

BAB IV

HASIL DAN PEMBAHASAN

4.1 Subjek Penelitian

Subjek penelitian merujuk pada kebutuhan data proses akreditasi mengenai lulusan yaitu data pelacakan studi UMY. Subjek penelitian ini difokuskan pada data alumni pelacakan studi tahun 2001 sampai 2014.

4.2 Analisis Kebutuhan

Pada penelitian ini analisis kebutuhan berasal dari database dari Biro Sistem Informasi, database tersebut diberi nama `cdc_web`, kemudian database tersebut dianalisis dengan cara melihat pengisian kuisioner dari pelacakan studi yang diajukan oleh alumni-alumni UMY, dengan melihat kuisioner tersebut data yang menjadi subjek penelitian dengan mudah diketahui fungsi dari masing-masing pertanyaan yang diajukan dalam kuisioner tersebut, kemudian untuk mendapatkan informasi kebutuhan penelitian dilakukan analisis pada data tersebut agar nantinya data tersebut dengan mudah dapat dibentuk sebagai pelaporan yang jelas tentang data alumni yang mengarah pada pertanyaan-pertanyaan dari sumbernya. Deskripsi serta analisis kebutuhan mengenai database `cdc_web` tersebut dapat dilihat pada tabel 4.1.

Tabel 4.1 Deskripsi dan Analisis basis data `cdc_web`

Nama Tabel	Deskripsi	Keterangan
<code>dbo.agenda</code>	Berisi data agenda alumni yang berisi id, title dan content, dalam data ini tidak terdapat informasi apapun yang berarti data kosong (tidak ada isi)	Tidak Diperlukan
<code>Dbo.alumni</code>	Berisi data mengenai informasi tentang alumni seperti nama, nim, tempat lahir, prodi, jabatan alumni, dsb	Diperlukan

Dbp.alumni	Berisi data mengenai informasi tentang alumni seperti nama, nim, tempat lahir, prodi, jabatan alumni, dsb	Diperlukan
Dbp.alumni_computer_competetion	Berisi data kolom id alumni dan id komputer	Tidak Diperlukan
Dbp.alumni_language_competition	Berisi data kolom id alumni dan id language	Tidak Diperlukan
Dbp.fakultas	Berisi data id fakultas beserta nama fakultas	Diperlukan
Dbp.jenis_pekerjaan	Berisi data id yang mengarah pada Id keterangan pekerjaan contohnya belum bekerja, Pemerintah, Swasta, Wiraswasta, Dll	Diperlukan
Dbp.job_position	Berisi kolom id yang mengarah pada posisi pekerjaan, Title yang berisi informasi tentang bagaimana mendapatkan pekerjaan contohnya lowongan kerja, dan Tanggal diterima kerja	Diperlukan
Dbp.news	Berisi kolom title yang menjelaskan tentang informasi atau berita mengenai lowongan pekerjaan	Tidak Diperlukan
Dbp.news_category	Berisi tentang keterangan informan yang mengisi data dari dbp.news seperti info alumni, info rekrutmen, pengumuman, dll	Tidak Diperlukan
Dbp.prodi	Berisi id Fakultas beserta Nama Fakultas	Diperlukan

Dboprofile	Berisi mengenai konten yang ada pada isian web seperti title sambutan ketua, sejarah, visi dan misi, dll	Tidak Diperlukan
Dbosalary_range	Berisi id Range dan Keterangan Range atau jumlah gaji	Diperlukan
Dbosend_jobs	Berisi data id yang mengarah pada id alumni, id job posisi, dan tanggal pengirimannya	Diperlukan
Dbouser	Berisi username dan password dari data alumni yang mengisi data dari pelacakan studi	Diperlukan
Dbouserrole	Berisi data User Role Id beserta keterangan dari dbouser seperti Administrator, Back Office, User, dan Alumni	Diperlukan

Sebelum melakukan perancangan pangkalan data tahap yang dilakukan adalah mendefinisikan kebutuhan terlebih dahulu. Kebutuhan pengguna yang dimaksud dalam penelitian ini adalah data lulusan alumni yang dibutuhkan dalam proses akreditasi, untuk mengetahui kebutuhan pengguna terhadap proses akreditasi dilakukan metode wawancara. Metode wawancara menghasilkan kebutuhan data yang dapat dilihat pada tabel 4.2. Terdapat data yang tidak tersedia dalam sistem, seperti masa tunggu untuk mendapatkan pekerjaan, data tersebut nantinya akan dilakukan perhitungan kesesuaian antara bulan lulus dan tanggal mulai kerja.

Tabel 4.2 Hasil Wawancara berupa kebutuhan data

Kategori	Jenis Informasi
Data lulusan Alumni	Profil Alumni
	IPK Alumni

Kategori	Jenis Informasi
	Masa tunggu untuk mendapatkan pekerjaan
	Gaji yang diterima (besarannya)
	Kesesuaian program studi dengan pekerjaan

4.3 Perancangan Pangkalan Data

Perancangan pangkalan data merupakan proses untuk menentukan isi dan pengaturan data yang nantinya akan dibutuhkan sebagai pendukung perancangan system. Sumber data yang digunakan dalam perancangan ini berasal dari tabel-tabel basis data pelacakan studi Universitas Muhammadiyah Yogyakarta. Berdasarkan pada hasil analisis kebutuhan, basis data yang digunakan adalah *cdc_web* yang telah dilakukan pemilihan data seperti yang dijelaskan pada tabel 1 diatas.

4.3.1 Pembuatan data Stage

Pangkalan data dengan arsitektur NDS+DDS ini memiliki *data store* yaitu Stage, NDS, dan DDS. Stage adalah basis data baru yang digunakan sebagai tempat persiapan data sebelum dilakukan proses ETL kedalam *data store* selanjutnya. Pada proses ETL menggunakan *Basic Transformation* sebagai transformasinya. Hasil dari analisis Kebutuhan ada beberapa data yang diperlukan dan tidak diperlukan, maka didalam stage ini berisi data yang diperlukan pada proses pembuatan pangkalan data selanjutnya. Stage berfungsi sebagai *data store* yang pertama pada arsitektur ini yang memuat data dari sumber data ke penyimpanan lokal. Data yang dimasukkan kedalam Stage belum dilakukan perubahan data atau transformasi Pada Stage ini dilakukan perubahan nama data yang dijelaskan pada Tabel 4.3.

Tabel 4.3 Perubahan nama sumber pada *Stage*

Sumber Data	Stage
dbo.fakultas	dbo.stage_fakultas
dbo.jenis_pekerjaan	dbo.stage_jenis_pekerjaan

dbo.prodi	dbo_stage_prodi
dbo.salary_range	dbo.stage_salary_range
dbo.send_jobs	dbo.stage_send_jobs
dbo.UserRole	dbo.stage_UserRole
dbo.user	dbo.stage_user
dbo.job_position	dbo.stage_job.position
dbo.alumni	dbo.stage_alumni

4.3.2 Pembuatan data *Normalized Data Store* (NDS)

4.3.2.1 Proses ETL NDS

Dalam proses Pembuatan *Normalized Data Store* (NDS) menggunakan data yang diambil dari data stage yang sebelumnya telah dibuat, ada beberapa kriteria yang perlu dilakukan persiapan, salah satunya adalah proses Extract, Transform, Load (ETL) dilakukan beberapa pemeriksaan terhadap data-datanya, tipe data yang digunakan dalam tabel, noise data yang ada dalam tabel. Noise data dicari dengan menggunakan query yang terdapat pada SQL Server, Noise data berupa null atau tidak terdapat isi data, duplikasi data, data yang tidak lengkap, data yang bersifat typo (salah dalam penggunaan kata), dan lain sebagainya. Proses pembuatan NDS ini akan dijelaskan satu persatu antar tabel yaitu sebagai berikut :

- NDS Fakultas

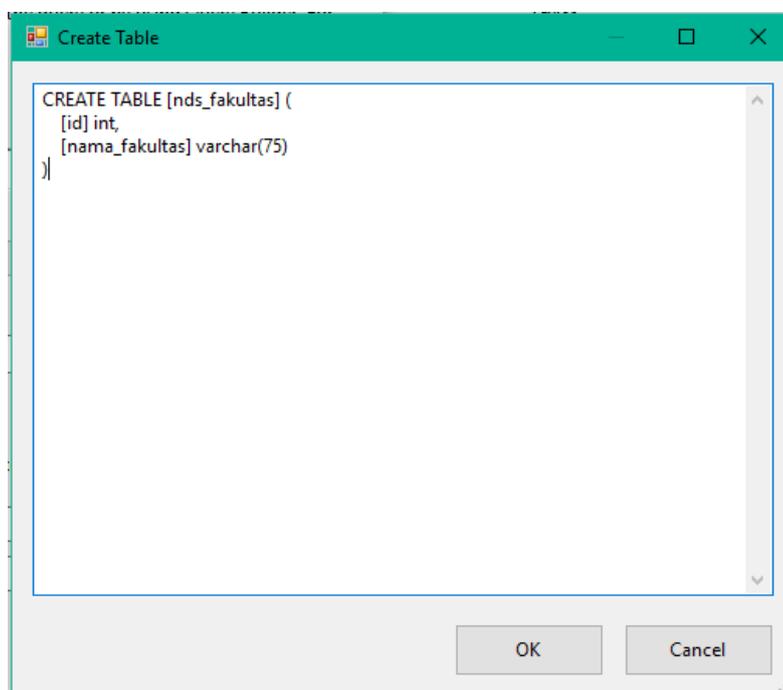
Tabel dbo.stage_fakultas memiliki 2 kolom dengan 11 data mengenai fakultas. Hasil pemeriksaan kolom dapat dilihat pada tabel 4.4

Tabel 4.4 Hasil pemeriksaan tabel dbo.stage_fakultas

Nama Kolom	Tipe Data	Data Noise	Keterangan
id	PK, int	0	Diperlukan
Nama_fakultas	Varchar(75)	0	Diperlukan

Pada tabel 4.4 data noise tidak ditemukan pada kedua kolom dari tabel tersebut, sehingga semua data tetap digunakan pada *data store*

selanjutnya karena semua kolom memiliki informasi yang diperlukan, didalam proses ini tidak terjadi proses ETL oleh sebab itu pada proses NDS ini dimulai dari menentukan tabel input, yaitu tabel pada stage yang akan ditransformasikan ke *data store* NDS. Pada OLE DB Source mengambil data source dari data dbo.stage_fakultas selanjutnya dipindahkan kedalam tabel NDS, sebelum mengambil tabel tersebut dibuat terlebih dahulu koneksi ke basis data pada SQL Server. Tabel dbo.stage_fakultas pada Stage diambil menggunakan query seperti pada Gambar 4.1



Gambar 4.1 Query proses pembuatan data baru nds_fakultas

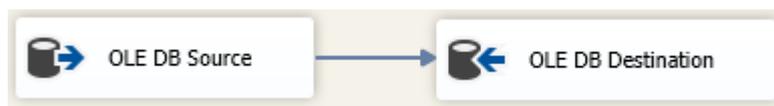
- NDS Jenis_pekerjaan

Tabel dbo.stage_jenis_pekerjaan memiliki 2 kolom dengan 8 data mengenai jenis pekerjaan. Hasil pemeriksaan kolom dapat dilihat pada tabel 4.5.

Tabel 4.5 Hasil pemeriksaan tabel dbo.stage_jenis_pekerjaan

Nama Kolom	Tipe Data	Data Noise	Keterangan
Id	PK, int	0	Diperlukan
keterangan	Varchar(75)	0	Diperlukan

Tidak ditemukan data noise pada kedua tabel tersebut, sehingga semua data tetap digunakan pada *data store* yang selanjutnya karena semua kolom memiliki informasi yang diperlukan. Pada proses ini tidak terjadi proses ETL oleh sebab itu pada proses NDS ini DB Source `dbo.stage_jenis_pekerjaan` dipindah ke dalam DB Destination menjadi `nds_jenis_pekerjaan`, proses pemindahan DB Source ke DB Destination melakukan proses yang sama pada proses sebelumnya, Hasil dari proses ini dapat dilihat pada gambar 4.2.



Gambar 4.2 Proses NDS tabel `dbo.stage_jenis_pekerjaan`

- NDS Prodi

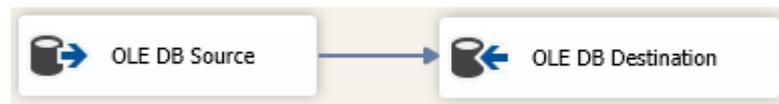
Tabel `dbo.stage_prodi` memiliki 3 kolom dengan 38 data mengenai prodi. Hasil pemeriksaan kolom dapat dilihat pada tabel 4.6.

Tabel 4.6 Hasil pemeriksaan tabel `dbo.stage_prodi`

Nama Kolom	Tipe Data	Data Noise	Keterangan
Id	PK, int	0	Diperlukan
Id_fakultas	Int	0	Diperlukan
Nama_prodi	Varchar(75)	0	Diperlukan

Tidak ditemukan data noise pada ketiga tabel tersebut, sehingga semua data tetap digunakan pada *data store* yang selanjutnya karena semua kolom memiliki informasi yang diperlukan. Pada proses ini

tidak terjadi proses ETL oleh sebab itu pada proses NDS ini DB Source `dbo.stage_prodi` dipindah ke dalam DB Destination menjadi `nds_prodi`, proses pemindahan DB Source ke DB Destination melakukan proses yang sama dengan proses sebelumnya yaitu menggunakan query, hasil dari proses pada pembuatan tabel NDS ini dapat dilihat pada gambar 4.3.



Gambar 4.3 Proses NDS tabel `dbo.stage_prodi`

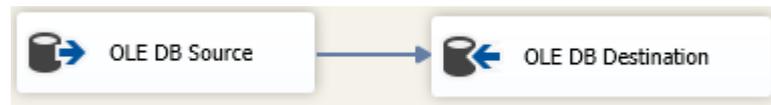
- NDS `Salary_range`

Tabel `dbo.stage_salary_range` memiliki 2 kolom dengan 8 data mengenai jumlah besar dan kecilnya range (gaji). Hasil pemeriksaan kolom dapat dilihat pada tabel 4.7.

Tabel 4.7 Hasil pemeriksaan tabel `dbo.stage_salary_range`

Nama Kolom	Tipe Data	Data Noise	Keterangan
Id	PK, int	0	Diperlukan
range	Varchar(75)	0	Diperlukan

Tidak ditemukan data noise pada ketiga tabel tersebut, sehingga semua data tetap digunakan pada *data store* yang selanjutnya karena semua kolom memiliki informasi yang diperlukan. Pada proses ini tidak terjadi proses ETL oleh sebab itu pada proses NDS ini DB Source `dbo.stage_salary_range` dipindah ke dalam DB Destination menjadi `nds_salary_range`, proses pemindahan DB Source ke DB Destination melakukan proses yang sama dengan proses sebelumnya yaitu menggunakan query, hasil dari proses pada pembuatan tabel NDS ini dapat dilihat pada gambar 4.4.



Gambar 4.4 Proses NDS tabel dbo.stage_salary_range

- NDS Send_jobs

Tabel dbo.stage_send_jobs memiliki 4 kolom dengan 309 data mengenai data id yang merujuk pada data alumni dan data job_position. Hasil pemeriksaan kolom dapat dilihat pada tabel 4.8. Tidak ditemukan data noise pada keempat tabel tersebut, sehingga tidak ada perubahan data pada tabel dbo.stage_send_jobs ini, sehingga semua data tetap digunakan pada *data store* yang selanjutnya karena semua kolom memiliki informasi yang diperlukan

Tabel 4.8 Hasil pemeriksaan tabel dbo.stage_send_jobs

Nama Kolom	Tipe Data	Data Noise	Keterangan
Id	PK, int	0	Diperlukan
Id_alumni	Int	0	Diperlukan
Id_job_position	Int	0	Diperlukan
Date_of_send	Datetime	0	Diperlukan

Pada proses ini tidak terjadi proses ETL oleh sebab itu pada proses NDS ini DB Source dbo.stage_send_jobs dipindah ke dalam DB Destination menjadi nds_send_jobs, proses pemindahan DB Source ke DB Destination melakukan proses yang sama dengan proses sebelumnya yaitu menggunakan query, hasil dari proses pada pembuatan tabel NDS ini dapat dilihat pada gambar 4.5.



Gambar 4.5 Proses NDS tabel dbo.stage_send_jobs

- NDS UserRole

Tabel `dbo.stage_UserRole` memiliki 2 kolom dengan 4 data mengenai data yang mengarah pada keterangan dari `dbo.stage_user`. Hasil pemeriksaan kolom dapat dilihat pada tabel 4.9.

Tabel 4.9 Hasil pemeriksaan tabel `dbo.stage_UserRole`

Nama Kolom	Tipe Data	Data Noise	Keterangan
UserRoleId	PK, int	0	Diperlukan
UserRoleName	Varchar(50)	0	Diperlukan

Tidak ditemukan data noise pada tabel 4.9, sehingga semua data tetap digunakan pada *data store* yang selanjutnya karena semua kolom memiliki informasi yang diperlukan. Pada proses ini tidak terjadi proses ETL oleh sebab itu pada proses NDS ini DB Source `dbo. stage_UserRole` dipindah ke dalam DB Destination menjadi `nds_ UserRole`, proses pemindahan DB Source ke DB Destination melakukan proses yang sama dengan proses sebelumnya yaitu menggunakan query, hasil dari proses pada pembuatan tabel NDS ini dapat dilihat pada gambar 4.6.



Gambar 4.6 Proses NDS tabel `dbo.stage_UserRole`

- NDS User

Tabel `dbo. stage_user` memiliki 8 kolom dengan 15 data mengenai data user . Hasil pemeriksaan kolom dapat dilihat pada tabel 4.10.

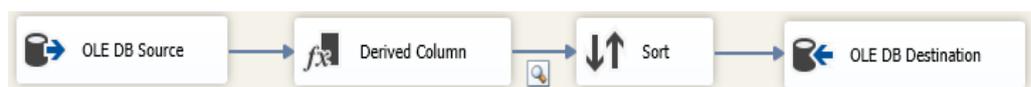
Tabel 4.10 Hasil pemeriksaan tabel `dbo.stage_user`

Nama Kolom	Tipe Data	Data Noise	Keterangan
Id	PK, int	0	Diperlukan
Username	Varchar(25)	0	Diperlukan

Nama Kolom	Tipe Data	Data Noise	Keterangan
Password	Varchar(250)	0	Diperlukan
nama	Varchar(75)	0	Diperlukan
Alamat	Varchar(250)	4	Diperlukan
Profile	Varchar(max)	15	Tidak Diperlukan
Role	Varchar(10)	0	Diperlukan
UserRoleId	int	4	Diperlukan

	id	username	nama	alamat	profile	role	UserRoleId
1	2	admin	Administrator	NULL	NULL	Admin	1
2	3	wahyu	Wahyu Fran Sudomo	NULL	NULL	Admin	1
3	5	MasBas	Muhammad Basuki	UMY	NULL	Admin	1
4	6	triyana	Triyana	Bausasran DN 3/560 Yogyakarta	NULL	Admin	1
5	9	bsi	User Tes	NULL	NULL	Admin	1
6	10	Anin	Anindita	CDC UMY	NULL	Admin	1
7	11	Fitri	Nia	Ilmu Komunikasi UMY	NULL	Admin	1
8	12	trista	CV. Tristanisaahmada	Gamping, Sleman, Yogyakarta	NULL	Perusahaan	NULL
9	13	okta	Oktavia	Tanubayan, Bantul	NULL	Admin	1
10	14	cholid	Cholid Fadliah	Ketanggungan, Brebes	NULL	Fakultas	NULL
11	15	khairul	Khairul Anwar	Jurusan IP Fisipol UMY	NULL	Admin	1
12	16	Purwadi	Purwadi	Biro SDM UMY	NULL	Admin	1
13	17	titiek	Titiek Nur Hanifah	CDC UMY	NULL	Admin	1
14	19	Mita	Ramita Paraswati	Bantul	NULL	Admin	NULL
15	22	Amir	Amir Hamzah	NULL	NULL	Admin	NULL

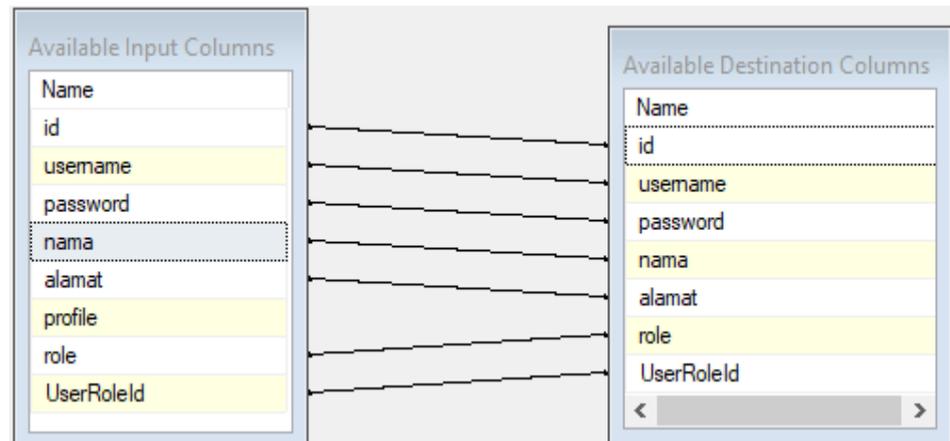
Gambar 4.7 Data keseluruhan pada dbo.stage_User



Gambar 4.8 Proses NDS tabel dbo.stage_User

Data Stage yang ditunjukkan pada Gambar 4.7 ditemukan data noise pada kolom Alamat, Profile, dan UserRoleId. Noise pada kolom tersebut bernilai null. Pada proses ini terjadi proses ETL seperti yang terlihat pada gambar 4.8. Pada tools Derived Column terjadi perubahan data, seperti perubahan data Alamat dan data UserRoleId, data Alamat pada proses selanjutnya sangat penting untuk dijadikan

informasi dari User maka digunakan *Transaction Basic Transformations* dengan Cleaning data noise pada kolom Alamat diganti dengan “-“ karena memungkinkan untuk tidak memanipulasi datanya dan mengurangi data yang bersifat null. Data null pada kolom UserRoleId diganti dengan nilai yang merujuk pada kolom Role, Role Admin pada UserRoleId bernilai 1, Role Fakultas pada UserRoleId diganti dengan nilai 2, dan Role Perusahaan pada UserRoleId diganti dengan nilai 3. Sedangkan pada kolom Profile informasinya tidak diperlukan pada penelitian ini, maka kolom tersebut dihapus pada *data store* selanjutnya, seperti yang ditunjukkan pada gambar 4.9, pada kolom profile tidak mempunyai tujuan ke dalam DB Destination. Pada proses NDS ini terjadi proses Sorting pada kolom ID dengan menggunakan tipe sorting ascending agar pencarian Id lebih mudah, kemudian hasil akhir DB Source `dbo.stage_user` dipindah ke dalam DB Destination menjadi `nds_user` dapat dilihat hasil isi tabel databasenya pada Gambar 4.10.



Gambar 4.9 Mapping tabel `dbo.stage_user`

	id	username	nama	alamat	role	UserRoleid
1	2	admin	Administrator	-	Admin	1
2	3	wahyu	Wahyu Fran Sudomo	-	Admin	1
3	5	MasBas	Muhammad Basuki	UMY	Admin	1
4	6	triyana	Triyana	Bausasran DN 3/560 Yogyakarta	Admin	1
5	9	bsi	User Tes	-	Admin	1
6	10	Anin	Anindita	CDC UMY	Admin	1
7	11	Fitri	Nia	Ilmu Komunikasi UMY	Admin	1
8	12	trista	CV. Tristanisaahmada	Gamping, Sleman, Yogyakarta	Perusahaan	3
9	13	okta	Oktavia	Tanubayan, Bantul	Admin	1
10	14	cholid	Cholid Fadilah	Ketanggungan, Brebes	Fakultas	2
11	15	khairul	Khairul Anwar	Jurusan IP Fisipol UMY	Admin	1
12	16	Purwadi	Purwadi	Biro SDM UMY	Admin	1
13	17	titiek	Titiek Nur Hanifah	CDC UMY	Admin	1
14	19	Mita	Ramita Paraswati	Bantul	Admin	1
15	22	Amir	Amir Hamzah	-	Admin	1

Gambar 4.10 Data keseluruhan pada dbo.nds_user

- NDS Job_position

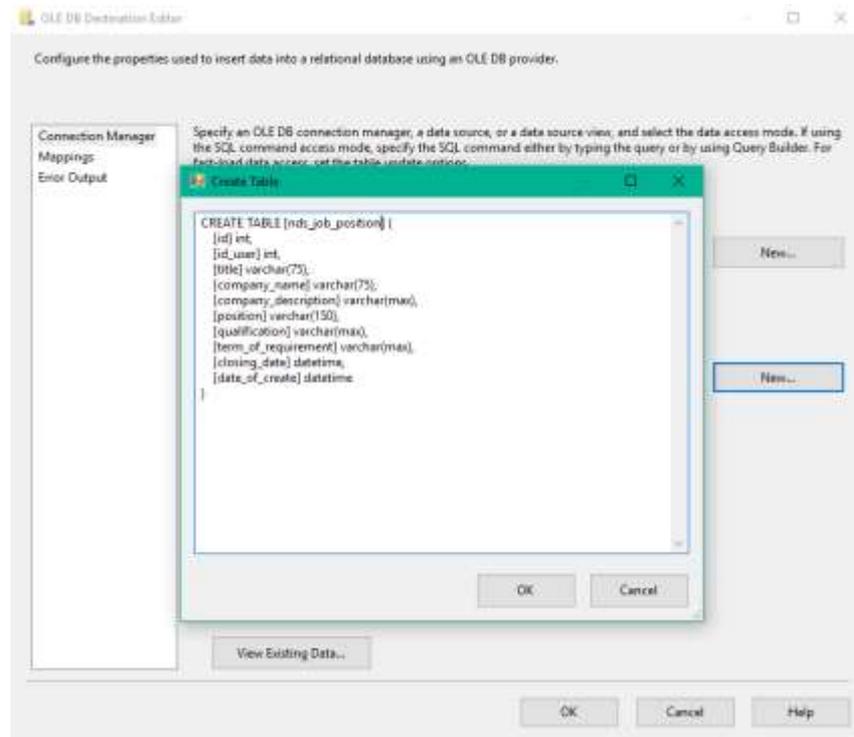
Tabel dbo.stage_job_position mempunyai 10 kolom dengan 494 data mengenai posisi pekerjaan alumni. Hasil pemeriksaan kolom dapat dilihat pada tabel 4.11.

Tabel 4.11 Hasil pemeriksaan tabel dbo.stage_job_position

Nama Kolom	Tipe Data	Data Noise	Keterangan
Id	PK, int	0	Diperlukan
Id_user	Int	0	Diperlukan
title	Varchar(75)	0	Tidak Diperlukan
Company_name	Varchar(75)	11	Diperlukan
Company_description	Varchar(max)	371	Tidak Diperlukan
position	Varchar(150)	4	Diperlukan
qualification	Varchar(max)	4	Tidak Diperlukan
Term_of_requirement	Varchar(max)	164	Tidak Diperlukan

Nama Kolom	Tipe Data	Data Noise	Keterangan
Closing_date	Datetime	125	Tidak Diperlukan
Date_of_create	Datetime	0	Diperlukan

Pada tabel 4.11 ini ditemukan data noise pada kolom company_name, company_description, position, qualification, term_of_requirement, dan closing_date. Data noise pada kolom tersebut berupa null. data noise pada kolom company_name dan position diganti dengan “-“ agar mengurangi data yang bersifat null (kosong). Sedangkan untuk data yang tidak diperlukan dibiarkan seperti data asli (sumbernya) karena data tersebut tidak memiliki informasi yang diperlukan. Pada proses NDS ini terjadi proses Sorting pada kolom ID dengan menggunakan tipe sorting ascending agar pencarian Id lebih mudah. Data DB Source dari dbo.stage_job position setelah dilakukan proses ETL kemudian dipindah ke dalam DB Destination nds_job_position yang dibuat terlebih dahulu koneksi ke basis data pada SQL Server. Tabel dbo.stage_fakultas pada Stage diambil menggunakan query. Proses pembuatan query ini dijelaskan pada Gambar 4.11, Hasil akhir dari Proses NDS pada tabel dbo,stage_job_position ditunjukkan pada Gambar 4.12.



Gambar 4.11 Tabel *Query* proses pembuatan data baru nds_job_position



Gambar 4.12 Proses NDS tabel dbo.stage_job_position

- NDS Alumni

Tabel dbo.Stage Alumni memiliki 1000 Data pengisi data Pelacakan Studi, didalam 1000 Data tersebut ada beberapa data yang disetiap kolomnya ada yang hanya mengisi nama dan selebihnya tidak terisi datanya sama sekali atau null, kemudian untuk mempermudah dalam proses ETLnya maka dibuat kedalam bentuk View yang diberi nama dbo.View_1. Dbo.Stage_Alumni dan dbo.View_1 isi kolomnya sama, yang membedakan adalah jumlah

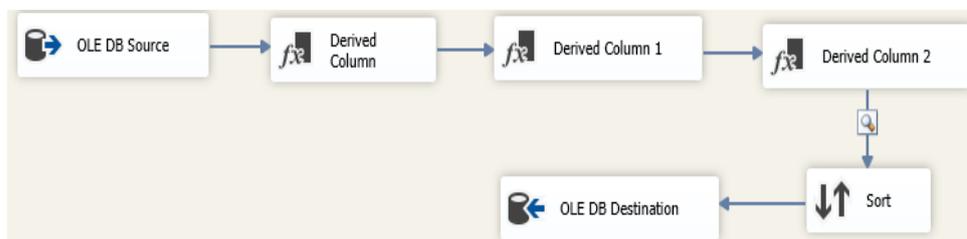
data pengisinya, didalam dbo.View_1 sudah terjadi penyaringan data dimana data yang dipakai hanya data yang bersifat lengkap, dalam artian terdapat informasi atau isi disetiap kolomnya, yang sebelumnya berjumlah 1000 data disaring menjadi 643 Data.

Tabel 4.12 Hasil pemeriksaan tabel dbo.View_1

Nama Kolom	Tipe Data	Data Noise	Keterangan
Id	int	0	Diperlukan
Nim	varchar(25)	0	Diperlukan
nama	varchar(75)	0	Diperlukan
Tempat_lahir	varchar(250)	9	Diperlukan
Jenis_kelamin	varchar(1)	0	Diperlukan
Id_prodi	Int	0	Diperlukan
Bulan_lulus	Int	9	Diperlukan
Tahun_lulus	Int	6	Diperlukan
Ipk	Float	7	Diperlukan
Judul_skripsi	Varchar(250)	574	Tidak Diperlukan
Password	Varchar(250)	0	Diperlukan
Alamat	Varchar(250)	9	Diperlukan
Email	Varchar(50)	9	Diperlukan
Nomor_hp	Varchar(50)	9	Diperlukan
Nomor_tlp_rumah	Varchar(50)	633	Tidak Diperlukan
Id_jenis_pekerjaan	Int	0	Diperlukan
Nama_institusi	Varchar(75)	19	Diperlukan
Alamat_institusi	Varchar(max)	26	Tidak Diperlukan
Bagian	Varchar(75)	38	Diperlukan

Nama Kolom	Tipe Data	Data Noise	Keterangan
Masa_tunggu	Float	643	Tidak Diperlukan
Id_salary_range	Int	0	Diperlukan
Tanggal_lahir	Datetime	2	Diperlukan
Tanggal_mulai_kerja	Datetime	32	Diperlukan
Tanggal_akhir_kerja	Datetime	643	Tidak Diperlukan
Nama_institusi_pertama	Varbinary(75)	643	Tidak Diperlukan
Alamat_institusi_pertama	Varchar(max)	643	Tidak Diperlukan
Bagian_pertama	Varbinary(75)	643	Tidak Diperlukan
Relevansi_pendidikan	Bit	9	Diperlukan
Alasan_relevansi	Varchar(max)	545	Tidak Diperlukan
Saran	Varchar(max)	152	Tidak Diperlukan
UserRoleId	Int	0	Diperlukan
Angkatan	Int	2	Diperlukan
Jabatan	Varchar	22	Diperlukan
IsMengisi	Bit	9	Diperlukan
GetJobYear	Int	32	Tidak Diperlukan
GetJobMonth	Int	32	Tidak Diperlukan
GetJobDay	Int	32	Tidak Diperlukan

dbo.View 1 memiliki 37 kolom dengan 643 Data mengenai alumni. Pada Tabel 4.12 terdapat data noise pada kolom tempat_lahir, bulan_lulus, tahun_lulus, ipk, judul_skripsi, alamat, email, nomor_hp, nomor_tlp_rumah, nama_institusi, alamat_institusi, bagian, masa_tunggu, tanggal_lahir, tanggal_mulai_kerja, tanggal_akhir_kerja, nama_institusi_pertama, alamat_institusi_pertama, bagian_pertama, relevansi_pendidikan, alasan_relevansi, saran, UserRoleId, angkatan, jabatan. Noise pada kolom tersebut berupa nilai null. Data pada kolom tempat_lahir, alamat, email, nomor_hp, nomor_tlp_rumah, bagian diganti dengan “-“, diganti dengan tanda “-“ karena memungkinkan untuk tidak memanipulasi datanya dan mengurangi data yang bersifat null. Sedangkan pada kolom noise yang lainnya dibiarkan tetap null, karena datanya tidak diperlukan informasinya dan dipakai. Pada kolom nama dan bagian ditemukan beberapa data dengan trailing space, Oleh karena itu untuk menanganinya digunakan fungsi RTRIM, RTRIM berfungsi untuk menghapus banyak spasi antar teks. Pada kolom Id juga dilakukan sorting data dengan tipe Ascending, agar data pada idurut sesuai nomornya dan dapat dengan mudah untuk dilakukan pencarian, proses tersebut dapat dilihat pada Gambar 4.13. Data DB Source dari dbo.stage_alumni setelah dilakukan proses ETL kemudian dipindah ke dalam DB Destination nds_alumni menggunakan query sama seperti yang dijelaskan pada tabel sebelumnya.

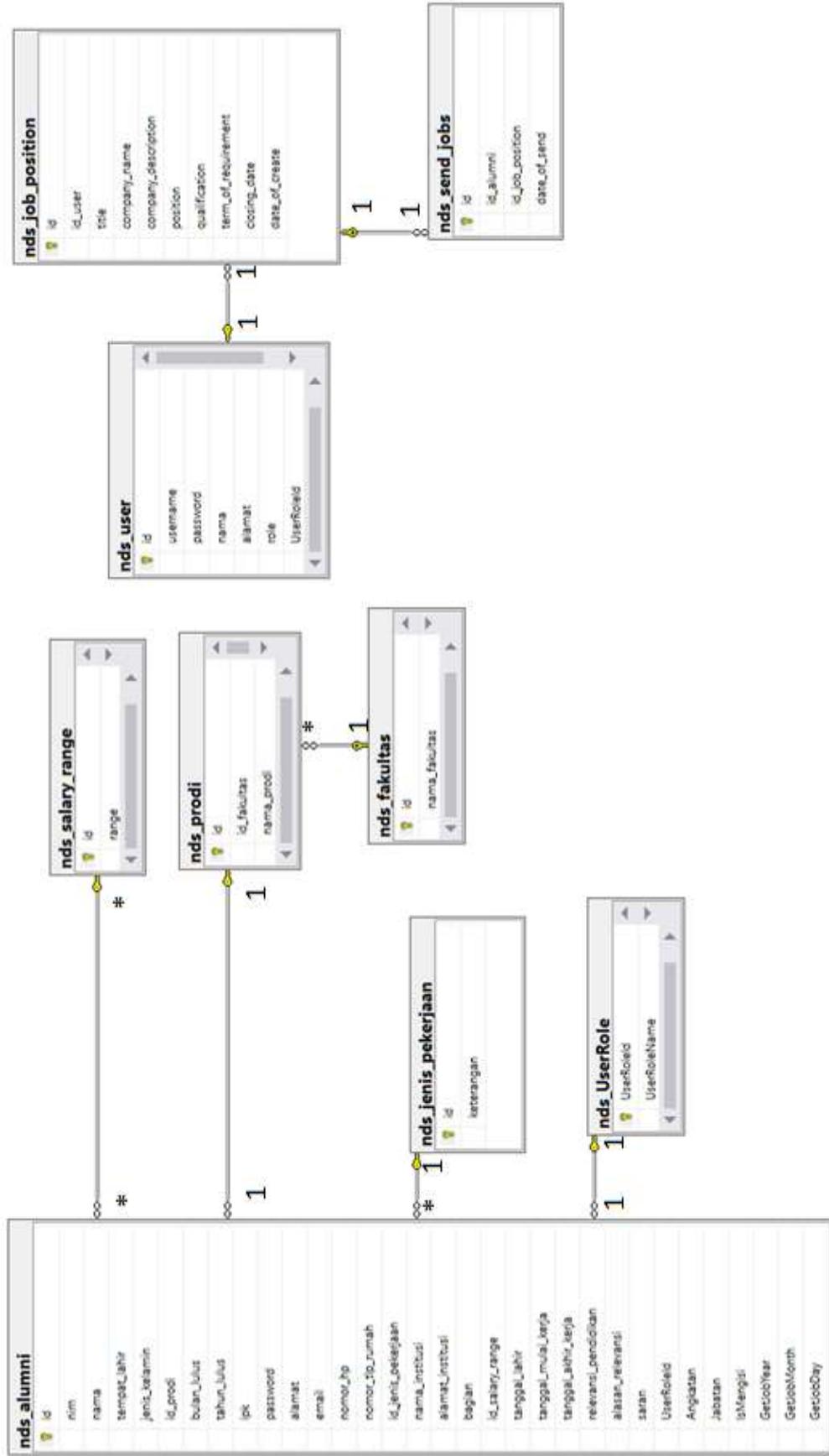


Gambar 4.13 Proses NDS tabel dbo.stage_alumni

4.2.2.2 Relasi Tabel

Relasi pada *Normalized Data Store* (NDS) menggunakan seluruh tabel yang telah dilakukan proses ETL dimana telah dibersihkan dari data noise. Diagram relasi NDS dapat dilihat pada Gambar 4.14. Dari gambar relasi tersebut dapat dilihat kebergantungan tabel transaksi terhadap tabel master. Tabel Transaksi mendapatkan informasi lebih dengan adanya tabel master, namun pada diagram ini ditunjukkan bahwa tabel transaksi pada tabel `dbo.user`, `dbo.job_position` dan `dbo.send_jobs` berdiri sendiri karena mencakup informasi yang berbeda dengan tabel master. Ada beberapa relasi dari Gambar 4.14 yaitu :

- Many to one : `nds_alumni` dan `nds_salary_range`
banyak alumni dapat mendapatkan banyak gaji yang didapat
- one to one : tabel `nds_alumni` dan `nds_prodi`
satu alumni mempunyai satu prodi
- many to one : tabel `nds_prodi` dan `nds_fakultas`
pada satu fakultas mempunyai banyak prodi
- many to one : `nds_alumni` dan `nds_jenis_pekerjaan`
banyak alumni dapat mendapatkan satu jenis pekerjaan
- one to one : `nds_alumni` dan `ndsUserRole`
setiap alumni atau satu alumni adalah bagian dari satu jenis user role
- one to many : `nds_user` dan `nds_job_position`
satu user dapat mempunyai banyak informasi dari informasi mengenai jon atau pekerjaan
- one to one : `nds_job_position` dan `nds_send_jobs`
satu jenis informasi pekerjaan dikirim dengan satu waktu.



Gambar 4.14 Diagram relasi pada Normalized Data Store (NDS)

4.2.3 Pembuatan data *Dimensional Data Store* (DDS)

4.2.3.1 Tabel Dimensi

a) Tabel dbo.dim_fakultas

Dimensi dbo.dim_fakultas terdiri dari 3 kolom dengan data sebanyak 11 data. Atribut pada tabel dbo.dim_fakultas dapat dilihat pada Tabel 4.13

Tabel 4.13 Tabel dbo.dim_fakultas

Nama Kolom	Tipe Data
Fakultas_key	Int
Id	Int
Nama_fakultas	Varchar(75)

Pada pembuatan dim_fakultas proses yang pertama adalah pada OLE DB Source mengambil data source dari data dbo.nds_fakultas yang selanjutnya dipindahkan kedalam tabel DDS, sebelum mengambil tabel tersebut dibuat terlebih dahulu koneksi ke basis data pada SQL Server, koneksi tersebut dapat dibuat menggunakan query pada OLE DB Destination, data dbo.nds_fakultas dipindahkan ke dalam database yang telah dibuat dengan query yang diberi nama dim_fakultas, ditambahkan kolom fakultas_key yang merupakan surrogate key sekaligus sebagai primary key pada tabel ini, hasil data dari query tersebut dapat dilihat pada Gambar 4.15



	fakultas_key	id	nama_fakultas
1	1	1	TEKNIK
2	2	2	PERTANIAN
3	3	3	KEDOKTERAN
4	4	4	EKONOMI
5	5	5	ISIPOL
6	6	6	HUKUM
7	7	7	AGAMA ISLAM
8	8	8	PENDIDIKAN BAHASA
9	9	10	PASCA SARJANA
10	10	20	PASCA S3
11	11	30	VOKASI D3

Gambar 4.15 Hasil tabel dbo.dim_fakultas

b) Tabel dbo.dim_jenis_pekerjaan

Dimensi dbo.dim_ jenis_pekerjaan terdiri dari 3 kolom dengan data sebanyak 8 data. Kolom Atribut pada tabel dbo.dim_jenis_pekerjaan dapat dilihat pada Tabel 4.14

Tabel 4.14 Tabel dbo.dim_jenis_pekerjaan

Nama Kolom	Tip e Data
jenispekerjaan_key	Int
Id	Int
keterangan	Varchar(75)

Pada pembuatan dim_jenis_pekerjaan proses yang pertama adalah pada OLE DB Source mengambil data source dari data dbo.nds_jenis_pekerjaan yang selanjutnya dipindahkan kedalam tabel DDS, sebelum mengambil tabel tersebut dibuat terlebih dahulu koneksi ke basis data pada SQL Server, koneksi tersebut dapat dibuat menggunakan query pada OLE DB Destination, data dbo.nds_fakultas dipindahkan ke dalam database yang telah dibuat dengan query yang diberi nama dim_ jenis_pekerjaan, ditambahkan kolom jenispekerjaan_key yang merupakan surrogate key sekaligus sebagai primary key pada tabel ini, hasil data dari query tersebut dapat dilihat pada Gambar 4.16

	jenispekerjaan_key	id	keterangan
1	1	1	Belum Bekerja
2	2	2	Pemerintah (Pusat / Departemen)
3	3	3	Pemerintah (Daerah)
4	4	4	Pemerintah (BUMN, BHMN)
5	5	5	Swasta (Jasa)
6	6	6	Swasta (Manufaktur)
7	7	7	Wiraswasta
8	8	8	Lain-Lain

Gambar 4.16 Hasil tabel dbo.dim_jenis_pekerjaan

c) Tabel dbo.dim_prodi

Dimensi dbo.dim_prodi terdiri dari 4 kolom dengan data sebanyak 38 data. Atribut pada tabel dbo.dim_prodi dapat dilihat pada Tabel 4.15

Tabel 4.15 Tabel dbo.dim_prodi

Nama Kolom	Tipe Data
Prodi_key	Int
Id	Int
Id_fakultas	int
keterangan	Varchar(75)

Pada pembuatan dim_prodi proses yang pertama pada OLE DB Source mengambil data source dari data dbo.nds_prodi yang selanjutnya dipindahkan kedalam tabel DDS, sebelum mengambil tabel tersebut dibuat terlebih dahulu koneksi ke basis data pada SQL Server, koneksi tersebut dapat dibuat menggunakan query pada OLE DB Destination, data dbo.nds_prodi dipindahkan ke dalam database yang telah dibuat dengan query yang diberi nama dim_prodi, ditambahkan kolom prodi_key yang merupakan surrogate key sekaligus sebagai primary key pada tabel ini, hasil data dari query tersebut dapat dilihat pada Gambar 4.17

prodi_key	id	id_fakultas	nama_prodi
1	11	1	S1. TEKNIK SIPIL
2	12	1	S1. TEKNIK ELEKTRO
3	13	1	S1. TEKNIK MESIN
4	14	1	S1. TEKNOLOGI INFORMASI
5	21	2	S1. AGROTEKNOLOGI
6	22	2	S1. AGROBISNIS
7	31	3	S1. PENDIDIKAN DOKTER
8	32	3	S1. ILMU KEPERAWATAN
9	34	3	S1. KEDOKTERAN GIGI
10	35	3	S1. FARMASI
11	41	4	S1. EKONOMI MANAJEMEN
12	42	4	S1. AKUNTANSI
13	43	4	S1. ILMU EKONOMI
14	51	5	S1. ILMU HUBUNGAN INT...
15	52	5	S1. ILMU PEMERINTAHAN
16	53	5	S1. ILMU KOMUNIKASI
17	54	5	S1. PEND. BAHASA INGGRIS
18	61	6	S1. ILMU HUKUM

CP-APP85IT (13.0 RIM) sa [54] DatawarehouseCDC 00:00:00 38 rows

Gambar 4.17 Hasil tabel dbo.dim_prodi

d) Tabel dbo.dim_salary_range

Dimensi dbo.dim_salary_range terdiri dari 3 kolom dengan data sebanyak 8 data. Kolom salaryrange_key merupakan surrogate key sekaligus primary key pada tabel ini. Atribut pada tabel dbo.dim_fakultas dapat dilihat pada Tabel 4.16

Tabel 4.16 Tabel dbo.dim_salary_range

Nama Kolom	Tipe Data
salaryrange_key	Int
Id	Int
range	Varchar(75)

Pada pembuatan dim_salary_range proses yang pertama pada OLE DB Source mengambil data source dari data dbo.nds_salary_range yang selanjutnya dipindahkan kedalam tabel DDS, sebelum mengambil tabel tersebut dibuat terlebih dahulu koneksi ke basis data pada SQL Server, koneksi tersebut dapat dibuat menggunakan query pada OLE DB Destination, data dbo.nds_salary_range dipindahkan ke dalam database yang telah dibuat dengan query yang diberi nama dim_salary_range, ditambahkan kolom salaryrange_key yang merupakan surrogate key sekaligus sebagai primary key pada tabel ini, hasil data dari query tersebut dapat dilihat pada Gambar 4.18

	salaryrange_key	id	range
1	1	1	< 1.000.000
2	2	2	1.000.000 - 3.000.000
3	3	3	3.000.000 - 5.000.000
4	4	4	5.000.000 - 7.500.000
5	5	5	7.500.000 - 10.000.000
6	6	6	10.000.000 - 12.500.000
7	7	7	12.500.000 - 15.000.000
8	8	8	> 15.000.000

Gambar 4.18 Hasil tabel dbo.dim_salary_range

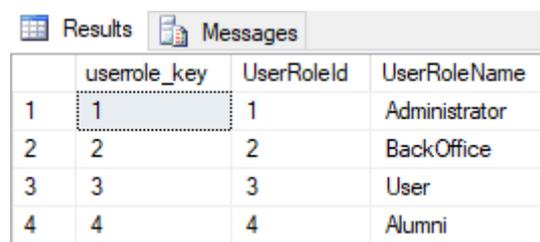
e) Tabel dbo.dim_userrole

Dimensi dbo.dim_userrole terdiri dari 3 kolom dengan data sebanyak 4 data. Kolom userrole_key merupakan surrogate key sekaligus primary key pada tabel ini. Atribut pada tabel dbo.dim_fakultas dapat dilihat pada Tabel 4.17

Tabel 4.17 Tabel dbo.dim_userrole

Nama Kolom	Type Data
userrole_key	Int
UserRoleId	Int
UserRoleName	Varchar(75)

Pada pembuatan dim_userrole proses yang pertama pada OLE DB Source mengambil data source dari data dbo.nds_UserRole yang selanjutnya dipindahkan kedalam tabel DDS, sebelum mengambil tabel tersebut dibuat terlebih dahulu koneksi ke basis data pada SQL Server, koneksi tersebut dapat dibuat menggunakan query pada OLE DB Destination, data dbo.nds_UserRole dipindahkan ke dalam database yang telah dibuat dengan query yang diberi nama dim_userrole, ditambahkan kolom userrole_key yang merupakan surrogate key sekaligus sebagai primary key pada tabel ini, hasil data dari query tersebut dapat dilihat pada Gambar 4.19

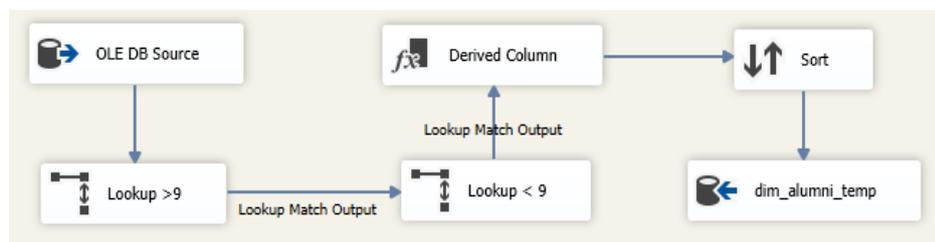


The screenshot shows a SQL Server Results window with a table containing four rows of data. The columns are userrole_key, UserRoleId, and UserRoleName. The first row is highlighted with a dotted border.

	userrole_key	UserRoleId	UserRoleName
1	1	1	Administrator
2	2	2	BackOffice
3	3	3	User
4	4	4	Alumni

Gambar 4.19 Hasil tabel dbo.dim_userrole

f) Tabel dim_alumni_temp



Gambar 4.20 Proses pembuatan tabel dim_alumni_temp

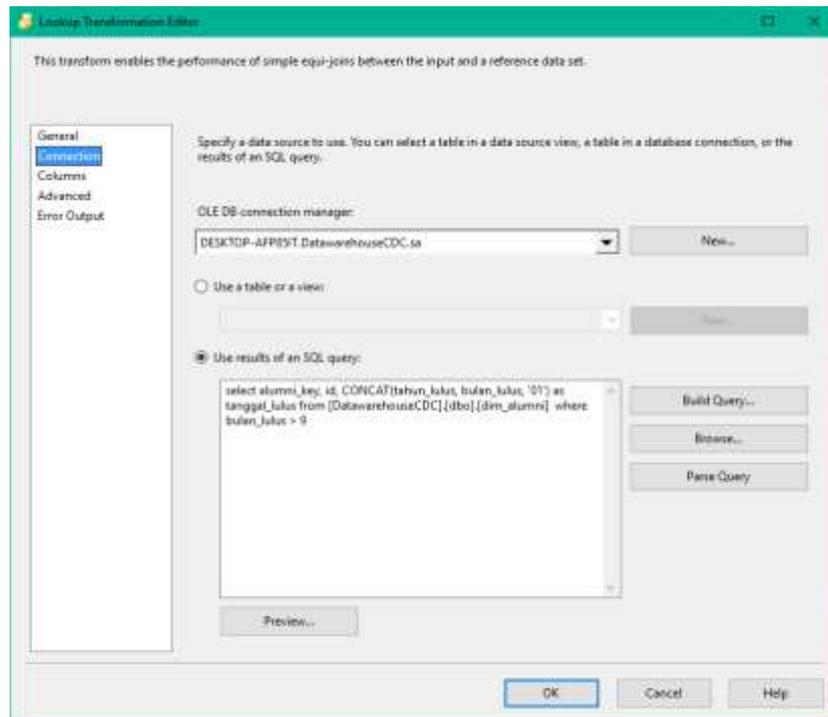
Tabel dim_alumni_temp dibuat untuk menambahkan tabel, berdasarkan keterangan pada Tabel 4.18 ditambahkan tabel Tanggal_lulus, Penambahan tabel Tanggal_lulus dibuat upaya untuk menentukan perhitungan dari masa tunggu alumni untuk mendapatkan pekerjaan, terjadi proses Lookup yang dapat dilihat pada Gambar 36, terjadi dua proses lookup yaitu lookup > 9 dan lookup < 9 proses ini ditambahkan query seperti pada Gambar 4.21 dan Gambar 4.22, penjelasan gambar sebagai berikut :

- Gambar 4.21 (Query penambahan hasil fungsi Lookup >9)
Ketika saat ada keterangan bulan lulus lebih dari 2 angka misal 10,11,12 maka ditulis dengan query tersebut, dengan menyatukan (tahun_lulus, bulan_lulus, '01') contoh : 20111001, pada kolom penambahan '01' bermaksud untuk semua sama rata dijadikan menjadi tanggal 01.
- Gambar 4.22 (Query penambahan hasil fungsi Lookup <9)
Ketika ada keterangan bulan lulus hanya satu angka yaitu bulan 1, 2, 3, 4, 5, 6, 7, 8, dan 9 maka ditulis dengan query tersebut, dengan menyatukan (tahun_lulus, 0, bulan_lulus, 01) penambahan angka 0 setelah tahun_lulus dimaksudkan untuk menjadikan bulan lulus menjadi bentuk puluhan, misal 01,02,03 dan seterusnya, contoh : 20110201.

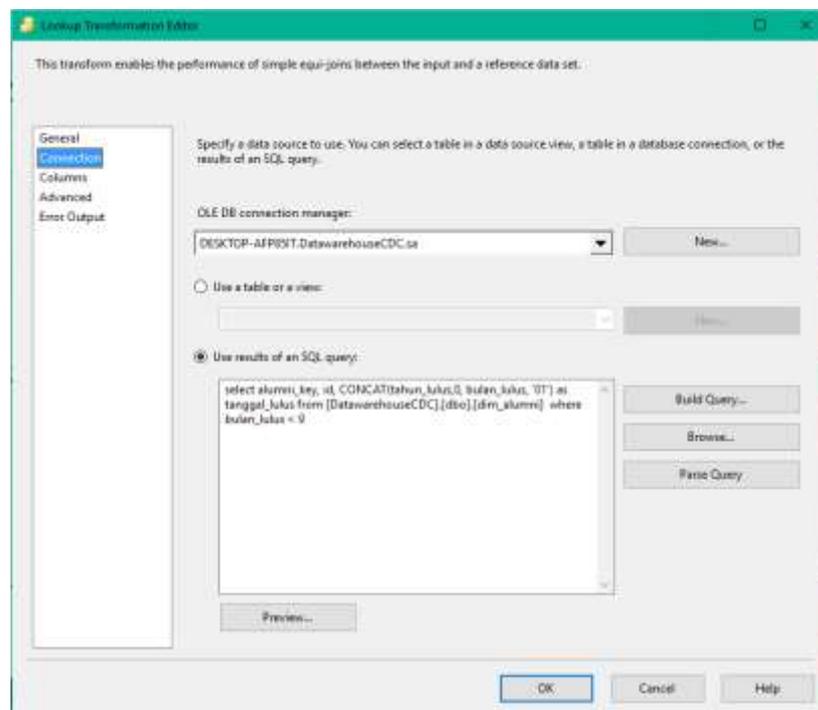
fungsi lookup dengan query tersebut agar tanggal_lulus dibuat seperti tipe data datetime agar nantinya dapat dihitung hasil Masa_tunggu, karena berdasarkan keterangan yang ada pada tabel 4.18, Masa_tunggu dihitung dari perhitungan Tanggal_mulai_kerja dikurangi dengan Tanggal_lulus, sedangkan tipe data pada Tanggal_mulai_kerja adalah datetime, oleh sebab itu dibuat Tanggal_lulus dengan bentuk tipe data datetime agar dapat dijadikan perhitungan Masa_tunggu. Hasil dari query tersebut dapat dilihat pada Gambar 4.23.

Tabel 4.18 Penambahan tabel pada dbo.dim_alumni_temp

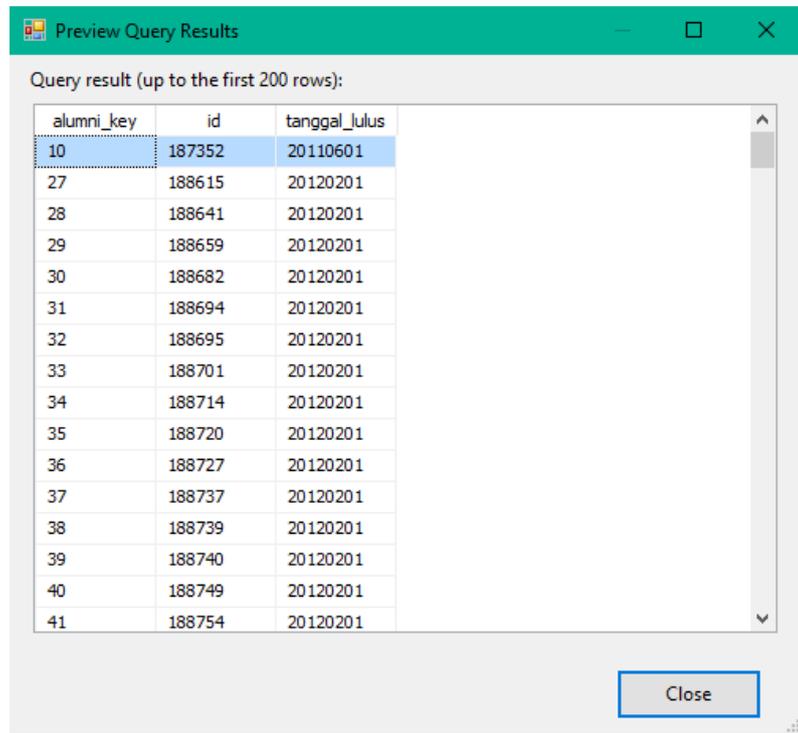
Nama Kolom	Tipe Data	Keterangan
Tanggal_lulus	Varchar(26)	Tabel tanggal lulus dibuat agar dapat menghitung masa tunggu
MasaTunggu	Int	Tabel Masa Tunggu dibuat untuk mengetahui masa tunggu alumni untuk mendapatkan pekerjaan, hasil data dari tabel Masa Tunggu dihitung dengan menghitung dari tanggal_mulai_kerja dikurangi dengan Tanggal_lulus.



Gambar 4.21 Query penambahan pada fungsi Lookup >9



Gambar 4.22 Query penambahan pada fungsi Lookup < 9



Query result (up to the first 200 rows):

alumni_key	id	tanggal_lulus
10	187352	20110601
27	188615	20120201
28	188641	20120201
29	188659	20120201
30	188682	20120201
31	188694	20120201
32	188695	20120201
33	188701	20120201
34	188714	20120201
35	188720	20120201
36	188727	20120201
37	188737	20120201
38	188739	20120201
39	188740	20120201
40	188749	20120201
41	188754	20120201

Gambar 4.23 Hasil tabel dbo.dim_alumni_temp

g) Tabel dim_alumnus

Dimensi dbo.dim_alumni terdiri dari 34 kolom dengan data sebanyak 643 data. Kolom Alumnus_key merupakan surrogate key sekaligus primary key pada tabel ini. Atribut pada tabel dbo.dim_alumni dapat dilihat pada Tabel 4.19.

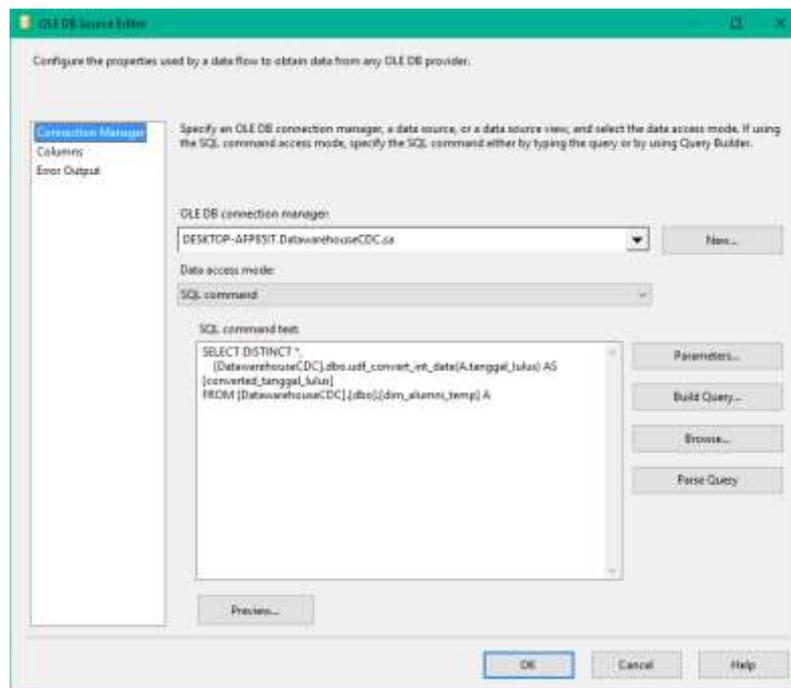
Tabel 4.19 Tabel dbo.dim_alumnus

Nama Kolom	Tipe Data
Alumnus_key	Int
Id	Int
Nim	Int
Nama	Varchar(75)
Tempat_lahir	Varchar(250)
Jenis_kelamin	Varchar(1)
Id_prodi	Int
Bulan_lulus	Int

Nama Kolom	Tipe Data
Tahun_lulus	Int
Ipk	Float
Password	Varchar(250)
Alamat	Varchar(250)
Email	Varchar(50)
Nomor_hp	Varchar(50)
Nomor_tlp_rumah	Varchar(50)
Id_jenis_pekerjaan	Int
Nama_institusi	Varchar(75)
Alamat_institusi	Varchar(max)
Bagian	Varchar(75)
Id_salary_range	Int
Tanggal_lahir	Datetime
Tanggal_mulai_kerja	Datetime
Tanggal_akhir_kerja	Datetime
Relevansi_pendidikan	Bit
Alasan_relevansi	Varchar(max)
Saran	Varchar(max)
UserRoleId	Int
Angkatan	Int
Jabatan	Varchar(100)
IsMengisi	Bit
GetJobYear	Int
GetJobMonth	Int
GetJobDay	Int
MasaTunggu	Int



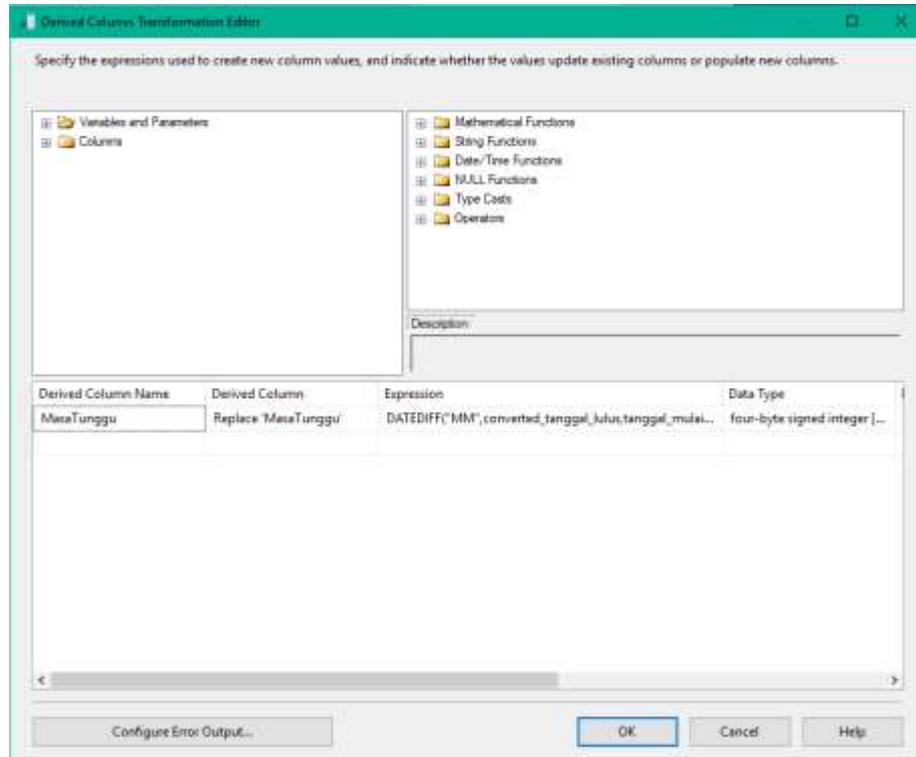
Gambar 4.24 Proses pembuatan tabel dbo.dim_alumnus



Gambar 4.25 Query convert data tanggal_lulus menjadi tipe data *datetime*

Pada pembuatan dim_userrole dijelaskan pada Gambar 4.24, proses yang pertama pada OLE DB Source ditujukan kepada kolom Tanggal_lulus yang telah dibuat sebelumnya pada dim_alumni_temp di convert atau diubah kedalam bentuk tipe data datetime menggunakan query SQL command pada Connection Manager seperti pada Gambar 4.25 terdapat query DISTINCT untuk *select* data keseluruhan dari data sumber DatawarehouseCDC. Class pada query A adalah mengambil data dari [DatawarehouseCDC].[dbo].[dim_alumni_temp]. Setelah berhasil diubah kedalam bentuk tipe data datetime dilakukan perhitungan pada Derived Column menggunakan fungsi Expression DATEDIFF yang dapat dilihat pada Gambar 4.26, perhitungan DATEDIFF disini menghitung lamanya

MasaTunggu berdasarkan bulan, Masa Tunggu dibuat untuk menghitung masa tunggu alumni untuk mendapatkan pekerjaan.



Gambar 4.26 Proses perhitungan Masa Tunggu

4.2.3.2 Tabel Fakta

Setelah seluruh tabel dimensi dibuat, tahap selanjutnya adalah membuat tabel fakta, berdasarkan kebutuhan data dan rancangan star schema penelitian ini terdapat sebuah tabel fakta yang diberi nama Tabel `dbo.fact_alumni`, tabel `dbo.fact_alumni` berisi foreign key dari primary key 6 tabel dimensi yang berelasi. Atribut pada tabel `dbo.fact_alumni` dapat dilihat pada Tabel 4.20.

Tabel 4.20 Tabel `fact_alumni`

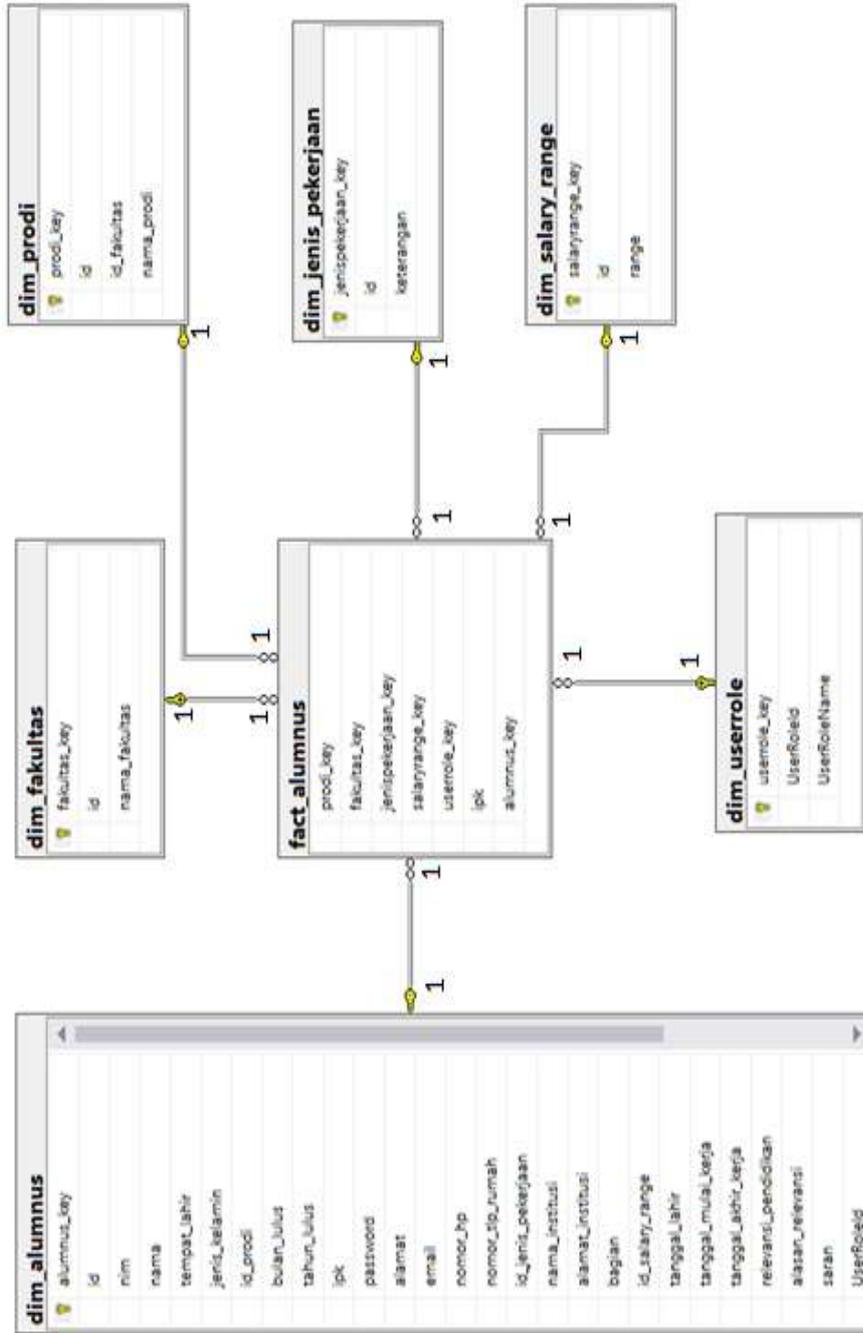
No.	Nama Kolom	Tipe Data
1	Alumni_key	int
2	Prodi_key	int
3	Fakultas_key	int

No.	Nama Kolom	Tipe Data
4	Jenispekerjaan_key	int
5	Salaryrange_key	int
6	Userrole_key	int
7	ipk	float

Proses pembuatan tabel fakta menggunakan tabel input dari tabel transaksional yang ada pada NDS dan tabel dimensi. Dalam tabel transaksional dikombinasikan dengan menggunakan fungsi Lookup pada seluruh tabel dimensi yang saling berelasi untuk mengambil surrogate key yang akan digunakan sebagai foreign key. Pada fungsi Lookup ditentukan kolom mana yang akan dijadikan acuan nilai yang sama pada kedua tabel dan kolom surrogate key yang kemudian akan diambil untuk tabel fakta, data-data tersebut kemudian ditampung di tabel output pada OLE DB Destination.

4.2.3.3 Relasi Tabel DDS

Setelah seluruh tabel dimensi dan tabel fakta dibuat, tahap selanjutnya adalah merelasikan tabel-tabel tersebut. Diagram relasi dari tabel dimensi dan tabel fakta dapat dilihat pada Gambar 4.27. terdapat satu relasi pada Gambar 4.27 yaitu one to one, karena tiap satu key hanya mempunyai satu hubungan dari tiap tabel.

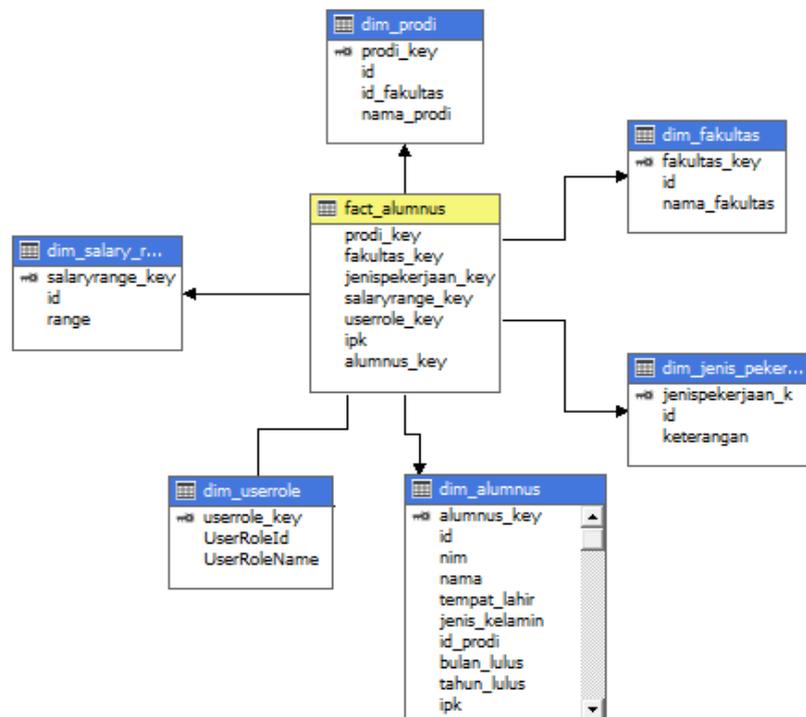


Gambar 4.27 Diagram Relasi pada DDS

4.2.4 Cube Pangkalan Data

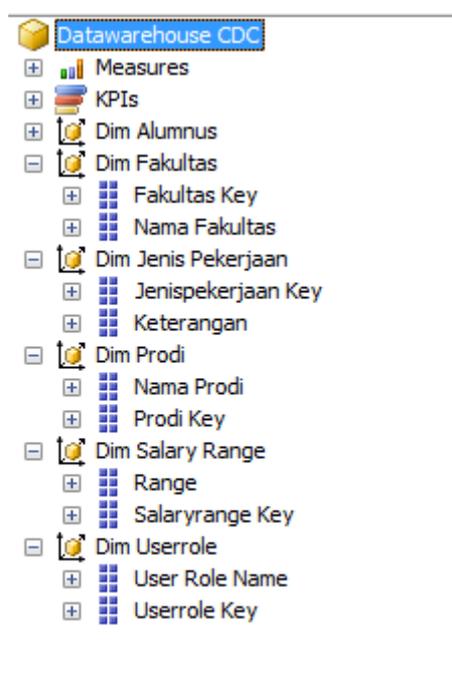
Pada Pangkalan data terdapat data-data yang perlu diorganisasi dan disimpan ke dalam bentuk multidimensi, data multidimensi adalah ketika dapat melihat sebuah data dari berbagai sudut pandang atau dimensi, data multidimensi ini memudahkan untuk pencarian kembali data untuk OLAP, setelah itu dilakukan analisa multidimensi yang dapat memberikan kemampuan untuk membuat laporan (*reporting*) atau menggunakan query dengan mudah tentang apa yang ingin diperlihatkan terhadap data-data tersebut.

Salah satu cara untuk melihat data dengan multidimensi tersebut adalah Cube. Cube disini menjadi struktur OLAP utama yang digunakan untuk melihat data-data pada pangkalan data. Sumber data dari struktur cube dapat dilihat pada Gambar 4.28, terdapat sumber data cube yang berbentuk skema bintang (star schema)



Gambar 4.28 Tampilan sumber data *cube*

Dalam cube terdapat istilah measure dan dimension, Measure adalah nilai quantitative atau data numeric database yang ingin di analisa, measure dihitung berdasarkan dimensi-dimensi dari cube. Sedangkan dimension adalah parameter terhadap measure sehingga dapat mendefinisikan suatu transaksi. Dimension memiliki kedudukan yang penentuannya bergantung pada proses drill down dan roll up yang ingin dilakukan saat melakukan OLAP. measure dan dimension dari cube yang telah dibuat dapat dilihat pada Gambar 4.29.



Gambar 4.29 Measure dan Dimensions pada Cube

4.4 Analisis dan Pembuatan laporan Pangkalan Data

Analisis pangkalan data pada penelitian ini dilakukan dengan cara pembuatan *reporting*, *reporting* ini dibuat agar mempermudah pengguna Kebutuhan dalam memahami dan melihat hasil data yang ada pada pangkalan data. Informasi yang akan di analisa pada pangkalan data ini disajikan dalam bentuk cube pada SQL Server Analysis Services.

Ada beberapa cara pembentukan laporan pada pangkalan data, antara lain dengan menggunakan *SQL Server Reporting Service (SSRS)*, *Analysis Services Multidimensional and Data Mining* pada *Visual Studio 2013*, *Microsoft Office Excel*, *Power BI Dekstop*, dan aplikasi lain yang terkoneksi dengan basis data. Dalam pembuatan laporan hasil analisis pada pangkalan data ini digunakan *Power BI Dekstop* karena *Power BI Dekstop* merupakan sebuah aplikasi perangkat lunak yang dapat terhubung langsung dengan basis data *SQL Server Management Studio*, pada aplikasi ini penggunaannya lebih mudah dan bervariasi, proses *reporting* data juga dapat dengan mudah di publikasikan ke dalam bentuk dashboard. Hasil analisis pada pangkalan data ini dapat dibuat dengan bentuk tabel, grafik, diagram, dan yang lainnya.

Informasi mengenai lulusan UMY dapat dilihat dari berbagai dimensi, yaitu alumni, fakultas, prodi, jenis pekerjaan, gaji, dan lain sebagainya. Adapun analisa kebutuhan yang informasinya tersedia pada cube alumni adalah sebagai berikut :

- a. Jumlah total alumni yang mengisi kuisioner dari pelacakan studi
- b. Data alumni (program studi, fakultas, dll)
- c. Nilai IPK mahasiswa
- d. Masa tunggu alumni untuk mendapatkan pekerjaan
- e. Gaji yang diterima (besarannya)
- f. Kesesuaian program studi dengan pekerjaan.

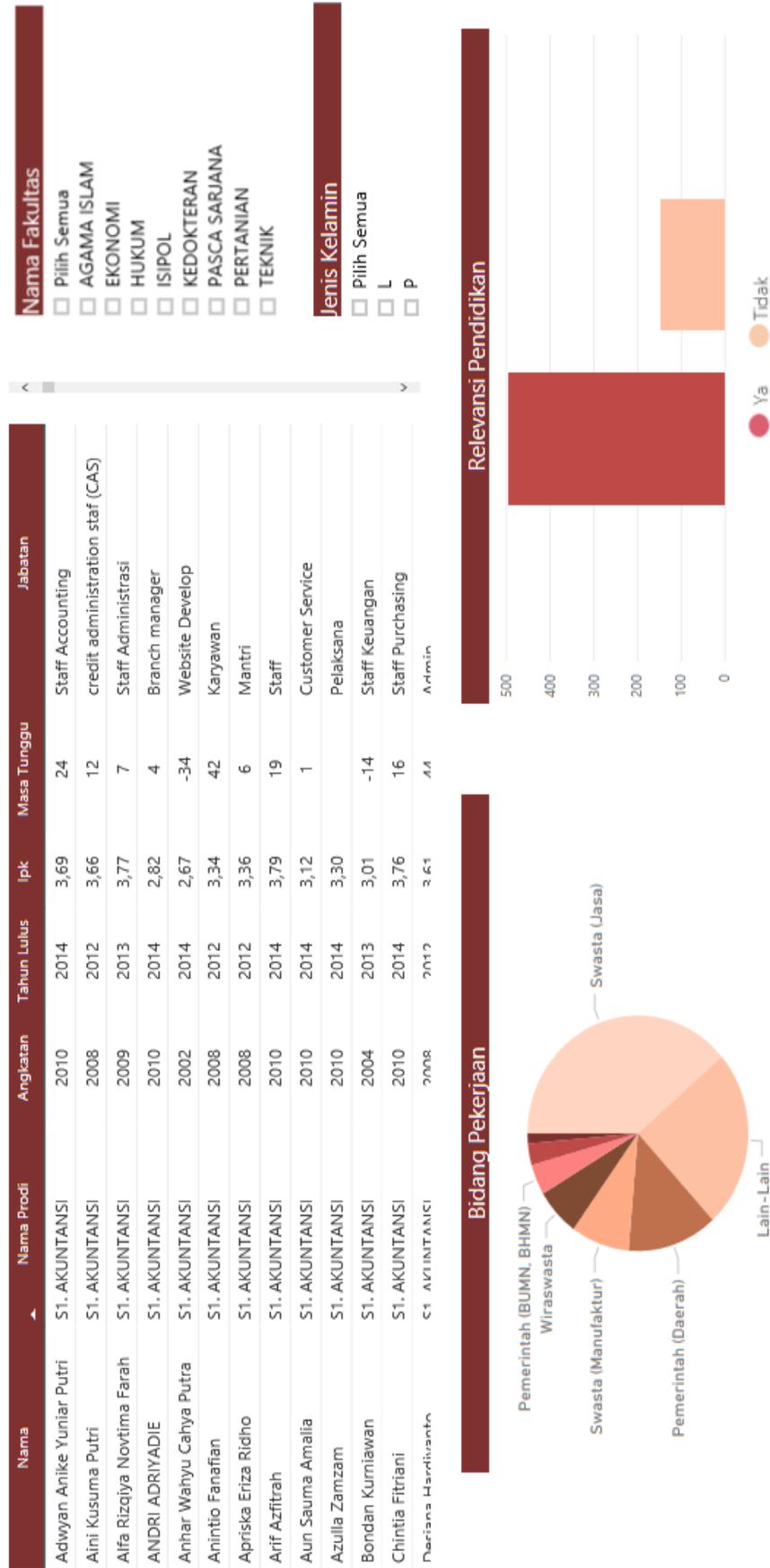
Eks Prodi Alumni Prodi	Eks Prodi Prodi Baru	Eks Salary Gaji (Rupiah)
01. ILMU KOMPUTER	0	< 1.000.000
02. TEKNIK SIPIL	3	< 1.000.000
03. TEKNIK INFORMATIKA	10	< 1.000.000
04. Sastra Sastra Perencanaan	30	< 1.000.000
05. Sastra Sastra Perencanaan	30	< 1.000.000
06. Sastra Sastra Perencanaan	30	< 1.000.000
07. Sastra Sastra Perencanaan	30	< 1.000.000
08. Sastra Sastra Perencanaan	30	< 1.000.000
09. Sastra Sastra Perencanaan	30	< 1.000.000
10. ILMU KOMPUTER	30	< 1.000.000 - 1.000.000
11. ILMU KOMPUTER	30	< 1.000.000 - 1.000.000
12. TEKNIK (Konsentrasi) Agensi Sistem	30	< 1.000.000
13. TEKNIK (Konsentrasi) Agensi Sistem	30	< 1.000.000
14. TEKNIK (Konsentrasi) Agensi Sistem	30	< 1.000.000
15. TEKNIK (Konsentrasi) Agensi Sistem	30	< 1.000.000

Gambar 4.30 Proses pemilihan data pada pelaporan Power BI

Tahap pembuatan *Report Dashboard* pada perangkat lunak Power BI ini yang pertama kali dilakukan adalah impor data database SQL Server Analysis Services Pada dengan mencentang seluruh komponen tabel data yang dipakai dalam pelaporan seperti pada Gambar 4.30, Hasil akhir pada proses pembuatan *dashboard* dipublikasikan dalam bentuk link sehingga mempermudah pengguna untuk membuka data secara langsung, tidak perlu membuka aplikasi maupun software-software yang lainnya. Tahap ini data sampel yang digunakan adalah pada keseluruhan pengisi data pelacakan studi. Berikut yang ditunjukkan pada Gambar 4.31 dan Gambar 4.32 adalah tampilan data keseluruhan pelacakan studi pada tahun 2001 sampai 2014 yang dapat digunakan sesuai dengan kebutuhan. Hasil akhir pelaporan *dashboard* dapat dibuka dengan link <https://bit.ly/2MDGt98>

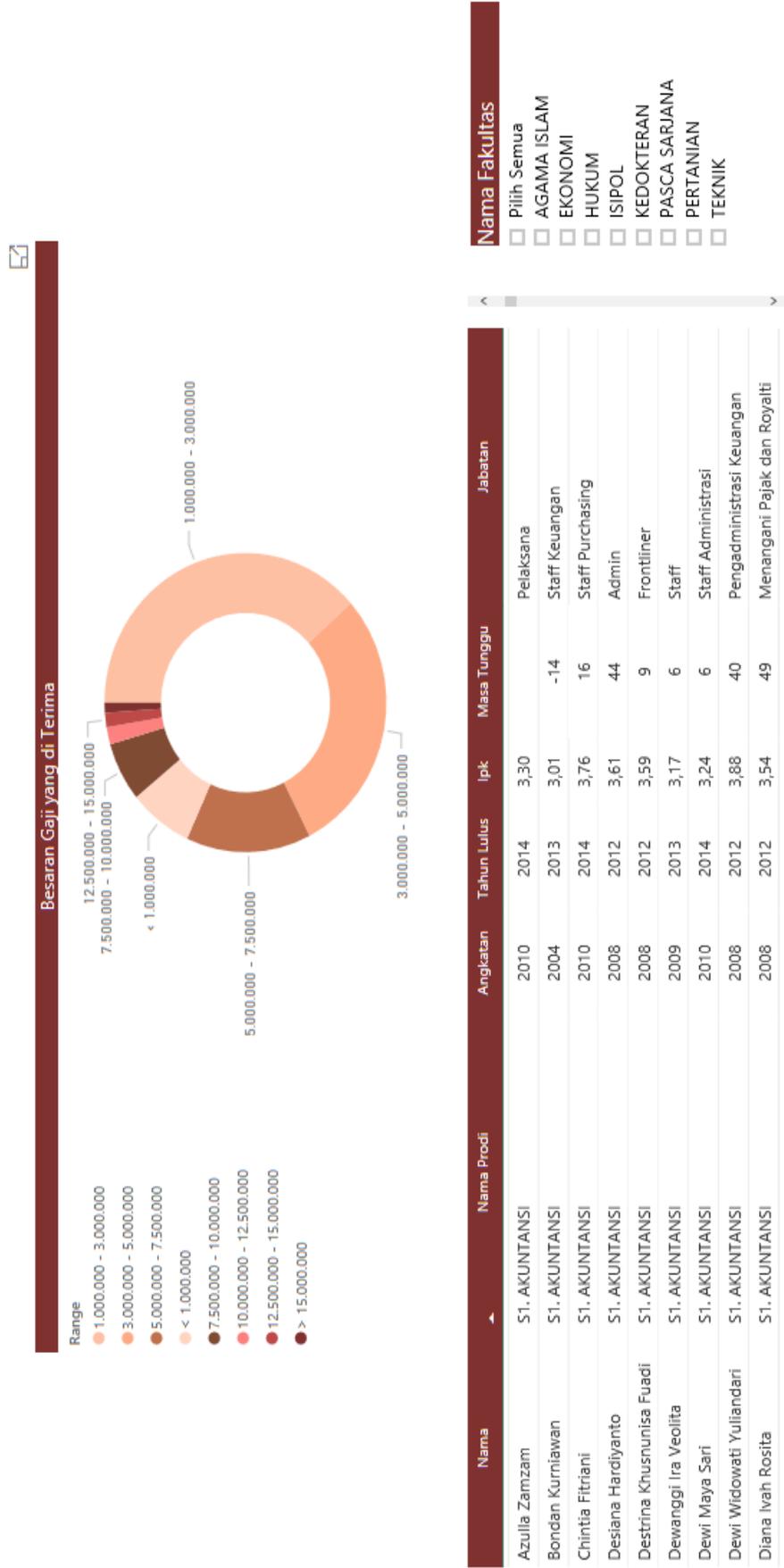
1. Gambar 4.31 (Tampilan data keseluruhan pelacakan studi pada tahun 2001 sampai 2014)
 - Pada tabel ditunjukan profil mahasiswa yang berisi Nama, nama prodi, angkatan, tahun lulus, ipk, masa tunggu, dan jabatan.
 - Pada tabel list Nama Fakultas dan Jenis Kelamin, ketika dipilih salah satu sebagai contoh pilih agama islam, maka seluruh data tersebut berganti menjadi data dari agama islam
 - Pada diagram Bidang pekerjaan dan Relevansi pendidikan, ketika di klik salah satu dari diagram tersebut, maka seluruh tabel berganti sesuai dengan yang dipilih, sebagai contoh mengklik pada bidang pekerjaan swasta, maka secara otomatis data tabel akan berubah menjadi profil alumni yang bekerja pada bidang tersebut.
2. Gambar 4.32 (Tampilan data keseluruhan pelacakan studi berdasarkan besaran jumlah gaji)
 - Pada diagram Besaran Gaji secara otomatis *responsive* dengan tabel profil mahasiswa, ketika di klik salah satu besaran gaji, maka dalam tabel akan muncul data alumni berdasarkan gaji yang telah di pilih.

Data Keseluruhan Alumni Tracer Study UMY



Gambar 4.31 Tampilan data keseluruhan pelacakan studi pada tahun 2001 sampai 2014

Data Keseluruhan Alumni Tracer Study UMY berdasarkan Besaran Jumlah Gaji



Gambar 4.32 Tampilan data keseluruhan pelacakan studi berdasarkan besaran jumlah gaji

4.5 Pengujian

Pangkalan data yang telah selesai dibangun memerlukan adanya pengujian. Pengujian dilakukan dengan beberapa metode yaitu sebagai berikut :

a. ETL Testing

Pengujian pada *ETL Testing* dilakukan dengan membandingkan hasil data. Setelah dilakukan perbandingan data kemudian akan menghasilkan data yang *valid* yang artinya ada kecocokan pada nilai data. Berikut adalah salah satu pengujian ETL menggunakan data sampel perhitungan Masa Tunggu pada tabel `dbo.dim_alumnus`

nama	tanggal_mulai_kerja	converted_tanggal_lulus	Masa Tunggu
Dini Aprilia Sari	2015-02-02 00:00:00.000	2011-10-01 00:00:00.000	40
Adhitya Wardhana	2016-03-28 00:00:00.000	2011-10-01 00:00:00.000	53
Aulia Dalila	2016-04-01 00:00:00.000	2011-10-01 00:00:00.000	54
Citra Selviana Febriany	2013-01-01 00:00:00.000	2011-10-01 00:00:00.000	15
Mifta Nuraida Fahmi	2012-06-21 00:00:00.000	2011-10-01 00:00:00.000	8
Rozi Oktori	2012-01-04 00:00:00.000	2011-10-01 00:00:00.000	3
Muhamad Yusvin Mustar	2015-02-02 00:00:00.000	2011-10-01 00:00:00.000	40
Muhammad Saleh	2012-07-31 00:00:00.000	2011-10-01 00:00:00.000	9
Rizki Yudhi Fitrianto	2014-10-06 00:00:00.000	2012-02-01 00:00:00.000	32
Wahyu Setiawan Nugroho	2016-01-04 00:00:00.000	2012-02-01 00:00:00.000	47
Burhan Hardoyo Putra	2015-08-10 21:21:00.000	2012-02-01 00:00:00.000	42
Sukardin	2013-01-15 00:00:00.000	2012-02-01 00:00:00.000	11
Yudhi Kristian	2016-12-01 00:00:00.000	2012-02-01 00:00:00.000	58
Surya Darma	2016-09-05 00:00:00.000	2012-02-01 00:00:00.000	55
Raqiburrahman .Mz	2015-11-05 00:00:00.000	2012-02-01 00:00:00.000	45

Gambar 4.33 Tabel `dbo.dim_alumnus` pada SQL

nama	tanggal mulai kerja	converted tanggal lulus	Perhitungan perBulan	Masa Tunggu
Dini Aprilia Sari	02/02/2015	01/10/2011	3 Tahun 4 Bulan (3x12) + 4 = 40	40
Adhitya Wardhana	28/03/2016	01/10/2011	4 Tahun 5 Bulan (4x12) + 5 = 53	53
Aulia Dalila	01/04/2016	01/10/2011	4 Tahun 6 Bulan (4x12) + 6 = 54	54
Citra Selviana Febriany	01/01/2013	01/10/2011	1 Tahun 3 Bulan (1x12) + 3 = 15	15
Mifta Nuraida Fahmi	21/06/2012	01/10/2011	0 Tahun 8 Bulan (0x12) + 8 = 8	8
Rozi Oktori	04/01/2012	01/10/2011	0 Tahun 3 Bulan (0x12) + 3 = 3	3
Muhamad Yusvin Mustar	02/02/2015	01/10/2011	3 Tahun 4 Bulan (3x12) + 4 = 40	40
Muhammad Saleh	31/07/2012	01/10/2011	0 Tahun 9 Bulan (0x12) + 9 = 9	9
Rizki Yudhi Fitrianto	06/10/2014	01/02/2012	2 Tahun 8 Bulan (2x12) + 8 = 32	32
Wahyu Setiawan Nugroho	04/01/2016	01/02/2012	3 Tahun 11 Bulan (3x12) + 11 = 47	47
Burhan Hardoyo Putra	10/08/2015	01/02/2012	3 Tahun 6 Bulan (3x12) + 6 = 42	42
Sukardin	15/01/2013	01/02/2012	0 Tahun 11 Bulan (0x12) + 11 = 11	11
Yudhi Kristian	01/12/2016	01/02/2012	4 Tahun 10 Bulan (4x12) + 10 = 58	58
Surya Darma	05/09/2016	01/02/2012	4 Tahun 7 Bulan (4x12) + 7 = 55	55
Raqiburrahman .Mz	05/11/2015	01/02/2012	3 Tahun 9 Bulan (3x12) + 9 = 45	45

Gambar 4.34 Tabel `dbo.dim_alumnus` pada Microsoft Excel

Pada Gambar 4.33 menunjukkan nilai Masa tunggu beberapa Alumni pada tabel *dbo.alumnus*. Perhitungan Masa tunggu berasal dari sistem yang sebelumnya telah dilakukan proses ETL menggunakan *query* dan proses *converted*. Dari gambar tersebut didapatkan hasil nilai Masa Tunggu terhitung berdasarkan bulan.

Untuk melihat keakuratan proses ETL dilakukan uji coba dengan melakukan perhitungan Masa Tunggu secara manual menggunakan *Microsoft Excel*. Hasil yang didapatkan adalah perhitungan manual berdasarkan perhitungan Tahun dan bulan, sebagai contoh pada Gambar 4.34 di kolom pertama tertulis Tanggal Mulai Kerja 02/02/2015 dan Tanggal Lulus 01/10/2011 jika dihitung maka terhitung Masa Tunggu untuk mendapatkan pekerjaan 3 Tahun 4 Bulan, 1 Tahun terdiri dari 12 Bulan, maka jika dijumlahkan ke seluruhnya maka Masa Tunggunya adalah 40 Bulan. Perbandingan kedua hasil perhitungan memberikan kesimpulan bahwa proses ETL yang telah dilakukan menghasilkan kecocokan data.

b. Functional Testing

Untuk meyakinkan apakah pangkalan data yang dibangun sudah sesuai dengan kebutuhan pengguna, pengujian ini dilakukan dengan membuat data *reporting dashboard*. *Reporting* yang menampilkan keseluruhan kebutuhan pengguna ditampilkan dengan menggunakan data sampel pada data Fakultas Ilmu Sosial dan Ilmu Politik (Fisipol) pada tahun 2001 sampai tahun 2014. Pada Gambar 4.35 ditunjukkan data pada *dashboard* ini antara lain Profil Alumni, IPK Alumni, Masa tunggu untuk mendapatkan pekerjaan, dan kesesuaian program studi dengan pekerjaannya, sedangkan pada Gambar 4.36 merujuk pada Data Alumni berdasarkan Gaji yang diterima, berikut adalah penjelasan dari tiap gambar :

1. Gambar 4.35 (Tampilan data alumni Fisipol UMY berdasarkan Jenis pekerjaan dan Relevansi Pendidikan

- Pada Tabel dijelaskan profil alumni yang menampilkan nama, nama prodi, tahun lulus, Ipk, Masa tunggu, jabatan. Masing-masing dari tabel tersebut berelasi dengan satu sama lainnya
 - Pada Diagram Lingkaran dijelaskan jumlah keseluruhan Jenis Pekerjaan, saat memilih sebagai contoh pada pilihan wiraswasta maka akan secara otomatis tabel dan yang lainnya menampilkan data sesuai dengan jenis pekerjaan wiraswasta.
 - Pada Diagram kolom ditunjukkan tingkat relevansi dan tidak relevansi antara kesesuaian program studi dengan pekerjaannya, ditunjukkan Ya dan Tidak.pada saat memilih Ya maka secara otomatis data keseluruhan yang ditampilkan juga berupa data yang berelevansi.
 - Pada kolom list dapat dipilih berdasarkan prodi dan jenis kelamin
2. Gambar 4.36 (Tampilan data Alumni Fisipol UMY berdasarkan gaji yang diterima (besarannya))
- Pada Grafik donat ditampilkan jumlah besaran gaji yang diterima berdasarkan rata-rata dari tiap prodi fisipol, pada saat memilih besaran gaji 1.000.000-3.000.000 maka data yang ditampilkan adalah data alumni yang mempunyai besaran gaji yang telah dipilih.
 - Pada tabel juga berisi profil alumni lengkap yang saling berelasi satu dengan yang lainnya.

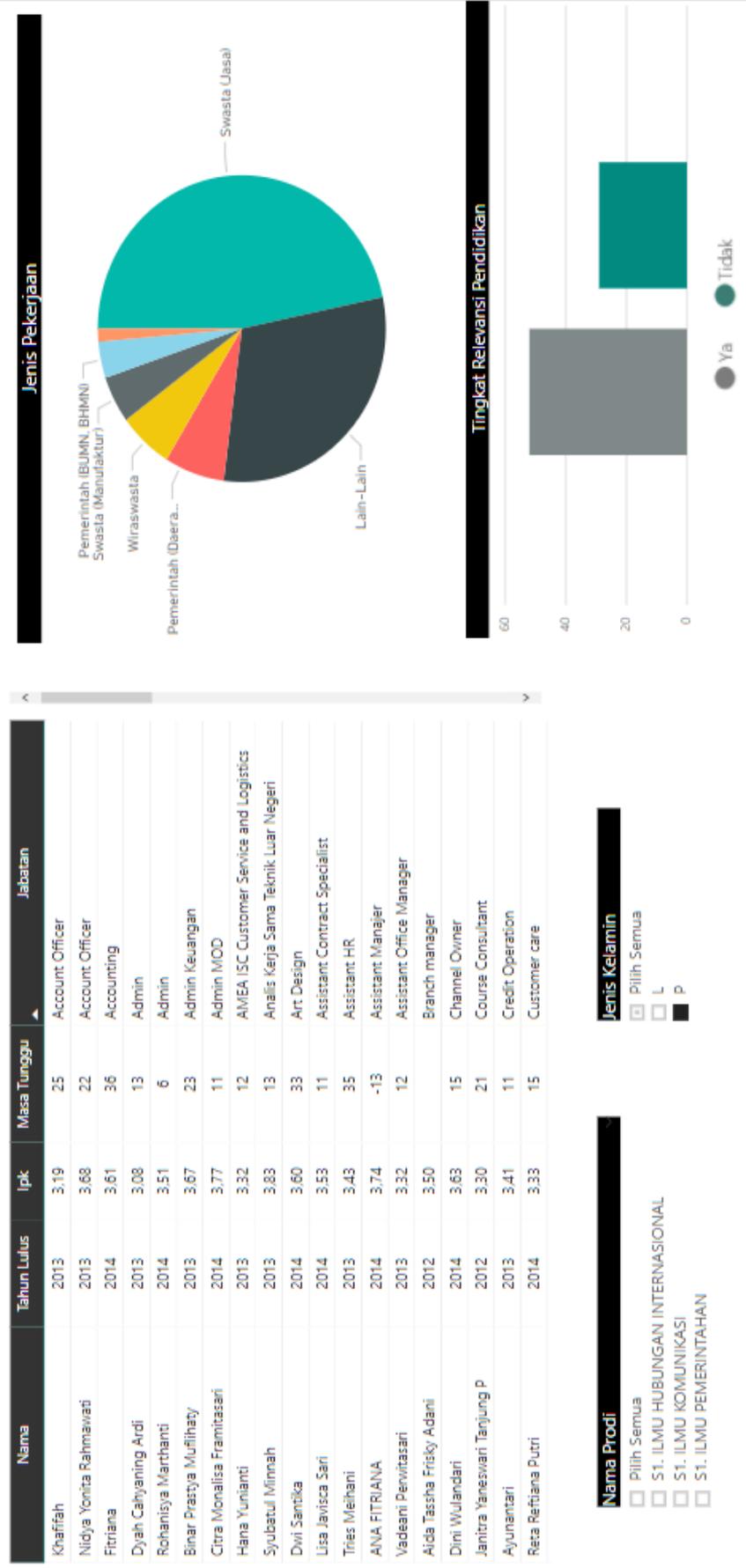
Merujuk pada Hasil wawancara kepada kepada pihak Lembaga Pengembangan Kemahasiswaan dan Alumni (LPKA) yang merupakan pihak pengguna data pelacakan studi yang berkaitan dengan pengelolaan data alumni pada proses akreditasi, maka dilakukan kembali wawancara terhadap respon atau kepuasan pengguna data, ketika wawancara penulis mendemokan hasil pelaporan yang ada pada link <https://bit.ly/2MDGt98> , respon pengguna terhadap data yang dibutuhkan adalah kebutuhan

pengguna dari kriteria yang telah diberikan sebelumnya semuanya terpenuhi dengan baik, dengan menampilkan data-data yang dibutuhkan dalam proses pelaporan, bentuk pelaporan menggunakan *dashboard* juga diberikan komentar yang baik pula, karena pengguna data yaitu Lembaga Pengembangan Kemahasiswaan dan Alumni (LPKA) diberikan kemudahan untuk membuat pelaporannya, karena data yang ditampilkan secara sistematis telah dapat ditunjuk hasilnya dan diketahui rata-rata tiap fakultas maupun prodi. kebutuhan tersebut dijelaskan pada Tabel 4.21.

Tabel 4.21 Hasil Pengujian kebutuhan pengguna

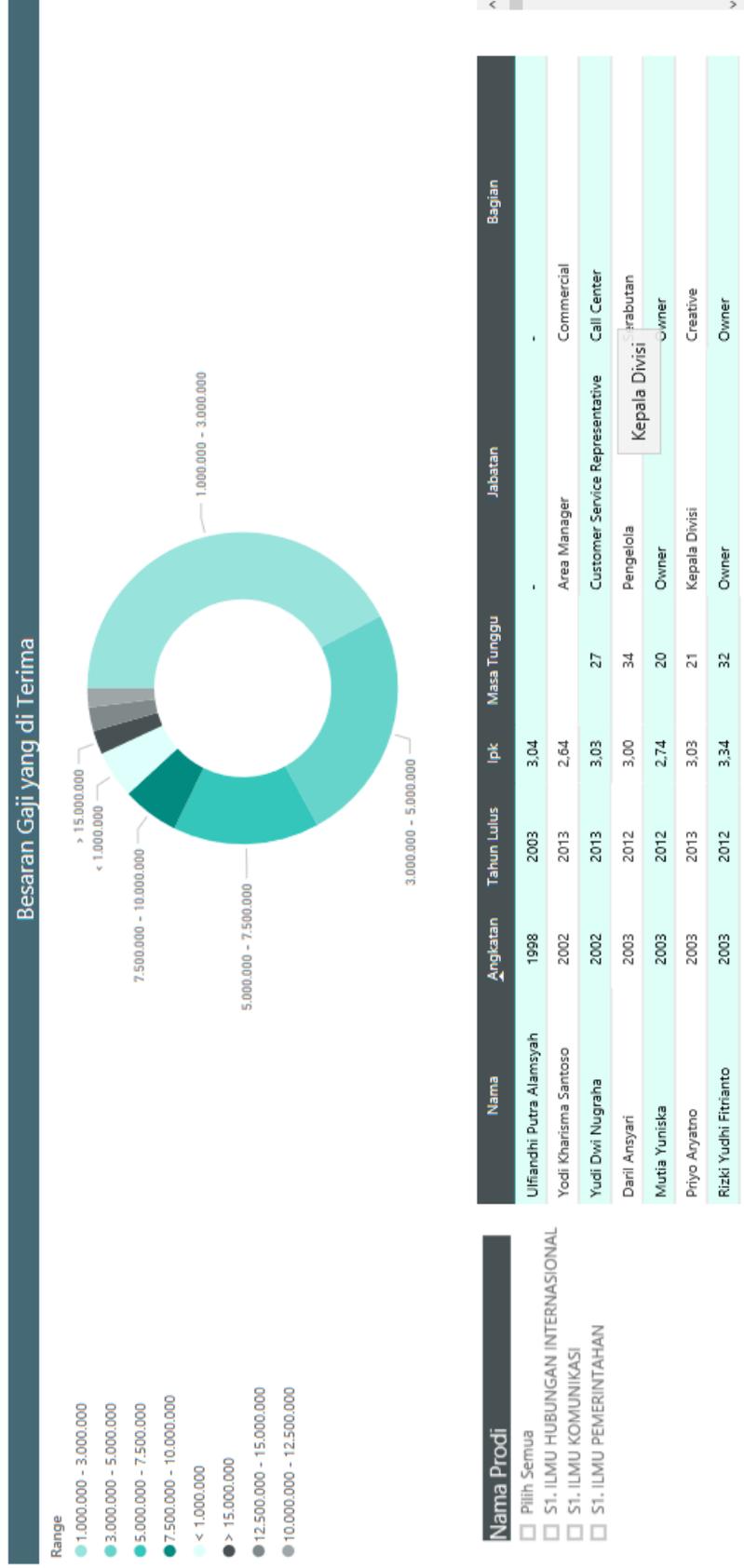
Kriteria Pengujian	Hasil Pengujian
Menampilkan Profil Alumni	Terpenuhi
Menampilkan IPK Alumni	Terpenuhi
Menampilkan masa tunggu untuk mendapatkan pekerjaan	Terpenuhi
Menampilkan Gaji yang diterima (besarannya)	Terpenuhi
Menampilkan kesesuaian program studi dengan pekerjaan.	Terpenuhi

Data Alumni Fakultas Fisipol UMY berdasarkan Jenis pekerjaan dan Relevansi Pendidikan



Gambar 4.35 Tampilan data Alumni Fisipol UMY berdasarkan Jenis pekerjaan dan Relevansi pendidikan

Data Fakultas Alumni Fisipol UMY berdasarkan Gaji yang di terima



Gambar 4.36 Tampilan data Alumni Fisipol UMY berdasarkan Gaji yang diterima (besarannya)