

# *Data Warehouse* Penerimaan Mahasiswa Baru

## TUGAS AKHIR

Oleh:

**Nur Yaqin**

**3310812034**

**Nina Krismawati**

**3310912002**

Disusun untuk memenuhi syarat kelulusan Program Diploma III



**POLITEKNIK NEGERI Batam**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**POLITEKNIK NEGERI BATAM**

**BATAM**

**2012**

# **LEMBAR PENGESAHAN**

Batam, 29 June 2012

**Pembimbing,**

**Hilda Widyastuti, ST, MT**

**NIK. 102020**

## **LEMBAR PERNYATAAN**

Dengan ini, saya:

NIM : 3310812034

Nama : Nur Yaqin

adalah mahasiswa Teknik Informatika Politeknik Negeri Batam yang menyatakan bahwa tugas akhir dengan judul:

### **DATA WAREHOUSE PENERIMAAN MAHASISWA BARU**

disusun dengan:

1. tidak melakukan plagiat terhadap naskah karya orang lain
2. tidak melakukan pemalsuan data
3. tidak menggunakan karya orang lain tanpa menyebut sumber asli atau tanpa ijin pemilik

Jika kemudian terbukti terjadi pelanggaran terhadap pernyataan di atas, maka saya bersedia menerima sanksi apapun termasuk pencabutan gelar akademik.

Lembar pernyataan ini juga memberikan hak kepada Politeknik Negeri Batam untuk mempergunakan, mendistribusikan ataupun memproduksi ulang seluruh hasil Tugas Akhir ini.

Batam, 29 Juni 2012

**Nur Yaqin**  
3310812034

## **LEMBAR PERNYATAAN**

Dengan ini, saya:

NIM : 3310912002

Nama : Nina Krismawati

adalah mahasiswa Teknik Informatika Politeknik Negeri Batam yang menyatakan bahwa tugas akhir dengan judul:

### **DATA WAREHOUSE PENERIMAAN MAHASISWA BARU**

disusun dengan:

1. tidak melakukan plagiat terhadap naskah karya orang lain
2. tidak melakukan pemalsuan data
3. tidak menggunakan karya orang lain tanpa menyebut sumber asli atau tanpa izin pemilik

Jika kemudian terbukti terjadi pelanggaran terhadap pernyataan di atas, maka saya bersedia menerima sanksi apapun termasuk pencabutan gelar akademik.

Lembar pernyataan ini juga memberikan hak kepada Politeknik Negeri Batam untuk mempergunakan, mendistribusikan ataupun memproduksi ulang seluruh hasil Tugas Akhir ini.

Batam, 29 Juni 2012

**Nina Krismawati**  
3310912002

## KATA PENGANTAR

Puji dan syukur kami ucapkan atas kehadiran Tuhan Yang Maha Esa, karena berkat dan rahmat serta hidayat-Nya, penyusun dapat menyelesaikan Tugas Akhir yang berjudul ***Data Warehouse* Penerimaan Mahasiswa Baru**. Analisis ini dibuat untuk memberikan sebuah tampilan informasi penerimaan mahasiswa baru yang sederhana, cepat dan akurat yang ada di dalam studi kasus Politeknik Negeri Batam. Dalam kesempatan ini, penyusun mengucapkan terima kasih kepada:

1. Bapak Ari Wibowo selaku koordinaator Tugas Akhir.
2. Ibu Hilda Widyastuti selaku pembimbing Tugas Akhir kami yang telah banyak memberi masukan dan saran.
3. Bapak/Ibu dosen program studi Informatika atas bimbingannya.
4. Kedua orang tua tercinta serta keluarga yang telah memberi dukungan moral dan doa.
5. Teman-teman seperjuangan Teknik Informatika 2008 yang telah memberikan dukungan, semangat dan doa.
6. Semua pihak yang telah membantu hingga Tugas Akhir ini selesai.

Dalam penulisan ini, penyusun mengakui bahwa masih banyak terdapat kekurangan dalam penyusunannya. Oleh karena itu, penyusun sangat mengharapkan bantuan dari berbagai pihak berupa kritik ataupun saran guna penyempurnaan selanjutnya. Akhir kata penyusun ucapkan terima kasih, semoga laporan ini bermanfaat bagi pembaca yang ingin mengembangkan sebuah analisis yang serupa.

Batam, 29 June 2012

Penulis

## **ABSTRAK**

### **DATA WAREHOUSE PENERIMAAN MAHASISWA BARU**

Panitia penerimaan mahasiswa baru Politeknik Negeri Batam menerima mahasiswa baru. Data penerimaan tersebut menumpuk setiap tahunnya, dan menjadi masalah dalam menyajikan informasi yan cepat dan menarik. Teknologi *data warehouse* dan On Line Analysis Processing (OLAP) dapat mengelola tumpukan data tersebut dan mempresentasikannya untuk membantu proses pengambilan keputusan. Tujuan penelitian ini adalah membangun *data warehouse* dan operasi-operasi OLAP, serta membangun aplikasi untuk memvisualisasikan hasil operasi-operasi OLAP. Data yang digunakan adalah data PMB dalam satu tahun.

*Data warehouse* membentuk skema star dengan satu fakta dan lima dimensi. data dari *data warehouse* dianalisis dengan operasi-operasi OLAP. Aplikasi OLAP dibangun dengan bahasa pemrograman PHP dan berbasis web sehingga akses data menjadi cepat. hasil dari eksplorasi data Politeknik Negeri Batam melalui web ini disajikan dalam bentuk grafik yang cepat dan menarik.

Kata kunci: Data Penerimaan Mahasiswa Baru, *data warehouse*, OLAP

## **ABSTRACT**

### **DATA WAREHOUSE ADMISSION OF NEW STUDENTS**

Admission committee of Polytechnic Batam accepts new students. Data revenue is accumulated each year, and the problem of presenting the information fast and exciting. Technology of data warehouse and On Line Analysis Processing (OLAP) can manage the stack data and present it to assist the decision making process. The purpose of this study was to build a data warehouse and OLAP operations, and build applications to visualize the results of OLAP operations. The data used is the data in a single year PMB.

Star schema of the data warehouse consists of one fact and five dimensions. Analyzed data from a data warehouse with OLAP operations. OLAP applications built with PHP programming language and web-based so that data access becomes faster. The results of data exploration Polytechnic Batam through this web are presented in graphical form a fast and exciting.

Keywords: Admissions Data, *data warehouse*, OLAP

## DAFTAR ISI

Bab I	Pendahuluan.....	1
I.1	Latar Belakang.....	1
I.2	Rumusan Masalah.....	2
I.3	Batasan Masalah.....	2
I.4	Tujuan.....	2
I.5	Sistematika Penulisan.....	3
Bab II	Landasan Teori.....	4
II.1	Penerimaan mahasiswa baru.....	4
II.2	Data warehouse.....	5
II.3	Perbedaan Data warehouse dengan OLTP.....	8
II.4	Keuntungan dan Kerugian Menggunakan Data warehouse.....	10
II.5	Struktur Data warehouse.....	11
II.6	Database, DBMS dan RDBMS.....	15
II.7	MySQL.....	16
II.8	Extract, Transform, Load (ETL).....	17
II.9	Metodologi penelitian.....	18
II.10	Metodologi pengujian.....	19
Bab III	Analisis dan Perancangan.....	21
III.1	Analisis Kebutuhan Pengguna.....	21
III.2	Perancangan Data warehouse.....	22
III.2.1	Perancangan Arsitektur Data Warehouse.....	22
III.2.2	Perancangan Sumber Data.....	22
III.2.3	Pemodelan Data Dimensional.....	24
III.3	Kapabilitas Data warehouse.....	28
Bab IV	Implementasi dan Pengujian.....	29
IV.1	Implementasi.....	29
IV.1.1	Lingkungan Pengembangan.....	29
IV.1.2	9 Tahap Metodologi.....	30

IV.1.3	Implementasi Basis Data .....	35
IV.1.4	Implementasi Antar Muka .....	37
IV.1.5	Pengujian Sistem .....	40
IV.1.6	Kasus dan Hasil Pegujian .....	42
Bab V	Kesimpulan dan Saran .....	44
V.1	Kesimpulan .....	44
V.2	Saran .....	44
	Daftar Pustaka .....	46
	Lampiran A Transformasi Data .....	47
1.	Jumlah Peserta Ujian Lokal .....	47
2.	Jumlah Peserta Ujian Lokal Diterima .....	50
3.	Jumlah Peserta Ujian Nasional .....	54
4.	Jumlah Peserta Ujian Nasional Diterima .....	57
5.	Jumlah Peserta PMDK .....	61
6.	Jumlah Peserta PMDK Diterima .....	64
7.	Jumlah Peserta Ujian .....	68
8.	Jumlah Peserta Ujian Diterima .....	71

## **DAFTAR GAMBAR**

Gambar 1 Arsitektur Data Warehouse .....	6
Gambar 2 Arsitektur Data Warehouse dengan Staging Area .....	7
Gambar 3 Skema bintang penerimaan mahasiswa baru.....	24
Gambar 4 Halaman menu utama (sebelum akses) .....	37
Gambar 5 Halaman menu utama (setelah akses) .....	38
Gambar 6 Hasil ekstraksi data menampilkan jumlah peserta ujian PMDK.....	39
Gambar 7 Halaman analisis data penerimaan mahasiswa baru .....	40

## DAFTAR TABEL

Tabel 1 Perbedaan <i>Data Warehouse</i> dan OLTP .....	9
Tabel 2 Data operasional PMB .....	23
Tabel 3 Tabel fakta penerimaan mahasiswa baru .....	25
Tabel 4 Tabel dimensi asal sekolah .....	25
Tabel 5 Tabel dimensi calon mahasiswa.....	26
Tabel 6 Tabel dimensi gelombang .....	27
Tabel 7 Tabel dimensi program studi .....	27
Tabel 8 Tabel dimensi jalur penerimaan mahasiswa baru .....	27
Tabel 9 Tabel pemilihan proses .....	30
Tabel 10 Tabel pemilihan grain .....	30
Tabel 11 Tabel penentuan dimensi .....	31
Tabel 12 Tabel penentuan fakta .....	31
Tabel 13 Tabel calonmhs_dim .....	33
Tabel 14 Tabel pmbasalsekolah_dim.....	33
Tabel 15 Tabel pmbjalurpenerimaan_dim .....	33
Tabel 16 Tabel pmbprodi_dim.....	33
Tabel 17 Tabel pmbgelombang_dim .....	33
Tabel 18 Tabel calon mahasiswa .....	35
Tabel 19 Tabel gelombang.....	36
Tabel 20 Tabel penerimaan mahasiswa baru .....	36
Tabel 21 Tabel program studi .....	36
Tabel 22 Tabel asal sekolah .....	36
Tabel 23 Tabel jalur penerimaan .....	37
Tabel 24 Tabel pengujian alfa.....	41
Tabel 25 Tabel pengujian akses data warehouse .....	42
Tabel 26 Tabel pengujian akses Extract, Taransform & Load .....	43

## **ABSTRAK**

### **DATA WAREHOUSE PENERIMAAN MAHASISWA BARU**

Panitia penerimaan mahasiswa baru Politeknik Negeri Batam menerima mahasiswa baru. Data penerimaan tersebut menumpuk setiap tahunnya, dan menjadi masalah dalam menyajikan informasi yan cepat dan menarik. Teknologi *data warehouse* dan On Line Analysis Processing (OLAP) dapat mengelola tumpukan data tersebut dan mempresentasikannya untuk membantu proses pengambilan keputusan. Tujuan penelitian ini adalah membangun *data warehouse* dan operasi-operasi OLAP, serta membangun aplikasi untuk memvisualisasikan hasil operasi-operasi OLAP. Data yang digunakan adalah data PMB dalam satu tahun.

*Data warehouse* membentuk skema star dengan satu fakta dan lima dimensi. data dari *data warehouse* dianalisis dengan operasi-operasi OLAP. Aplikasi OLAP dibangun dengan bahasa pemrograman PHP dan berbasis web sehingga akses data menjadi cepat. hasil dari eksplorasi data Politeknik Negeri Batam melalui web ini disajikan dalam bentuk grafik yang cepat dan menarik.

Kata kunci: Data Penerimaan Mahasiswa Baru, *data warehouse*, OLAP

## **ABSTRACT**

### **DATA WAREHOUSE ADMISSION OF NEW STUDENTS**

Admission committee of Polytechnic Batam accepts new students. Data revenue is accumulated each year, and the problem of presenting the information fast and exciting. Technology of data warehouse and On Line Analysis Processing (OLAP) can manage the stack data and present it to assist the decision making process. The purpose of this study was to build a data warehouse and OLAP operations, and build applications to visualize the results of OLAP operations. The data used is the data in a single year PMB.

Star schema of the data warehouse consists of one fact and five dimensions. Analyzed data from a data warehouse with OLAP operations. OLAP applications built with PHP programming language and web-based so that data access becomes faster. The results of data exploration Polytechnic Batam through this web are presented in graphical form a fast and exciting.

Keywords: Admissions Data, *data warehouse*, OLAP

# Bab I Pendahuluan

## I.1 Latar Belakang

Penerimaan mahasiswa baru sudah menjadi agenda tahunan di setiap universitas termasuk di Politeknik Negeri Batam. Prosedur penerimaan mahasiswa baru yang ada di Politeknik Negeri Batam adalah calon mahasiswa mengisi borang pendaftaran baik yang melalui jalur PMDK ataupun mengikuti seleksi ujian masuk, melakukan pendaftaran ulang setelah dinyatakan lulus, serta mengikuti matrikulasi dan ospek. Dokumen-dokumen yang berhubungan dengan pelaksanaan penerimaan mahasiswa baru dimasukkan ke Sistem Informasi Politeknik Batam (SIMPOL), dan beberapa dokumen yang lain disimpan di server Politeknik seperti, dokumen perencanaan dan evaluasi penerimaan mahasiswa baru, hasil ujian serta daftar hadir, dan hanya dapat diakses oleh panitia pelaksana dan manajemen.

Dari kegiatan penerimaan mahasiswa baru dari tahun ke tahun tentunya membuat ukuran *database* semakin bertambah besar, sehingga akan membebani sistem pemrosesan data yang ada. Selain itu, sistem informasi yang ada belum terintegrasi secara sempurna yang mengakibatkan informasi yang tersaji kurang lengkap dan akurat. Contoh, data mahasiswa yang mendaftar dan data hasil ujian disimpan di *database* yang berbeda, padahal pihak manajemen sebagai pembuat keputusan (*decision maker*) membutuhkan akses ke semua data yang ada. Dan juga, data-data yang ada di Sistem Informasi Politeknik Batam (SIMPOL) merupakan data operasional sehingga untuk mendapatkan informasi langsung, dilakukan proses *query* secara manual. Kendalanya adalah, tidak semua panitia pelaksana dan manajemen mampu melakukan proses *query*, tentunya hal ini membuat proses *query* tidak praktis dan efisien. Data yang tersimpan di Sistem Informasi Politeknik Batam (SIMPOL) dan server Politeknik Batam tidak hanya digunakan untuk keperluan operasional, namun dapat menjadi acuan pengambilan keputusan pada saat pihak manajemen ingin mendapatkan informasi yang bersifat *customized report* guna meningkatkan kualitas proses penerimaan mahasiswa baru

ditahun mendatang. Misalnya saja informasi mengenai pola kehadiran mahasiswa pada saat matrikulasi yang berubah dari tahun ke tahun, merupakan pertimbangan penting dalam mengambil keputusan.

Pembangunan *data warehouse* merupakan salah satu cara untuk mengekstrak informasi yang penting dari data yang tersebar di beberapa sistem informasi, dan dapat menyimpan data yang bervolume besar serta menyimpan data histori sehingga data tersebut dapat dimanfaatkan menjadi sebuah informasi yang berguna di masa yang akan datang.

## **I.2 Rumusan Masalah**

1. Bagaimana menyajikan laporan data penerimaan mahasiswa baru secara terintegrasi untuk memudahkan analisa pengambilan keputusan.
2. Bagaimana merancang skema bintang dan membangun data warehouse penerimaan mahasiswa baru di Politeknik Negeri Batam

## **I.3 Batasan Masalah**

1. Hanya menampilkan data dalam satu dimensi.
2. Hanya menampilkan data dalam satu periode penerimaan mahasiswa baru.

## **I.4 Tujuan**

Penelitian ini bertujuan untuk membangun *data warehouse* untuk penyimpanan data Penerimaan Mahasiswa Baru Politeknik Negeri Batam. Hasil yang diharapkan sebagai berikut:

1. Memudahkan pihak manajemen dalam melakukan *query* dan *costumized report* sesuai dengan kebutuhan yang diinginkan.
2. Memudahkan pihak kampus dalam mengambil keputusan yang tepat guna untuk meningkatkan kualitas kegiatan penerimaan mahasiswa baru di tahun selanjutnya.

## **I.5 Sistematika Penulisan**

Laporan ini terdiri dari bab pendahuluan, landasan teori, analisis , implementasi dan pembahasan, kesimpulan dan saran serta lampiran yang berhubungan dengan proses penerapan.

Bab 1 Pendahuluan berisi penjelasan mengenai latar belakang masalah dalam proses penelitian, perumusan masalah, batasan masalah dalam proses penelitian, tujuan penelitian, dan sistematika penulisan yang memberikan gambaran isi laporan tugas akhir.

Bab 2 Landasan Teori berisi mengenai proses penerimaan mahasiswa baru, data warehouse, *Extract Transform Loading* dan *metodologi* penelitian.

Bab 3 Analisa dan perancangan berisi mengenai analisis kebutuhan pengguna dan kebutuhan informasi, arsitektur dan rancangan data warehouse penerimaan mahasiswa baru.

Bab 4 Implementasi dan pengujian berisi mengenai proses implementasi dari hasil pemilihan dan perancangan.

Bab 5 Kesimpulan, Saran dan Solusi berisi mengenai penyimpulan hasil dari proses implementasi dan pengujian pada bab sebelumnya, saran sebagai bahan pertimbangan untuk pengembangan penelitian selanjutnya.

## **Bab II      Landasan Teori**

### **II.1    Penerimaan mahasiswa baru**

Berdasarkan survei Politeknik Negeri Batam setiap tahunnya selalu menaikkan standar nilai kelulusan baik melalui jalur ujian lokal dan ujian nasional maupun PMDK. Bagian program studi akan melakukan sesi wawancara untuk calon mahasiswa yang melalui jalur PMDK dengan mempertimbangkan nilai raport, asal sekolah, nilai non akademik dan hasil wawancara. Sementara untuk ujian lokal dan ujian nasional yang mengikuti ujian masuk, berdasarkan data tahun lalu nilai tertinggi adalah jurusan Teknik Informatika yang setiap tahun peminatnya semakin bertambah, kemudian Teknik Elektro dan Akutansi. Tahapan yang dilakukan oleh calon mahasiswa dalam pendaftaran mahasiswa baru di Politeknik Negeri Batam yaitu melakukan pendaftaran melalui ujian lokal dan ujian nasional atau PMDK, mengikuti ujian, kemudian melakukan daftar ulang setelah dinyatakan lulus dan yang terakhir melakukan konfirmasi pendaftaran. Sedangkan untuk petugas penerimaan mahasiswa baru Politeknik Negeri Batam adalah, manajemen memberikan keputusan kelulusan untuk calon mahasiswa, memberikan dan mencatat hasil dari proses daftar ulang sebagai bukti perjanjian, yang terakhir memberikan nilai induk mahasiswa dan mencatat bukti pendaftaran untuk dikonfirmasi.

Calon mahasiswa mengisi borang pendaftaran yang tersedia baik ujian lokal dan ujian nasional maupun PMDK. Data yang dimasukkan akan disimpan ke dalam database. Petugas akan menerima data pendaftar yang kemudian akan dikelola kembali. Untuk ujian lokal dan ujian nasional petugas menghitung nilai hasil ujian kemudian apabila sudah memenuhi syarat data tersebut akan dibandingkan dengan data pendaftar PMDK. Kemudian manajemen Politeknik Batam memiliki hak untuk menentukan pendaftar yang berhak lulus atau tidak.

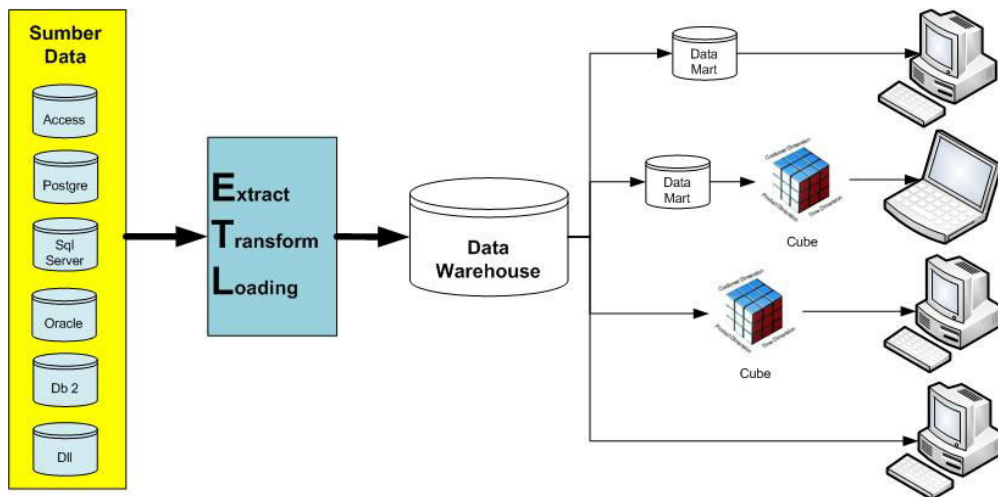
## II.2 Data warehouse

*Data warehouse* merupakan tempat penampungan data perusahaan atau intitusi yang disusun sedemikian rupa sehingga mengandung makna dan untuk analisis dan pelaporan. Sebuah *data warehouse* merupakan sumber informasi yang datanya diperoleh dari *Online Transaction Processing* (OLTP). Biasanya *data warehouse* ini menyimpan data yang bersifat historis. Seperti yang dikatakan oleh Turban, Aronson and Ting (2006, p305), *data warehouse* adalah sebuah basisdata komprehensif yang mendukung semua analisis keputusan yang diperlukan oleh suatu organisasi dengan menyediakan ringkasan dan rincian informasi. Sedangkan menurut Connolly dan Begg (2002, p1047), *data warehouse* adalah suatu kumpulan data yang bersifat :

- *Subject-oriented* artinya *data warehouse* didesain untuk menganalisa data berdasarkan subyek tertentu dalam organisasi, bukan pada proses atau fungsi aplikasi tertentu. Dalam kasus penerimaan mahasiswa baru yang menjadi subyeknya adalah pihak panitia dan manajemen. Dimana pihak-pihak tersebut yang menjadi pengguna dalam *data warehouse*.
- *Integrated Data Warehouse* dapat menyimpan data-data yang berasal dari sumber-sumber yang terpisah kedalam suatu format yang konsisten dan saling terintegrasi satu dengan lainnya. Dengan demikian data tidak bisa dipecah-pecah karena data yang ada merupakan suatu kesatuan yang menunjang keseluruhan konsep *data warehouse* itu sendiri. Dalam hal ini sumber data berasal dari Sistem Informasi Politeknik (SIMPOL) dan server Politeknik Batam.
- *Time-variant* artinya Seluruh data pada *data warehouse* dapat dikatakan akurat atau valid pada rentang waktu tertentu. Untuk melihat interval waktu yang digunakan dalam mengukur keakuratan suatu *data warehouse*.
- *Non-volatile* artinya data pada *data warehouse* tidak di-*update* secara *real time* tetapi di *refresh* dari sistem operasional secara *reguler*. Data yang baru selalu ditambahkan sebagai *suplemen* bagi *database* itu sendiri dari pada sebagai sebuah perubahan. *Database* tersebut secara kontinyu menyerap data

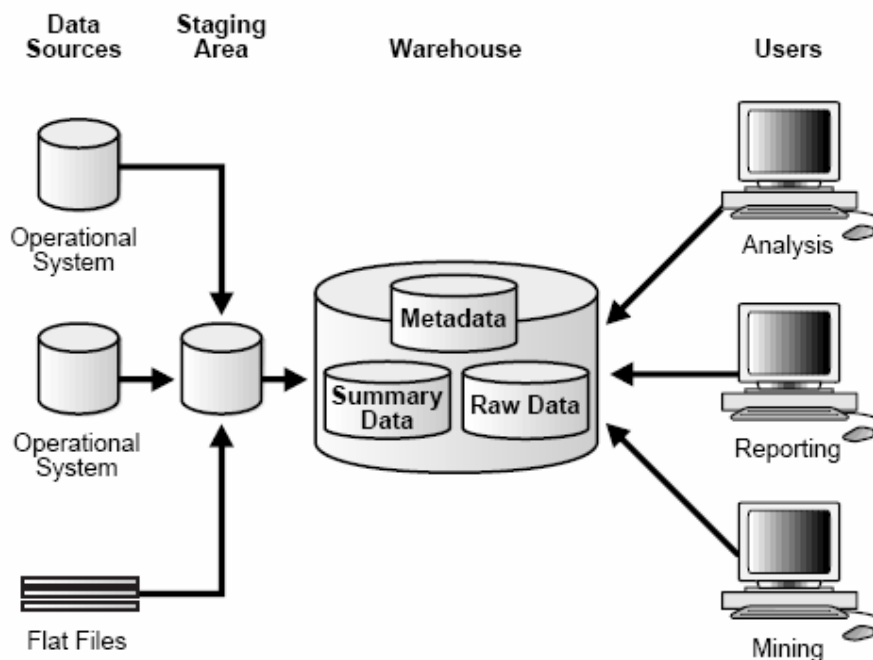
baru ini, kemudian secara *incremental* disatukan dengan data sebelumnya. Berbeda dengan *database* operasional yang dapat melakukan *update, insert* dan *delete* terhadap data yang mengubah isi dari *database* sedangkan pada data warehouse hanya ada dua kegiatan memanipulasi data yaitu *loading* data (mengambil data) dan akses data (mengakses data warehouse seperti melakukan *query* atau menampilkan laporan yang dibutuhkan, tidak ada kegiatan *updating* data).

Arsitektur *data warehouse* (lihat gambar 1 dibawah) terdiri dari sumber data, *Extract Transform Loading*, data mart dan cube. Sumber data, merupakan data operasional yang disimpan dalam *database*, yang akan diproses (*Extract Transform Loading*) dan diintegrasikan kedalam *data warehouse*. Sedangkan data mart dan cube berisi data-data yang mendukung fungsi bisnis, seperti yang dikatakan oleh Inmon (2005, p321) data mart adalah sub-set dari *data warehouse* yang umumnya terdiri dari sebuah subjek tunggal. Jadi, data mart merupakan serangkaian data yang hanya menjelaskan satu fungsi dari operasi perusahaan.



Gambar 1 Arsitektur Data Warehouse

Dari gambar 1, terlihat aliran data dari operational ke data warehouse, dilanjutkan dengan pengembangan aplikasi (*reporting*) yang akan digunakan untuk menganalisis dan mengevaluasi bisnis. Seperti apa yang disampaikan oleh Inmon , data mengalir dari lingkungan operasional ke dalam data warehouse dimana data mengalami transformasi dari tingkatan operasional ke tingkatan data warehouse. Setiap universitas dalam mencapai tujuannya menggunakan strategi yang berbeda, hal ini membuat jenis data dan tipe data bahkan arsitektur dan proses bisnisnya ikut berbeda. Dalam melakukan perancangan *data warehouse* harus ditentukan arsitektur yang cocok untuk pengembangan *data warehouse*. Dari gambar 3 dibawah ini, terlihat proses pengolahan data operasional sebelum dimasukkan ke dalam *data warehouse* melalui *staging area* terlebih dahulu. *Staging area* ini digunakan untuk memudahkan dalam melakukan integrasi dan pembersihan data sehingga dapat menghasilkan data yang berkualitas. Karena didalam *Staging area* terdapat proses untuk penggabungan data, pembersihan data dan standarisasi data.[1]



Gambar 2 Arsitektur Data Warehouse dengan Staging Area

### II.3 Perbedaan Data warehouse dengan OLTP

Berikut beberapa penjelasan perbedaan antara *data warehouse* dan OLTP (*Online Transaction Processing*) berdasarkan:

- Karakteristik

OLTP mempunyai *database* yang sering kita jumpai dalam aplikasi-aplikasi yang berhubungan dengan aktivitas sehari-hari dari suatu toko atau perusahaan, contoh: *database* yang digunakan pada sistem informasi penjualan. Sehingga *database* tersebut didesain atau dirancang khusus untuk menangani kegiatan operasional atau transaksi dari suatu toko atau perusahaan.

Hal ini jelas berbeda dengan *data warehouse* karena *data warehouse* menyimpan data yang bersifat *historical* dan digunakan untuk menangani kegiatan penyajian informasi.

- Orientasi

OLTP lebih berorientasi pada transaksi, sedangkan *data warehouse* lebih diorientasikan pada proses untuk melakukan analisis.

- Struktur Data

Karena OLTP diorientasikan pada proses transaksi, maka *query* yang digunakan lebih dioptimalkan untuk mendukung proses transaksi tersebut. Sehingga harus dihindarkan dari data yang berulang-ulang atau redudan. Dalam hal ini, perancangannya membutuhkan normalisasi. Sedangkan dalam *data warehouse* yang berorientasi pada proses *query* untuk menghasilkan suatu informasi yang lengkap dan akurat, maka memungkinkan terjadinya data yang berulang-ulang karena dibutuhkan akses yang cepat dalam *query*-nya. Dalam hal ini, perancangannya haruslah denormalisasi.

- Tipe Data

Dalam hal tipe data, data yang dimiliki oleh OLTP bersifat bernilai sekarang. Hal ini disebabkan karena OLTP menyimpan data operasional atau transaksi yang terjadi secara terus-menerus dalam suatu perusahaan atau toko.

Sedangkan *data warehouse* memiliki data yang bersifat *historical*, karena data tersebut hanya akurat pada saat proses ETL (*update*) dilaksanakan atau berdasarkan periode yang telah ditentukan.

- Frekuensi Akses

Frekuensi dalam mengakses OLTP sangat tinggi karena data terus-menerus dibutuhkan untuk mendukung kegiatan operasional atau transaksi dalam suatu perusahaan atau toko. Contoh: dalam sistem informasi penjualan, dimana data yang terdapat dalam tabel yang berisi informasi barang akan sangat ditinggi diakses untuk mendukung proses transaksi (misalnya harga barang).

Sedangkan dalam *data warehouse*, frekuensi aksesnya sangat sedang dan bahkan dapat dikatakan rendah, hal ini disebabkan karena *data warehouse* hanya diakses pada saat proses analisis dibutuhkan.

Untuk melihat perbedaan yang lainnya antara *Data Warehouse* dan OLTP, Anda dapat melihat pada tabel di bawah ini:

Perbedaan	<i>Data Warehouse</i>	<i>OLTP</i>
Karakteristik	Proses penyajian informasi	Proses operasional
Orientasi	Analisis	Transaksi
Struktur data	Dioptimasi untuk <i>query</i> , denormalisasi	Dioptimasi untuk transaksi, normalisasi
Tipe data	<i>History</i> , hanya akurat pada saat <i>update</i>	Bernilai sekarang atau <i>up-to-date</i>
Frekuensi akses	Sedang-rendah	Tinggi
Tipe akses	<i>Read</i>	<i>Read, update, delete</i>
Penggunaan	<i>Update</i> secara periodic	<i>Update</i> secara terus-menerus
Pengguna	Lebih sedikit	Banyak
Jumlah <i>record</i> yang diakses	Jutaan	Ribuan
Desain <i>database</i>	Data multidimensi	Model E-R
Ukuran <i>database</i>	100 GB hingga TB	100 MB hingga GB

**Tabel 1 Perbedaan *Data Warehouse* dan OLTP**

## II.4 Keuntungan dan Kerugian Menggunakan Data warehouse

- Keuntungan menggunakan *data warehouse*
  1. *Data warehouse* menyediakan model data yang umum untuk semua data tidak tergantung sumber datanya. Hal ini mempermudah pembuatan laporan dan analisis informasi dibandingkan jika diperlukan banyak model data untuk menerima informasi seperti faktur penjualan, kuitansi pemesanan, tagihan-tagihan lainnya.
  2. Saat meload data ke dalam *datawarehouse*, ketidakkonsistenan akan teridentifikasi dan diperbaiki. Hal ini akan mendukung juga dalam proses pembuatan laporan dan analisis.
  3. Informasi yang disimpan dalam *data warehouse* dibawah kontrol dari pengguna data warehouse, sehingga sekalipun sumber data dibersihkan pada suatu waktu, informasi itu akan tetap tersimpan dengan aman di *data warehouse*.
  4. Karena *data warehouse* terpisah dari sistem operasional, maka *data warehouse* dapat menerima data tanpa memperlambat kerja kerja sistem operasional
  5. *Data warehouse* menyediakan fasilitas yang mendukung pengambilan keputusan seperti laporan berdasarkan tren (misalnya : barang yang paling banyak terjual di suatu area dalam 2 tahun terakhir), laporan perkecualian, dan laporan yang menampilkan pencapaian di lapangan yang sesungguhnya dibandingkan dengan gol yang telah ditetapkan.
- Kerugian menggunakan *data warehouse*
  1. *Data warehouse* bukan merupakan lingkungan yang optimal untuk data yang tidak terstruktur
  2. Data perlu untuk diekstrak, diubah, dan diload ke *data warehouse*, sehinggann terdapat delay (tenggat waktu) di mana data yang dimasukkan ke dalam *data warehouse* belum terdeteksi.

3. Semakin lama masa hidupnya, maka *data warehouse* dapat menyebabkan biaya yang besar. *Data warehouse* umumnya tidak statis. Biaya perawatannya cukup tinggi.
4. *Data warehouse* dapat menjadi ketinggalan dari data terbaru relatif cepat. Maka, akan ada resiko bahwa data yang akan dianalisis di *data warehouse* merupakan data yang tidak optimal.

## II.5 Struktur Data warehouse

Metadata merupakan suatu bentuk informasi yang berisi data yang akan digunakan, tipe data, panjang datanya serta sumber datanya yang akan digunakan dalam *data warehouse*. Seperti apa yang dikatakan oleh Inmon (2002, p393), metadata adalah data mengenai data atau deskripsi dari struktur, isi, kunci, indeks, dan lain-lain mengenai data. Sedangkan Mallach (2002, p474) mengatakan metadata adalah data tentang data yang berguna sebagai pusat penyimpanan informasi untuk menjelaskan kepada user tentang apakah *data warehouse* itu, dari mana asalnya dan siapa yang bertanggung jawab atas hal tersebut dan sebagainya. Menurut Connolly (2005, p1055), Metadata digunakan untuk berbagai tujuan meliputi :

- Proses ekstraksi dan loading. Metadata digunakan untuk memetakan sumber data ke dalam pandangan umum dari data dalam warehouse.
- Proses manajemen warehouse. Metadata digunakan untuk mengotomatiskan pembuatan tabel ringkasan.
- Sebagai bagian dari proses manajemen query. Metadata digunakan untuk menghubungkan suatu query dengan sumber data yang tepat.

Dalam penyusunan metadata, tidaklah dilakukan dengan semauanya, tetapi harus memenuhi syarat-syarat yang berlaku. Syarat-syarat dalam pembuatan metadata menurut Inmon (2005, p102) :

- Struktur data yang dikenal programmer
- Struktur data yang dikenal analis *Decision Support System (DSS)*

- Sumber data yang membantu data warehouse
- Transformasi data ketika dilewatkan ke *data warehouse*
- Model data
- Hubungan antara model data dan *data warehouse*
- Histori dari ekstrak

Metadata dalam data warehouse dibagi menjadi 3 kategori (Ponniah, 2001, p36), yaitu :

- Metadata operasional, berisi mengenai informasi tentang sumber data operasional yang memiliki struktur data yang berbeda, ukuran field yang berbeda, dan tipe data yang berbeda.
- Metadata ekstraksi dan transformasi, berisi mengenai data ekstraksi dari sumber data, penamaan, frekuensi ekstraksi, metode ekstraksi, dan peraturan untuk ekstraksi.
- Metadata pengguna akhir, adalah sebuah peta navigasi dari data warehouse. Ini memungkinkan pengguna akhir untuk menemukan informasi dari data warehouse.

Berdasarkan Kimball seperti yang dikutip oleh Connolly dan Begg (2002, p1083), terdapat 9 tahap *metodologi* dalam membangun *data warehouse* yang terdiri dari:

1. Memilih proses
2. Memilih grain
3. Mendefinisikan dan menyesuaikan dimensi
4. Memilih fakta
5. Menyimpan pre kalkulasi dalam table
6. Melengkapi table dimensi
7. Memilih durasi waktu
8. Melacak perubahan dimensi secara perlahan
9. Menentukan prioritas dan model query.

Untuk pemodelan *data warehouse*, lebih digunakan teknik pemodelan dimensional. Dengan teknik ini, dapat dibuat tabel fakta, tabel dimensi, dan membangun relasi antara masing-masing tabel dimensi dan tabel fakta. Ada beberapa hasil pemodelan tersebut, antara lain :

### **Skema Bintang (*Star Schema*)**

Menurut Ponniah (2001, pp210-216), skema bintang (*star schema* ) adalah teknik dasar perancangan data untuk *data warehouse*. Struktur skema bintang adalah suatu struktur yang dapat dengan mudah dipahami dan digunakan oleh pengguna. Struktur tersebut mencerminkan bagaimana pengguna biasanya memandang ukuran-ukuran kritis mengikuti dimensi-dimensi bisnis yang ada. Dalam skema bintang tergambar dua jenis tabel, yaitu tabel dimensi dan tabel fakta. Kedua table tersebut mempunyai karakteristik sebagai berikut; [2]

- Tabel dimensi
  - ✓ *Key* tabel dimensi, merupakan *primary key* dari tabel dimensi yang mengidentifikasi setiap baris dalam tabel secara unik.
  - ✓ Merupakan tabel yang lebar. Tabel dimensi memiliki jumlah kolom atau atribut yang banyak, oleh karena itu tabel dimensi bersifat lebar.
  - ✓ Atribut berupa teks. Dalam tabel dimensi, jarang ditemukan nilai numerik untuk perhitungan, atribut umumnya berupa teks yang merepresentasikan deskripsi tekstual dari komponen-komponen dalam dimensi bisnis.
  - ✓ Atribut-atribut tidak berhubungan secara langsung.
  - ✓ Tidak dinormalisasi. Untuk kinerja *query* yang efektif, paling baik jika *query* mengambil dari tabel dimensi dan langsung ke tabel fakta tanpa melalui tabel perantara yang akan terbentuk jika tabel dimensi dinormalisasi.
  - ✓ Kemampuan *drill-down dan roll-up*. Atribut-atribut dalam tabel dimensi menyediakan kemampuan untuk mendapatkan detail dari tingkat tinggi agregasi sampai tingkat detail yang rendah.

- ✓ Terdapat beberapa hirarki. Berbagai bagian perusahaan dapat mengelompokkan dimensi dengan cara yang berbeda, sehingga terbentuk lebih dari 1 hirarki.
  - ✓ Jumlah *record* yang lebih sedikit. Tabel dimensi umumnya memiliki jumlah *record* atau baris yang lebih sedikit dari tabel fakta.
- Tabel fakta
    - ✓ *Concatenated key*. Baris dalam tabel fakta diidentifikasi dengan menggunakan *primary key* dari tabel tabel dimensi, maka *primary key* dari tabel fakta merupakan gabungan *primary key* dari semua tabel dimensi.
    - ✓ *Data grain*, merupakan tingkat detail untuk pengukuran. Sebagai contoh, jumlah mahasiswa yang baru lulus berhubungan dengan banyaknya jumlah pendaftar calon mahasiswa baru, dan pendaftar yang melalui jalur PMDK.
    - ✓ *Fully additive measures*. Agregasi dari *fully additive measures* dilaksanakan dengan penjumlahan sederhana nilai-nilai atribut tersebut.
    - ✓ *Semiadditive measures*. *Semiadditive measures* merupakan nilai yang tidak dapat langsung dijumlahkan, sebagai contoh persentase keuntungan.
    - ✓ Tabel besar, tidak lebar. Tabel fakta umumnya memiliki lebih sedikit atribut daripada tabel dimensi, namun memiliki jumlah *record* yang lebih banyak.
    - ✓ *Sparse data*. Tabel fakta tidak perlu menyimpan *record* yang nilainya null. Maka tabel fakta dapat memiliki gap.
    - ✓ *Degenerate dimensions*. Terdapat elemen-elemen data dari sistem operasional yang bukan merupakan fakta ataupun dimensi, status mahasiswa yang bekerja dan yang belum bekerja, namun atribut-atribut tersebut dapat berguna dalam analisis tertentu. Sebagai contoh, mencari rata-rata status jumlah calon mahasiswa yang bekerja dan yang belum bekerja maka status tersebut harus dihubungkan disetiap gelombang penerimaan mahasiswa baru untuk mendapatkan rata-rata.

## II.6 Database, DBMS dan RDBMS

Basis data (atau database) adalah kumpulan informasi yang disimpan di dalam computer secara sistematis sehingga dapat diperiksa menggunakan suatu program computer untuk memperoleh informasi dari basis data tersebut. Database digunakan untuk menyimpan informasi atau data yang terintegrasi dengan baik di dalam komputer.

Untuk mengelola database diperlukan suatu perangkat lunak yang disebut DBMS (*Database Management System*). DBMS merupakan suatu sistem perangkat lunak yang memungkinkan *user* untuk membuat, memelihara, mengontrol dan mengakses database secara praktis dan efisien. Dengan DBMS, *user* akan lebih mudah mengontrol dan memanipulasi data yang ada.

Sedangkan RDBMS atau *Relationship Database Management System* merupakan salah satu jenis DBMS yang mendukung adanya *relationship* atau hubungan antar table. Disamping RDBMS, terdapat jenis DBMS lain, misalnya *Hierarchy DBMS*, *Object Oriented DBMS*, dsb.

Beberapa software atau perangkat lunak DBMS yang sering digunakan dalam aplikasi program antara lain:

- DB2
- Microsoft SQL Server
- Oracle
- Sybase
- Interbase
- Teradata
- Firebird
- MySQL
- PostgreSQL

## II.7 MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL atau DBMS yang *multithread*, *multi-user*, dengan sekitar 6 juta instalasi diseluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis di bawah lisensi GNU *General Public License* (GPL), tetapi mereka juga menjual dibawah lisensi komersial untuk kasus-kasus dimana penggunaanya tidak cocok dengan penggunaan GPL.

Beberapa kelebihan MySQL antara lain:

- Free
- Stabil dan tangguh
- Fleksibel dengan berbagai pemrograman
- *Security* yang baik
- Dukungan dari banyak komunitas
- Kemudahan *management database*
- Mendukung transaksi
- Perkembangan software yang cukup cepat.

Berikut ini beberapa tools yang biasa digunakan dalam mempermudah administrasi server MySQL. Tools berikut ini hanya digunakan untuk mempermudah administrasi MySQL, jadi tidak harus digunakan.

- **MySQL-Front** merupakan *front-end* MySQL berbasis Windows yang cukup banyak digunakan. MySQL-Front memiliki user interface yang cukup mudaj digunakan, bahkan oleh *user* pemula. Pada awalnya MySQL-Front merupakan software yang *free*, namun mulai versi 3.0 ke atas, software ini menjadi software yang bersifat *shareware* dengan masa percobaan selama 30 hari.
- **PHP MYAdmin** merupakan *front-end* MySQL berbasis Web. PHPMYAdmin dibuat dengan menggunakan PHP. Saat ini, PHPMyAdmin banyak digunakan dalam hamper semua penyedia hosting yang ada di internet.
- **SQLYog**
- **MySQL Client**

## II.8 Extract, Transform, Load (ETL)

ETL merupakan proses yang sangat penting dalam *data warehouse*, dengan ETL inilah data dari operational dapat dimasukkan ke dalam *data warehouse*. ETL juga dapat digunakan untuk mengintegrasikan data dengan sistem yang sudah ada sebelumnya (Untuk lebih jelasnya dapat dilihat pada Gambar 1 diatas).

Tujuan ETL adalah mengumpulkan, menyaring, mengolah, dan menggabungkan data-data yang relevan dari berbagai sumber untuk disimpan ke dalam data warehouse. Hasil dari proses ETL adalah dihasilkannya data yang memenuhi kriteria data warehouse seperti data yang historis, terpadu, terangkum, statis, dan memiliki struktur yang dirancang untuk keperluan proses analisis.

- Ekstrak

Langkah pertama pada proses ETL adalah mengekstrak data dari sumber-sumber data. Kebanyakan proyek data warehouse menggabungkan data dari sumber-sumber yang berbeda. Sistem-sistem yang terpisah sangat mungkin menggunakan format data yang berbeda. Ekstraksi adalah mengubah data ke dalam suatu format yang berguna untuk proses transformasi.

- *Transform*

Tahapan transformasi menggunakan serangkaian aturan atau fungsi untuk mengekstrak data dari sumber dan selanjutnya akan dimasukkan ke data warehouse. Berikut adalah hal-hal yang dapat dilakukan dalam tahapan transformasi:

- ✓ Hanya memilih kolom tertentu saja untuk dimasukkan ke dalam data warehouse.
- ✓ Menterjemahkan nilai-nilai yang berupa kode.
- ✓ Mengkodekan nilai-nilai ke dalam bentuk bebas (Contohnya memetakan “Pria” dengan “P” dan “Wanita” ke dalam “W”).
- ✓ Melakukan perhitungan nilai-nilai baru (Contohnya nilai = qty \* harga\_satuan).
- ✓ Menggabungkan data secara bersama-sama dari berbagai sumber.
- ✓ Membuat ringkasan dari sekumpulan baris data.

- ✓ Mengubah sekumpulan kolom menjadi sekumpulan baris atau sebaliknya.
  - ✓ Memisahkan sebuah kolom menjadi berbagai kolom.
- *Load*

Fase *load* merupakan tahapan yang berfungsi untuk memasukkan data ke dalam target akhir, yang biasanya ke dalam suatu data warehouse. Jangka waktu proses ini tergantung pada kebutuhan organisasi. Beberapa data warehouse dapat setiap minggu mengisi keseluruhan informasi yang ada secara kumulatif, data diubah, sementara data warehouse yang lain (atau bagian lain dari data warehouse yang sama) dapat menambahkan data baru dalam suatu bentuk yang *historikal*, contohnya setiap jam. Waktu dan jangkauan untuk mengganti atau menambah data tergantung dari perancangan data warehouse pada waktu menganalisis keperluan informasi. Fase *load* berinteraksi dengan suatu database, *constraint* didefinisikan dalam skema database sebagai suatu trigger yang diaktifkan pada waktu me-load data (Contohnya *uniqueness*, *referential integrity*, *mandatory fields*), yang juga berkontribusi untuk keseluruhan tampilan dan kualitas data dari proses *ETL*. Masalah-masalah yang terjadi dalam *ETL* adalah sumber-sumber data umumnya sangat bervariasi diantaranya:

    - ✓ Platform mesin dan sistem operasi yang berlainan.
    - ✓ Mungkin melibatkan sistem kuno dengan teknologi basis data yang sudah ketinggalan zaman.
    - ✓ Kualitas data yang berbeda-beda.
    - ✓ Aplikasi sumber data mungkin menggunakan nilai data (representasi) internal yang sulit dimengerti.[2]

## II.9 Metodologi penelitian

Metodologi yang digunakan dalam penulisan ini adalah sebagai berikut:

- Metode Analisis dilakukan melalui tahap:
  - ✓ Survey atas sistem yang berjalan dengan melakukan wawancara.
  - ✓ Analisis terhadap data yang didapatkan dari survey.

- ✓ Mengidentifikasi informasi yang dibutuhkan untuk pembuatan data warehouse yaitu dengan mengidentifikasi masalah yang terjadi dan pemecahannya.
- ✓ Identifikasi persyaratan sistem yang akan dibangun.
- Metode Perancangan Data warehouse yang digunakan adalah metode perancangan data warehouse menurut Kimball yang dikutip oleh Connolly dan Begg, dengan tahap-tahap sebagai berikut:
  - ✓ Memilih proses, pada proses ini mengacu pada data-data penerimaan mahasiswa baru. Data-data penerimaan mahasiswa baru akan dibangun harus tepat waktu, dan sesuai anggaran.
  - ✓ Memilih grain, memutuskan dengan pasti apa yang menjadi fakta. Dengan memilih grain dapat mengidentifikasi fakta tabel dimensi.
  - ✓ Membangun set dimensi yang akan memudahkan data mart untuk dipahami dan digunakan.
  - ✓ Memilih fakta yang digunakan dalam *data mart*
  - ✓ Mengecek ulang fakta-fakta yang terpilih.
  - ✓ Melengkapi table dimensi dengan menambahkan keterangan yang dapat dimengerti oleh pengguna
  - ✓ Memilih durasi waktu berapa lama kebutuhan untuk penyimpanan data.

## II.10 Metodologi pengujian

- Pengujian *Black box*

Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak. Disebut juga pengujian behavioral atau pengujian partisi. Pengujian *black box* memungkinkan perekayasa perangkat lunak mendapatkan serangkaian input yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program. Pengujian *black box* berusaha menemukan :

- ✓ Fungsi-fungsi yang tidak benar atau hilang
- ✓ Kesalahan *interface*
- ✓ Kesalahan dalam struktur data atau akses database eksternal.

- ✓ Kesalahan kinerja
- ✓ Inisialisasi dan kesalahan terminasi.

Dengan mengaplikasikan teknik *black box*, maka kita menarik serangkaian *test case* yang memenuhi kriteria berikut :

- ✓ *Test case* yang mengurangi, dengan harga lebih dari satu, jumlah *test case* tambahan yang harus di desain untuk mencapai pengujian yang dapat dipertanggungjawabkan.
- ✓ *Test case* yang member tahu kita sesuatu mengenai kehadiran atau ketidakhadiran kelas kesalahan, daripada member tahu kesalahan yang berhubungan hanya dengan pengujian spesifik.

- Pengujian Alfa dan Beta

Pengujian merupakan suatu proses yang tidak bisa dihindari. Pengujian berfungsi untuk melihat sejauh mana aplikasi berjalan dan untuk menemukan kesalahan yang terdapat pada aplikasi. Menurut Tay Vaughan dalam bukunya yang berjudul *multimedia making it work*, pengujian dibedakan menjadi dua, yaitu pengujian alfa dan pengujian beta. Pengujian alfa dan pengujian beta digunakan oleh pengembang aplikasi untuk mendeskripsikan tingkatan perkembangan.

Pengujian alfa merupakan pengujian yang dilakukan oleh pembuat aplikasi atau orang-orang yang terlibat didalamnya. Sedangkan beta dilakukan oleh orang-orang yang tidak terlibat dalam pengembangan proyek atau pengujian ini dilakukan dari sisi *user*, baik seorang maupun beberapa orang.

## **Bab III Analisis dan Perancangan**

### **III.1 Analisis Kebutuhan Pengguna**

Dari pengumpulan data dan informasi yang dilakukan dalam rangka mengetahui kebutuhan pengguna melalui wawancara didapat beberapa hal yang dibutuhkan oleh pihak-pihak yang akan menjadi pengguna yang akan memanfaatkan *data warehouse* penerimaan mahasiswa baru. Beberapa hal tersebut adalah keinginan akan kebutuhan ketersediaan data yang disimpan dalam sebuah *database*, ketersediaan akan informasi mengenai laporan penerimaan mahasiswa baru yang disajikan secara lengkap dari 3-5 tahun terakhir dalam bentuk grafik.

Informasi-informasi itu diharapkan bisa menampilkan kebutuhan data untuk laporan penerimaan mahasiswa baru antara lain:

1. Jumlah peserta ujian masuk lokal per gelombang, per program studi, per asal sekolah, per asal kota/kabupaten dan totalnya.
2. Jumlah peserta ujian masuk lokal yang dinyatakan diterima per gelombang, per program studi, per asal sekolah, per asal kota/kabupaten dan totalnya.
3. Jumlah peserta ujian masuk secara nasional per program studi, per asal sekolah, per asal kota/kabupaten dan totalnya.
4. Jumlah peserta ujian masuk secara nasional yang dinyatakan diterima per program studi, per asal sekolah, per asal kota/kabupaten dan totalnya.
5. Jumlah peserta PMDK per program studi, per asal sekolah, per asal kota/kabupaten dan totalnya.
6. Jumlah peserta PMDK yang dinyatakan diterima per program studi, per asal sekolah, per asal kota/kabupaten dan totalnya.
7. Jumlah peserta seluruh jalur seleksi penerimaan mahasiswa baru baik dari ujian lokal, nasional dan PMDK per asal sekolah, per asal kota/kabupaten.
8. Jumlah peserta seluruh jalur seleksi penerimaan mahasiswa baru baik dari ujian lokal, nasional dan PMDK yang telah dinyatakan lulus dan diterima per asal sekolah, per asal kota/kabupaten.

### **III.2 Perancangan Data warehouse**

Tahap perancangan ini meliputi perancangan arsitektur data warehouse, perencanaan sumber data, serta pemodelan data dimensional.

#### **III.2.1 Perancangan Arsitektur Data Warehouse**

Sumber data operasional yang digunakan adalah *database* Sistem Informasi Politeknik Batam (SIMPOL) yang merupakan *database* berisi semua data akademik di Politeknik Negeri Batam. Dari sumber data tersebut, kemudian dilakukan pemilihan data dan selanjutnya dimuatkan ke *database* terpisah yaitu DBPMB agar *database* operasional tidak terganggu kerja sistemnya. Dalam teori data warehouse, DBPMB disebut juga sebagai *data staging*. Dalam proses pemuatan data ke *database* DBPMB, selalu dilakukan proses pengecekan terlebih dahulu untuk memastikan bahwa data valid dan dapat dimuat ke *database* DBPMB. Hasil dari proses tersebut kemudian disimpan ke dalam data warehouse.

#### **III.2.2 Perancangan Sumber Data**

Politeknik Negeri Batam telah mengembangkan beberapa aplikasi untuk menunjang pelaksanaan kegiatan akademis yang dapat digunakan oleh mahasiswa, dosen maupun staf administrasi. Aplikasi yang telah dikembangkan merupakan aplikasi berbasis intranet (hanya dapat diakses di lingkungan kampus dengan teknik *client server*) dan juga aplikasi yang berbasis web yang dapat diakses dari internet. Aplikasi tersebut antara lain:

- Sistem Informasi Politeknik Negeri Batam (SIMPOL) yang merupakan aplikasi akademik yang utama. Dengan aplikasi ini dosen dapat melakukan pengisian nilai secara online. Mahasiswa dapat melakukan pendaftaran ulang dan melihat nilai secara *online*.
- Calon mahasiswa yang mendaftar harus mengisi form pendaftaran, form ini yang kemudian akan di masukkan kedalam database oleh panitia penerimaan mahasiswa baru sebagai acuan. Berikut merupakan data identitas mahasiswa dari form pendaftaran yang di masukkan kedalam database.

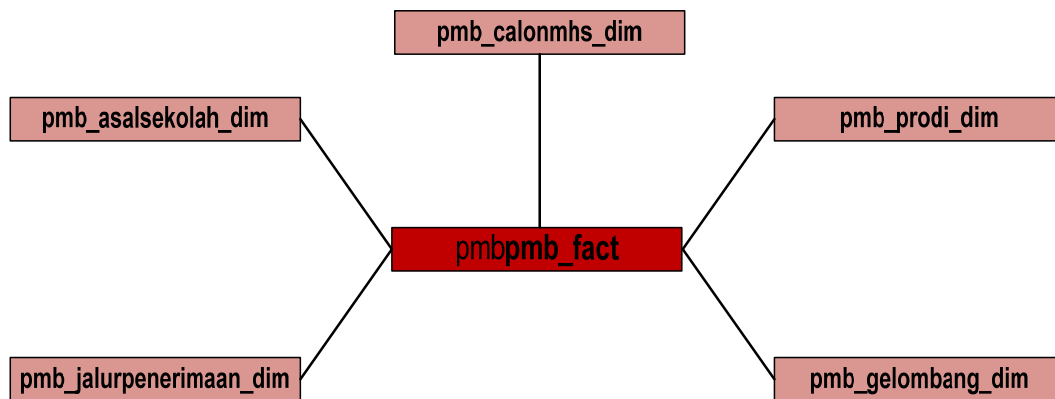
dataoperasionalPMB
<b>no_ujian : varchar(20)</b>
nama : varchar(30)
jenis_kelamin : enum('laki-laki','perempuan')
tempat_lahir : varchar(30)
tanggal_lahir : date
agama : varchar(11)
kewarganegaraan : varchar(11)
alamat : varchar(30)
kode_pos : int(11)
telepon : varchar(30)
kota_asal : varchar(30)
jenis_kelas : enum('reguler','karyawan')
tahun_mulai_sekolah : varchar(4)
tahun_tamat_sekolah : varchar(4)
jurusan_sekolah : varchar(30)
nilai_UAN : int(11)
nilai_STK : int(11)
tahun_masuk_perg_tinggi : year(4)
tahun_keluar_perg_tinggi : year(4)
jurusan_perg_tinggi : varchar(30)
IPK : varchar(4)
semester : int(11)
nama_perusahaan : varchar(100)
posisi/jabatan : varchar(30)
tahun_kerja : int(4)
nama_orang_tua : varchar(30)
hubungan_orang_tua : varchar(30)
alamat_orang_tua : varchar(30)
telepon_orang_tua : varchar(30)
informasi_pendaftaran : enum('iklan','TV','koran','radio','sosialisasi')
nilai_semester1 : int(11)
nilai_semester2 : int(11)
nilai_semester3 : int(11)
nilai_semester4 : int(11)
nilai_semester5 : int(11)
nilai_semester6 : int(11)
prestasi_akademik_semester1 : int(11)
prestasi_akademik_rata1 : float
prestasi_akademik_peringkat_kelas1 : int(11)
prestasi_akademik_peringkat_sekolah1 : int(11)
prestasi_non_akademik_tahun1 : year(4)
prestasi_non_akademik_nama_penghargaan : varchar(100)
prestasi_non_akademik_tingkat1 : varchar(100)
prestasi_organisasi_tahun1 : varchar(10)
prestasi_organisasi_jabatan1 : varchar(100)
prestasi_organisasi_nama_organisasi1 : varchar(100)
prestasi_non_formal_tahun1 : varchar(10)
prestasi_non_formal_penyelenggara1 : varchar(100)
prestasi_non_formal_jenis1 : varchar(100)

**Tabel 2 Data operasional PMB**

### III.2.3 Pemodelan Data Dimensional

Skema yang digunakan untuk pemodelan data adalah skema bintang dimana terdapat satu tabel fakta dan beberapa tabel dimensi. Penggunaan skema bintang memungkinkan proses *query* yang lebih ringan dan memudahkan penjelajahan terhadap tabel dimensinya. Tabel fakta yang terbentuk dari perancangan data warehouse ini merupakan tabel yang berhubungan dengan kegiatan pendaftaran mahasiswa baru. Tabel dimensi merupakan tabel yang berisi data yang menunjukkan tinjauan dari tabel fakta. Tabel fakta dan tabel dimensi yang ada dalam *data warehouse* PMB adalah sebagai berikut:

- Skema bintang penerimaan mahasiswa baru



Gambar 3 Skema bintang penerimaan mahasiswa baru

- **Tabel fakta penerimaan mahasiswa baru**

Table pmb\_fact berisi semua data yang berhubungan dengan penerimaan mahasiswa baru yang meliputi data pribadi, status penerimaan, dan asal sekolah calon mahasiswa dan jumlah mahasiswa yang mendaftar.

<b>pmbpmb_fact</b>
id_calonmhs : varchar(20)
id_asalsekolah : varchar(20)
id_jalurpenerimaan : varchar(10)
id_prodidipilihan1 : varchar(20)
id_prodidipilihan2 : varchar(20)
id_prodiditerima : varchar(20)
id_gelombang : int(10)
jumlah_mhs: int(10)
jumlah_mhs_diterima: int(10)

**Tabel 3 Tabel fakta penerimaan mahasiswa baru**

- **Tabel dimensi asal sekolah**

Data yang termasuk kedalam dimensi asal sekolah adalah data asal sekolah calon mahasiswa tersebut antara lain nama sekolah, alamat dan lain sebagainya.

<b>pmbasalsekolah_dim</b>
id_asalsekolah : varchar(20)
nama_sekolah : varchar(20)
alamat_sekolah : varchar(30)
no_telepon_sekolah : varchar(20)
kab/kota_sekolah : varchar(20)

**Tabel 4 Tabel dimensi asal sekolah**

- **Tabel dimensi calon mahasiswa**

Data yang termasuk dalam dimensi mahasiswa adalah data pribadi calon mahasiswa.

<b>pmbcalonmhs_dim</b>
id_calonmhs : varchar(20)
nama : varchar(30)
jenis_kelamin : enum ('laki-laki', 'perempuan')
tempat_lahir : varchar(30)
tanggal_lahir : date
agama : varchar(11)
kewarganegaraan : varchar(11)
alamat : varchar(30)
kode_pos : int(11)
telepon : varchar(30)
kota_asal : varchar(30)
jenis_kelas : enum('reguler', 'karyawan')
nama_prodi : varchar(20)
nilai_UAN : int(11)
nilai_STK : int(11)
nama_perusahaan : varchar(100)
posisi/jabatan : varchar(30)
tahun_kerja : int(4)
nama_orang_tua : varchar(30)
hubungan_orang_tua : varchar(30)
alamat_orang_tua : varchar(30)
telepon_orang_tua : varchar(30)
informasi_pendaftaran : enum('iklan', 'TV', 'koran', 'radio', 'sosialisasi')

**Tabel 5 Tabel dimensi calon mahasiswa**

- **Tabel dimensi gelombang**

Data yang termasuk kedalam dimensi gelombang adalah tahun akademik, semester (ganjil dan genap).

<b>pmbgelombang_dim</b>
id_gelombang : int(10)
tahun_ajaran : int(11)
semester : enum('genap', 'ganjil')

**Tabel 6 Tabel dimensi gelombang**

- **Tabel dimensi program studi**

Data yang termasuk dalam dimensi prodi adalah program studi yang ada di Politeknik Negeri Batam antara lain Teknik Informatika, Teknik Elektro dan Akutansi.

<b>pmbprodi_dim</b>
id_prodi : varchar(20)
nama_prodidipilihan1 : varchar(20)
nama _prodipilihan2 : varchar(20)

**Tabel 7 Tabel dimensi program studi**

- **Tabel dimensi jalur penerimaan mahasiswa baru**

Data yang termasuk dalam jalur penerimaan mahasiswa baru adalah jalur masuk yang ada di Politeknik Negeri Batam antara lain jalur masuk ujian lokal, jalur masuk ujina nasional dan jalur masuk PMDK.

<b>pmbjalurpenerimaan_dim</b>
id_jalurpenerimaan : varchar(10)
nama_jalur : int(11)

**Tabel 8 Tabel dimensi jalur penerimaan mahasiswa baru**

### **III.3 Kapabilitas Data warehouse**

Kemampuan penyediaan informasi dari *data warehouse* ini digunakan untuk melakukan analisa terhadap kegiatan penerimaan mahasiswa baru yang berlangsung di Politeknik Negeri Batam. Sehingga direktur Politeknik Batam dapat menentukan langkah apa saja yang perlu di ambil untuk meningkatkan penerimaan mahasiswa baru yang ada di Politeknik Negeri Batam.

#### **Penyediaan Informasi**

*Data warehouse* Politeknik Negeri Batam dibangun agar dapat menyediakan informasi yang cepat, tepat dan akurat sehingga pimpinan kampus dapat mengambil keputusan yang tepat dalam menjalankan proses kegiatan penerimaan mahasiswa baru. Karena menggunakan aplikasi berbasis *web* maka informasi yang tersedia ini dapat diakses dari komputer manapun yang mempunyai koneksi internet. Selain itu kemampuan lain yang cukup bermanfaat dari adanya *data warehouse* adalah:

- Kemampuan membuat query sendiri sesuai dengan kebutuhan.
- Kemampuan *report customization* sesuai dengan kebutuhan informasi.
- Kemampuan untuk membuat *chart* atau grafik sesuai dengan laporan yang diinginkan.

## **Bab IV Implementasi dan Pengujian**

Pada bab ini akan dipaparkan mengenai implementasi dan pengujian *data warehouse* yang dibangun. Hasil dari perancangan pada tahap sebelumnya diimplementasikan dalam bahasa pemrograman PHP dan query SQL. Setelah tahap implementasi selesai, dilanjutkan dengan tahap pengujian terhadap *data warehouse*.

### **IV.1 Implementasi**

Setelah sistem dianalisis dan rancangan *data warehouse* didesain, maka selanjutnya menuju tahap implementasi. Tahapan implementasi dilakukan untuk mengkonfirmasi rancangan dan pembangunan *data warehouse* pada pelaku sistem sehingga pengguna dapat memberi masukan dan saran.

#### **IV.1.1 Lingkungan Pengembangan**

##### **IV.1.1.1 Perangkat Lunak Pembangun**

Perangkat lunak yang digunakan untuk membangun *data warehouse* penerimaan mahasiswa baru yaitu:

1. Sistem Operasi Windows 7.
2. Bahasa Pemrograman PHP dan sebagai perangkat pembangunan *data warehouse (tools)*.
3. MySQL Versi 3.4 sebagai Database Manajemen Sistem (DBMS).

##### **IV.1.1.2 Perangkat Keras Pembangun**

Spesifikasi perangkat keras yang digunakan dalam membangun *data warehouse* penerimaan mahasiswa baru yaitu:

1. Processor Intel Dual Core 2,2 Ghz
2. RAM 2 GB
3. Harddisk dengan ruang kosong 2 GB
4. Monitor dengan resolusi 1280 x 800 pixels

#### IV.1.2 9 Tahap Metodologi

Berdasarkan Kimball perancangan *data warehouse* yang baik dilakukan dengan menggunakan 9 tahap metodologi. Tahapan-tahapan yang dilakukan akan diuraikan sebagai berikut:

##### IV.1.2.1 Memilih Proses Bisnis

Proses ini meliputi proses yang terlibat dalam sistem *data warehouse*. Dalam kasus ini adalah proses penerimaan mahasiswa baru.

Proses	Deskripsi	Fungsi yang terlibat
Penerimaan mahasiswa baru	Mengenai data-data yang berhubungan dengan mahasiswa pada saat melakukan pendaftaran dan mahasiswa yang mengikuti ujian dan telah dinyatakan diterima di Politeknik Negeri Batam.	Panitia penerimaan mahasiswa baru dan Pihak manajemen kampus.

Tabel 9 Tabel pemilihan proses

##### IV.1.2.2 Memilih Grain

Setelah proses dipilih, maka langkah berikutnya adalah menentukan *grain* dari proses tersebut yang nantinya akan dijadikan sebagai tabel fakta. Yang menjadi *Grain* dalam kasus ini adalah nilai penerimaan mahasiswa baru.

Grain / Fakta	Deskripsi	Proses yang terkait
Penerimaan mahasiswa baru	Nilai penerimaan mahasiswa baru yang dimaksudkan adalah jumlah mahasiswa mendaftar dan jumlah mahasiswa diterima per asal sekolah, per gelombang, per prodi dan per kabupaten.	Panitia penerimaan mahasiswa baru dan Pihak manajemen kampus.

Tabel 10 Tabel pemilihan grain

#### IV.1.2.3 Mendefinisikan dan Menyesuaikan Dimensi

Setelah pemilihan *grain* maka yang perlu dilakukan selanjutnya adalah mengidentifikasi dimensi yang terkait dengan fakta. Dimensi yang ada pada proses ini adalah dimensi calon mahasiswa, dimensi asal sekolah, dimensi jalur penerimaan, dimensi prodi pilihan 1, dimensi prodi pilihan 2, dimensi prodi diterima dan dimensi gelombang.

Dimensi	Deskripsi	Grain
Calon mahasiswa	Keterangan identitas calon mahasiswa yang mendaftar.	Penerimaan mahasiswa baru
Asal sekolah	Keterangan asal sekolah calon mahasiswa	Penerimaan mahasiswa baru
Jalur penerimaan	Keterangan tentang jenis jalur penerimaan mahasiswa	Penerimaan mahasiswa baru
Prodi pilihan 1	Program studi pilihan pertama	Penerimaan mahasiswa baru
Prodi pilihan 2	Program studi pilihan kedua	Penerimaan mahasiswa baru
Prodi diterima	Program studi yang dinyatakan lulus	Penerimaan mahasiswa baru
Gelombang	Tahun calon mahasiswa mendaftar	Penerimaan mahasiswa baru

Tabel 11 Tabel penentuan dimensi

#### IV.1.2.4 Memilih Fakta

Apabila sudah menentukan dimensi yang terkait. Langkah selanjutnya adalah memastikan tabel fakta yang dapat diperoleh dalam proses *grain*. Fakta pada perusahaan ini antara lain adalah fakta penerimaan mahasiswa baru.

Fakta	Deskripsi	Dimensi
Penerimaan mahasiswa baru	Fakta penerimaan mahasiswa baru meliputi identitas calon mahasiswa, asal sekolah, jalur penerimaan, prodi pilihan dan gelombang masuk calon mahasiswa.	Dimensi calon mahasiswa, dimensi asal sekolah, dimensi jalur penerimaan, dimensi prodi pilihan 1, dimensi prodi pilihan 2, dimensi prodi diterima dan dimensi gelombang.

Tabel 12 Tabel penentuan fakta

#### IV.1.2.5 Menyimpan Pre Kalkulasi dalam Tabel

Untuk fakta Penerimaan mahasiswa baru, *agregasi* -nya adalah:

1. Jumlah mahasiswa adalah total jumlah mahasiswa yang mendaftar berdasarkan jalur penerimaan.
2. Jumlah mahasiswa diterima adalah total jumlah mahasiswa yang mendaftar berdasarkan jalur penerimaan dan telah dinyatakan diterima.

#### IV.1.2.6 Melengkapi Tabel Dimensi

Tabel dimensi yang digunakan adalah:

1. calonmhs\_dim

Atribut	Tipe Data	Panjang
Id_calonmhs	Varchar	20
Nama	Varchar	30
Jenis_kelamin	Enum	Laki-laki, perempuan
Tempat_lahir	Varchar	30
Tanggal_lahir	Date	
Agama	Varchar	11
Kewarganegaraan	Varchar	11
Alamat	Varchar	30
Kode_pos	Int	11
Telepon	Varchar	30
Kota_asal	Varchar	30
Jenis_kelas	Enum	Reguler, karyawan
Nilai_ujian_mtk	Int	11
Nilai_ujian_bindo	Int	11
Nilai_ujian_bing	Int	11
Nama_prodi	Varchar	20
Status	Enum	Diterima, tidak diterima
Tahun_mulai_sekolah	Varchar	4
Tahun_tamat_sekolah	Varchar	4
Jurusan_sekolah	Varchar	30
Nilai_UAN	Int	11
Nilai_STK	Int	11
Nama_perusahaan	Varchar	100
Posisi/jabatan	Varchar	30
Tahun_kerja	Int	4
Nama_orang_tua	Varchar	30
Hubungan_orang_tua	Varchar	30
Alamat_orang_tua	Varchar	30

Telepon_orang_tua	Varchar	30
Informasi_pendaftaran	Enum	Iklan, TV, koran, radio, sosialisasi

**Tabel 13 Tabel calonmhs\_dim**

2. pmbasalsekolah\_dim

Atribut	Tipe Data	Panjang
Id_asalsekolah	Varchar	20
Nama_sekolah	Varchar	20
Alamat_sekolah	Varchar	30
No_telepon_sekolah	Varchar	15
Kab/kota_sekolah	Varchar	30

**Tabel 14 Tabel pmbasalsekolah\_dim**

3. pmbjalurpenerimaan\_dim

Atribut	Tipe Data	Panjang
Id_jalurpenerimaan	Varchar	10
Nama_jalur	Int	11

**Tabel 15 Tabel pmbjalurpenerimaan\_dim**

4. pmbprodi\_dim

Atribut	Tipe Data	Panjang
Id_prodi	Varchar	20
Nama_prodi pilihan1	Varchar	20
Nama_prodi pilihan2	Varchar	20

**Tabel 16 Tabel pmbprodi\_dim**

5. pmbgelombang\_dim

Atribut	Tipe Data	Panjang
Id_gelombang	Int	10
Tahun_ajaran	Int	11
Semester	Enum	Genap, ganjil

**Tabel 17 Tabel pmbgelombang\_dim**

#### **IV.1.2.7 Memilih Durasi Waktu**

*Data warehouse* yang akan dibuat memiliki durasi selama dua tahun dari tahun 2007 sampai dengan 2008. Dimana database yang digunakan merupakan database yang diambil dari data OLTP dan dipindahkan ke dalam OLAP melalui proses transformasi data. Yang hasilnya berupa tabel fakta dan tabel dimensi untuk menjawab kebutuhan pihak manajemen kampus.

#### **IV.1.2.8 Melacak Perubahan Dimensi Secara Perlahan**

*Slowly changing dimension* (SCD) terbagi menjadi 2 jenis, yaitu SCD tipe 1 dan SCD tipe 2. Pada SCD tipe 1 apabila ada data yang berubah, maka data yang sebelumnya juga akan dirubah sesuai dengan yang baru sedangkan pada SCD tipe 2 data yang lama tetap sama tetapi diberikan batasan waktu *expired date*.

#### **IV.1.2.9 Menentukan Prioritas dan Model Query**

Pada tahap ini yang harus dilakukan adalah membuat rancangan fisik untuk menampilkan hasil dari kebutuhan pihak kampus. Yang paling penting dalam rancangan fisik adalah urutan dari perintah fisik yang akan ditampilkan berdasarkan table fakta dan table dimensi.

### IV.1.3 Implementasi Basis Data

Pembuatan *database* dilakukan dengan menggunakan *tools Microsoft My SQL*.

Implementasi *database* dalam bahasa SQL adalah sebagai berikut :

#### IV.1.3.1 Tabel Calon Mahasiswa

```
CREATE TABLE IF NOT EXISTS `calonmhs_con` (  
  `no_ujian` varchar(11) NOT NULL,  
  `nama` varchar(30) NOT NULL,  
  `jk` enum('Laki-laki','Perempuan') NOT NULL,  
  `tmpt_lahir` varchar(30) NOT NULL,  
  `tg_lahir` date NOT NULL,  
  `agama` varchar(11) NOT NULL,  
  `KWN` varchar(11) NOT NULL,  
  `alamat` varchar(30) NOT NULL,  
  `kode_pos` int(11) NOT NULL,  
  `telp` varchar(15) NOT NULL,  
  `kota_asal` varchar(20) NOT NULL,  
  `jn_kelas` enum('Reguler','Karyawan') NOT NULL,  
  `nilai_ujian_mtk` int(11) NOT NULL,  
  `nilai_ujian_bindo` int(11) NOT NULL,  
  `nilai_ujian_bing` int(11) NOT NULL,  
  `id_pilihan1` enum('1','2','3') NOT NULL,  
  `status` enum('Diterima','Tidakditerima') NOT NULL,  
  `thn_mulai_sekolah` varchar(4) NOT NULL,  
  `thn_tamat_sekolah` varchar(4) NOT NULL,  
  `jurusan_sekolah` varchar(20) NOT NULL,  
  `nilai_UAN` int(11) NOT NULL,  
  `nilai_stk` int(11) NOT NULL,  
  `nama_perusahaan` varchar(100) NOT NULL,  
  `posisi/jabatan` varchar(30) NOT NULL,  
  `tahun_kerja` int(4) NOT NULL,  
  `nama_ortu` varchar(30) NOT NULL,  
  `hubungan_ortu` varchar(30) NOT NULL,  
  `alamat_ortu` varchar(100) NOT NULL,  
  `telp_ortu` varchar(15) NOT NULL,  
  `informasi_pendaftaran`  
enum('Iklan','TV','Koran','Radio','SosialisasiKesekolah','Lain-  
lain') NOT NULL,  
  `id_jalurpenerimaan` int(11) NOT NULL,  
  `id_gelombang` int(11) NOT NULL,  
  `id_asal_sekolah` int(11) NOT NULL,  
  PRIMARY KEY (`no_ujian`)
```

**Tabel 18** Tabel calon mahasiswa

#### IV.1.3.2 Tabel Gelombang

```
CREATE TABLE IF NOT EXISTS `gelombang_dim` (  
  `id_gelombang` int(11) NOT NULL,  
  `tahun_ajaran` int(11) NOT NULL,  
  `semester` enum('genap','ganjil') NOT NULL,  
  PRIMARY KEY (`id_gelombang`)
```

**Tabel 19 Tabel gelombang**

#### IV.1.3.3 Tabel Penerimaan mahasiswa baru

```
CREATE TABLE IF NOT EXISTS `pmb_fact` (  
  `id_calon` varchar(30) NOT NULL,  
  `id_pilihan1` varchar(30) NOT NULL,  
  `id_pilihan2` varchar(30) NOT NULL,  
  `id_asal_sekolah` varchar(30) NOT NULL,  
  `id_gelombang` varchar(30) NOT NULL,  
  `status` enum('Lulus','Tidak') NOT NULL,  
  `id_jalurpenerimaan` int(11) NOT NULL,  
  `jumlah_mhs` int(11) NOT NULL,  
  PRIMARY KEY (`id_fakta`),
```

**Tabel 20 Tabel penerimaan mahasiswa baru**

#### IV.1.3.4 Tabel Program Studi

```
CREATE TABLE IF NOT EXISTS `prodi_dim` (  
  `id_prodi` varchar(20) NOT NULL,  
  `nama_prodi_pilihan1` varchar(20) NOT NULL,  
  `nama_prodi_pilihan2` varchar(20) NOT NULL,  
  PRIMARY KEY (`id_prodi`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

**Tabel 21 Tabel program studi**

#### IV.1.3.5 Tabel Asal Sekolah

```
CREATE TABLE IF NOT EXISTS `tasalsekolah` (  
  `id_sekolah` varchar(20) NOT NULL,  
  `nama_sekolah` varchar(20) NOT NULL,  
  `alamat_sekolah` varchar(30) NOT NULL,  
  `no_telepon_sekolah` varchar(15) NOT NULL,  
  `kab/kota_sekolah` varchar(30) NOT NULL,  
  PRIMARY KEY (`id_sekolah`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

**Tabel 22 Tabel asal sekolah**

#### IV.1.3.6 Jalur penerimaan

```
CREATE TABLE IF NOT EXISTS `tjalurpenerimaan` (  
  `id_jalurpenerimaan` varchar(10) NOT NULL,  
  `nama_jalur` varchar(10) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

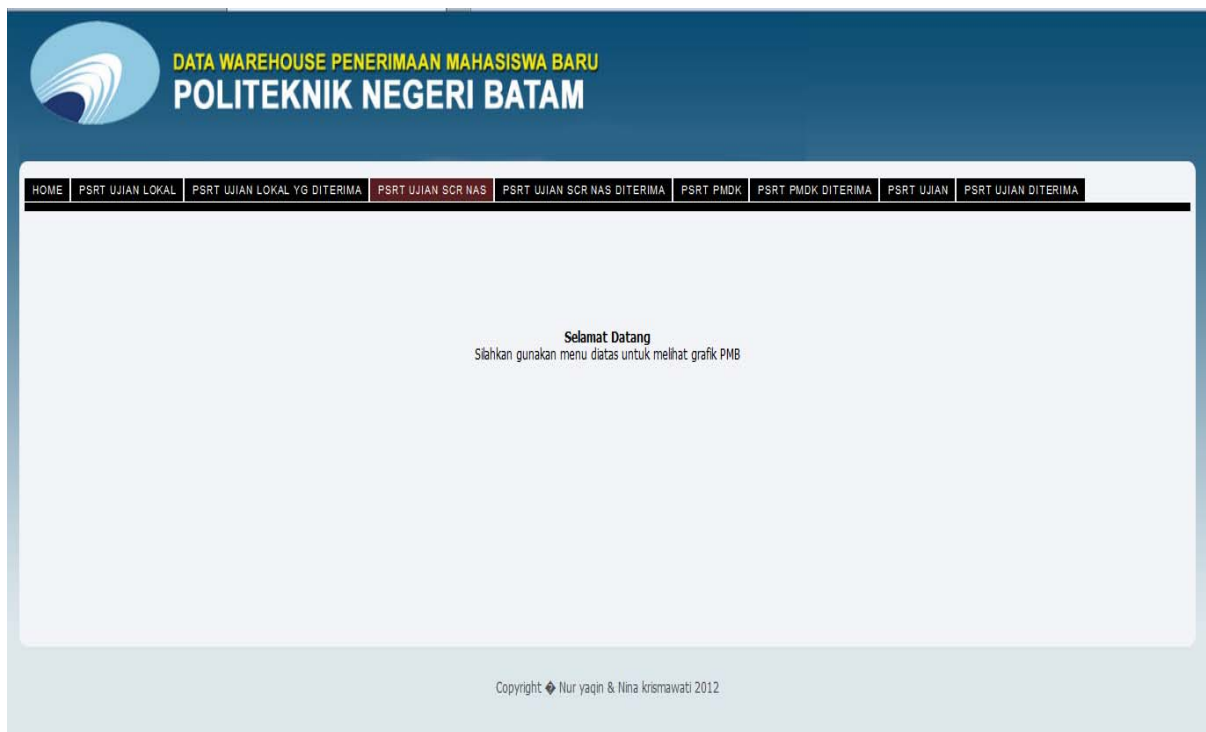
Tabel 23 Tabel jalur penerimaan

#### IV.1.4 Implementasi Antar Muka

Hasil dari perancangan antar muka yang dilakukan pada tahap perancangan diimplementasikan pada antarmuka sebagai berikut:

##### IV.1.4.1 Halaman Utama

Halaman utama (sebelum akses) akan tampil ketika user memasukkan alamat/ nomor IP lokal *data warehouse* penerimaan mahasiswa baru. Selanjutnya akan tampil layar sebagai berikut.



Gambar 4 Halaman menu utama (sebelum akses)

#### IV.1.4.2 Halaman Utama (setelah akses)

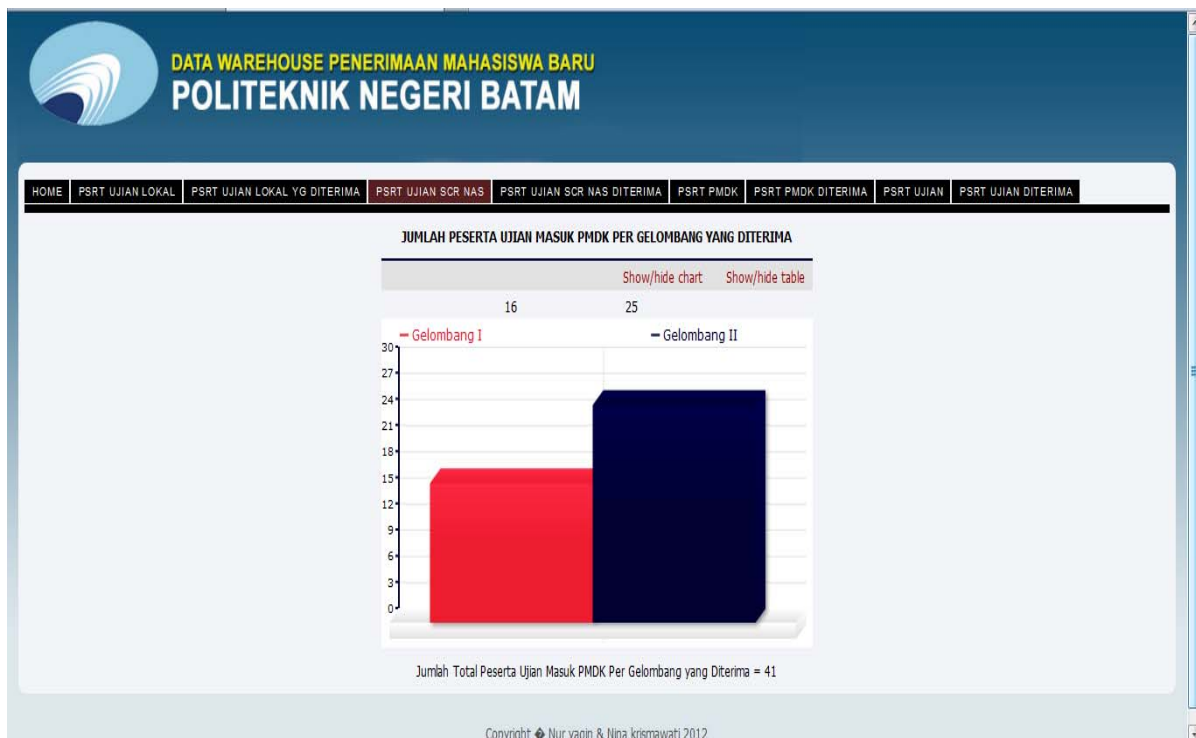
Tampilan halaman menu utama (setelah akses) akan tampil ketika user sudah memasukkan alamat lokal di browser maka akan dapat dilihat gambar yang memperlihatkan menu-menu sebagai berikut :



Gambar 5 Halaman menu utama (setelah akses)

#### IV.1.4.3 Halaman ETL (Extract Transform Load)

Halaman ETL akan tampil ketika user mengklik menu ETL pada halaman utama. Halaman ETL digunakan untuk mengekstrak data dari database operasional ke dalam database *data warehouse*. Untuk ekstrak data dari database bisa dilakukan di tiap-tiap menu dan langsung menampilkan grafik sesuai dengan informasi yang diharapkan, yang mana tampilannya sebagai berikut:



**Gambar 6 Hasil ekstraksi data menampilkan jumlah peserta ujian PMDK**

Hasil ekstrak data peserta ujian masuk PMDK per gelombang yang diterima dari database MySQL. Maka dapat dilihat bahwa jumlah siswa yang masuk untuk gelombang 1 sebanyak 16 orang dan gelombang dua sebanyak 25 orang. Tampilan ini memperlihatkan jumlah keseluruhan siswa.

#### IV.1.4.4 Analisis data penerimaan mahasiswa baru

Halaman analisis data penerimaan mahasiswa baru akan tampil ketika user mengklik beberapa menu sesuai dengan informasi yang dibutuhkan. Data grafik dapat dilihat secara umum dan dapat dilihat secara detail dengan cara mengklik tanda show/hide chart, tampilan untuk jumlah peserta ujian masuk per asal sekolah sebagai berikut:



Gambar 7 Halaman analisis data penerimaan mahasiswa baru

#### IV.1.5 Pengujian Sistem

Agar hasil implementasi benar-benar memenuhi kebutuhan maka dilakukan pengujian terhadap *data warehouse* yang dibangun. Teknik pengujian yang dilakukan yaitu pengujian *Alpha* dan *Beta*. Pengujian *alpha* dilakukan dengan metode *black box* yang berfokus pada persyaratan fungsional perangkat lunak sedangkan pengujian *beta* dilakukan dengan melakukan wawancara.

#### IV.1.5.1 Pengujian Alfa

Pengujian *data warehouse* yang dibangun menggunakan metode pengujian *black box*. Adapun rencana pengujian selengkapnya dapat dilihat pada tabel berikut ini:

Item Uji	Detail Pengujian	Jenis Uji
Akses	Menampilkan menu utama	Black box
<i>Extract, Transform, Load (ETL)</i>	Proses <i>Extract, Transform, Load (ETL)</i> dari data base	<i>Black box</i>
Hasil <i>Extract, Transform, Load (ETL)</i>	Proses hasil <i>Extract, Transform, Load (ETL)</i>	<i>Black box</i>

Tabel 24 Tabel pengujian alfa

#### IV.1.5.2 Pengujian Beta

Pada pengembangan *data warehouse* penerimaan mahasiswa baru, penulis menguji sistem dengan menggunakan teknik pengujian beta dengan melibatkan beberapa teman penulis, panitia penerimaan mahasiswa baru dan pihak manajemen kampus selaku pengambil keputusan.

#### IV.1.6 Kasus dan Hasil Pegujian

Berdasarkan rencana pengujian yang telah dibuat, maka dilakukan pengujian seperti berikut:

##### IV.1.6.1 Pengujian akses data warehouse

<b>Kasus dan Hasil Uji (Data Normal)</b>			
<b>Data</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Kesimpulan</b>
Memasukkan Alamat Lokal	Tampil menu utama	Dapat melihat beberapa menu-menu yang tersedia sehingga user bisa memilih	Diterima : Tidak terjadi error
Memilih menu	Menampilkan grafik	Dapat melihat grafik Data Penerimaan Mahasiswa berdasarkan Gelombang masuk, daerah asal, jumlah peserta ujian dan jumlah mahasiswa berdasarkan prodi	Diterima ; Tidak terjadi Error
<b>Kasus dan Hasil Uji (Data Salah)</b>			
<b>Data</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Kesimpulan</b>
Salah alamat local	Tidak bisa masuk ke menu pmb, dan menghasilkan alamat error	Tidak bisa melihat grafik data tentang data warehouse	Diterima
Salah alamat local	Menampilkan grafik	Dapat melihat grafik Data Penerimaan Mahasiswa berdasarkan Gelombang masuk, daerah asal, jumlah peserta ujian dan jumlah mahasiswa berdasarkan prodi	Diterima ; Tidak terjadi Error

Tabel 25 Tabel pengujian akses data warehouse

#### IV.1.6.2 Pengujian akses Extract, Transform & Load

Kasus dan Hasil Uji (Data Normal)			
Data	Yang Diharapkan	Pengamatan	Kesimpulan
Memilih Tabel Dimensi atau Tabel Fakta	Pilihan Tabel dimensi atau tabel fakta yang tercantum pada Menu Utama dan menampilkan data hasil ETL pada Grafik Batang	Dapat mengisikan/memilih Tabel Dimensi atau Tabel Fakta pada Menu Utama Sesuai yang diharapkan.	Diterima
Kasus dan Hasil Uji (Data Salah)			
Data	Yang Diharapkan	Pengamatan	Kesimpulan
Tidak memilih Tabel dimensi atau tabel fakta pada Menu Utama	Tidak menampilkan data hasil ETL pada Grafik batang	Tidak menampilkan data hasil ETL. Sesuai yang diharapkan.	Diterima

Tabel 26 Tabel pengujian akses Extract, Taransform & Load

Melihat kesimpulan akhir dari pengamatan selama pengujian yang dilakukan terhadap perangkat lunak yang dibangun dengan metode pengujian *black box* adalah sesuai dengan yang diharapkan, maka disimpulkan *Data Warehouse* Penerimaan Mahasiswa Baru Pada Politeknik Negeri Batam yang dibangun diterima.

## **Bab V Kesimpulan dan Saran**

### **V.1 Kesimpulan**

Kesimpulan yang dapat diambil dari implementasi sebuah *data warehouse* untuk menunjang kegiatan penerimaan mahasiswa baru di Politeknik Negeri Batam adalah sebagai berikut:

1. Penelitian ini menghasilkan *data warehouse* penerimaan mahasiswa baru menggunakan perancangan skema bintang. *Data warehouse* tersebut berisi nilai-nilai agregasi calon mahasiswa yang mendaftar dan telah dinyatakan lulus melalui jalur ujian nasional, ujian local dan jalur PMDK.
2. Dengan adanya *data warehouse*, proses penyusunan laporan di Politeknik Negeri Batam menjadi lebih sederhana, karena pengguna bisa melakukan *customization report* sesuai dengan yang diinginkan, sehingga ada efisiensi waktu yang dari sebelumnya memerlukan waktu satu bulan untuk membuat program baru atau satu minggu untuk mengerjakan secara manual menjadi hanya satu hari
3. Untuk *data warehouse* Politeknik Negeri Batam, media presentasi yang digunakan adalah aplikasi berbasis *web* sehingga pemenuhan kebutuhan laporan menjadi lebih fleksibel untuk para pimpinan Politeknik Negeri Batam karena dapat diakses melalui internet.

### **V.2 Saran**

Diharapkan:

1. Pembangunan *data warehouse* tidak hanya menangani penerimaan mahasiswa baru tetapi juga dapat menangani seluruh kegiatan akademik yang ada di Politeknik Negeri Batam.
2. Penelitian ini dapat dikembangkan lebih lanjut untuk mengarah ke pemanfaatan *data warehouse* untuk data mining. Dengan telah terbentuknya

informasi yang terintegrasi dalam *data warehouse*, pola-pola didalamnya dapat digali untuk informasi yang berharga untuk pengambilan keputusan.

3. Dikembangkan dengan menambahkan fasilitas login. Dengan fasilitas tersebut hak akses pengguna diatur sehingga aplikasi hanya dapat diakses oleh pihak-pihak yang berkepentingan dan kerahasiaan data terjaga.

## Daftar Pustaka

- [1] Subhan, 2009, *Pengenalan Data warehouse*, <http://subhan.blog.binusian.org> , diakses tanggal 2 Maret 2011.
- [2] Connolly T & Begg C, 2002, *Database Systems: A Practical Approach to Design, Implementation, and Management*. USA: Addison Wesley
- [3] Suparto Darudiato, 2010, *Perancangan Data warehouse Penjualan untuk Mendukung Kebutuhan Informasi Eksekutif Cemerlang Skin Care*, Seminar Nasional Informatika, diakses tanggal 20 Maret 2011.
- [4] [http://en.wikipedia.org/wiki/data\\_warehouse](http://en.wikipedia.org/wiki/data_warehouse), diakses tanggal 8 Juni 2012
- [5] [http://download.oracle.com/docs/cd/B28359\\_01/server.111/b28313/concept.htm](http://download.oracle.com/docs/cd/B28359_01/server.111/b28313/concept.htm)  
diakses tanggal 1 Juli 2012
- [6] <http://www.cs.bris.ac.uk/maintain/OracleDocs/server.816/a76994/concept.htm>  
diakses tanggal 1 Juni 2012
- [7] [http://www.datawarehouse4u.info/Data\\_warehouse.htm](http://www.datawarehouse4u.info/Data_warehouse.htm), diakses tanggal 8 Juni 2012
- [8] Silberschatz, Database System Concepts, Fourth Edition

## Lampiran A Transformasi Data

### 1. Jumlah Peserta Ujian Lokal

```
<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Model_jml_peserta_msk_lokal extends CI_Model
{
    function gelombang1()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_gelombang',1);
        $this->db->where('id_jalurpenerimaan',1);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function gelombang2()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_gelombang',2);
        $this->db->where('id_jalurpenerimaan',1);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma1_sekupang()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',1);
        $this->db->where('id_jalurpenerimaan',1);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma3_nongsa()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',2);
        $this->db->where('id_jalurpenerimaan',1);
```

```

        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma4_tiban()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',3);
        $this->db->where('id_jalurpenerimaan',1);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma1_pamedan()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',4);
        $this->db->where('id_jalurpenerimaan',1);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma1_kartini()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',5);
        $this->db->where('id_jalurpenerimaan',1);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma8()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',6);
        $this->db->where('id_jalurpenerimaan',1);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_smk1()

```

```

{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('id_asal_sekolah',7);
    $this->db->where('id_jalurpenerimaan',1);
    $query = $this->db->get();
    $query->num_rows();
    return $query->num_rows();
}

function prodi_informatika()
{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('id_pilihan1',1);
    $this->db->where('id_jalurpenerimaan',1);
    $query = $this->db->get();
    $query->num_rows();
    return $query->num_rows();
}

function prodi_elektro()
{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('id_pilihan1',2);
    $this->db->where('id_jalurpenerimaan',1);
    $query = $this->db->get();
    $query->num_rows();
    return $query->num_rows();
}

function prodi_akutansi()
{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('id_pilihan1',3);
    $this->db->where('id_jalurpenerimaan',1);
    $query = $this->db->get();
    $query->num_rows();
    return $query->num_rows();
}

function kab_batam()
{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('kota_asal','Batam');
    $this->db->where('id_jalurpenerimaan',1);

```

```

        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function kab_tanjung_pinang()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('kota_asal','Tanjung Pinang');
        $this->db->where('id_jalurpenerimaan',1);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function gelombang1_pergram_studi()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_gelombang',1);
        $this->db->where('id_jalurpenerimaan',1);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function gelombang2_pergram_studi()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_gelombang',1);
        $this->db->where('id_jalurpenerimaan',2);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
}

```

## 2. Jumlah Peserta Ujian Lokal Diterima

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Model_jml_peserta_msk_lokal_diterima extends CI_Model
{
    function gelombang1()

```

```

{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('id_gelombang',1);
    $this->db->where('status','Diterima');
    $this->db->where('id_jalurpenerimaan',1);
    $query = $this->db->get();
    $query->num_rows();
    return $query->num_rows();
}

function gelombang2()
{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('id_gelombang',2);
    $this->db->where('status','Diterima');
    $this->db->where('id_jalurpenerimaan',1);
    $query = $this->db->get();
    $query->num_rows();
    return $query->num_rows();
}

function asal_sekolah_sma1_sekupang()
{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('id_asal_sekolah',1);
    $this->db->where('status','Diterima');
    $this->db->where('id_jalurpenerimaan',1);
    $query = $this->db->get();
    $query->num_rows();
    return $query->num_rows();
}

function asal_sekolah_sma3_nongsa()
{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('id_asal_sekolah',2);
    $this->db->where('status','Diterima');
    $this->db->where('id_jalurpenerimaan',1);
    $query = $this->db->get();
    $query->num_rows();
}

```

```

        return $query->num_rows();
    }
    function asal_sekolah_sma4_tiban()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',3);
        $this->db->where('status','Diterima');
        $this->db->where('id_jalurpenerimaan',1);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma1_pamedan()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',4);
        $this->db->where('status','Diterima');
        $this->db->where('id_jalurpenerimaan',1);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma1_kartini()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',5);
        $this->db->where('status','Diterima');
        $this->db->where('id_jalurpenerimaan',1);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma8()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',6);
        $this->db->where('status','Diterima');
        $this->db->where('id_jalurpenerimaan',1);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }

```

```

}
function asal_sekolah_smk1()
{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('id_asal_sekolah',7);
    $this->db->where('status','Diterima');
    $this->db->where('id_jalurpenerimaan',1);
    $query = $this->db->get();
    $query->num_rows();
    return $query->num_rows();
}

function prodi_informatika()
{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('id_pilihan1',1);
    $this->db->where('status','Diterima');
    $this->db->where('id_jalurpenerimaan',1);
    $query = $this->db->get();
    $query->num_rows();
    return $query->num_rows();
}

function prodi_elektro()
{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('id_pilihan1',2);
    $this->db->where('status','Diterima');
    $this->db->where('id_jalurpenerimaan',1);
    $query = $this->db->get();
    $query->num_rows();
    return $query->num_rows();
}

function prodi_akutansi()
{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('id_pilihan1',3);
    $this->db->where('status','Diterima');
    $this->db->where('id_jalurpenerimaan',1);
    $query = $this->db->get();
    $query->num_rows();
    return $query->num_rows();
}
}

```

```

function kab_batam()
{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('kota_asal','Batam');
    $this->db->where('status','Diterima');
    $this->db->where('id_jalurpenerimaan',1);
    $query = $this->db->get();
    $query->num_rows();
    return $query->num_rows();
}

function kab_tanjung_pinang()
{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('kota_asal','Tanjung Pinang');
    $this->db->where('status','Diterima');
    $this->db->where('id_jalurpenerimaan',1);
    $query = $this->db->get();
    $query->num_rows();
    return $query->num_rows();
}
}

```

### 3. Jumlah Peserta Ujian Nasional

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');
class Model_jml_peserta_msk_nasional extends CI_Model
{
    function gelombang1()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_gelombang',1);
        $this->db->where('id_jalurpenerimaan',2);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }

    function gelombang2()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_gelombang',2);
    }
}

```

```

        $this->db->where('id_jalurpenerimaan',2);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma1_sekupang()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',1);
        $this->db->where('id_jalurpenerimaan',2);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma3_nongsa()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',2);
        $this->db->where('id_jalurpenerimaan',2);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma4_tiban()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',3);
        $this->db->where('id_jalurpenerimaan',2);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma1_pamedan()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',4);
        $this->db->where('id_jalurpenerimaan',2);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
}

```

```

function asal_sekolah_sma1_kartini()
{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('id_asal_sekolah',5);
    $this->db->where('id_jalurpenerimaan',2);
    $query = $this->db->get();
    $query->num_rows();
    return $query->num_rows();
}

function asal_sekolah_sma8()
{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('id_asal_sekolah',6);
    $this->db->where('id_jalurpenerimaan',2);
    $query = $this->db->get();
    $query->num_rows();
    return $query->num_rows();
}

function asal_sekolah_smk1()
{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('id_asal_sekolah',7);
    $this->db->where('id_jalurpenerimaan',2);
    $query = $this->db->get();
    $query->num_rows();
    return $query->num_rows();
}

function prodi_informatika()
{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('id_pilihan1',1);
    $this->db->where('id_jalurpenerimaan',2);
    $query = $this->db->get();
    $query->num_rows();
    return $query->num_rows();
}

function prodi_elektro()
{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('id_pilihan1',2);

```

```

        $this->db->where('id_jalurpenerimaan',2);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function prodi_akutansi()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_pilihan1',3);
        $this->db->where('id_jalurpenerimaan',2);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function kab_batam()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('kota_asal','Batam');
        $this->db->where('id_jalurpenerimaan',2);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function kab_tanjung_pinang()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('kota_asal','Tanjung Pinang');
        $this->db->where('id_jalurpenerimaan',2);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
}

```

#### 4. Jumlah Peserta Ujian Nasional Diterima

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Model_jml_peserta_msk_nasional_diterima extends CI_Model
{
    function gelombang1()

```

```

{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('id_gelombang',1);
    $this->db->where('status','Diterima');
    $this->db->where('id_jalurpenerimaan',2);
    $query = $this->db->get();
    $query->num_rows();
    return $query->num_rows();
}

function gelombang2()
{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('id_gelombang',2);
    $this->db->where('status','Diterima');
    $this->db->where('id_jalurpenerimaan',2);
    $query = $this->db->get();
    $query->num_rows();
    return $query->num_rows();
}

function asal_sekolah_sma1_sekupang()
{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('id_asal_sekolah',1);
    $this->db->where('status','Diterima');
    $this->db->where('id_jalurpenerimaan',2);
    $query = $this->db->get();
    $query->num_rows();
    return $query->num_rows();
}

function asal_sekolah_sma3_nongsa()
{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('id_asal_sekolah',2);
    $this->db->where('status','Diterima');
    $this->db->where('id_jalurpenerimaan',2);
    $query = $this->db->get();
    $query->num_rows();
    return $query->num_rows();
}

function asal_sekolah_sma4_tiban()
{

```

```

        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',3);
        $this->db->where('status','Diterima');
        $this->db->where('id_jalurpenerimaan',2);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma1_pamedan()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',4);
        $this->db->where('status','Diterima');
        $this->db->where('id_jalurpenerimaan',2);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma1_kartini()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',5);
        $this->db->where('status','Diterima');
        $this->db->where('id_jalurpenerimaan',2);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma8()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',6);
        $this->db->where('status','Diterima');
        $this->db->where('id_jalurpenerimaan',2);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_smk1()
    {
        $this->db->select('*');

```

```

        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',7);
        $this->db->where('status','Diterima');
        $this->db->where('id_jalurpenerimaan',2);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function prodi_informatika()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_pilihan1',1);
        $this->db->where('status','Diterima');
        $this->db->where('id_jalurpenerimaan',2);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function prodi_elektro()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_pilihan1',2);
        $this->db->where('status','Diterima');
        $this->db->where('id_jalurpenerimaan',2);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function prodi_akutansi()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_pilihan1',3);
        $this->db->where('status','Diterima');
        $this->db->where('id_jalurpenerimaan',2);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function kab_batam()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');

```

```

        $this->db->where('kota_asal','Batam');
        $this->db->where('status','Diterima');
        $this->db->where('id_jalurpenerimaan',2);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function kab_tanjung_pinang()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('kota_asal','Tanjung Pinang');
        $this->db->where('status','Diterima');
        $this->db->where('id_jalurpenerimaan',2);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
}

```

## 5. Jumlah Peserta PMDK

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Model_jml_peserta_msk_pmdk extends CI_Model
{
    function gelombang1()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_gelombang',1);
        $this->db->where('id_jalurpenerimaan',3);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function gelombang2()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_gelombang',2);
        $this->db->where('id_jalurpenerimaan',3);
        $query = $this->db->get();
        $query->num_rows();
    }
}

```

```

        return $query->num_rows();
    }
    function asal_sekolah_sma1_sekupang()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',1);
        $this->db->where('id_jalurpenerimaan',3);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma3_nongsa()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',2);
        $this->db->where('id_jalurpenerimaan',3);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma4_tiban()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',3);
        $this->db->where('id_jalurpenerimaan',3);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma1_pamedan()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',4);
        $this->db->where('id_jalurpenerimaan',3);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma1_kartini()
    {
        $this->db->select('*');

```

```

        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',5);
        $this->db->where('id_jalurpenerimaan',3);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma8()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',6);
        $this->db->where('id_jalurpenerimaan',3);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_smk1()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',7);
        $this->db->where('id_jalurpenerimaan',3);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function prodi_informatika()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_pilihan1',1);
        $this->db->where('id_jalurpenerimaan',3);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function prodi_elektro()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_pilihan1',2);
        $this->db->where('id_jalurpenerimaan',3);
        $query = $this->db->get();
        $query->num_rows();
    }

```

```

        return $query->num_rows();
    }
    function prodi_akutansi()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_pilihan1',3);
        $this->db->where('id_jalurpenerimaan',3);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function kab_batam()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('kota_asal','Batam');
        $this->db->where('id_jalurpenerimaan',3);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function kab_tanjung_pinang()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('kota_asal','Tanjung Pinang');
        $this->db->where('id_jalurpenerimaan',3);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
}

```

## 6. Jumlah Peserta PMDK Diterima

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Model_jml_peserta_msk_pmdk_diterima extends CI_Model
{
    function gelombang1()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
    }
}

```

```

        $this->db->where('id_gelombang',1);
        $this->db->where('status','Diterima');
        $this->db->where('id_jalurpenerimaan',3);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function gelombang2()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_gelombang',2);
        $this->db->where('status','Diterima');
        $this->db->where('id_jalurpenerimaan',3);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma1_sekupang()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',1);
        $this->db->where('status','Diterima');
        $this->db->where('id_jalurpenerimaan',3);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma3_nongsa()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',2);
        $this->db->where('status','Diterima');
        $this->db->where('id_jalurpenerimaan',3);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma4_tiban()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',3);

```

```

        $this->db->where('status','Diterima');
        $this->db->where('id_jalurpenerimaan',3);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma1_pamedan()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',4);
        $this->db->where('status','Diterima');
        $this->db->where('id_jalurpenerimaan',3);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma1_kartini()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',5);
        $this->db->where('status','Diterima');
        $this->db->where('id_jalurpenerimaan',3);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma8()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',6);
        $this->db->where('status','Diterima');
        $this->db->where('id_jalurpenerimaan',3);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_smk1()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',7);
        $this->db->where('status','Diterima');
    }

```

```

        $this->db->where('id_jalurpenerimaan',3);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function prodi_informatika()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_pilihan1',1);
        $this->db->where('status','Diterima');
        $this->db->where('id_jalurpenerimaan',3);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function prodi_elektro()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_pilihan1',2);
        $this->db->where('status','Diterima');
        $this->db->where('id_jalurpenerimaan',3);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function prodi_akutansi()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_pilihan1',3);
        $this->db->where('status','Diterima');
        $this->db->where('id_jalurpenerimaan',3);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function kab_batam()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('kota_asal','Batam');
        $this->db->where('status','Diterima');
        $this->db->where('id_jalurpenerimaan',3);
    }

```

```

        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function kab_tanjung_pinang()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('kota_asal','Tanjung Pinang');
        $this->db->where('status','Diterima');
        $this->db->where('id_jalurpenerimaan',3);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
}

```

## 7. Jumlah Peserta Ujian

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Model_jml_peserta_msk extends CI_Model
{
    function gelombang1()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_gelombang',1);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function gelombang2()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_gelombang',2);

        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma1_sekupang()
    {

```

```

        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',1);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma3_nongsa()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',2);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma4_tiban()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',3);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma1_pamedan()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',4);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma1_kartini()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',5);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma8()
    {

```

```

        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',6);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_smk1()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',7);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function prodi_informatika()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_pilihan1',1);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function prodi_elektro()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_pilihan1',2);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function prodi_akutansi()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_pilihan1',3);
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function kab_batam()
    {

```

```

        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('kota_asal','Batam');
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function kab_tanjung_pinang()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('kota_asal','Tanjung Pinang');
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
}

```

### 8. Jumlah Peserta Ujian Diterima

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Model_jml_peserta_msk_diterima extends CI_Model
{
    function gelombang1()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_gelombang',1);
        $this->db->where('status','Diterima');
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function gelombang2()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_gelombang',2);
        $this->db->where('status','Diterima');
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
}

```

```

function asal_sekolah_sma1_sekupang()
{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('id_asal_sekolah',1);
    $this->db->where('status','Diterima');
    $query = $this->db->get();
    $query->num_rows();
    return $query->num_rows();
}

function asal_sekolah_sma3_nongsa()
{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('id_asal_sekolah',2);
    $this->db->where('status','Diterima');
    $query = $this->db->get();
    $query->num_rows();
    return $query->num_rows();
}

function asal_sekolah_sma4_tiban()
{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('id_asal_sekolah',3);
    $this->db->where('status','Diterima');
    $query = $this->db->get();
    $query->num_rows();
    return $query->num_rows();
}

function asal_sekolah_sma1_pamedan()
{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('id_asal_sekolah',4);
    $this->db->where('status','Diterima');
    $query = $this->db->get();
    $query->num_rows();
    return $query->num_rows();
}

function asal_sekolah_sma1_kartini()
{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('id_asal_sekolah',5);

```

```

        $this->db->where('status','Diterima');
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_sma8()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',6);
        $this->db->where('status','Diterima');
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function asal_sekolah_smk1()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_asal_sekolah',7);
        $this->db->where('status','Diterima');
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function prodi_informatika()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_pilihan1',1);
        $this->db->where('status','Diterima');
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
    function prodi_elektro()
    {
        $this->db->select('*');
        $this->db->from('tcalonmhs_dim');
        $this->db->where('id_pilihan1',2);
        $this->db->where('status','Diterima');
        $query = $this->db->get();
        $query->num_rows();
        return $query->num_rows();
    }
}

```

```
function prodi_akutansi()
{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('id_pilihan1',3);
    $this->db->where('status','Diterima');
    $query = $this->db->get();
    $query->num_rows();
    return $query->num_rows();
}

function kab_batam()
{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('kota_asal','Batam');
    $this->db->where('status','Diterima');
    $query = $this->db->get();
    $query->num_rows();
    return $query->num_rows();
}

function kab_tanjung_pinang()
{
    $this->db->select('*');
    $this->db->from('tcalonmhs_dim');
    $this->db->where('kota_asal','Tanjung Pinang');
    $this->db->where('status','Diterima');
    $query = $this->db->get();
    $query->num_rows();
    return $query->num_rows();
}
}
```