

Chart Customization for Developing Decision Support System at Petra Christian University

Siget Wibisono¹, Rolly Intan², Yulia³

Program Studi Teknik Informatika, Fakultas Teknologi Industri, Universitas Kristen Petra

Jl. Siwalankerto 121 – 131 Surabaya 60236

Telp. (031) – 2983455, Fax. (031) – 8417658

E-mail: sigetwibisono@gmail.com¹, rintan@petra.ac.id², yulia@petra.ac.id³

ABSTRAK

Data akademik calon mahasiswa dan mahasiswa Universitas Kristen Petra (UKP) yang telah disimpan selama ini belum digunakan untuk menghasilkan informasi. *Top-level management* UKP memerlukan cara untuk menghasilkan informasi dari data tersebut. Informasi yang dihasilkan diharapkan dapat mendukung proses pengambilan keputusan oleh *top-level management*.

Sebelum memulai pengembangan aplikasi, dilakukan analisa dan desain terhadap data akademik mahasiswa dan kebutuhan *top-level management*. Tahap desain menghasilkan sejumlah pemodelan yang nantinya digunakan untuk membuat aplikasi.

Hasil akhir dari pengembangan aplikasi ini adalah sebuah aplikasi *Decision Support System* Akademik UKP. Aplikasi yang dikembangkan mencakup fitur *Extract, Transform, and Load* (ETL), kustomisasi grafik, dan *maintain* pengguna.

Kata Kunci: Aplikasi, *Data Warehouse*, *Decision Support System*

ABSTRACT

Academic records of student candidates and students of Petra Christian University (PCU) which have been stored so far have not been used to generate information. PCU's Top-level management needs a way to generate information from the records. The generated information is expected to support the decision-making process of Top-level management.

Before starting the application development, analysis and design of the student academic records and the needs of Top-level management are done. The design stage produces a number of modeling that will be used to create the application.

The final result of the development is an Academic Decision Support System application for PCU. The developed application includes Extract, Transform, and Load (ETL), customizable graphs, and user maintenance features.

Keywords: *Data Warehouse*, *Decision Support System*, *Software*

1. PENDAHULUAN

Universitas Kristen Petra (UKP) mempunyai mahasiswa yang berasal dari banyak sekolah. Sekolah-sekolah tersebut tentu memiliki standar yang berbeda-beda, sehingga nilai tinggi yang diperoleh seorang mahasiswa pada masa sekolah tidak menjamin mahasiswa tersebut akan memperoleh indeks prestasi kumulatif

(IPK) yang tinggi pada saat berkuliah di UKP. Selama ini, UKP telah menyimpan data akademik calon mahasiswa yang mendaftar di UKP, berupa nilai matematika dan bahasa Inggris pada masa sekolah. Selain itu, setelah memasuki masa perkuliahan, UKP akan menyimpan IPK dari seluruh mahasiswa.

Data akademik calon mahasiswa dan mahasiswa UKP yang telah disimpan tersebut selama ini belum digunakan untuk menghasilkan informasi. *Top-level management* UKP merasakan perlu adanya cara untuk menghasilkan informasi dari data tersebut. Informasi yang dihasilkan diharapkan dapat mendukung proses pengambilan keputusan oleh *Top-level management*.

Dengan data calon mahasiswa dan data mahasiswa, UKP dapat memperoleh informasi yang dapat mendukung proses pengambilan keputusan oleh *Top-level management*. Informasi yang dikeluarkan dapat berupa grafik yang menampilkan informasi data mahasiswa secara statistik. Oleh karena itu, UKP memerlukan suatu perangkat lunak yang dapat mengeluarkan informasi yang dibutuhkan *Top-level management* berkaitan dengan data calon mahasiswa dan mahasiswa di UKP.

2. TINJAUAN PUSTAKA

2.1 Data Warehouse

Data Warehouse merupakan sekumpulan teknologi yang dapat membantu mengambil keputusan, diarahkan untuk membantu *knowledge worker* (*executive, manager, analyst*) untuk membuat keputusan yang lebih cepat dan tepat [4]. *Data Warehouse* adalah suatu konsep dan kombinasi teknologi yang memfasilitasi organisasi untuk mengelola dan memelihara data historis yang diperoleh dari sistem atau aplikasi operasional. *Data Warehouse* adalah suatu paradigma baru di lingkungan pengambilan keputusan strategis. *Data Warehouse* bukan suatu produk tetapi suatu lingkungan di mana pengguna dapat menemukan informasi strategis [3]. *Data Warehouse* merupakan tempat menyimpan informasi yang dikhususkan untuk membantu mengambil keputusan [1]. *Data Warehouse* berisi kumpulan data-data logis yang terpisah dengan *database* operasional dan merupakan suatu ringkasan. *Data Warehouse* memungkinkan integrasi berbagai macam jenis data dari berbagai macam aplikasi atau sistem. Hal ini menjamin mekanisme akses satu pintu bagi manajemen untuk memperoleh informasi dan menganalisisnya untuk pengambilan keputusan.

Data Warehouse memiliki beberapa karakteristik antara lain [3,4]:

1. Berorientasi subyek
2. Data yang terintegrasi

Sumber data yang ada dalam *Data Warehouse* tidak hanya berasal dari *database* operasional (*internal source*) tetapi juga berasal dari data di luar sistem (*external source*).

3. *Nonvolatile*

Data dalam *database* operasional akan secara berkala atau periodik dipindahkan ke dalam *Data Warehouse* sesuai dengan jadwal yang sudah ditentukan. Misal per hari, per minggu, per bulan, dan lain-lain.

4. *Time-Variant*

Data Warehouse mengandung data tidak hanya data terkini tetapi juga data *history* yang akan digunakan dalam analisis dan pengambilan keputusan. Waktu adalah dimensi penting yang harus didukung oleh semua *Data Warehouse*. Data untuk analisis dari berbagai sumber berisi berbagai nilai waktu, misalkan harian, mingguan, dan bulanan.

5. Tidak ternormalisasi

Data di dalam sebuah *Data Warehouse* biasanya tidak ternormalisasi dan sangat *redundant*.

Pada *Data Warehouse* digunakan teknik pemodelan data yang disebut *dimensional modelling technique*. Pemodelan *dimensional* adalah suatu model berbasis pemanggilan yang mendukung akses *query* tingkat tinggi. *Star Schema* merupakan salah satu bentuk skema pemodelan *dimensional* dan berisi sebuah *fact table* pada pusatnya. *Fact table* berisi atribut deskriptif yang digunakan untuk proses *query* dan *foreign key* untuk menghubungkan ke *dimension table*. Atribut analisis keputusan terdiri dari ukuran performa, metrik operasional, ukuran agregat, dan semua metrik yang lain yang diperlukan untuk menganalisis performa organisasi. *Fact table* menunjukkan apa yang didukung oleh *Data Warehouse* untuk analisis keputusan. *Dimension table* mengelilingi *fact table* pusat. *Dimension table* berisi atribut yang menguraikan data yang dimasukkan dalam *fact table*.

2.2 *Extract, Transform, and Load*

Extract, Transform, and Load (ETL) merupakan proses integrasi data yang melakukan ekstraksi data dari sumber luar, mengubah data tersebut sesuai kebutuhan bisnis, dan menyimpannya ke dalam *Data Warehouse* [2]. Data yang digunakan dalam proses ETL dapat berasal dari berbagai sumber antara lain aplikasi *Enterprise Resource Planning* (ERP), *flat file*, dan *Excel spreadsheet*. Proses ETL terbagi menjadi tiga subproses, antara lain:

1. *Extract*, merupakan proses membaca data dari sumber *database* yang telah ditentukan dan melakukan ekstraksi sebagian data yang diinginkan
2. *Transform*, merupakan proses mengubah/mengkonversi data yang diinginkan dari bentuk awalnya menjadi bentuk yang diperlukan sehingga data tersebut dapat ditempatkan pada *database* lain. Di dalam proses ini terjadi proses penggabungan dan pembersihan data sehingga dapat digunakan sesuai dengan kebutuhan
3. *Load*, merupakan proses menulis atau memindahkan data ke dalam *database* yang diinginkan

2.3 *Decision Support Systems*

Decision Support Systems (DSS) merupakan sistem berbasis komputer yang membantu pengambil keputusan menggunakan data dan model yang tersedia untuk menyelesaikan masalah-masalah [5]. DSS berfungsi menggabungkan sumber daya dari setiap

individu dengan kemampuan dari komputer untuk meningkatkan kualitas dari keputusan. DSS membutuhkan data yang berasal dari berbagai sumber untuk menyelesaikan masalah. Setiap masalah yang harus diselesaikan dan setiap kesempatan dan strategi yang dianalisis membutuhkan data. Data merupakan komponen pertama dari arsitektur DSS. Data berhubungan dengan suatu keadaan yang dapat dimanipulasi menggunakan model yang merupakan komponen kedua dari arsitektur DSS. Beberapa sistem juga memiliki pengetahuan (*knowledge*) yang merupakan komponen ketiga dari arsitektur DSS. Pengguna (*users*) yang merupakan komponen keempat berinteraksi dengan sistem melalui sebuah user interface yang merupakan komponen kelima dalam arsitektur DSS. Dalam membangun DSS, diperlukan perencanaan sistem yang matang disertai dengan penyusunan dan penggabungan komponen-komponen dengan baik.

Beberapa karakteristik dan kemampuan yang dimiliki oleh DSS antara lain [5]:

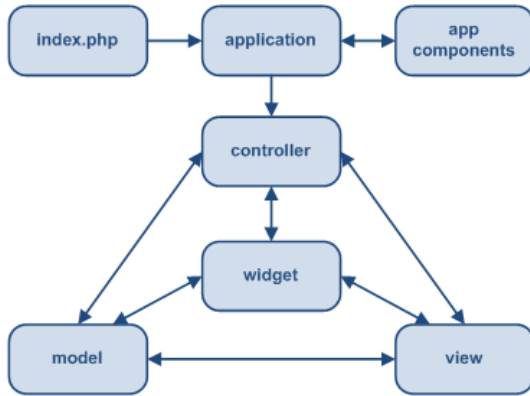
1. Membantu pemegang keputusan, dengan menyatukan cara pikir manusia dan informasi yang terkomputerisasi.
2. Membantu semua level manajerial, dimulai dari *top executives* hingga *line managers*.
3. Membantu individu atau kelompok dalam mengambil keputusan. Masalah yang kurang terstruktur sering membutuhkan kesediaan individu dari beberapa departemen berbeda atau bahkan dari organisasi berbeda. DSS dikembangkan untuk membantu individu maupun kelompok dalam menyelesaikan permasalahan tersebut.
4. Mendukung dalam pembuatan keputusan yang saling berkaitan dan/atau beruntutan. Keputusan dapat dibuat sekali, beberapa kali, atau berturut-turut.
5. Membantu dalam setiap fase proses pengambilan keputusan: *intelligence, design, choice, dan implementation*.
6. Mendukung berbagai jenis proses dan gaya pengambilan keputusan
7. Dapat beradaptasi dan fleksibel terhadap perubahan kondisi
8. *User-friendliness*
9. Lebih meningkatkan efektivitas dari pengambilan keputusan (akurasi, aktualitas, kualitas) daripada efisiensi (biaya dan waktu dari pengambilan keputusan).
10. Pemegang keputusan tetap memegang kendali secara penuh terhadap setiap langkah pengambilan keputusan dalam menyelesaikan permasalahan. DSS berfungsi untuk membantu bukan menggantikan pemegang keputusan.
11. *End users* dapat mengembangkan dan memodifikasi sistem.
12. Model digunakan untuk menganalisa kondisi pengambilan keputusan.
13. Akses diberikan pada berbagai sumber, format, dan tipe data
14. Dapat digunakan sebagai *standalone* tool atau didistribusikan dalam organisasi. DSS dapat pula diintegrasikan dengan aplikasi lain.

2.4 *Yii Framework*

Yii ("Yes It Is!") merupakan sebuah *framework open source* yang disusun dengan menggunakan PHP5 untuk pengembangan aplikasi berbasis *web*. Yii sering digunakan untuk pengembangan sebuah aplikasi yang sifatnya *rapid development* dan juga sangat membantu untuk menghasilkan sebuah aplikasi yang *extremely efficient, extensible, dan maintainable*.

2.4.1 Model, View, dan Controller

Yii mengimplementasi pola desain *model, view, dan controller* (MVC) dimana telah diadopsi secara luas oleh sejumlah pemrograman *web*. MVC bertujuan untuk memisahkan logika bisnis (*business logic*) dari tampilan aplikasi (*user interface*), sehingga para pengembang aplikasi dapat lebih mudah untuk mengubah masing-masing bagian tanpa mempengaruhi yang lain. Struktur statis dari aplikasi Yii berupa MVC dapat dilihat pada Gambar 1. *Model* menggambarkan informasi (data) dan aturan (*business rules*). *View* berisikan elemen-elemen yang digunakan untuk membentuk tampilan aplikasi (*user interface*) seperti teks atau kolom inputan. Sedangkan *controller* berfungsi untuk memajemen komunikasi antara *model* dan *view*.



Gambar 1. Struktur Statis dari Aplikasi Yii

2.4.2 Gii

Yii dilengkapi dengan dengan sebuah *Web-based code generator tool* yang disebut Gii. Gii merupakan sebuah *tool* yang berfungsi untuk membuat *model, controller, form, dan view* melalui tampilan aplikasi, sehingga pengembang tidak perlu membuat MVC secara manual. Gii dapat di kembangkan secara bebas oleh pengguna Yii seperti mengubah isi dari *script* yang dihasilkan dengan menggunakan *generator* dasar Yii atau dengan membuat *generator* yang baru.

3. ANALISIS DAN DESAIN

3.1 Analisis Permasalahan

Permasalahan yang dihadapi oleh UKP antara lain:

1. Pimpinan UKP mengambil keputusan dalam bidang promosi/kerjasama hanya berdasarkan perkiraan dan kebiasaan, belum memanfaatkan data akademik yang ada.
2. Tim Promosi Studi UKP tidak diperlengkapi dengan data atau fakta tentang keadaan akademik UKP saat melakukan promosi ke sekolah-sekolah.
3. Kurangnya sarana yang membantu dalam menghasilkan informasi akademik mahasiswa dengan cepat, lengkap, dan sesuai dengan kebutuhan pimpinan UKP, hal ini dikarenakan bentuk laporan grafik yang disediakan pada sistem informasi akademik merupakan laporan grafik yang statis dan tidak dapat dikustomisasi. Bila Pimpinan UKP ingin mendapatkan informasi tentang akademik mahasiswa yang tidak disediakan di sistem informasi akademik, Pimpinan UKP harus meminta informasi tersebut kepada Biro Administrasi Akademik terlebih dahulu.

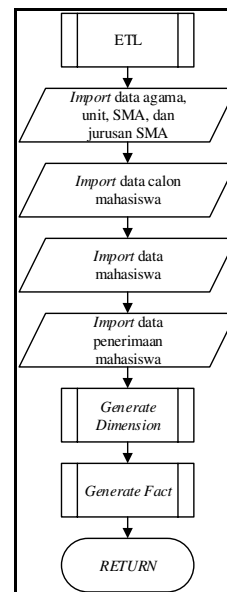
3.2 Analisis Kebutuhan

Dari permasalahan-permasalahan yang ada di atas, dapat disimpulkan bahwa Pimpinan UKP membutuhkan suatu sistem yang berbasis komputer untuk membantu dalam menghasilkan informasi akademik mahasiswa UKP. Sistem yang dibutuhkan merupakan sistem informasi berbasis *Data Warehouse* yang dapat melakukan proses (ETL) data mentah (pada sistem pendaftaran mahasiswa baru dan Sistem Informasi Akademik UKP) dan menghasilkan informasi akademik mahasiswa dalam bentuk grafik. Sistem ini mendapatkan data calon mahasiswa pada sistem pendaftaran mahasiswa baru dan data mahasiswa pada sistem informasi akademik UKP melalui proses *import file excel*.

3.3 Desain Flowchart

3.3.1 Proses ETL

Proses ETL merupakan proses yang dapat diakses oleh *administrator*. *Flowchart* proses ETL dapat dilihat pada Gambar 2. Proses ini diawali dengan melakukan *import* data ke dalam aplikasi. Data yang di-*import* adalah data master (agama, unit/program studi, SMA/sekolah, dan jurusan SMA/sekolah), data calon mahasiswa, data mahasiswa, dan data penerimaan mahasiswa. Setelah *import* selesai, aplikasi akan masuk ke dalam proses *Generate Dimension*, kemudian dilanjutkan ke dalam proses *Generate Fact*. Proses *Generate Dimension* merupakan proses persiapan data master (agama, unit/program studi, SMA/sekolah, dan jurusan SMA/sekolah). dan dilakukan agar data yang akan diolah menjadi lebih valid, sedangkan proses *Generate Fact* merupakan proses persiapan data calon mahasiswa, mahasiswa, dan penerimaan mahasiswa yang diawali dengan melakukan *join table* calon mahasiswa, mahasiswa, dan penerimaan mahasiswa kemudian melakukan pengecekan dan pengolahan *field*.

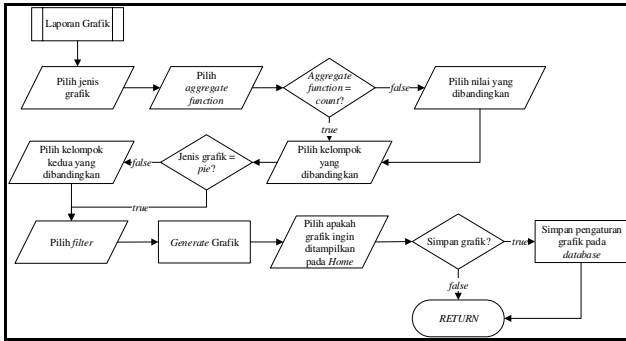


Gambar 2. Flowchart Proses ETL

3.3.2 Proses Laporan Grafik

Proses laporan grafik merupakan proses yang dapat diakses oleh semua jenis *login*. Laporan grafik yang dihasilkan disesuaikan dengan jenis *login*, yaitu *administrator*, pimpinan universitas, dan tim promosi studi dapat mengakses semua data mahasiswa,

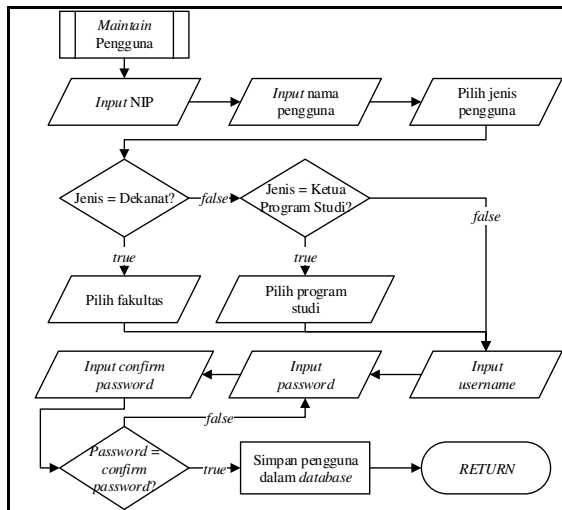
dekanat hanya dapat mengakses data mahasiswa dalam satu fakultas, dan ketua program studi hanya dapat mengakses data mahasiswa dalam satu program studi. *Flowchart* proses laporan grafik dapat dilihat pada Gambar 3. Proses ini diawali dengan meminta pengguna untuk memberikan *input* berupa jenis grafik, *aggregate function*, nilai yang dibandingkan, kelompok yang dibandingkan, dan *filter*. Setelah *input* selesai aplikasi akan mengeluarkan grafik sesuai dengan permintaan pengguna dan memberikan pilihan kepada pengguna untuk menampilkan grafik tersebut pada *Home* atau tidak. Kemudian, pengguna dapat memilih menyimpan pengaturan grafik tersebut pada *database*.



Gambar 3. *Flowchart* Proses Laporan Grafik

3.3.3 Proses Maintain Pengguna

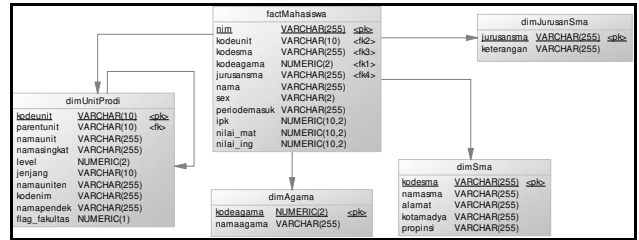
Proses laporan *maintain* pengguna merupakan proses yang dapat diakses oleh *administrator*. *Flowchart* proses *maintain* pengguna dapat dilihat pada Gambar 4. Proses ini diawali dengan meminta *administrator* untuk meng-*input* NIP pengguna, nama pengguna, dan memilih jenis pengguna. Jika jenis pengguna yang dipilih adalah dekanat, maka *administrator* diminta untuk memilih unit fakultas yang dipegang oleh pengguna tersebut. Jika jenis pengguna yang dipilih adalah ketua program studi, maka pengguna diminta untuk memilih unit program studi yang dipegang oleh pengguna tersebut. Kemudian, *administrator* diminta meng-*input* *username*, *password*, dan *confirm password*. Jika *password* dan *confirm password* tidak sama, maka *administrator* akan diberi notifikasi dan diminta untuk meng-*input* kembali. Setelah *input* selesai, aplikasi akan menyimpan data pengguna tersebut dalam *database*.



Gambar 4. *Flowchart* Proses Maintain Pengguna

3.4 Desain *Schema* Data Warehouse

Data Warehouse pada aplikasi ini menggunakan desain *Schema* berupa *star Schema*. Desain *Schema* yang digunakan terdiri atas satu *fact table* yaitu *factMahasiswa* (Tabel Mahasiswa pada ERD) dan empat *dimension table* yaitu *dimUnitProdi* (Tabel Unit_Prodi pada ERD), *dimAgama* (Tabel Agama pada ERD), *dimSma* (Tabel SMA pada ERD), dan *dimJurusanSma* (Tabel Jurusan SMA pada ERD). *Star Schema* yang digunakan dapat dilihat pada Gambar 5.



Gambar 5. *Star Schema* Data Warehouse

3.5 Desain Hak Akses Pengguna

Aplikasi ini memiliki lima macam jenis pengguna, antara lain *administrator*, rektorat, dekanat, ketua program studi, dan tim promosi studi. Setiap jenis pengguna memiliki hak akses yang berbeda-beda. Daftar hak akses yang dimiliki setiap jenis pengguna dapat dilihat pada Tabel 1.

Tabel 1. Daftar Hak Akses Pengguna

No.	Jenis Pengguna	Hak Akses
1.	<i>Administrator</i>	Proses ETL
		Kustomisasi Grafik
		Manipulasi Pengguna
2.	Rektorat	Kustomisasi Grafik (semua data)
3.	Dekanat	Kustomisasi Grafik (data dalam satu fakultas)
4.	Ketua Program Studi	Kustomisasi Grafik (data dalam satu program studi)
5.	Tim Promosi Studi	Kustomisasi Grafik (semua data)

4. HASIL

4.1 Laporan Grafik

Pengguna dapat memilih lima macam grafik, antara lain *pie chart*, *bar chart*, *column chart*, *line chart*, dan *scatter chart*. Kustomisasi untuk grafik yang dapat dilakukan oleh pengguna akan berbeda untuk setiap jenis grafik. Untuk *pie chart*, pengguna dapat memilih *aggregate function*, satu jenis pembandingan, dan *filter*. Sebagai contoh, dilakukan kustomisasi grafik seperti pada Gambar 6, sebagai berikut:

1. Jenis grafik adalah *pie chart*
2. *Aggregate function* yang dipilih adalah jumlah
3. Pembandingan yang dipilih adalah fakultas, dengan anggota yang dipilih adalah ekonomi, ilmu komunikasi, sastra, seni dan desain, teknik sipil dan perencanaan, dan teknologi industri
4. *Filter* yang dipilih adalah IPK, dengan kondisi lebih dari sama dengan 3.51

Jenis grafik yang digunakan: **Pie Chart**

Pilih aggregate function: **Jumlah (Count)**

Pilih Pembanding: **Fakultas**

Pilih Fakultas:

- Pilih Semua
- EKONOMI
- ILMU KOMUNIKASI
- SASTRA
- SENI DAN DESAIN
- TEKNIK SIPIL DAN PERENCANAAN
- TEKNOLOGI INDUSTRI

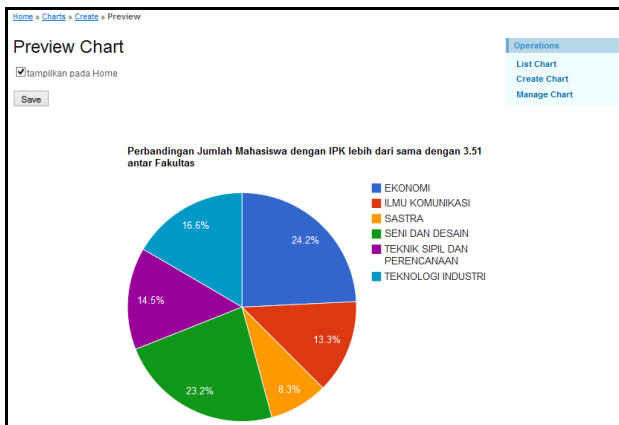
Pilih filter: **IPK** lebih dari sama dengan (\geq) **3.51**

- abaikan mahasiswa dengan IPK 0
- abaikan mahasiswa dengan Nilai Matematika 0
- abaikan mahasiswa dengan Nilai Bahasa Inggris 0

Preview

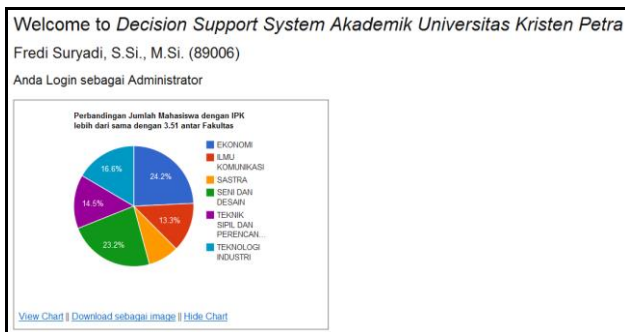
Gambar 6. Contoh Input Kustomisasi Pie Chart

Melalui kustomisasi grafik tersebut didapatkan grafik perbandingan jumlah mahasiswa dengan IPK lebih dari sama dengan 3.51 antar fakultas yang dapat dilihat pada halaman *preview chart* seperti pada Gambar 7.



Gambar 7. Contoh Output Kustomisasi Pie Chart

Pengguna dapat memilih untuk menampilkan grafik yang telah dibuat pada halaman *Home* dengan memberikan centang pada *checkbox* tampilkan pada *Home* yang tersedia. Kemudian, pengguna dapat menyimpan grafik dalam *database* dengan menekan tombol *Save*. Sebagai contoh, pengguna memilih untuk menyimpan dan menampilkan grafik pada halaman *Home*, sehingga halaman *Home* pengguna akan menjadi seperti pada Gambar 8.



Gambar 8. Tampilan Grafik pada Halaman Home

Untuk *scatter chart*, pengguna dapat memilih atribut untuk sumbu X, atribut untuk sumbu Y, satu jenis pembanding, dan *filter*. Sebagai contoh, dilakukan kustomisasi grafik seperti pada Gambar 9, sebagai berikut:

1. Jenis grafik adalah *scatter chart*
2. Atribut untuk sumbu X adalah Nilai Matematika
3. Atribut untuk sumbu Y adalah IPK
4. Pembanding yang dipilih adalah agama, dengan anggota yang dipilih adalah Budha, Hindu, Islam, Katholik, dan Kristen
5. *Filter* yang dipilih adalah fakultas, dengan anggota yang dipilih adalah teknologi industri

Jenis grafik yang digunakan: **Scatter Chart**

Pilih atribut untuk Sumbu X: **Nilai Matematika**

Pilih atribut untuk Sumbu Y: **IPK**

Pilih Pembanding: **Agama**

Pilih Agama:

- Pilih Semua
- AGAMA LAIN
- AGAMA LAIN (3)
- AGAMA LAIN (4)
- Budha
- Hindu
- Islam
- Katholik
- Kristen

Pilih filter: **Fakultas**

Pilih Fakultas:

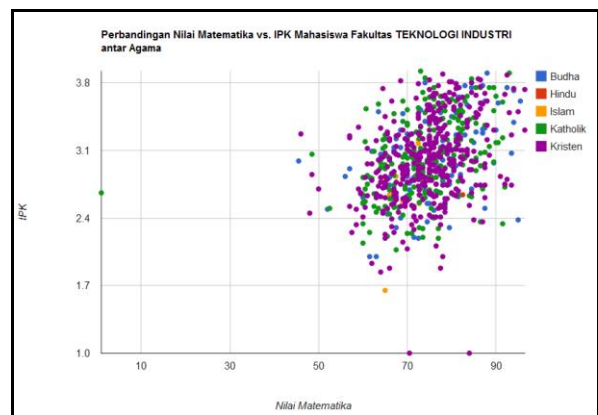
- Pilih Semua
- EKONOMI
- ILMU KOMUNIKASI
- SASTRA
- SENI DAN DESAIN
- TEKNIK SIPIL DAN PERENCANAAN
- TEKNOLOGI INDUSTRI

- abaikan mahasiswa dengan IPK 0
- abaikan mahasiswa dengan Nilai Matematika 0
- abaikan mahasiswa dengan Nilai Bahasa Inggris 0

Preview

Gambar 9. Contoh Input Kustomisasi Scatter Chart

Melalui kustomisasi tersebut, didapatkan grafik perbandingan Nilai Matematika vs. IPK mahasiswa fakultas teknologi industri antar agama yang dapat dilihat pada Gambar 10.



Gambar 10. Contoh Output Kustomisasi Scatter Chart

4.2 Penilaian Aplikasi

Untuk mengetahui penilaian pengguna tentang aplikasi ini, dilakukan penelitian terhadap penggunaan aplikasi ini. Sampel dari penilaian ini adalah lima orang pejabat struktural UKP. Untuk mengumpulkan data, disebarakan kuesioner yang berisi indikator-indikator penilaian terhadap penggunaan aplikasi. Indikator yang dinilai melingkupi tampilan aplikasi, fitur kustomisasi grafik, kemudahan penggunaan aplikasi, kemampuan untuk menjawab kebutuhan pengguna, dan keseluruhan aplikasi. Detail penilaian terhadap penggunaan aplikasi dapat dilihat pada Tabel 2.

Tabel 2. Tabel Penilaian terhadap Penggunaan Aplikasi

Indikator	1	2	3	4	5
Tampilan aplikasi				3	2
Fitur kustomisasi grafik				1	4
Kemudahan menggunakan aplikasi				4	1
Aplikasi menjawab kebutuhan			2	1	2
Keseluruhan aplikasi				2	3

Keterangan skala penilaian:

- Nilai 1: Sangat buruk
- Nilai 2: Buruk
- Nilai 3: Cukup
- Nilai 4: Baik
- Nilai 5: Sangat baik

Penilaian secara keseluruhan terhadap kelayakan program:

1. Tampilan aplikasi 100% baik
2. Fitur kustomisasi grafik 100% baik
3. Kemudahan menggunakan aplikasi 100% baik
4. Aplikasi menjawab kebutuhan 60% baik dan 40% cukup
5. Keseluruhan aplikasi 100% baik

5. KESIMPULAN

Dari hasil perancangan dan pembuatan aplikasi, dapat diambil kesimpulan antara lain:

1. Aplikasi yang dikembangkan menyediakan tiga fitur utama, antara lain ETL, kustomisasi grafik, dan *maintain* pengguna.
2. Fitur-fitur yang disediakan aplikasi cenderung lebih berguna untuk *administrator*, rektorat, dan tim promosi studi karena informasi yang disediakan aplikasi berupa *summary*. Sementara itu, ketua program studi lebih membutuhkan informasi yang terperinci misalnya berupa rincian nilai per matakuliah dan dosen yang mengajar daripada *summary* rata-rata nilai mahasiswa dalam bentuk IPK.
3. Pengguna aplikasi menilai tampilan aplikasi 100% baik, fitur kustomisasi grafik 100% baik, kemudahan menggunakan aplikasi 100% baik, aplikasi menjawab kebutuhan 60% baik, dan keseluruhan aplikasi 100% baik. Hal ini menunjukkan bahwa aplikasi yang dikembangkan memiliki manfaat bagi UKP.

6. DAFTAR PUSTAKA

- [1] Golfarelli, M. & Rizzi, S. 2009. *A Survey on Temporal Data Warehousing*. International Journal of Data Warehousing & Mining, 5(1), 1-17.
- [2] Gour, V., Sarangdevot, S. S., Tanwar, G. S., & Sharma, A. 2010. *Improve Performance of Extract, Transform and Load (ETL) in Data Warehouse*. International Journal on Computer Science and Engineering, 2(3), 786-789.
- [3] Ponniah, P. 2010. *Data Warehouse Fundamentals: a Comprehensive Guide for IT Professional* (2nd ed.). New York: The McGraw-Hill Companies.
- [4] Reddy, G. S., Srinivasu, R., Rao, M. P., & Rikkula, S. R. 2010. *Data Warehousing, Data Mining, OLAP and OLTP Technologies Are Essential Elements To Support Decision-Making Process In Industries*. International Journal on Computer Science and Engineering, 2(9), 2865-2873.
- [5] Turban, E., Aronson, J. E., Liang, T., & Sharda, R. 2014. *Decision Support and Business Intelligence Systems* (10th ed.). New Jersey: Pearson Education, Inc.