

Implementasi *Stored Procedure* Dan *View* Basis DBMS Dalam Membangun Sistem Informasi Tugas Akhir Mahasiswa Politeknik Kampar

Andri Nofiar.Am¹

¹ Jurusan Teknik Informatika Politeknik Kampar
Jln. Tengku Muhammad KM 2 Bangkinang INDONESIA

¹andrinofiar@poltek-kampar.ac.id

Intisari— Tugas akhir di Politeknik Kampar merupakan karya ilmiah yang dibuat oleh seorang atau sekelompok mahasiswa tingkat III semester VI untuk menyelesaikan Studi Jenjang Diploma-III. Untuk dapat mengetahui informasi tentang tugas akhir yang telah dibuat, mahasiswa terlebih dahulu meminjam dokumentugas akhir ke Ka. Prodi atau perpustakaan. Namun, sistem pengarsipan tugas akhir yang berjalan dinilai masih kurang efektif. Tujuan dari penelitian ini adalah untuk merancang sebuah sistem informasi dengan mengimplementasikan *stored procedure* dan *view basis dbms* tentang tugas akhir mahasiswa Politeknik Kampar berbasis *web*. Tahapan yang digunakan untuk membangun aplikasi *web* ini menggunakan tahapan model SDLC air terjun (*waterfall*). Dengan mengimplementasikan *stored procedure* dan *view basis dbms* dalam pembuatan program sistem informasi tugas akhir ini, dapat memudahkan dalam pengembangan sistem dikemudian hari. Pengimplementasian *sotred procedure* dan *view basis dbms* telah berhasil di uji cobakan pada sistem informasi tugas akhir mahasiswa Politeknik Kampar dan menyediakan informasi kepada mahasiswa yang ingin mengajukan judul tugas akhir secara cepat, tepat dan akurat.

Abstract— *Final project in Politeknik Kampar is a scientific work who made by one or group students grade III in semester VI for finished Study Diploma III. To know information about final project that have made, initially student borrow document final project to Ka.Prodi or library. But, the final project archiving system that runs less effective. The purpose of this research is design an information system use implementing a stored procedure and view dbms basis about students' final project Politeknik Kampar web based. Stages are used to build web applications using stage waterfall SDLC models. With implementing stored procedure and view basis dbms in manufacturing information systems program final project, can facilitate the development of the system in the future. Implementation stored procedure and view basis dbms has been successfully tested on the information system student's final project Politeknik Kampar and provide information for students who wish to submit a final project tittle in a rapid, precise and accurate.*

Keywords— *DBMS, Final Project, Information System, Stored Procedure, View*

I. PENDAHULUAN

Tugas akhir di Politeknik Kampar merupakan karya ilmiah yang dibuat oleh seorang atau sekelompok mahasiswa tingkat III semester VI untuk menyelesaikan Studi Jenjang Diploma-III. Setelah menyelesaikan tugas akhir, mahasiswa diwajibkan untuk mengumpulkan *soft file* tugas akhir ke ketua program studi (Ka. Prodi), akademik dan perpustakaan. Namun, sistem pengarsipan tugas akhir yang berjalan dinilai masih kurang efektif.

Sistem informasi memberikan sebuah informasi kepada para pengguna yang membutuhkan informasi berupa data. Data yang diberikan sebagai informasi dapat berupa data teks maupun data gambar. Data teks sering digunakan untuk sebuah informasi berupa artikel maupun data yang lainnya yang berbentuk teks [1].

Berdasarkan dari sistem yang sedang berjalan saat ini, untuk dapat mengetahui informasi tentang tugas akhir yang telah dibuat, mahasiswa terlebih dahulu meminjam dokumen

tugas akhir ke Ka. Prodi atau perpustakaan. Sistem informasi dibutuhkan untuk menunjang dan mempermudah dalam penyampaian informasi mengenai tugas akhir agar dapat meningkatkan kualitas pendidikan yang lebih baik dari sebelumnya. Sistem tersebut nantinya dapat diakses oleh pengguna yang telah terkoneksi dengan *internet*.

Sistem Informasi Tugas Akhir Mahasiswa Politeknik Kampar akan dibuat dengan mengimplementasikan *Stored Procedure* dan *View* basis *DBMS*. Sebuah *stored procedure* disimpan dan di susun dikatalog *database* yang dapat dieksekusi lebih cepat dibandingkan dengan *SQL* yang disusun dari kode aplikasi, hal ini karena aplikasi cukup memanggil *stored procedure* dan mengirim *parameter* yang diperlukan, *stored procedure* lebih *flexisibel* karena ada *parameter* didalamnya, kemudian keuntungan lainnya dalam penggunaan *stored procedure* adalah apabila terjadi perubahan aturan *database* maka bisa dilakukan di *server* sehingga akan mempengaruhi semua *client* dan juga untuk

mempermudah saat perbaikan aplikasi apabila ada proses *query* yang berulang.

Stored procedure adalah sejumlah pernyataan *Structured Query Language* (SQL) yang dapat dianggap menjadi satu unit logika tunggal yang disimpan dalam *database* yang tersusun sehingga dapat dipanggil oleh sebuah program, *trigger* atau bahkan *stored procedure* [2].

View adalah tabel yang dibuat dengan *query* SQL *SELECT*, *view* bisa diasumsikan sebagai virtual tabel ataupun *stored query*. Data yang bisa diakses lewat *view* tidak disimpan dalam *database* sebagai objek yang nyata, yang disimpan dalam *database* hanya perintah *SELECT*nya. Hasil dari perintah *SELECT* tersebut yang membentuk tabel virtual yang ditampilkan *view* [3].

Beberapa keuntungan dari penggunaan *stored procedure* yaitu membantu kita dalam menggabungkan kode *transact-sql* ketika terjadi penambahan data. Hanya terdapat satu tempat penyimpanan untuk mencari kode *sql* atau *sql batch* sehingga pengguna dapat menemukan bagian yang perlu diperbaiki atau bermasalah. *Stored procedure* juga dapat mengurangi lalu lintas jaringan pada saat permintaan data yang lebih besar. Selain itu, penggunaan *stored procedure* memungkinkan pengguna untuk menyamakan pengambilan data, dan *stored procedure* juga mendukung penggunaan kembali kode dan menstabilkan waktu respon *query* [4].

PHP singkatan dari PHP *Hypertext Processor* yang digunakan sebagai bahasa script server-side dalam pengembangan Web yang disisipkan pada dokumen HTML. Penggunaan PHP memungkinkan Web dapat dibuat dinamis sehingga *maintenance* situs Web tersebut menjadi lebih mudah dan efisien. PHP merupakan *software Open-Source* yang disebar dan dilisensikan secara gratis serta dapat didownload secara bebas dari situs resminya <http://www.php.net>. PHP ditulis dengan menggunakan bahasa C.

MySQL dikembangkan oleh sebuah perusahaan Swedia bernama MySQL AB yang pada saat itu bernama TcX DataKonsult AB sekitar tahun 1994-1995, namun cikal bakal kodenya sudah ada sejak 1979. Awalnya TcX membuat MySQL dengan tujuan mengembangkan aplikasi web untuk klien. TcX merupakan perusahaan pengembang software dan konsultan *database*. Saat ini MySQL sudah diakuisi oleh Oracle Corp. MySQL adalah salah satu jenis *database server* yang sangat terkenal dan banyak digunakan untuk membangun aplikasi web yang *database* sebagai sumber dan pengelolaan datanya. Kepopuleran MySQL antara lain karena MySQL menggunakan SQL sebagai bahasa dasar untuk mengakses *database*-nya sehingga mudah untuk digunakan. MySQL juga bersifat *open source* dan *free* pada berbagai *platform* kecuali pada *windows* yang bersifat *shareware*. MySQL didistribusikan dengan lisensi *open source* GPL (General Public License) mulai versi 3.23[3].

II. METODOLOGI PENELITIAN

Adapun langkah-langkah dalam metodologi penelitian yang mendukung terhadap data-data yang diperlukan serta rencana proses penelitian antara lain terdiri dari :

1. Unit Penelitian

Pengambilan data untuk penelitian ini diperoleh dari sebuah instansi yaitu pada prodi Teknik Informatika Politeknik Kampar yang beralamatkan di Jl. Tengku Muhammad KM.2 Bangkinang, Riau.

2. Metode Pengembangan/Pembangunan Perangkat Lunak.

A. Jenis Data

Jenis data yang diperlukan penulis yaitu data *primer*. data *primer* yang diperoleh dengan cara mengumpulkannya secara langsung dengan melakukan observasi dan wawancara, yaitu data asli dan *up to date* berupa data tugas akhir mahasiswa prodi Teknik Informatika Politeknik Kampar

B. Sumber Data

Sumber data dalam penelitian ini diperoleh dari lembaga atau instansi prodi Teknik Informatika Politeknik Kampar.

C. Pengumpulan Data

Dalam hal ini penulis menggunakan teknik pengumpulan data sebagai berikut :

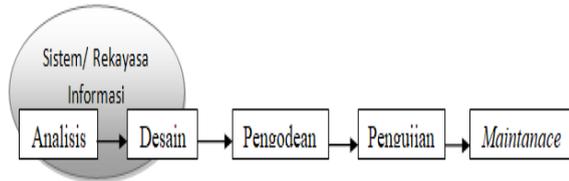
- a. *Observasi*, yaitu teknik pengumpulan data dengan cara melakukan pengamatan secara langsung terhadap pelaksanaan penelitian guna memperoleh data atau gambaran serta keterangan pihak prodi Teknik Informatika Politeknik Kampar.
- b. *Interview*, yaitu pengumpulan data dengan cara berkomunikasi secara langsung dengan sumber data. Penulis melakukan proses tanya jawab dengan ketua prodi Teknik Informatika Politeknik Kampar sehingga penulis dapat mengetahui permasalahan yang ada, serta mengetahui sistem yang digunakan.
- c. Studi pustaka, merupakan penelitian dengan menggunakan *literature*, buku – buku, jurnal, karya tulis ilmiah dan situs yang berkaitan dengan penelitian guna menunjang hasil penelitian, tata bahasa, maupun pengkajian penelitian.

D. Pembangunan Perangkat Lunak

Perangkat lunak (*software*) adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara penggunaan (*user manual*).

Proses Pengembangan Perangkat Lunak (*Software Development Process*) adalah suatu penerapan struktur pada pengembangan suatu perangkat lunak (*Software*), yang bertujuan untuk mengembangkan sistem dan memberikan panduan untuk menyelesaikan proyek pengembangan sistem melalui tahapan-tahapan tertentu.

Tahapan yang digunakan untuk membangun aplikasi web ini menggunakan tahapan model SDLC air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequential linear*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara *sekuensial* atau terurut dimulai dari analisis, desain, pengodean, pengujian dan tahap pendukung (*support*) [6]. Berikut adalah model gambar air terjun :



Gambar 1. Ilustrasi model *Waterfall*

III. HASIL DAN PEMBAHASAN

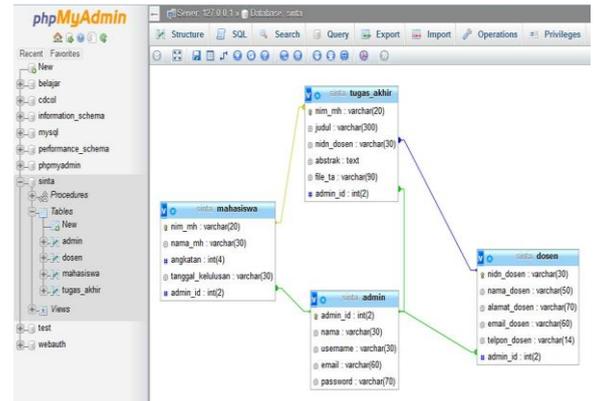
Hasil dan pembahasan merupakan tahap dimana sistem siap dioperasikan pada keadaan yang sebenarnya, dari sini akan diketahui apakah sistem yang dibuat benar-benar dapat menghasilkan tujuan yang diinginkan yaitu mengimplementasikan *stored procedure* dan *view basis dbms* dalam membangun sistem informasi tugas akhir mahasiswa. Langkah – langkah dalam membangun *stored procedure* dan *view basis dbms* adalah :

- A. Terlebih dahulu Membuat *query sql* baik itu *stored procedure* maupun *view basis dbms*.
- B. Setelah *query* dibuat, buka *phpmyadmin*, masuk ke *database*, klik tab *sql* lalu salin *query* yang telah dibuat tadi.
- C. Tambahkan (;) di *Bookmark this sql query* dan tambahkan (/) di *delimiter* lalu *go*.
- D. Untuk menjalankan *stored procedure* yang telah dibuat di *database*, jalankan *query sql* dengan perintah *Call nama_stored*.
- E. Untuk menjalankan *view basis dbms* yang telah dibuat di *database*, jalankan *query sql* dengan perintah *Select * from nama_view*.

1. Hasil Implementasi Database Di *Phpmyadmin*

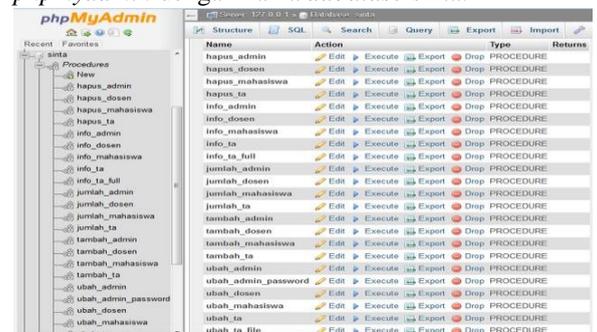
A. Hasil Implementasi Database.

Dibawah ini merupakan gambar tampilan *design database* di *phpmyadmin* dengan nama *dabatase* *sinta*.



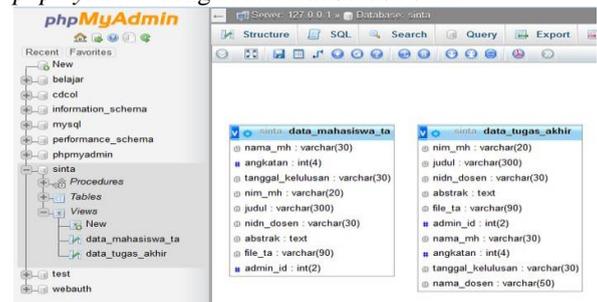
Gambar 2. Hasil Implementasi Database

- B. Hasil Implementasi *Stored Procedure* Di Database. Dibawah ini merupakan gambar tampilan implementasi *stored procedure* di *database phpmyadmin* dengan nama *dabatase* *sinta*.



Gambar 3. Hasil Implementasi *Stored Procedure* di Database

- C. Hasil Implementasi *View Basis DBMS* Di Database. Dibawah ini merupakan gambar tampilan implementasi *view basis dbms* di *database phpmyadmin* dengan nama *dabatase* *sinta*.



Gambar 4. Hasil Implementasi *View Basis DBMS* di Database

2. Kode Program dan Hasil Tampilan *Interface User*

- A. Kode Program dan *Interface* Hasil Pencarian Tugas Akhir

```

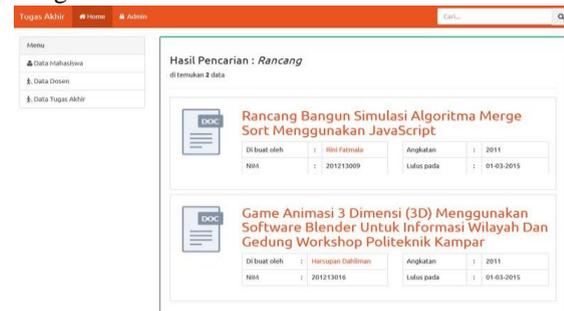
1 create view data_tugas_akhir as
2     select ta.*
3         , m.nama_mh
4         , m.angkatan
5         , m.tanggal_kelulusan
6         , d.nidn_dosen
7         , d.nama_dosen
8     from tugas_akhir ta
9     inner join mahasiswa m
10        on ta.nim_mh=m.nim_mh
11     inner join dosen d
12        on ta.dosen_pembimbing=d.nidn_dosen
    
```

Baris ke 1 untuk membuat *view* dengan nama *data_tugas_akhir*, baris ke 2 sampai ke 12 untuk menjalankan *query sql select* guna mendapatkan data di tabel *tugas_akhir*, tabel *mahasiswa* dan tabel *dosen*. Penghubung ketiga tabel ini menggunakan *nim mahasiswa* dan *nidn dosen* yang ada di tabel *tugas_akhir*.

Proses kode program pemanggilan semua *data_tugas_akhir view basis dbms* yang diurutkan berdasarkan *nim_mh* sebagai berikut :

```
$all_ta = $db->fetch_multiple("select * from data_tugas_akhir order by nim_mh ASC");
```

Dari hasil kode program diatas maka dihasilkan *Interface* hasil pencarian tugas akhir seperti gambar sebagai berikut :



Gambar 5. Interface Hasil Pencarian Tugas Akhir

B. Kode Program dan *Interface* Hasil Informasi Data Mahasiswa

```

1 CREATE PROCEDURE info_mahasiswa(IN nim_mh_i varchar(50))
2 BEGIN
3     select * from mahasiswa where nim_mh=nim_mh_i;
4 END
    
```

Baris ke 1 untuk membuat *stored procedure* dengan nama *info_mahasiswa* dengan *input parameter* *nim_mh_i* baris ke 2 merupakan *tag* pembuka, baris ke 3 untuk mendapatkan data dari tabel *mahasiswa* berdasarkan *nim* yang di inputkan, Baris ke 4 sebagai penutup.

Proses kode program pemanggilan *stored procedure* *info_mahasiswa* sebagai berikut :

```
$data_mahasiswa = $db->fetch("call info_mahasiswa('$nim_mh')");
```

Kode program diatas berfungsi untuk memanggil *stored procedure* dengan nama *info_mahasiswa* berdasarkan *nim_mh* dan semua data mahasiswa dengan *nim* tersebut akan di dapatkan.

Dari kode program diatas maka dihasilkan *Interface* informasi data mahasiswa yang merupakan halaman bagi *user* untuk melihat informasi data mahasiswa dan merupakan implementasi dari *stored procedure* yang tersimpan dalam *database* dengan nama *info_mahasiswa*. *Interface* hasil informasi data mahasiswa sebagai berikut :



Gambar 6. Interface Hasil Informasi Data Mahasiswa

C. Kode Program dan *Interface* Hasil Informasi Data Dosen

```

1 CREATE PROCEDURE info_dosen(IN nidn_dosen_i varchar(50))
2 BEGIN
3     select * from dosen where nidn_dosen=nidn_dosen_i;
4 END
    
```

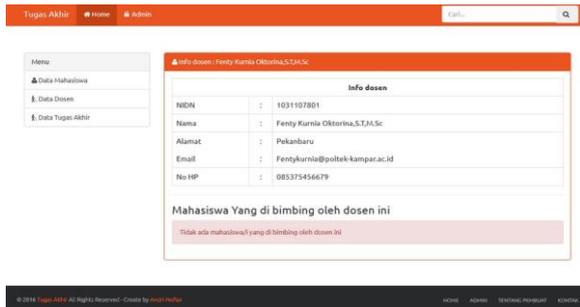
Baris ke 1 untuk membuat *stored procedure* dengan nama *info_dosen* dengan *input parameter* *nidn_dosen_i* baris ke 2 merupakan *tag* pembuka, baris ke 3 untuk mendapatkan data dari tabel *dosen* berdasarkan *nidn* yang di inputkan, Baris ke 4 sebagai penutup.

Proses kode program pemanggilan *stored procedure* *info_dosen* sebagai berikut :

```
$data_dosen = $db->fetch("call info_dosen('$nidn_dosen')");
```

Kode program diatas berfungsi untuk memanggil *stored procedure* dengan nama *info_dosen* berdasarkan *nidn_dosen* dan semua data dosen dengan *nidn* tersebut akan di dapatkan.

Dari kode program diatas maka dihasilkan *Interface* informasi data dosen yang merupakan halaman bagi *user* untuk melihat informasi data dosen dan merupakan implementasi dari *stored procedure* yang tersimpan dalam *database* dengan nama *info_dosen*. *Interface* hasil informasi data dosen sebagai berikut :



Gambar 7. Interface Hasil Informasi Data Dosen

D. Kode Program dan Interface Hasil Informasi Data Tugas Akhir

```

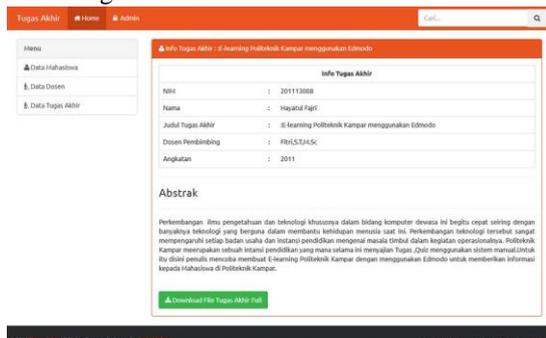
1 CREATE PROCEDURE info_ta_full(IN nim_mh_i varchar(50))
2 BEGIN
3 select * from `data_tugas_akhir` where nim_mh=nim_mh_i;
4 END
    
```

Baris ke 1 untuk membuat *stored procedure* dengan nama *info_ta_full* dengan *input parameter* *nim_mh_i* baris ke 2 merupakan *tag* pembuka, baris ke 3 untuk mendapatkan data dari tabel *data_tugas_akhir* berdasarkan *nim* yang di inputkan, Baris ke 4 sebagai penutup.

Proses kode program pemanggilan *stored procedure* *info_ta_full* sebagai berikut :

```
$data_ta = $db->fetch("call info_ta_full('$nim_mh')");
```

Dari kode program diatas maka dihasilkan *Interface* informasi data tugas akhir yang merupakan halaman bagi *user* untuk melihat informasi data tugas akhir dan merupakan implementasi dari *stored procedure* yang tersimpan dalam *database* dengan nama *info_ta_full*. *Interface* hasil informasi data tugas akhir sebagai berikut :



Gambar 8. Interface Hasil Informasi Data Tugas Akhir

UCAPAN TERIMAKASIH

Penelitian ini dapat dilaksanakan dengan baik, berkat bantuan dari berbagai pihak, untuk itu peneliti mengucapkan terimakasih kepada Bapak Edward Ali, M.Kom selaku Ketua Program Studi Teknik Informatika STMIK AMIK Riau dan pihak akademik Politeknik Kampar yang telah memberikan kerjasama yang baik dalam penelitian ini.

IV. KESIMPULAN

Berdasarkan hasil analisa dan pembahasan serta uraian dari bab-bab diatas, maka dapat disimpulkan: Pengimplementasian *stored procedure* dan *view basis dbms* telah berhasil di uji cobakan pada sistem informasi tugas akhir mahasiswa Politeknik Kampar dan menyediakan informasi kepada mahasiswa yang ingin mengajukan judul tugas akhir secara cepat, tepat dan akurat.

REFERENSI

[1]Nugroho, Fajar Adi, Kodrat Iman Satoto, and R Rizal Isnanto. 2013. "Perancangan Sistem Infoamsi Kerja Praktek Dan Tugas Akhir Teknik Elektro Universtias" *Transient* 2(3): 519–27.

[2]Kakade, Santosh, Rohan Thakare, Bhushan Sapare, and Dr. B.B. Meshram. 2012. "Maintaining Stored Procedures in Database Application." *International Journal of ...* 2(3): 400–403. <http://ijcscn.com/Documents/Volumes/vol2issue3/ijcscn2012020319.pdf>.

[3]Husni, Muchammad, Nunut Priyo Jatmiko, and Arief Prasetyo. 2005. "Database Sql Server Berbasis Web." 4(1): 40–45.

[4]Nasair, Mysql. 2005. "Stored Procedures." In *Stored Procedures*, , 271–87.

[5]Suhartanto, Medi. 2012. "Kata Kunci : Pembuatan Website Sekolah, PHP, 1.1." *Journal Speed-Sentra Penelitian Enginerring dan Edukasi* 4(1): 1–8. <http://ijns.org/journal/index.php/speed/article/view/1156/1144>.

[6] Rosa, A, and M Shalahuddin. 2013. "Rekayasa Perangkat Lunak Terstruktur Dan Berorientasi Objek." : 2–165.