

## **PENERAPAN *FRAMEWORK* DENGAN ARSITEKTUR *MODEL-VIEW-CONTROLLER* PADA PENGELOLAAN DATA PENELITIAN DAN PENGABDIAN KEPADA MASYARAKAT**

**Sumiyatun<sup>1</sup>, Femi Dwi Astuti<sup>2</sup>, Syamsu Windarti<sup>3</sup>**

<sup>1,2</sup> Program studi Teknik Informatika STMIK AKAKOM Yogyakarta

<sup>3</sup> Program studi Manajemen Informatika STMIK AKAKOM Yogyakarta

Jl. Raya Janti No.143 Banguntapan Bantul Yogyakarta 55198

[sumiyatun@akakom.ac.id](mailto:sumiyatun@akakom.ac.id)<sup>1</sup>, [femi@akakom.ac.id](mailto:femi@akakom.ac.id)<sup>2</sup>, [wiiwien@akakom.ac.id](mailto:wiiwien@akakom.ac.id)<sup>3</sup>

### **ABSTRAK**

*Teknologi web yang saat ini sedang berkembang dan mulai banyak digunakan adalah dengan menggunakan framework. Framework merupakan kumpulan script (terutama class dan function) yang dapat membantu developer/programmer dalam menangani berbagai masalah-masalah dalam pemrograman.*

*Penelitian ini mengimplementasikan framework dengan arsitektur model view controller untuk mengelola data penelitian dan pengabdian pada masyarakat di STMIK AKAKOM. Framework yang digunakan dalam penelitian ini adalah Code Igniter. Code Igniter merupakan salah satu framework yang dibangun menggunakan konsep MVC (Model-View-Controller) development pattern.*

*Aplikasi pengelolaan data penelitian dan pengabdian pada masyarakat menjadi lebih mudah dikelola, lebih terstruktur dan memiliki standar pemrograman yang jelas setelah dibangun dengan menggunakan konsep MVC.*

Kata kunci : *Code Igniter, Framework, MVC, Penelitian, Pengabdian*

### **PENDAHULUAN**

Informasi adalah data yang telah diolah menjadi bentuk yang berguna. Secara umum informasi selalu mendukung pengambilan keputusan. Untuk itu informasi harus :

- a. Benar dan bisa dibuktikan, ini berarti informasi harus akurat dan bisa dicek kebenarannya
- b. Lengkap dan padat, lengkap berarti informasi harus menyertakan semua data yang sesuai, padat berarti informasi hanya menyertakan data yang sesuai.
- c. Efektif biaya, informasi bisa didapatkan secara efisien dan informasi bisa dengan mudah dipahami.

- d. Terbaru, ini berarti informasi harus tepat waktu dan sesuai dengan kebutuhan, didasarkan pada kebutuhan informasi masa lalu, saat ini dan yang akan datang.
- e. Bisa diakses, informasi bisa diperoleh dengan cepat dan mudah.

Salah satu cara untuk menyajikan informasi adalah dengan menggunakan web. Dengan demikian informasi dapat diakses dengan mudah dan cepat. Teknologi web yang saat ini sedang berkembang dan mulai banyak digunakan adalah dengan menggunakan *framework*. Teknologi ini akan diimplementasikan dalam pengolahan data

penelitian, publikasi ilmiah dan pengabdian kepada masyarakat.

Sesuai dengan Tri Dharma Perguruan Tinggi, tugas seorang dosen ada tiga yaitu mengajar, penelitian dan pengabdian kepada masyarakat. Di STMIK Akakom penelitian dan pengabdian kepada masyarakat dikelola oleh bagian puslitbang dan PPM. Sampai saat ini belum ada sistem untuk pengolahan data tersebut, sehingga ketika data tersebut dibutuhkan tidak dapat segera disajikan.

Pada kenyataannya informasi penelitian, publikasi ilmiah dan pengabdian kepada masyarakat sering kali dibutuhkan. Misalnya

- a. Penelitian apa saja yang dihasilkan oleh dosen tertentu?
- b. Berapa persen dosen yang melakukan penelitian pada tahun ini?
- c. Siapa saja yang belum menyelesaikan penelitiannya?
- d. Judul penelitian apa saja yang sudah dipublikasikan?

Selama ini dibutuhkan waktu yang lama untuk memperoleh informasi – informasi tersebut sebab data penelitian belum diolah dengan baik. Untuk informasi publikasi ilmiah yang juga tidak disediakan. Sehingga ketika informasi tersebut dibutuhkan tidak dapat segera disajikan. Selain itu tidak ada publikasi penelitian yang diupload sehingga hasil penelitian tidak dapat dimanfaatkan oleh orang lain yang membutuhkan referensi untuk penelitian berikutnya.

Tri Dharma yang ketiga adalah pengabdian kepada masyarakat. Pengabdian kepada masyarakat dapat dilakukan secara mandiri dan dilakukan secara bersama – sama

yang dikelola oleh puslitbang dan PPM. Permintaan pengabdian kepada masyarakat yang diajukan kepada STMIK Akakom cukup banyak, sehingga perlu pengelolaan yang baik dan terjadwal agar pelaksanaan pengabdian juga berjalan lancar.

Dalam penelitian ini akan digunakan salah satu teknologi web yang mulai banyak digunakan pada saat ini yaitu *framework*. *Framework* adalah sebuah struktur konseptual dasar yang digunakan untuk memecahkan sebuah permasalahan, bahkan isu - isu kompleks yang ada. Sebuah *framework* telah berisi sekumpulan arsitektur yang dapat mempermudah dalam pemecahan sebuah permasalahan. *Framework* yang digunakan dalam penelitian ini adalah Codeigniter.

Memperhatikan permasalahan yang ada pada latar belakang masalah maka penelitian ini merumuskan bagaimana menerapkan *framework* dengan arsitektur Model view Controller pada pengelolaan data penelitian dan pengabdian masyarakat berbasis web sehingga memudahkan penyajian informasi penelitian dan pengabdian.

Mengingat luasnya permasalahan yang timbul maka diperlukan batasan untuk menghindari meluasnya masalah dalam pembahasan yaitu :

1. Aplikasi yang akan dibangun menangani proses pencatatan data penelitian, pengabdian pada masyarakat dan publikasi
2. Aplikasi tidak menangani proses pemberian dana

Tujuan utama dari penelitian ini adalah menerapkan salah satu teknologi web yaitu *framework* untuk mengelola data penelitian, publikasi dan pengabdian kepada masyarakat sehingga proses pengelolaan data menjadi lebih baik sehingga mendapatkan informasi yang akurat, relevan dan tepat waktu.

Manfaat dari penelitian ini adalah memberikan gambaran nyata tentang cara menyajikan informasi informasi tentang penelitian, publikasi dan pengabdian yang dilakukan oleh dosen di STMIK Akakom dapat diperoleh dengan mudah dan cepat melalui web

## TINJAUAN PUSTAKA

Dalam Purnomosidi, B (2010) disebutkan bahwa suatu *framework*, dalam konteks *Web Engineering* adalah sekumpulan pustaka kelas, konfigurasi, aturan-aturan, serta konvensi yang digunakan untuk mengembangkan suatu software. Biasanya *framework* ini dirancang untuk mempermudah serta menyeragamkan pekerjaan pengembangan software dengan menyediakan abstraksi serta berbagai komponen-komponen lainnya untuk mempermudah pengembangan aplikasi.

Arief, Hidayat (2012) dalam penelitiannya menerapkan arsitektur model view controller dalam rancang bangun sistem kuis online. Rancang bangun sistem ini akan lebih mudah dikembangkan jika menggunakan pola desain berarsitektur model view controller (MVC). Arsitektur ini membagi aplikasi menjadi tiga bagian secara konsep yang terpisah yaitu model, view, dan

controller, masing-masing dapat dikembangkan secara terpisah antara satu dengan yang lainnya, sehingga perubahan pada satu bagian memiliki dampak minimal pada bagian lain. Bagian model digunakan untuk mendefinisikan suatu cara dimana data dapat diakses, bagian view menghasilkan keluaran jika diberikan data, dan bagian controller menerima perintah dan mengatur aplikasi untuk tugas dan tampilan yang sesuai. Hasil dari rancang bangun ini adalah sebuah sistem penilaian siswa berdasarkan kemampuan, pengetahuan dan pilihan dari masing-masing siswa secara online.

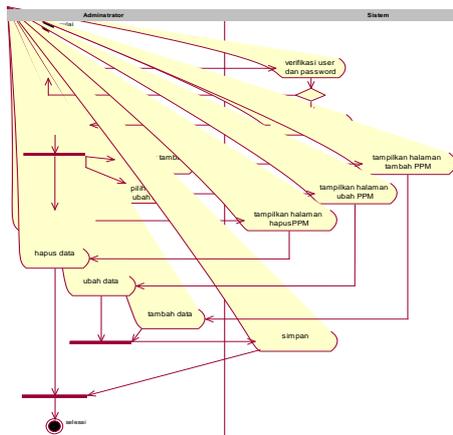
Beberapa kelebihan penggunaan *framework* ini terutama berkaitan dengan kemudahan serta fasilitas pustaka kelas dan keseragaman pengembangan aplikasi. Dalam hal pengelolaan tim, penggunaan *framework* ini akan sangat mengurangi tingkat ketidakpastian dalam tim yang bersangkutan. Meskipun demikian, penggunaan *framework* ini juga menuntut proses belajar dan adaptasi yang seringkali bukan merupakan hal yang mudah.

## METODE PENELITIAN

Metode penelitian yang digunakan pada penelitian ini adalah:

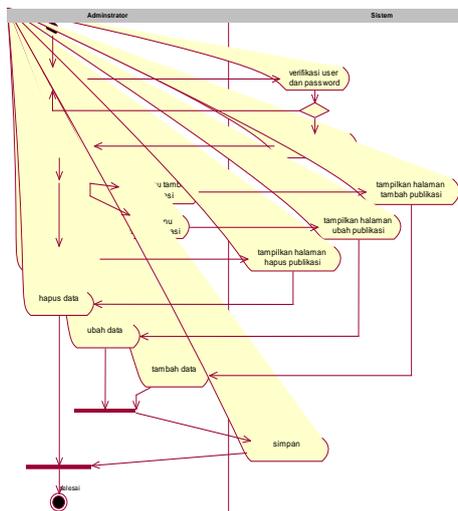
1. Melakukan studi kepustakaan yang berkaitan dengan permasalahan ini
2. Mengumpulkan data-data yang dibutuhkan
3. Metode Pembangunan Perangkat Lunak. Metode ini terdiri dari Analisis Kebutuhan Perangkat Lunak,





Gambar 4.3. Activity Diagram olah data PPM

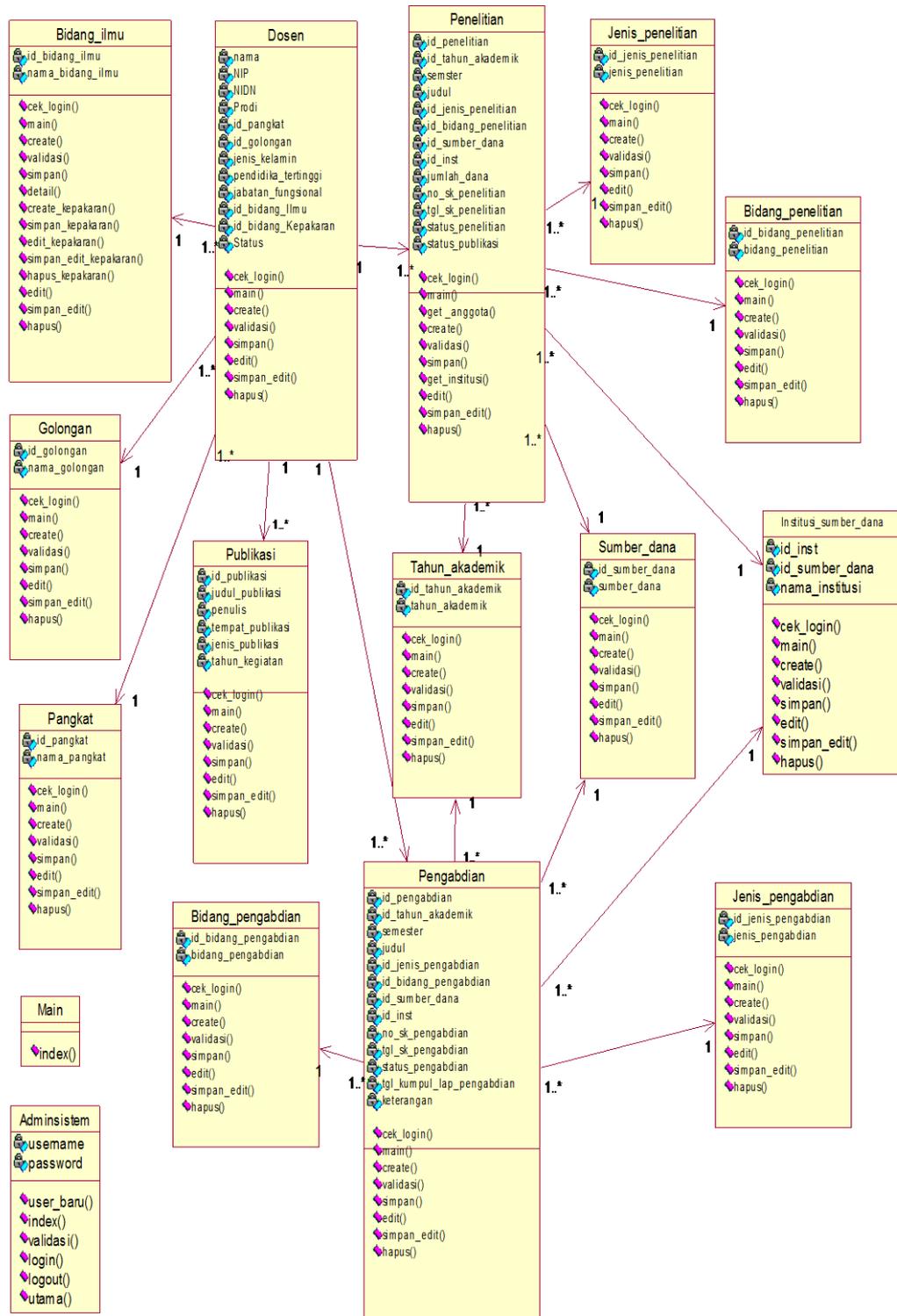
pada sistem ini dapat dilihat pada Gambar 4.5.



Gambar 4.4. Activity Diagram olah data publikasi

#### 4.1.3. Class Diagram

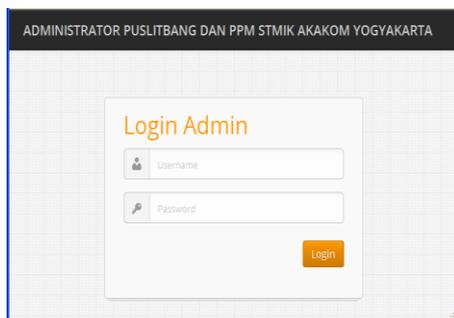
*Class* Diagram merupakan diagram yang digunakan untuk menampilkan beberapa kelas serta paket-paket yang ada dalam sistem/perangkat lunak yang sedang kita gunakan. *Class* diagram memberi gambaran (diagram statis) tentang sistem/perangkat lunak dan relasi-relasi yang ada didalamnya. *Class* Diagram



Gambar 4.5. Class Diagram

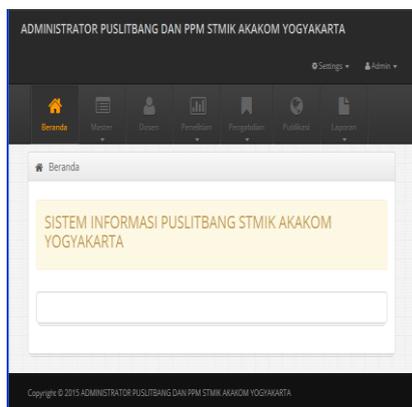
### a. Implementasi Sistem

Aplikasi ini dibuat untuk menangani proses pengelolaan data penelitian dan pengabdian dengan menerapkan model view controller. Pada saat ada pengajuan penelitian dan pengabdian, semua data harus terekam dengan baik. Berikut adalah implementasi sistem yang sudah dibangun :



Gambar 5.1. Halaman Login Administrator

Untuk dapat masuk ke sistem ini, user admin harus melakukan login terlebih dahulu dengan memasukkan username dan password. Jika admin sudah memasukkan username dan password yang sesuai maka akan muncul halaman seperti pada Gambar 5.2.



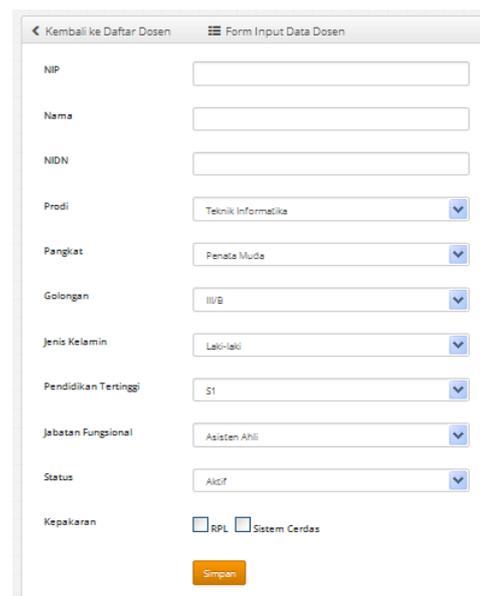
Gambar 5.2. Halaman Utama Administrator

Menu-menu yang disediakan untuk administrator meliputi : Beranda, Master, Dosen, Penelitian, Pengabdian, Publikasi, Laporan. Menu Beranda untuk menampilkan halaman utama. Menu Master bertujuan untuk melakukan pengelolaan data baik input, edit atau delete dari data master. Data master berisi Sub Menu Pangkat, Golongan, Bidang Ilmu, Sumber Dana, Institusi Sumber Dana, dan Tahun Akademik. Contoh pengelolaan data master dapat dilihat pada gambar 5.3.

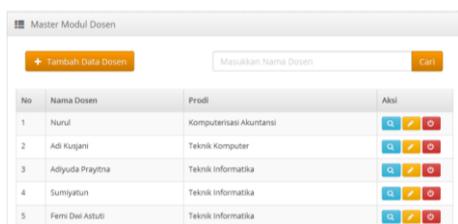


Gambar 5.3. Menu Sumber Dana

Menu Dosen digunakan untuk melakukan pengelolaan terhadap data dosen seperti terlihat pada gambar 5.4.



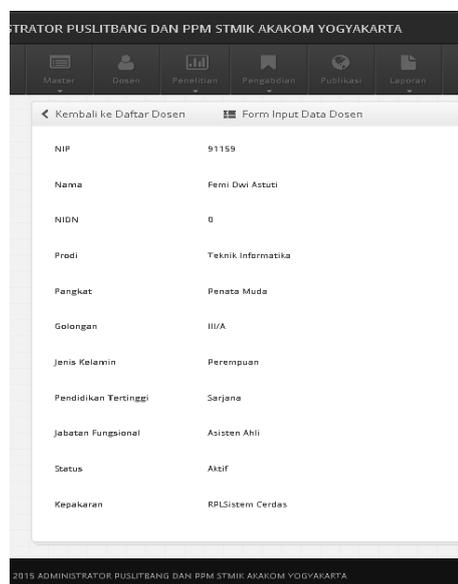
Gambar 5.4. Input Data Dosen



Gambar 5.5. Data Dosen

Gambar 5.5 menampilkan halaman olah data dosen. Dari halaman ini seorang admin dapat menambahkan data dosen dengan memilih tombol Tambah Data Dosen. Sedangkan untuk mengubah atau menghapus data dosen dapat memilih link edit atau link hapus.

Pada data dosen terdapat link berupa icon untuk melihat detail dosen. Apabila link detail dosen ini diklik maka akan menampilkan informasi data dosen sesuai id dosen yang dipilih. Tampilan detail dosen dapat dilihat pada Gambar 5.6.



Gambar 5.6. Detail dosen

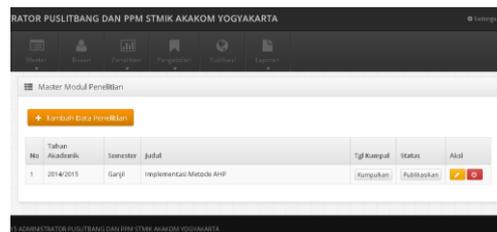
Menu penelitian bertujuan untuk mengelola data penelitian, menu ini berisi sub menu jenis penelitian, bidang penelitian dan daftar penelitian. Data jenis penelitian

terdiri dari penelitian mandiri atau kelompok. Sub menu bidang penelitian berisi data mengenai bidang-bidang penelitian apa saja yang ada, diantaranya Sistem Cerdas, RPL, Mobile dan lain-lain. Sedangkan untuk sub menu daftar penelitian berisi data rekapitulasi penelitian yang pernah dilakukan oleh dosen-dosen di STMIK AKAKOM. Data-data ini dapat ditambah, diubah maupun dihapus. Tampilan menu jenis penelitian dapat dilihat pada Gambar 5.7.



Gambar 5.7. Sub menu jenis penelitian

Daftar penelitian dapat digunakan untuk melihat seluruh penelitian yang ada beserta status penelitian maupun status publikasi. Daftar penelitian dapat dilihat pada Gambar 5.8.



Gambar 5.8. Daftar Penelitian

Pada saat dipilih Tambah Data penelitian maka akan ditampilkan form seperti pada Gambar 5.9.

STRATOR PUSLIBANG DAN PPM STMIK AKAKOM YOGYAKARTA

Kembali ke Daftar Penelitian Form Input Data Penelitian

Tahun Akademik: 2014/2015  
Semester: Ganjil  
Judul:   
Jenis Penelitian: kelompok  
Bidang penelitian: rpl  
Sumber Dana: Dalam Negeri  
Institusi Sumber Dana: Pemerintah  
Jumlah Dana:   
No SK Penelitian:   
Tgl SK Penelitian:   
Dosen (Ketua): Sumiyatun

+ Tambah Anggota  
Simpan

Gambar 5.9. Input Data Penelitian

Program simpan data penelitian dibuat dengan menggunakan arsitektur *model view controller*. Dalam program simpan data penelitian, function Simpan dibuat di bagian model. Pembuatan *function* ini berisi *query* untuk menyimpan data, sedangkan deskripsi dari *function* Simpan() ini ditulis di bagian *Controller* dengan memanggil *function* yang telah dibuat sebelumnya.

Seperti halnya menu penelitian, menu pengabdian juga bertujuan untuk mengelola data pengabdian, menu ini berisi sub menu jenis pengabdian, bidang pengabdian dan daftar pengabdian. Data jenis pengabdian terdiri dari pengabdian mandiri dan kelompok. Sub menu bidang pengabdian berisi data mengenai bidang-bidang pengabdian apa saja yang ada. Sedangkan untuk sub menu daftar pengabdian berisi data rekapitulasi pengabdian yang pernah dilakukan oleh dosen-dosen di STMIK AKAKOM. Data-data ini dapat ditambah, diubah maupun dihapus. Tampilan sub menu

jenis pengabdian dapat dilihat pada Gambar 5.10.

ADMINISTRATOR PUSLIBANG DAN PPM STMIK AKAKOM YOGYAKARTA

Materi Modul jenis Pengabdian

+ Tambah Data jenis Pengabdian

No	Jenis Pengabdian	Aksi
1	Mandiri	[ikon]

Gambar 5.10. Sub menu jenis pengabdian

STRATOR PUSLIBANG DAN PPM STMIK AKAKOM YOGYAKARTA

Materi Modul Pengabdian

+ Tambah Data Pengabdian

Tahun Akademik	Semester	Judul	Status Pengabdian	Aksi
1	2014/2015	Ganjil	Mengajar Anak-anak	Kumpulan [ikon]

Gambar 5.11. Daftar Pengabdian

Untuk menambah data pengabdian dapat di klik pada tombol Tambah Data Pengabdian. Adapun form untuk input data pengabdian seperti pada Gambar 5.12.

STRATOR PUSLIBANG DAN PPM STMIK AKAKOM YOGYAKARTA

Kembali ke Daftar Pengabdian Form Input Data Pengabdian

Tahun Akademik: 2014/2015  
Semester: Ganjil  
Judul:   
Jenis Pengabdian: Mandiri  
Bidang pengabdian: umum  
Sumber Dana: Dalam Negeri  
Institusi Sumber Dana: Pemerintah  
No SK Pengabdian:   
Tgl SK Pengabdian:   
Dosen (Ketua): Femi Dwi Astuti

+ Tambah Anggota  
Simpan

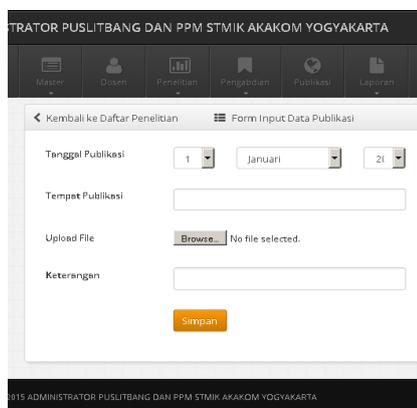
Keterangan:

Gambar 5.12. Input Data Pengabdian

Sama halnya dengan *function* simpan pada saat simpan data penelitian.

Dalam program simpan data pengabdian, *function* Simpan dibuat di bagian model. Pembuatan *function* ini berisi *query* untuk menyimpan data, sedangkan deskripsi dari *function* Simpan() ini ditulis di bagian *Controller* dengan memanggil *function* yang telah dibuat sebelumnya.

Menu publikasi berisi data-data penelitian yang telah dipublikasikan beserta dengan informasi mediadan tempat publikasinya. Penelitian yang telah dipublikasi berarti akan mengubah status penelitian menjadi telah dipublikasikan. Daftar publikasi secara otomatis muncul saat status penelitian dirubah ke sudah dipublikasikan. Untuk mengetahui waktu dan media publikasi dapat dimasukkan melalui form seperti pada Gambar 5.13.



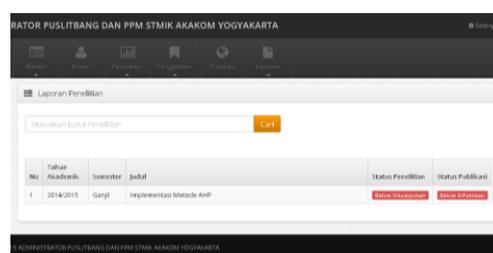
Gambar 5.13. Form Kumpul Publikasi

Setelah data pada form kumpul publikasi diisi, data publikasi akan muncul pada menu publikasi. Daftar publikasi dapat dilihat pada Gambar 5.14.

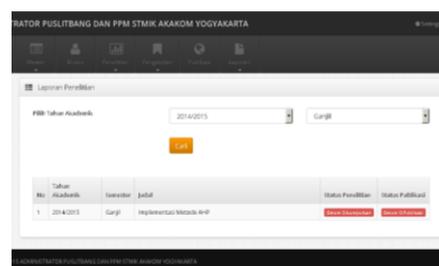


Gambar 5.14. Daftar Publikasi

Menu laporan bertujuan untuk melihat laporan penelitian, pengabdian maupun publikasi per periode, per dosen maupun semua data penelitian, pengabdian dan publikasi. Menu Laporan penelitian dapat dilihat pada Gambar 5.15. Dari laporan ini dapat dilihat juga status dari penelitian terkait dengan pengumpulan laporan dan pelaksanaan publikasi.

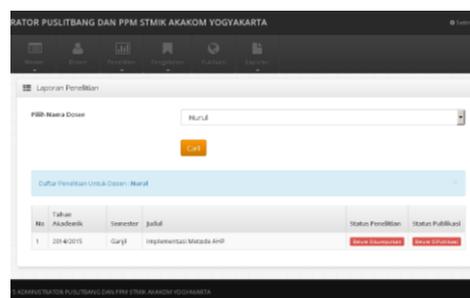


Gambar 5.15. Laporan Semua Penelitian



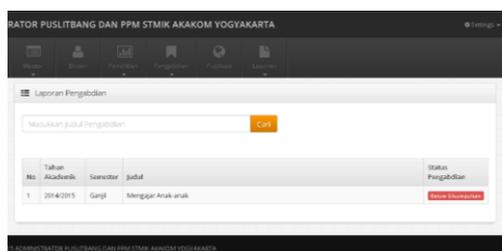
Gambar 5.16. Laporan Penelitian Per Periode

Gambar 5.16 menampilkan laporan penelitian per periode. Dari halaman ini dapat dilihat daftar penelitian pada periode tertentu, misalnya laporan penelitian pada tahun akademik 20114/2015 Ganjil.



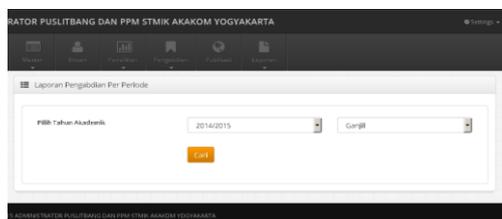
Gambar 5.17. Laporan Penelitian Per Dosen

Gambar 5.17 menampilkan laporan penelitian per dosen. Dari halaman ini dapat dilihat daftar penelitian yang dilakukan oleh seorang dosen.



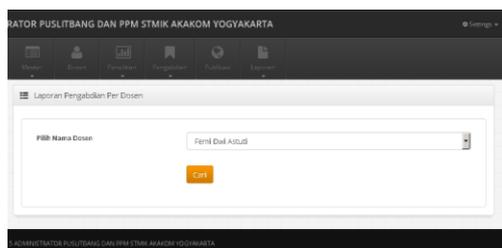
Gambar 5.18. Laporan semua pengabdian

Gambar 5.18 menampilkan laporan pengabdian secara keseluruhan yang dilakukan oleh dosen. Status pengabdian akan menampilkan data terkait dengan pengumpulan laporan. Adapun laporan yang menampilkan daftar pengabdian per periode dapat dilihat pada Gambar 5.19.



Gambar 5.19. Laporan pengabdian per periode

Gambar 5.20 menampilkan laporan pengabdian per dosen. Dari halaman ini dapat dilihat daftar pengabdian yang dilakukan oleh seorang dosen.



Gambar 5.20. Laporan pengabdian per dosen

Selain admin, aplikasi ini juga dapat diakses oleh pengunjung. Adapun informasi yang dapat diakses antara lain informasi tentang pedoman – pedoman penulisan penelitian dan pengabdian serta laporan penelitian dan pengabdian. Adapun halaman utama untuk pengunjung dapat dilihat pada Gambar 5.21.



Gambar 5.21. Halaman Utama Pengunjung

## KESIMPULAN

Setelah melalui tahap perancangan sistem dan implementasi, serta berdasarkan hasil dan pembahasan pada bab-bab sebelumnya maka dapat diambil kesimpulan :

1. Dengan mengimplementasikan MVC memungkinkan beberapa view menggunakan model yang sama sehingga komponen model sebuah aplikasi lebih mudah untuk diterapkan, diuji, dan dipelihara, karena semua akses ke model berjalan melalui komponen ini.
2. Proses pengelolaan data penelitian dan pengabdian menjadi lebih mudah dan cepat.

## DAFTAR PUSTAKA

Arief,Hidayat; 2012; *Penerapan Arsitektur Model View Controller (MVC) dalam*

*Rancang Bangun Sistem Kuis Online Adaptif, SENTIKA*

Balani,Naveen; 2002; *Web services architecture using MVC style*; (Online); (<http://www.webifysolutions.com?subject=Webservices> architecture using MVC style; diakses tanggal 1 April 2010)

Deacon,John; 2009; *Model-View-Controller Architecture*; (Online); (<http://www.jdl.co.uk/briefings/index.html/#mvc>; diakses tanggal 10 Maret 2010)

Kadir,Abdul; 2003; *Dasar Pemrograman Web Dinamis Menggunakan PHP*; Penerbit Andi; Yogyakarta

Komang,Wismakarma; 2010; *9 Langkah Menjadi Master Framework Codeigniter*; Toko Media; Yogyakarta

Purnomosidi,Bambang; 2010; *Web Engineering dalam Konteks Web Science:Isu Terkini dan Tantangan*; Prosiding SRITI; STMIK AKAKOM; Yogyakarta

Puslitbang; 2001; *Pedoman Pelaksanaan Penelitian Dosen STMIK AKAKOM*, STMIK AKAKOM; Yogyakarta

Satish; 2004; *Model View Controller (MVC) Architecture*; (Online);(<http://www.dotnetspider.com/resources/316-Model-View-Controller-MVC-architecture.aspx>; diakses tanggal 13 April 2015).

Wahana Komputer; 2011; *Mudah dan cepat membuat web dengan codeigniter*, Penerbit Andi; Yogyakarta

## BIODATA PENULIS



Sumiyatun, menyelesaikan studi S1 di Jurusan Teknik Informatika di STMIK AKAKOM. Saat ini bekerja sebagai staff pengajar di STMIK AKAKOM.

