbl_jenis_keluar_id
```

```

AND mk.tbl_mahasiswa_kategori_ta_id =
kta.tbl_mahasiswa_kategori_ta_id
AND mk.tbl_mahasiswa_keluar_tgl =
'2017-10-30'

GROUP BY
kta.tbl_mahasiswa_kategori_ta_id,
kta.tbl_mahasiswa_kategori_ta_nama

```

Gambar 8 Query SQL menampilkan Jumlah Alumni berdasarkan Tingkat Pendidikan pada Periode Wisuda Tertentu

Bentuk aljabar relasional untuk menampilkan jumlah Alumni pada periode Wisuda tertentu yaitu

$a,bG_c(\sigma_{i=j \wedge k=l \wedge m='2017-10-30'}(PxQxR))$
dimana

a = kta.tbl_mahasiswa_kategori_ta_id
 b = kta.tbl_mahasiswa_kategori_ta_nama
 c = COUNT(mk.tbl_jenis_keluar.tbl_jenis_keluar_id)
AS Jumlah_Alumni
 i = mk.tbl_jenis_keluar.tbl_jenis_keluar_id
 j = jk.tbl_jenis_keluar_id
 k = mk.tbl_mahasiswa_kategori_ta_id
 l = kta.tbl_mahasiswa_kategori_ta_id
 m = mk.tbl_mahasiswa_keluar_tgl
 P = tbl_mahasiswa_jenis_keluar **jk**
 Q = tbl_mahasiswa_kategori_ta **kta**
 R = tbl_mahasiswa_keluar **mk**

5. KESIMPULAN

Kesimpulan dalam penelitian ini yaitu basis data alumni Universitas Halu Oleo yang dibangun menggunakan operasi himpunan dan fungsi agregasi menjadikan informasi yang dibutuhkan terkait data alumni dapat diperoleh dan relasi tabel dibuat lebih efisien.

6. SARAN

Saran dari penelitian ini untuk penelitian selanjutnya yaitu untuk studi kasus dapat diintegrasikan dengan basis data keuangan dan basis data kepegawaian.

DAFTAR PUSTAKA

- [1] Funderburk, J. E., Kiernan, G., Shanmugasundaram, J., Sheki-ta, E., dan Wei, C. 2002, XTABLES: Bridging relational technology and XML. *IBM Systems Journal*, 41(4).
- [2] Aguilera, A., Mata-Toledo, R., Subero, A., Monger, M., dan Gupta, P. 2013, On an Extension of Fuzzy Aggregate Functions For Databases. *Journal of Information Systems & Operations Management*, , 1-8.
- [3] Sun, W., Chen, C., Zhu, L., Gao, Y., Jing, Y., dan Li, Q, 2015, On efficient aggregate nearest neighbor query processing in road networks. *Journal of Computer Science and Technology*, 30(4), 781-798.
- [4] Gray, J., Chaudhuri, S., Bosworth, A., Layman, A., Reichart, D., Venkatrao, M., dan Pirahesh, H., 1997, Data cube: A relational aggregation operator generalizing group-by, cross-tab, and sub-totals. *Data Mining and Knowledge Discovery*, 1(1), 29-53.
- [5] Wang, Y., Li, X., Li, X., dan Wang, Y., 2013, A survey of queries over uncertain data. *Knowledge and Information Systems*, 37(3), 485-530.
- [6] Adiwijaya. 2017. Matematika Diskrit. Sekolah Tinggi Teknologi Telkom <http://ishaq.staff.gunadarma.ac.id/Downloads/files/43844/Chapter+1.+Himpunan.pdf>, diakses tanggal 27 Agustus 2017.
- [7] Fatansyah. 2012. *Basis Data*. Informatika Bandung, Indonesia.