

RANCANG BANGUN SISTEM ADMINISTRASI KERJA PRAKTEK DAN TUGAS AKHIR BERBASIS WEB MENGGUNAKAN FRAMEWORK CODEIGNITER

Arkhan Subari, Dista Yoel Tadeus, Heru Winarno, Teguh Yuwono
Program Studi Diploma III Teknik Elektro
Sekolah Vokasi Universitas Diponegoro

ABSTRACT

Arkhan Subari, Dista Yoel Tadeus, Heru Winarno, Teguh Yuwono, in this paper explain that information system is a set of components that interconnect, collect, process, store, and distribute information to support decision making and supervision within an organization. Today the information system can not be separated with computer technology that feels growing rapidly along with our increasingly complex needs. These developments can be seen from the many organizations or institutions in various fields that use computerized systems to perform various jobs quickly and with great accuracy. The use of computerized systems is expected to minimize errors made by humans thereby reducing large losses. And can mengefisienkan time in completing the work quickly and accurately. One concept of building a computerized information system is dengfan MVC model. Model-View-Controller (MVC) is a concept introduced by the inventor of Smalltalk (Trygve Reenskaug) to encapsulate data along with processing (model), isolate from the process of manipulation (controller) and view to be represented on a user interface . The MVC architecture has the benefit of separating between model and view allowing multiple views using the same model. As a result, an application model component is easier to implement, test, and maintain, because all access to the model runs through this component. One of the frameworks that works based on the MVC concept is Codeigniter. Codeigniter is an open source web application framework that is used to build dynamic php applications. The main purpose of Codeigniter development is to help developers to work on applications faster than writing all the code from scratch. Because of the superiority of codeigniter that the system of practical work administration and the final task is built using the framework..

Keywords: web application, information system, MVC, codeigniter.

PENDAHULUAN

Latar Belakang

Sistem informasi merupakan sekumpulan komponen yang saling berhubungan, mengumpulkan, memproses, menyimpan, dan mendistribusikan informasi untuk menunjang pengambilan keputusan dan pengawasan dalam suatu organisasi.

Selain menunjang proses pengambilan keputusan, koordinasi dan pengawasan, sistem informasi juga dapat membantu manusia dalam menganalisis permasalahan, menggambarkan hal-hal yang rumit dan menciptakan produk baru.

Dewasa ini sistem informasi tidak dapat dipisahkan dengan teknologi komputer yang terasa berkembang semakin pesat seiring dengan kebutuhan kita yang semakin kompleks. Perkembangan tersebut dapat dilihat dari banyaknya organisasi atau institusi diberbagai bidang yang menggunakan sistem terkomputerisasi untuk melaksanakan berbagai pekerjaan secara cepat dan dengan ketelitian yang tinggi. Penggunaan sistem komputerisasi diharapkan dapat meminimalkan kesalahan yang dibuat oleh manusia sehingga mengurangi kerugian yang besar. Serta dapat mengefisienkan waktu dalam menyelesaikan pekerjaan tersebut secara cepat dan akurat.

Oleh karenanya pemanfaatan sistem informasi terkomputerisasi dalam administrasi kerja praktek dan tugas akhir dalam sebuah institusi

pendidikan sangat dibutuhkan. Dengan adanya sistem ini maka penyiapan sarana penunjang sebelum, selama dan setelah kegiatan kerja praktek dan tugas akhir dapat dilakukan dengan mudah, cepat dan akurat sesuai dengan peraturan yang berlaku.

Tujuan

Penelitian yang dilaksanakan ini bertujuan untuk merancang sebuah sistem administrasi menggunakan komputer yang mampu menyiapkan sarana penunjang dalam kegiatan kerja praktek dan tugas akhir baik sebelum, selama dan setelah kegiatan tersebut dengan mudah, cepat dan akurat sesuai dengan peraturan yang berlaku.

Permasalahan

Permasalahan yang ditemukan adalah bagaimana menyiapkan sarana penunjang dalam kegiatan kerja praktek dan tugas akhir baik sebelum, selama dan setelah kegiatan tersebut dengan mudah, cepat dan akurat sesuai dengan peraturan yang berlaku.

ARSITEKTUR MODEL-VIEW-CONTROLLER (MVC)

Model-View-Controller (MVC) adalah sebuah konsep yang diperkenalkan oleh penemu Smalltalk (Trygve Reenskaug) untuk meng-encapsulasi data bersama dengan pemrosesan

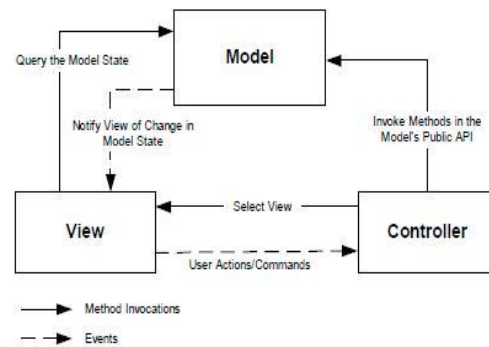
(model), mengisolasi dari proses manipulasi (controller) dan tampilan (view) untuk direpresentasikan pada sebuah user interface^[1]. MVC mengikuti pendekatan yang paling umum dari Layering. Layering hanyalah sebuah logika yang membagi kode kita ke dalam fungsi di kelas yang berbeda. Pendekatan ini mudah dikenal dan yang paling banyak diterima. Keuntungan utama dalam pendekatan ini adalah penggunaan ulang (reusability) kode^[2]. Definisi teknis dari arsitektur MVC dibagi menjadi tiga lapisan^[3], yaitu model, view dan controller. Model, view dan controller sangat erat terkait, oleh karena itu, mereka harus merujuk satu sama lain. Gambar 1. mengilustrasikan hubungan dasar Model-View-Controller.

Arsitektur MVC memiliki manfaat yaitu pemisahan antara model dan view memungkinkan beberapa view menggunakan model yang sama. Akibatnya, komponen model sebuah aplikasi lebih mudah untuk diterapkan, diuji, dan dipelihara, karena semua akses ke model berjalan melalui komponen ini^[4].

Model digunakan untuk mengelola informasi dan memberitahu pengamat ketika ada perubahan informasi^[5]. Hanya model yang mengandung data dan fungsi yang berhubungan dengan pemrosesan data. Sebuah model meringkas lebih dari sekedar data dan fungsi yang beroperasi di dalamnya. Pendekatan model yang digunakan untuk komputer model atau abstraksi dari beberapa proses dunia nyata. Hal ini tidak hanya menangkap keadaan proses atau sistem, tetapi bagaimana sistem bekerja. Sebagai contoh, programmer dapat menentukan model yang menjembatani komputasi back-end dengan front-end GUI (graphical user interface).

View bertanggung jawab untuk pemetaan grafis ke sebuah perangkat. View biasanya memiliki hubungan 1-1 dengan sebuah permukaan layar dan tahu bagaimana untuk membuatnya^[6]. View melekat pada model dan me-render isinya ke permukaan layar. Selain itu, ketika model berubah, view secara otomatis menggambar ulang bagian layar yang terkena perubahan untuk menunjukkan perubahan tersebut.

Controller menerima input dari pengguna dan menginstruksikan model dan view untuk melakukan aksi berdasarkan masukan tersebut. Sehingga, controller bertanggung jawab untuk pemetaan aksi pengguna akhir terhadap respon aplikasi^[7]. Sebagai contoh, ketika pengguna mengklik tombol atau memilih item menu, controller bertanggung jawab untuk menentukan bagaimana aplikasi seharusnya merespon.

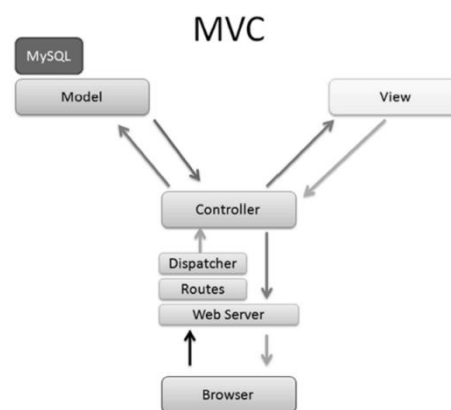


Gambar 1. Hubungan antara model, view, dan controller^[8]

CODEIGNITER

Codeigniter adalah sebuah web application framework yang bersifat open source yang digunakan untuk membangun aplikasi php dinamis. Tujuan utama pengembangan Codeigniter adalah untuk membantu developer untuk mengerjakan aplikasi lebih cepat daripada menulis semua code dari awal^[9]. Codeigniter menyediakan berbagai macam library yang dapat mempermudah dalam pengembangan. Codeigniter diperkenalkan kepada public pada tanggal 28 Februari 2006.

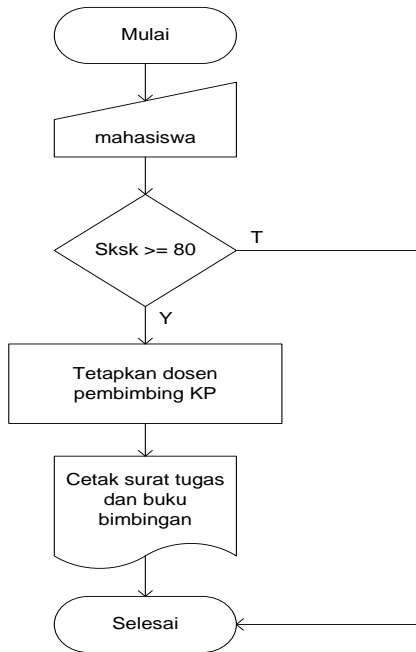
Codeigniter dibangun menggunakan konsep Model-View-Controller (MVC) development pattern. Dalam Codeigniter, browser berinteraksi melalui controller. Controller yang akan menerima dan membalas semua request dari browser. Ketika controller membutuhkan data, maka controller akan meminta ke Model. Sedangkan untuk tampilan ke user akan ditangani oleh View. Jadi otak dari aplikasi ada di controller, muka aplikasi ada di view dan data berada di model. Hal ini ditunjukkan pada Gambar 2.



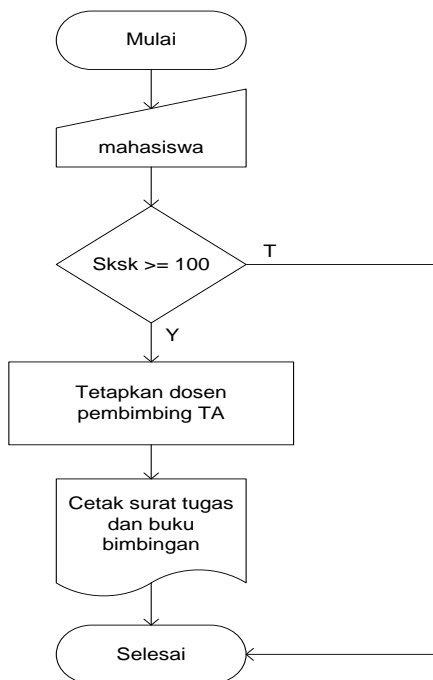
Gambar 2. Alur data codeigniter^[9]

DIAGRAM ALUR PERANGKAT LUNAK

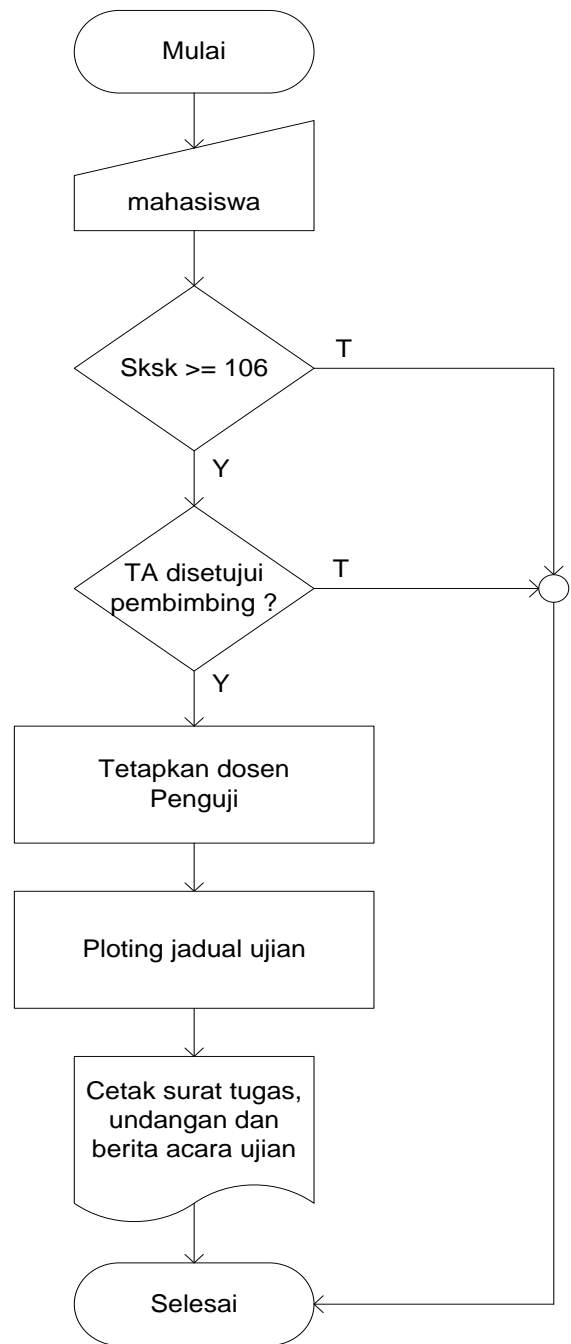
Diagram alur digunakan untuk memberikan gambaran jalanya aplikasi yang akan dibuat^[10]. Alur perangkat lunak dari aplikasi administrasi kerja praktek dan tugas akhir ini ditunjukkan pada Gambar 3,4 dan 5.



Gambar 3. Diagram alur penetapan pembimbing KP



Gambar 4. Diagram alur penetapan pembimbing TA



Gambar 5. Diagram alur administrasi pelaksanaan ujian TA

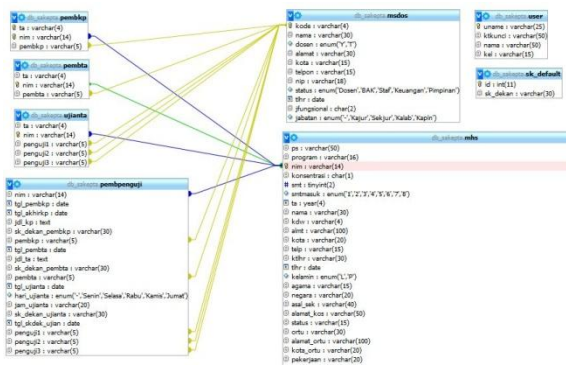
PERANCANGAN DATABASE

Dalam sebuah aplikasi, database berguna untuk mengatur keberadaan data yang dibutuhkan aplikasi. Pengaturan meliputi penyimpanan, perubahan, penghapusan serta pengaturan akses data oleh user. Data-data disimpan dalam tabel-tabel yang telah ditetapkan^[11].

Mengikuti konsep dari database, dalam penelitian ini setiap data yang dibutuhkan ditempatkan pada tabel-tabel. Jumlah tabel yang digunakan adalah 8 tabel, yaitu : mhs, msdos,

pembkp, pembpenguji, pembta, sk_default, ujianta dan user.

Setiap tabel yang dibuat dihubungkan satu sama lain dalam sebuah relationship yang dapat digambarkan seperti pada Gambar 6.



Gambar 6. Relationship tabel

PERANCANGAN ANTARMUKA DAN TAMPILAN APLIKASI

Perancangan antarmuka merupakan salah satu langkah penting dalam pengembangan aplikasi, dalam hal ini aplikasi sistem administrasi kerja praktek dan tugas akhir. Tujuan utama dari perancangan antarmuka adalah membangun antarmuka yang user friendly bagi pengguna, namun tetap memberikan fungsi sesuai dengan yang diinginkan^[10].

Dalam penelitian ini, setiap modul yang dibuat, dilakukan perancangan antarmuka terlebih dahulu. Total modul yang dibuat adalah 10 modul yang meliputi :

1. Modul Master Data Dosen
2. Modul Master Data Mahasiswa
3. Modul Penetapan Pembimbing Kerja Praktek
4. Modul Penetapan Pembimbing Tugas Akhir
5. Modul Penetapan Penguji Tugas Akhir
6. Modul Rekap Per Dosen
7. Modul Rekap Per Mahasiswa
8. Modul Dokumen Pembimbing Kerja Praktek
9. Modul Dokumen Pembimbing Tugas Akhir
10. Modul Dokumen Penguji Tugas Akhir

Berdasarkan rancangan antarmuka yang dibuat itulah, tampilan aplikasi akan dibuat. Gambar 7 dan 8 menunjukkan contoh perancangan antarmuka serta tampilan aplikasinya untuk modul rekap KP dan TA per mahasiswa.

Pilih Tahun

Rekap per mhs

Gambar 7. Desain antarmuka modul rekap per mahasiswa

NO	NIM	NAMA	PEMBIMBING KP	PEMBIMBING TA	PENGUJI TA
1	2106011006002	HERMANI TRIKARSIANA	Priyo Sasoko,ST,AM,Eng	Yunarto,ST,MT	Drs.Heru Winarno Drs.Eko Ariyanto Priyo Sasoko,ST,AM,Eng
2	2106011006001	FERDAN RAMADHAN	Ir. H.Safid Maman ,MT	Drs.Subali	Yunarto,ST,MT Drs.H. Teguh Yuvono,MKkom Ir. H.Safid Maman ,MT
3	2106011006002	PRIYO TRI HANDOKO	Drs.Subali	Drs.Heru Winarno	Arhan Subari, ST,AM,Kom Priyo Sasoko,ST,AM,Eng Drs.Subali
4	2106011006003	DWI AGUNG WULIYA	Drs.Heru Winarno		
5	2106011006004	RATNA PUSPA NINGRUM	Drs.Subali		
6	2106011006006	ANBAR TANDU FASILA	Drs.Subali	Drs.Heru Winarno	Arhan Subari, ST,AM,Kom Priyo Sasoko,ST,AM,Eng Drs.Subali
7	2106011006007	NOORROKHATI AGUS WIYATNI	Drs.Subali	Drs.Subali	Yunarto,ST,MT Drs.H. Teguh Yuvono,MKkom Ir. H.Safid Maman ,MT
8	2106011006009	ABDUL GHOFUR	Dr.Drs.Iwan Setiono,MSI	Drs.H. Teguh Yuvono,MKkom	Drs.Eko Ariyanto Ir. H.Safid Maman ,MT Arhan Subari, ST,AM,Kom Yunarto,ST,MT
9	2106011006010	SONA ADRIYAN	Dr.Drs.Iwan Setiono,MSI	Drs.Subali	Drs.H. Teguh Yuvono,MKkom Ir. H.Safid Maman ,MT

Gambar 8. Tampilan modul rekap per mahasiswa

PENGUJIAN APLIKASI

Tabel 1. menunjukkan hasil pengujian yang dilakukan terhadap fungsi-fungsi yang terdapat dalam sistem administrasi kerja praktek dan tugas akhir ini.

KESIMPULAN

Kesimpulan yang dapat diambil dari penelitian ini adalah :

1. Aplikasi sistem informasi administrasi kerja praktek dan tugas akhir memiliki dua kelompok pengguna, yaitu operator dan mahasiswa. Kelompok pengguna operator dapat mengakses semua menu pada aplikasi sedangkan kelompok pengguna mahasiswa hanya dapat mengakses menu laporan.
2. Pengujian aplikasi menunjukkan bahwa seluruh fungsi yang terdapat dalam aplikasi ini dapat bekerja sesuai dengan yang diharapkan

Tabel 1. Hasil pengujian aplikasi

No	Fungsi yang diuji	Bentuk pengujian	Hasil yang diharapkan	Hasil pengujian
1	Master data dosen	Pemilihan tampilkan daftar dosen	Daftar dosen ditampilkan	Berhasil
		Pemilihan link detail data dosen	Data detail dosen ditampilkan	Berhasil
2	Master data mahasiswa	Pemilihan angkatan mahasiswa	Daftar mahasiswa sesuai dengan angkatan yang dipilih ditampilkan	Berhasil
		Pemilihan link detail data mahasiswa	Data detail mahasiswa ditampilkan	Berhasil
3	Penetapan pembimbing kerja praktek	Pemilihan angkatan dan nama mahasiswa	Ditampilkan form pengisian pembimbing kerja praktek	Berhasil
		Pengisian data kerja praktek	Data yang diisikan dapat direkam dalam form	Berhasil
		Penyimpanan data kerja praktek	Data kerja praktek tersimpan dalam database	Berhasil
4	Penetapan pembimbing tugas akhir	Pemilihan tahun dan nama mahasiswa	Ditampilkan form pengisian pembimbing tugas akhir	Berhasil
		Pengisian data tugas akhir	Data yang diisikan dapat direkam dalam form	Berhasil
		Penyimpanan data tugas akhir	Data tugas akhir tersimpan dalam database	Berhasil
5	Rekap per mahasiswa	Pemilihan tahun	Rekap mahasiswa pada tahun yang dipilih ditampilkan	Berhasil

DAFTAR PUSTAKA

- Deacon, John. 2009. **Model-View-Controller Architecture, (Online)**.
- Satish. 2004. **Model View Controller (MVC) Architecture, (Online)**.
- Burbeck, Steven. 1992. **Application Programmings in Smaltalk's 80™: How To Use MVC, (Online)**.
- Balani, Naveen. 2002. **Web services architecture using MVC style, (Online)**.
- Daqiqil, I., 2011, **Framework Codeigniter: Sebuah Panduan dan Best Practice**, Pekanbaru.
- Hidayat,A., Surarso, B., 2012, **Penerapan Arsitektur Model View Controller (MVC) Dalam Rancang Bangun Sistem Kuis Online Adaptif**, Seminar Nasional Teknologi Informasi dan Komunikasi 2012 (SENTIKA 2012) pp. 57 - 64, Yogyakarta.
- Leff, A., Rayfield, J.T., 2001, **Web-Application Development Using the Model View Controller Design Pattern**, IEEE, pp. 118-127.
- Gulzar, Nadir. 2002. **Fast track to struts: what it does and how**, (Online).
- Tanjung, M.I., 2011, **Analisis Dan Perancangan Sistem Informasi Berbasis Website Menggunakan Arsitektur MVC Dengan Framework Codeigniter**, Amikom, Yogyakarta.
- Pressman, R.S., 2001, **Software Engineering : A Practitioner's Approach**, McGraw-Hill.
- Veglis, A., 2005, **PHP and SQL Made Simple**, IEEE Distributed Systems Online, pp. 4.