



**RANCANG BANGUN SISTEM INFORMASI PENGARSIPAN SURAT MASUK DAN SURAT
KELUAR MENGGUNAKAN MODEL VIEW CONTROLLER
DI STMIK PARNA RAYA MANADO**

**Christian Poluakan,
Franky Manoppo, Horas Marbun**
Parna Raya Manado
Jl. Sam Ratulangi 1 No. 2-3, (0431) 850677

ABSTRAK

CHRISTIAN POLUAKAN. *Rancang Bangun Sistem Informasi Pengarsipan Surat Masuk Dan Surat Keluar Menggunakan Model View Controller Di STMIK Parna Raya Manado.* (Dibimbing oleh Hasanuddin Sirat, Rosdiana Simbolon).

Penelitian bertujuan (1) untuk memperbaiki sistem pengarsipan yang masih berupa manual dan dapat mempermudah pengerjaan dalam penginputan data surat (2) sebagai pembelajaran pembuatan sistem informasi berbentuk *Web Base* menggunakan metode pemrograman *Model View Controller*.

Penelitian ini dilaksanakan pada kampus STMIK Parna Raya Manado. Metode yang digunakan dalam penelitian ini adalah pengumpulan data dengan cara pengamatan secara langsung pada objek penelitian mengenai cara kerja kampus dalam menangani pengarsipan surat masuk dan surat keluar. Selanjutnya data-data yang telah didapat akan dilengkapi dengan data wawancara mengenai pengarsipan yang ada di kampus STMIK Parna Raya.

Dari hasil penelitian yang telah dilaksanakan dapat disimpulkan bahwa dengan adanya sistem informasi pengarsipan surat masuk dan surat keluar ini dapat membantu, mempermudah pekerjaan dalam pengarsipan pada STMIK Parna Raya Manado.

Kata Kunci : Informasi, Pengarsipan, Surat, *WebBase*, *MVC*

ABSTRACT

The summary of this research is (1) to improve the archiving system is still manual and can the facilitate progress of the data input letter (2) as a learning of information system manufacture with the Web Base form using the Model of View Controller programming method.

The research was conducted on the campus STMIK Parna Raya Manado. The method used in this study is the collection of data by direct observation on the object of research on the workings of the campus to handle archiving incoming mail and outgoing mail. Furthermore, the data that have been obtained will be equipped premises of the interview data archiving in campus STMIK Parna Raya.

From the research that has been done can be concluded that the information system archiving incoming mail and outgoing mail can help, facilitate the work in archiving the STMIK Parna Raya Manado.

Keywords: Information, Archiving, Mail, *WebBase*, *MVC*

I. PENDAHULUAN

A. Latar Belakang

Perkembangan zaman sekarang, teknologi komunikasi berkembang begitu pesat, banyak bermuncullannya berbagai alat telekomunikasi atau perhubungan yang canggih, seperti telepon, televisi, radio, dan lain sebagainya. Namun masih ada komunikasi tertulis yang tidak dapat dilupakan keberadaannya, bahkan sampai sekarang masih tetap kokoh terpakai seolah tidak bisa tergantikan oleh berbagai peralatan komunikasi

yang canggih itu, komunikasi tertulis tersebut adalah surat. Namun masih banyak ditemukan dalam suatu instansi, perusahaan, atau universitas yang melakukan berbagai kesalahan dalam proses pengelolaan surat atau data-data penting yang ada. Seperti ditemukannya ada data atau surat yang tercecer ataupun rusak, sehingga dapat menyebabkan kerugian bagi perusahaan atau universitas tersebut.

Proses pengelolaan dokumen surat masuk dan surat keluar ini merupakan suatu pengurusan surat penting yang harus dilakukan

oleh suatu organisasi, dan proses dokumen tersebut dapat berbeda bagi setiap instansi. Pengurusan surat masuk dan surat keluar harus mendapatkan perhatian yang sungguh, karena isi dari surat pada perusahaan, instansi atau universitas akan menjadi sarana pencapaian tujuan organisasi atau universitas yang bersangkutan, maka dari itu perlu adanya pengolahan surat

Seperti halnya pada kampus Sekolah Tinggi Manajemen Informatika Komputer (STMIK) Parna Raya Manado masih memerlukan penanganan prosedur tata persuratan yang baik, untuk itu penulis membuat program untuk mempermudah dalam proses pengarsipan surat, dengan adanya program tersebut proses surat di kampus STMIK Parna Raya Manado akan lebih mudah dan dalam pencarian surat nantinya akan lebih cepat dari sebelumnya yang masih manual. Dalam pembuatan program, penulis membuat program yang dijalankan di *browser* atau disebut dengan (*WEB based*), dengan menggunakan teknik Pemrograman *Model View Controller (MVC)*, teknik pemrograman *MVC* tersebut dibagikan menjadi 3 bagian yaitu

1. *Model*

Objek *Model* adalah bagian dari aplikasi yang mengimplementasikan logika untuk domain data aplikasi. Umumnya, objek *Model* digunakan untuk mengambil data dari *database* atau menyimpan data ke dalam *database*.

2. *View*

View adalah komponen yang menampilkan antarmuka untuk pengguna (*User Interface, UI*) aplikasi. Antarmuka ini dibuat berdasarkan data dari *Model*.

3. *Controller*

Controller merupakan komponen yang digunakan untuk menangani interaksi pengguna, bekerja dengan *Model*, dan memilih *View* mana yang digunakan untuk merender data.

Dunia pemrograman saat ini, baik pemrograman *desktop* maupun *WEB based*, sudah semakin ramai pengerjaannya dengan menggunakan *framework*. *Framework* memang dikembangkan untuk mempermudah dalam *developing* suatu aplikasi. Selain itu, dengan *framework*, waktu yang dihabiskan untuk membangun aplikasi dapat dikurangi secara signifikan. (Upton David 2007)

Salah satu *framework* yang populer untuk pemrograman *WEB based* adalah *CodeIgniter*. *CI (CodeIgniter) Framework* merupakan sebuah *framework* berbasis *PHP* yang mengorganisasi *file* menjadi 3 (tiga) kelompok, yaitu *model* yang merupakan kelompok *file* yang mengatur konfigurasi *database*, *View* yang merupakan kelompok *file* yang menghubungkan *file-file Model* dengan *file-file View*. (Myer Thomas, 2008)

Metode *Model-View-Controller (MVC)* ini, pengorganisasian *file* dalam membangun suatu aplikasi akan lebih tertata rapi, sehingga meningkatkan stabilitas dan efisiensi dalam proses pemeliharaan dan pengembangan aplikasi demi memenuhi perkembangan kebutuhan dalam mencari solusi masalah yang ditemui di kemudian hari. Misalnya ketika diperlukannya pengembangan dari segi tampilan, seseorang *programmer* cukup mengubah *file* yang ada dalam kelompok *file View*, tanpa perlu mengubah *file* pada *Model* ataupun *Controller*. Begitu pula ketika mengubah *file* pada *Controller* atau *Model*.

Pemisahan ini, maka pembuatan satu program dapat dibagi-bagi kepada beberapa orang. Setiap bagian atau komponen yang dibuat secara terpisah diharapkan dapat digunakan tidak hanya oleh satu program saja, tetapi oleh banyak program, terutama bagian *Model* dan *View*. (Betha Sidik, 2012)

Berdasarkan pengamatan yang dilakukan pada kampus STMIK Parna Raya Manado, perlu adanya suatu sistem aplikasi yang dapat membantu dalam proses penanganan surat, dalam hal ini penulis melakukan penelitian dengan judul : "Rancang Bangun Sistem Informasi Pengarsipan Surat Masuk Dan Surat Keluar menggunakan *Model View Controller* di STMIK Parna Raya Manado". Dengan adanya sistem yang baik penanganan dan pengolahan data akan lebih mudah.

B. Rumusan Masalah

Rumusan masalah yang diangkat penulis berdasarkan uraian latar belakang diatas, maka diambil perumusan permasalahannya sebagai berikut :



1. Bagaimana mengelola sistem informasi pengarsipan menggunakan *Model View Controller* berbasis *web based* pada STMIK Parna Raya Manado.
2. Bagaimana membangun sistem pengarsipan dengan dasar *WEB Base* dengan metode pemrograman *Model View Controller*

C. Tujuan Penelitian

Tujuan penulis melakukan penelitian adalah sebagai berikut :

1. Untuk memperbaiki sistem pengarsipan yang masih berupa manual dan dapat mempermudah pengerjaan dalam penginputan data surat.
2. Sebagai pembelajaran dalam pembuatan sistem informasi berbentuk *WEB Base* dengan metode pemrograman *Model View Controller*

D. Manfaat Penelitian

Adanya manfaat dari penelitian ini adalah untuk membantu baik dari sisi penulis maupun objek penelitian yang akan dilaksanakan antara lain :

1. Agar dapat meningkatkan efisiensi waktu untuk pengelolaan dokumen dan pengarsipan.
2. Agar dapat mempercepat proses pengarsipan dan sebagai *bakup* arsip yang sangat efektif

E. Batasan Masalah

Agar memperoleh penelitian yang maksimal dan terfokus dalam rancangan pembuatan pengarsipan, maka perlu adanya batasan penelitian, yaitu :

1. Data surat yang akan dimasukan dalam sistem informasi pengarsipan adalah :
 - a. Surat Masuk
Surat yang masuk di kampus STMIK Parna Raya di arsip secara umum atau dalam satu buku besar.
 - b. Surat Keluar
Surat yang keluar dari kampus STMIK Parna Raya di arsip menjadi 3 Bagian yaitu :
 - 1) Surat Umum
 - 2) Surat Internal
 - 3) Surat Eksternal
2. Pencarian berkas surat berdasarkan perihal / isi ringkas surat
3. Input data surat masuk :
 - a. Tanggal surat

- b. Nomor surat
- c. Pengirim
- d. Perihal
- e. Jenis surat
- f. Kode surat
4. Input data surat keluar :
 - a. Tanggal surat
 - b. Nomor surat
 - c. Dikirim kepada
 - d. Tanggal keluar surat
 - e. Perihal
 - f. Jenis surat
 - g. Kode surat
5. Pembuatan aplikasi surat masuk dan surat keluar menggunakan *PHP*
6. Pembuatan *database* sistem aplikasi surat masuk dan surat keluar dengan menggunakan *MySQL*
7. Pengguna program harus terkoneksi dengan jaringan internet

II. TINJAUAN PUSTAKA

A. Sistem Informasi

Sistem sangatlah dibutuhkan dalam suatu perusahaan atau instansi pemerintahan, karena sistem sangatlah menunjang terhadap kinerja perusahaan atau instansi pemerintah, baik yang berskala kecil maupun besar. Supaya dapat berjalan dengan baik, dan di perlukan kerjasama di antara unsur-unsur yang terkait dalam sistem tersebut.

Sistem informasi merupakan sebuah proses yang menjalankan fungsi mengumpulkan data (*input*), memproses, menyimpan, menganalisis, dan menyebarkan informasi (*output*) untuk kepentingan tertentu. Sederhananya, sebuah sistem informasi akan menerima *input* berupa data, data ini akan di simpan (*storage*) dan kemudian diproses untuk menghasilkan sebuah output (Soeherman. Bonnie Dkk, 2010).

Sistem informasi suatu sistem didalam organisasi yang mempersatukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat menajerial, dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang di perlukan (Jack Febrian & Farida Andayani, 2008).

Sistem informasi merupakan suatu kombinasi dari orang – orang, fasilitas, teknologi, media, prosedur – prosedur, dan pengendalian yang ditujukan untuk mendapat jalur komunikasi penting, memproses tipe

transaksi rutin tertentu, memberi sinyal kepada manajemen dan yang lainnya terhadap kejadian – kejadian internal dan eksternal dan menyediakan suatu dasar informasi untuk pengambilan keputusan yang baik (Agus Mulyono, 2009).

Sistem informasi adalah “ suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan – laporan yang diperlukan. Nilai informasi ditentukan dari dua hal yaitu manfaat dan biaya mendapatkannya. Suatu informasi dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya. (Jogiyanto, 2009).

B. Pengarsipan

Pengarsipan merupakan proses, cara, atau perbuatan mengarsip. Istilah mengarsip, dalam hal ini, merujuk pada kegiatan penyimpanan berbagai catatan atau dokumen dengan menggunakan sistem dan tatacara tertentu sehingga, jika sewaktu-waktu diperlukan, catatan atau dokumen itu dengan mudah dapat ditemukan kembali. Sehubungan dengan hal itu, arsip merupakan suatu hal yang cukup penting karena sistem pengarsipan yang baik dapat menunjang suatu organisasi, instansi, atau perusahaan yang bersangkutan (Arifin Zaenal 2008).

Surat masuk adalah surat yang masuk ke dalam suatu instansi/perusahaan atau bagian dalam suatu instansi/perusahaan, baik yang berasal dari instansi/perusahaan lain atau dari bagian lain pada instansi/perusahaan yang sama. Dengan demikian surat masuk dapat berasal dari pihak ekstern maupun pihak intern instansi/perusahaan tersebut. Ada beberapa contoh kegiatan dalam penanganan surat masuk :

1. Surat penting
Menahan kartu kendali pertama sebagai pengganti buku agenda, menyampaikan surat beserta kartu kendali kedua dan ketiga kepada tata usaha atau unit pengolah, menerima kartu kendali kedua setelah diparaf sebagai tanda terima.
2. Surat biasa
Menyampaikan surat beserta dua lembar pengantar kepada tata usaha atau unit pengolah, menerima lembar

pengantar kedua setelah diparaf sebagai tanda terima.

3. Surat rahasia

Menyampaikan surat dalam keadaan tertutup beserta dua lembar pengantar kepada tata usaha atau unit pengolah, menerima lembar pengantar kedua setelah diparaf sebagai tanda terima.

Surat keluar adalah surat yang dikirim oleh suatu instansi/perusahaan atau antar bagian dalam instansi/perusahaan tersebut, ditunjukan kepada instansi/perusahaan lain atau ke bagian lain dalam instansi/perusahaan yang sama. Dengan demikian surat ekstern dapat ditunjukan ke pihak ekstern maupun intern dalam instansi/perusahaan tersebut. Ada beberapa contoh kegiatan dalam penanganan surat keluar : (Kanisius, 2008)

1. Penyampaian surat penting
Menyampaikan surat asli, pertinggal serta kartu kendali pertama dan kedua kepada unit kearsipan. Menerima dan menyampaikan pertinggal surat keluar setelah distempel tanggal dan jam pengiriman oleh unit kearsipan.
2. Penyampaian surat biasa
Menyampaikan surat biasa, pertinggal, dan lembar pengantar pertama dan kedua kepada unit kearsipan.
3. Penyampaian surat rahasia
Menyampaikan surat rahasia dalam keadaan tertutup dan menyampaikan lembar pengantar pertama dan kedua kepada unit kearsipan. Menyimpan lembar pengantar pertama. Menyimpan kartu kendali pertama, kedua, dan lembar pengantar kedua.

C. Pemrograman WEB

Membangun sebuah aplikasi WEB, diperlukan bahasa pemrograman yang dapat mendukung pembuatan aplikasi tersebut, berikut ini merupakan penjelasan secara singkat mengenai bahasa-bahasa pemrograman yang akan penulis gunakan :

1. **HTML (Hyper Text Markup Language)**
HTML merupakan salah satu elemen dasar yang akan penulis pakai dalam membangun sistem informasi ini. HTML (Hyper Text Markup Language) merupakan sebuah protocol WEB yang

akan digunakan untuk membuat format suatu dokumen *WEB* yang mampu dibaca dalam *browser* dari berbagai *platform* komputer. Dokumen *HTML* berbentuk *file* yang ada pada umumnya berekstensi *.htm* atau *.HTML*, dimana bahasa *HTML* tersebut tersusun atas *tag* yang berformat *<isi tag>*. (Christopher L, 2011)

Dokumen *HTML* merupakan *file* teks murni yang dapat dibuat dengan sembarang editor teks. *HTML* dewasa ini dikenal sebagai bahasa standar untuk membuat dokumen *WEB* yang sebenarnya *HTML* justru tidak dapat dibuat untuk mempublikasikan informasi pada *WEB* namun oleh karena itu kesederhanaan serta kemudahan penggunaannya, *HTML* kemudian dipilih orang untuk mendistribusikan informasi di *WEB*. (Jubilee E, 2011)

2. **PHP (Pemrograman Hypertext Preprocessor)**

PHP menurut dokumen resmi merupakan *Hypertext Preprocessor* dan bahasa berbentuk *script* yang ditempatkan dalam *server* dan diproses di *server* serta hasilnya dikirimkan ke klien, tempat pemakai menggunakan *browser*, secara khusus *PHP* dirancang untuk membentuk aplikasi *WEB* dinamis. Artinya, ia dapat membentuk suatu tampilan berdasarkan permintaan terkini. (Abdul Kadir, 2008)

D. Basis Data

Database memiliki beberapa *Model* diantaranya *Model* relasional. Dalam *Model* table-tabel yang terdapat dalam suatu *database* idealnya harus saling berelasi. (Budi Raharjo, 2011)

1. **MY SQL (My Structured Query Language)**

MySQL (My Structured Query Language) merupakan salah satu *software database* (basis data) *open source* yang dikembangkan sebuah komunitas bernama *MySQL AB* dengan tujuan membantu *user* untuk menyimpan data dalam table-tabel. Table terdiri atas *field* (kolom) yang mengelompokan data-data berdasarkan kategori tertentu, misalnya nama, alamat, nomor telepon, dan sebagainya. Bagian dari table adalah

record (baris) yang mencantumkan isi data yang sebenarnya. (Agus S, 2011)

MySQL banyak dipakai untuk kepentingan penanganan *database* karena handal dalam penanganan data-data. (Abdul Kadir, 2010)

2. **XAMPP (X Apache MySQL PHP Perl)**

XAMPP termasuk paket *PHP* yang berbasis *open source* yang dikembangkan oleh sebuah komunitas *open source*. Dengan menggunakan *XAMPP* kita tidak dapat melakukan penginstalan program-program yang lain, karena semua kebutuhan telah disediakan oleh *XAMPP*. Berikut ini adalah beberapa paket yang telah disediakan :

- a. Apache
- b. MySQL
- c. PHP
- d. FileZilla FTP Server 0.9 Beta
- e. PHPMyAdmin 2.6.1

XAMPP memiliki beberapa ikon terpisah untuk menjalankan program pendukungnya. *XAMPP* telah menyediakan satu ikon yang dapat digunakan untuk menjalankan (mengaktifkan) semua *service* yang telah terinstal. (Nugroho B, 2010)

E. Framework

Framework adalah kumpulan perintah atau fungsi dasar membentuk aturan-aturan tertentu saling berinteraksi satu sama lain sehingga dalam pembuatan aplikasi *WEB* kita harus mengikuti aturan dari *framework* tersebut.

Dengan *framework* (dalam hal ini *framework PHP*), kita tidak perlu memikirkan kode perintah atau fungsi dasar dari aplikasi *WEBSITE* kita. (Wardana, S.Hut. M.Si, 2010)

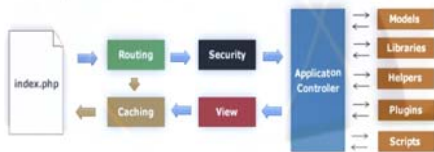
Pemrograman *WEB*, *framework* telah dikembangkan untuk bahasa pemrograman antara lain *PHP* dan *Java*. Untuk *PHP*, *framework* yang banyak digunakan misalnya *Zend framework* yang dikembangkan oleh *Zend Technologies*, *CodeIgniter* yang dikembangkan oleh *Ellislab, Inc*, dan *Symfony Framework* yang dikembangkan oleh *SesioLabs*. (Zaninotto, Francios and Potencier, Fabien 2010).

Bagaimana suatu proses data mengalir pada suatu sistem yang menggunakan *CodeIgniter Framework* dapat diilustrasikan pada gambar berikut : (Basuki Awan P, 2010)

F. CodeIgniter Framework

CodeIgniter adalah *framework* aplikasi WEB yang *open source* untuk bahasa pemrograman PHP. CodeIgniter memiliki banyak fitur yang membuatnya berbeda dengan *framework* lainnya. Tidak seperti beberapa *framework PHP* lainnya, dokumentasi untuk *framework* ini sangat lengkap, yang mencakup seluruh aspek dalam *framework*.

CodeIgniter juga mampu berjalan pada lingkungan *shared hosting* karena memiliki ukuran yang sangat kecil, namun memiliki kinerja yang sangat luar biasa. (Griffiths Adam, 2010).



Gambar 1. Application Flow Chart
CodeIgniter Framework

Keterangan :

1. **Index.PHP** berfungsi sebagai *front Controller*, menginisialisasi *base resource* untuk menjalankan CodeIgniter.
2. **Router** memeriksa *HTTP request* untuk menentukan apa yang harus dilakukan dengannya.
3. Jika **Chace** aktif, maka hasilnya akan langsung dikirimkan ke *browser* dengan mengabaikan aliran data normal.
4. **Security**. Sebelum *Controller* dimuat, *HTTP request* dan data yang dikirimkan *user* akan difilter untuk keamanan.
5. **Controller** memuat *Model*, *code libraries*, *plugins*, *helpers*, dan semua *resource* yang diperlukan untuk memproses *request*.
6. Akhirnya **View** yang dihasilkan akan dikirim ke *browser*. Jika *cache* aktif, maka *View* akan disimpan sebagai *cache* dahulu, sehingga pada *request* berikutnya langsung ditampilkan.

G. Model View Controller (MVC)

MVC merupakan sebuah *pattern* atau teknik pemrograman yang memisahkan antar pengembang aplikasi berdasarkan komponen utama pada sebuah aplikasi, seperti manipulasi data, *user interface* dan bagian yang menjadi control aplikasi.

Secara sederhana dapat dikatakan bahwa antara desain dan proses data berada pada tempat terpisah. Dan *Model View Controller*

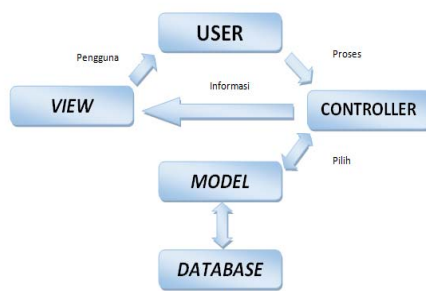
merupakan suatu konsep yang populer dalam pembangunan aplikasi WEB, yang berawal pada bahasa *small talk*. (YM.Kusuma Ardhana, 2013)

MVC merupakan pemisahan tampilan pengguna dari Kendali asupan pengguna *Model* informasi yang mendasarinya. Untuk hal tersebut, ada beberapa alasan pokok mengapa *Model MVC* menjadi sangat bermanfaat, yaitu :

1. Penggunaan ulang komponen-komponen antarmuka pengguna (*user interface reusable component*).
2. Kemampuan untuk mengembangkan aplikasi dengan antar muka pengguna secara terpisah.
3. Kemampuan untuk melakukan pewarisan (*inheritance*) dari berbagai bagian yang berbeda pada suatu hierarki kelas.
4. Kemampuan untuk mendefinisikan kelas-kelas pengaturan tampilan (*control style*) yang menyediakan fitur-fitur umum secara terpisah dengan bagaimana fitur-fitur itu akan ditampilkan oleh aplikasi yang di kembangkan.

Hal-hal yang disebutkan diatas mempunyai arti bahwa berbagai antarmuka yang berbeda dapat digunakan dalam aplikasi yang sama, tanpa aplikasi yang bersangkutan perlu mengetahuinya (Nugroho Adi, 2009).

Model View Controller (MVC) merupakan pola pemrograman yang bertujuan memisahkan logika bisnis, logika data dan logika tampilan (*interface*), atau secara sederhana memisahkan antara proses, data dan tampilan. MVC mengatur arsitektur sebuah aplikasi. Umumnya aplikasi yang dibangun dengan konsep MVC adalah aplikasi yang cukup besar, karena salah satu keuntungan dari MVC adalah kemudahan *maintenance*, dan pengembangan aplikasi tersebut. Dan harus memisahkan kode *database* ke folder *Model*, kode proses ke folder *Controller* dan kode tampilan ke folder *View*.



Gambar. 2

Hubungan Model View Controller (Zuchdi Darmiyanti, 2010)

Berdasarkan gambar diatas dapat kita ketahui ketika datang sebuah permintaan *user*, maka permintaan tersebut akan ditangani oleh *Controller*, kemudian *Controller* akan memanggil *Model* jika memang diperlukan operasi *database*. Hasil *query* oleh *Model* kemudian akan dikembalikan ke *Controller*. Selanjutnya *Controller* akan memanggil *View* yang tepat dan mengkombinasikannya dengan hasil *query Model*. Hasil akhir dari operasi ini akan ditampilkan ke *browser* yang selanjutnya bisa dilihat oleh *user*.

View, *file-file* yang ditempatkan pada bagian ini bertanggung jawab untuk menunjukkan data kami pada para pengunjung situs kita, atau pengguna dari aplikasi kita. Tidak ada logika pemrograman, tidak ada *query insert* atau *update* yang harus dijalankan disini, meskipun akses data bisa terjadi pada *file-file* ini. *File-file* disini hanya untuk menunjukkan hasil dari dua bagian lainnya. Jadi kita mengambil data dalam *Model*, dan menampilkan dalam *View*. (Blanco, Jose A. And Upton, David, 2009)

Model merupakan komponen MVC yang bertugas mengambil data dari *database* dan juga memasukan data ke *database*. Isi utamanya berupa perintah SQL. Hasilnya dikirim ke *Controller*.

Controller merupakan komponen MVC yang bertugas mengirim perintah ke *Model* untuk mendapatkan data yang di inginkan. *Controller* tidak mengetahui bagaimana data tersebut diambil dari *database*, karena *Controller* tidak berisi kode perintah SQL. Karena itu adalah tugas modal. *Controller* mengolah data dari inputan unser dan data dari modal kemudian data olah tersebut di kirim ke *View* untuk ditampilkan sesuai aturan *Controller*. *Controller* merupakan penghubung

antara *Model* dan *View* dan mengatur hubungan tersebut (Zuchdi Darmiyanti, 2010).

H. Printer



(sumber, Google image)

Gambar 3. Printer

Munculnya perangkat keras bernamakan printer di dunia teknologi memang sudah menghadirkan banyak manfaat. Perangkat mesin ketik yang sejak puluhan tahun telah tersisi. Karena kecanggihannya perangkat printer yang memiliki fungsi sebagaimana manusia-manusia di dunia saat ini inginkan untuk mencetak suatu dokumen untuk kepentingan pribadi maupun banyak orang, mulai dari teks, gambar serta gabungan teks dan gambar.

Printer merupakan perangkat keras (*hardware*) dimana perangkat itu akan bekerja apabila pengguna menghubungkannya dengan perangkat komputer, yang bisa digunakan untuk keperluan mencetak tulisan, gambar, dan grafik, kedalam bentuk kertas atau sejenisnya. Printer itu sendiri saat ini sering digunakan untuk mencetak dokumen penting baik itu perusahaan ataupun organisasi universitas dan lain sebagainya, dengan demikian kehadiran printer tentu saja sangat membantu keseharian para pekerja kantoran dan pelajar serta masyarakat lainnya yang memiliki keperluan mencetak suatu dokumen. (Anonymous, 2014)

Fungsi dari printer tersebut untuk mencetak tulisan, gambar, grafik dan lainnya dari komputer ke media kertas atau sejenisnya. Untuk program sistem informasi pengarsipan yang penulis buat, printer ini sangatlah penting dalam proses pengarsipan surat.

I. Scanner



(sumber, Google image)

Gambar 4. Scanner

Scanner merupakan sebuah alat yang dapat berfungsi untuk mengcopy atau menyalin gambar kedalam bentuk digital yang kemudian disimpan kedalam memori komputer, dari memori komputer selanjutnya disimpan kedalam harddisk komputer. Fungsi scanner ini mirip seperti mesin fotocopy dan perbedaannya adalah mesin hasilnya dapat dilihat pada kertas sedangkan scanner hasilnya yang sudah menjadi kedalam bentuk digital dapat ditampilkan melalui monitor terlebih dahulu sehingga kita dapat melakukan perbaikan atau modifikasi dan kemudian dapat disimpan kembali baik dalam bentuk teks maupun file gambar. (Anonymous, 2014)

Fungsi dari scanner ini untuk pengarsipan surat yang ada pada kampus STMIK ini adalah untuk menyalin teks dokumen surat kedalam bentuk data digital dan kemudian menjadi file gambar yang nantinya akan diupload kedalam program sistem informasi pengarsipan yang penulis buat yang ada di Website www.arsip.parnaraya.ac.id untuk selanjutnya akan diarsip.

J. Komputer



(sumber, Google image)

Gambar 5. Komputer

Komputer merupakan alat untuk mempercepat proses dalam suatu pekerjaan dengan komputer banyak pekerjaan yang dapat di selesaikan dengan mudah. Dan dengan adanya komputer sistem informasi pengarsipan

surat yang ada pada kampus STMIK Parna Raya Manado dapat mempercepat proses dalam pekerjaan pengarsipan.

Fungsi dari komputer ini untuk menjalankan program sistem informasi pengarsipan yang penulis buat yang ada di website www.arsip.parnaraya.ac.id yang terkoneksi dengan internet akses. Dan juga untuk mengolah data pengarsipan perlu adanya scanner dan printer. Printer untuk mencetak dokumen surat dan scanner untuk menyalin teks dokumen menjadi kedalam bentuk gambar yang nantinya akan akan diupload di www.arsip.parnaraya.ac.id sebagai arsip surat.

K. Sistem Pengarsipan Di STMIK Parna Raya Manado

Sekolah Tinggi Manajemen Ilmu Komputer (STMIK) Parna Raya Manado adalah suatu perguruan tinggi swasta yang bergerak dalam bidang pendidikan komputer di bawah naungan Yayasan Buana Parna. Dengan memiliki Visi "Menjadi perguruan tinggi yang unggul di Sulawesi Utara dan terkemuka di bidang teknologi informasi" dan Misi "Menyelenggarakan pendidikan berbasis teknologi informasi yang bermutu tinggi dan menghasilkan lulusan yang professional, disiplin, jujur dan bertaqwa kepada Tuhan Yang Maha Esa, untuk mencapai kualitas Nasional.

Informasi menjadi bagian yang sangat penting untuk mendukung proses kerja administrasi kampus di STMIK Parna Raya Manado, dan salah satu sumber informasi yang penting supaya dapat menunjang proses kegiatan administrasi adalah arsip. Arsip berfungsi sebagai pusat ingatan, alat bantu pengambilan keputusan, bukti eksistensi organisasi dan untuk kepentingan kampus STMIK Parna Raya Manado, berdasarkan fungsi arsip yang sangat penting tersebut maka harus mempunyai manajemen atau pengelolaan arsip yang baik.

Pengarsipan data surat yang menjadi objek penelitian yang diambil dari kampus STMIK Parna Raya Manado antara lain :

1. Surat Masuk
Surat Masuk di arsip secara umum
2. Surat Keluar
Surat Keluar di arsip menjadi 3 bagian yaitu
 - a. Surat Umum
 - b. Surat Internal
 - c. Surat Eksternal

Pengarsipan surat yang manual yaitu data surat yang masuk dan data surat yang keluar diarsipkan dalam empat buku besar masih mempersulit pekerja untuk pengarsipan.

III. METODOLOGI PENELITIAN

A. Tempat Dan Waktu Penelitian

1. Lokasi

Tempat penelitian dilakukan di STMIK Parna Raya Manado yang beralamat di jalan Sam Ratulangi 1 No. 2-3 Manado Sulawesi Utara.



(Sumber, google map)

(Gambar 6. Lokasi STMIK Parna Raya Manado)

B. Metode Penelitian

Teknik pengumpulan data yang penulis lakukan pada penelitian ini adalah Observasi, Wawancara, Studi pustaka, Penelitian laboratorium, sebagai berikut :

1. Observasi

Pembuatan suatu tulisan ilmiah maka diperlukan pengumpulan data bahan-bahan dengan membaca buku, laporan-laporan, arsip-arsip dan sebagainya. Dalam hal ini penulis mencari data dan informasi mengenai komponen-komponen dasar yang akan digunakan dalam merancang subjek (alat) yang dimaksud serta hal-hal lainnya yang mungkin dapat membantu.

2. Wawancara

Penulis mewawancarai pihak yang berwenang dari STMIK Parna Raya Manado untuk mendapatkan penjelasan dan data tentang pengarsipan.

3. Studi Pustaka

Metode kepustakaan merupakan metode yang digunakan penulis untuk memperoleh data dengan cara membaca dan meneliti dengan sejumlah buku, jurnal, artikel, majalah yang berkaitan

dan mendukung penelitian, wawancara dan juga referensi dari internet.

4. Penelitian laboratorium

Penulis melakukan uji pakai terhadap program untuk sistem informasi pengarsipan di STMIK Parna Raya Manado agar supaya bekerja sesuai perencanaan tujuan dari sistem.

C. Alat dan Bahan

Alat dan bahan yang peneliti gunakan dengan pemrograman *Model View Controller*, sebagai berikut :

1. Alat

a. Hardware (Perangkat Keras)

- 1) Komputer
- 2) Printer
- 3) Scan

b. Software (Perangkat Lunak)

- 1) Dreamweaver
- 2) MySQL
- 3) Notepad ++
- 4) XAMPP

2. Bahan

Bahan yang diperlukan dalam penelitian ini adalah data pengarsipan STMIK Parna Raya Manado. Pentingnya data tersebut sangat mempengaruhi program yang akan di buat.

IV. HASIL DAN PEMBAHASAN

A. Hasil Penelitian

Dengan adanya program sistem informasi pengarsipan yang dibuat penulis pada kampus STMIK Parna Raya Manado sangat membantu dalam pengarsipan yang dilakukan di STMIK Parna Raya Manado dan mempercepat dalam proses pengarsipan.

B. Perancangan Database

Tahap ini penulis membahas tentang *database system* mulai pada saat membuat *database*, *table* dan *field-field* yang ada dalam *table* yang berhubungan dengan sistem informasi pengarsipan yang penulis buat, dan tahap ini dijelaskan pula mengenai *table-tabel* dalam *database* sebagai berikut :



Tabel 1. Tabel Admin

Field	Type	Collation	Null	Default	Extra
Id	Int(2)		No	None	Auto_increment
Username	Var(15)	Latin1_swedish_ci	No	None	
Password	Var(75)	Latin1_swedish_ci	No	None	
Nama	Var(15)	Latin1_swedish_ci	No	None	
Nip	Var(25)	Latin1_swedish_ci	No	None	
Level	Enum('super admin', 'admin')	Latin1_swedish_ci			

(Sumber, perancangan & implementasi)

Tabel Admin berfungsi untuk menginput data admin, contoh admin yang akan dibuat lebih dari satu.

Tabel 2. Tabel Disposisi

Field	Type	Collation	Null	Default	Extra
Id	Int(6)		No	None	Auto increment
Id_surat	Int(6)		No	None	
Kpd_yth	Varchar(250)	Latin1_swedish_ci	No	None	
Isi_disposisi	Varchar(250)	Latin1_swedish_ci	No	None	
sifat	enum('Biasa', 'Segera', 'Perlu Perhatian Khusus', 'Perhatian Batas Waktu')	Latin1_swedish_ci	No	None	
Batas waktu	Date	Latin1_swedish_ci	No	None	
catatan	Varchar(250)	Latin1_swedish_ci	No	None	

(sumber, perancangan & implementasi)

Tabel disposisi berfungsi untuk menginput data surat yang masuk dan kemudian diberikan kepada yang bersangkutan dan diarsip kembali oleh admin.

Tabel 3. Tabel Surat Keluar

Field	Type	Collation	Null	Default	Extra
Id	Int(6)		No	None	Auto_increment
Kode	Varchar(20)		No	None	
no_agenda	Varchar(100)		No	None	
Isi_ringkas	Mediumtext		No	None	
tujuan	Varchar(250)		No	None	
no_surat	Varchar(100)		No	None	
tgl_surat	Date		No	None	
tgl_catat	Date		No	None	
keterangan	Varchar(200)		No	None	
file	Varchar(200)		No	None	
pengolah	Int(11)		No	None	

(sumber, perancangan & implementasi)

Tabel surat keluar untuk menampung semua data surat keluar kedalam form input data keluar. Dalam tabel surat keluar terdapat 11 field, yaitu : Id, Kode, No_Agenda, Tujuan, No_surat, Tgl_surat, Tgl_catatan, keterangan, file, pengolah.

Tabel 4. Tabel Surat Masuk

Field	Type	Collation	Null	Default	Extra
Id	Int(6)		No	None	Auto_increment
Kode	Varchar(50)	Latin1_swedish_ci	No	None	
No_agenda	Varchar(7)	Latin1_swedish_ci	No	None	
Indek berkas	Varchar(100)	Latin1_swedish_ci	No	None	
Isi_ringkas	Mediumtext	Latin1_swedish_ci	No	None	
Dari	Varchar(250)	Latin1_swedish_ci	No	None	
No_surat	Varchar(100)	Latin1_swedish_ci	No	None	
Tgl_surat	Date		No	None	
Tgl_terima	Date				
Keterangan	Varchar(200)	Latin1_swedish_ci			
File	Varchar(200)	Latin1_swedish_ci			
pengolah	Int(4)	Latin1_swedish_ci			

(sumber, perancangan & implementasi)

Tabel surat masuk berfungsi untuk menampung data-data surat yang masuk dan diinput dalam form surat masuk dan dalam tabel surat masuk terdapat 11 field yaitu : Id, Kode, No agenda, Indeks berkas, Isi ringkas, dari, No surat, Tanggal surat, Tanggal terima, Keterangan, File, Pengolah.

C. Perancangan Aplikasi

Perancangan sistem informasi pengarsipan menggunakan *Web Based* yang diakses lewat internet yang di hubungkan dengan web www.parnaraya.ac.id dan diakses di www.arsip.parnaraya.ac.id dan sebagai databasenya menggunakan *MySQL* untuk menampung data-data yang telah diinput oleh admin.

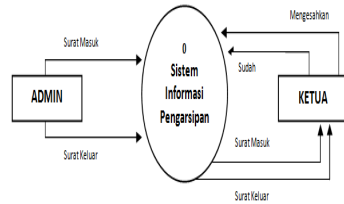
1. Alur Kerja Sistem

Untuk memulai membangun program yang berupa aplikasi *Web Based* mengenai sistem informasi pengarsipan pada kampus STMIK Parna Raya Manado terlebih dahulu penulis merancang alur kerja berdasarkan kebutuhan pengguna.

Aplikasi sistem informasi pengarsipan pada kampus STMIK Parna Raya Manado ini adalah suatu proses pendataan arsip surat masuk, surat keluar yang lama prosesnya. Perancangan sistem informasi pengarsipan merupakan usaha untuk memulai sistem yang baik.

a. Data Flow Diagram

DFD Level 0

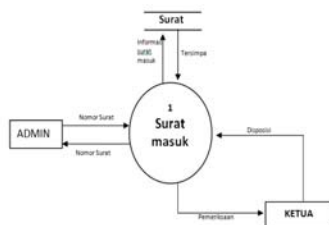


(Sumber, perancangan & implementasi)

Gambar 7. Data Flow Diagram Level 0

Admin bertugas menyimpan data surat masuk, surat keluar, menyimpan surat masuk, surat keluar dan mengecek surat masuk, surat keluar apabila sewaktu-waktu dibutuhkan data surat masuk atau surat keluar dapat dilihat

DFD Level 1 Surat Masuk

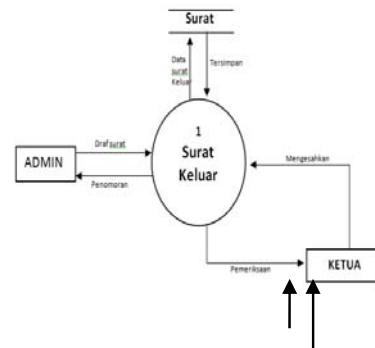


(Sumber, Perancangan & Implementasi)

Gambar 8. DFD Level 1 Surat Masuk

Admin bertugas mengisi nomor surat, menyimpan data surat masuk, dan setelah surat diperiksa surat sudah bisa didisposisi. Dan setelah didisposisi admin akan menyimpan surat di program sistem informasi pengarsipan.

DFD Level 1 Surat Keluar

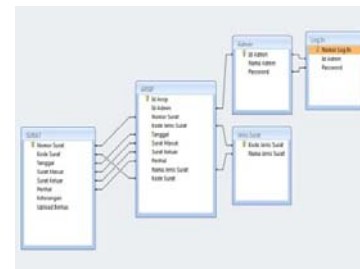


(sumber, perancangan & implementasi)

Gambar 9. DFD Level 1 Surat Keluar

Admin bertugas membuat konsep surat, penomoran surat, menyimpan data surat keluar, dan setelah diperiksa maka ketua STMIK mengesahkan surat yang akan dikeluarkan. Setelah ketua STMIK mengesahkan surat tersebut admin akan bertugas menyimpan data surat keluar yang akan diarsip.

b. Relasi antar tabel



(sumber, perancangan & implementasi)

Gambar 10. Perancangan database

Pada gambar di atas ada 5 tabel entitas yaitu entitas surat, arsip, admin, jenis surat, log in. di mana masing-masing entitas mempunyai hubungan dengan entitas lain, yaitu :

- 1) Nomor surat (*Primary Key*) mempunyai hubungan dengan tabel Arsip dan menjadi *Foreign Key* di tabel arsip.
- 2) Id Admin (*Primary Key*) mempunyai hubungan dengan tabel Arsip dan

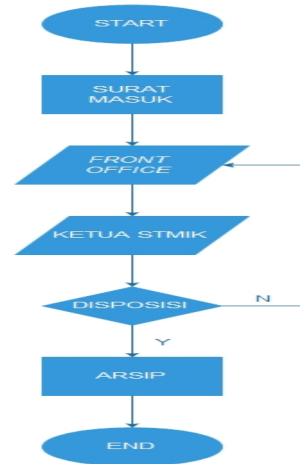
- menjadi *Foreign Key* di tabel arsip.
- 3) Kode Jenis Surat (*Primary Key*) mempunyai hubungan dengan tabel arsip dan menjadi *Foreign Key* di tabel arsip.
- 4) Id Admin (*Primary Key*) mempunyai hubungan dengan tabel log in dan menjadi *Foreign Key* di tabel log in.
- Selanjutnya merupakan atribut-atribut dari entitas surat, jenis surat ke entitas arsip dan entitas admin yang mempunyai atribut password ke log in password.

2. Flowchart

Flowchart atau bagan alir merupakan gambaran dari rangkaian kegiatan-kegiatan sistem yang penulis buat dari awal hingga akhir. Serta merupakan gambaran dari suatu program yang menyatakan arah alur program tersebut. (Sony ak, 2008).

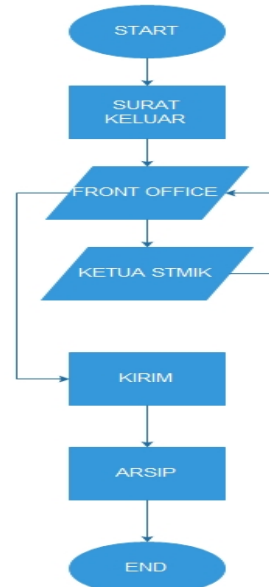
Pengertian lain yang penulis dapatkan yaitu system yang dibentuk berupa diagram alur dan algoritma-algoritma dalam suatu program, dan akhirnya menyatakan arah alur program itu. Sehingga *flowchart* atau bagan alur disebut juga metode untuk menggambarkan tahap-tahap penyelesaian masalah beserta aliran data dengan symbol-simbol standar yang mudah dipahami. (Soeherman.et.all, 2008)

a. Flowchart Surat Masuk



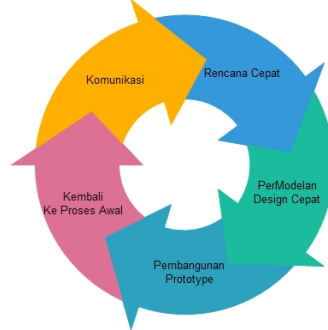
Gambar 11. Flowchart Surat Masuk

b. Flowchart Surat Keluar



Gambar 12. Flowchart Surat Keluar

3. Metode Prototype

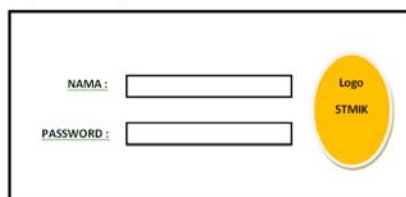


Gambar 13. Metode Prototype

- Komunikasi penulis dan admin STMIK Parna Raya Manado tentang sistem informasi pengarsipan di STMIK Parna Raya Manado.
- Melakukan perencanaan dan perModelan secara cepat berupa rancangan cepat dan kemudian akan memulai konstruksi pembuatan prototype.
- Pembuatan prototype dan melakukan evaluasi lebih lanjut setelah itu pembuatan program.
- Pembuatan *software* sesuai dengan prototype yang telah dievaluasi.
- Jika belum memenuhi kebutuhan dari STMIK Parna Raya maka akan kembali ke proses awal sampai kebutuhan dari STMIK telah terpenuhi.

4. Perancangan Tampilan Sistem

- Tampilan untuk Log In

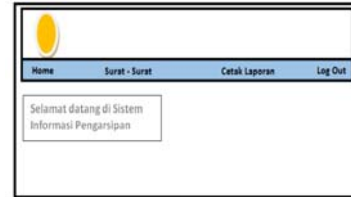


(Sumber, Perancangan & Implementasi)

Gambar 14. Tampilan depan untuk Log In

Gambar diatas merupakan rancangan tampilan saat awal membuka link

www.arsip.parnaraya.ac.id, pada form log in ini, admin hanya memasukan username dan password. Tampilan Menu Utama

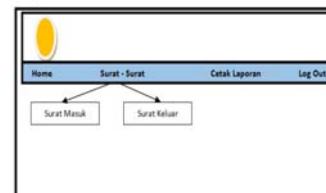


(Sumber, perancangan & implementasi)

Gambar 15. Tampilan Menu Utama

Gambar 16 merupakan rancangan tampilan menu utama, tampilan ini akan muncul ketika admin telah berhasil login dari program sistem informasi pengarsipan.

- Tampilan sub menu Surat - Surat



(Sumber, perancangan & implementasi)

Gambar 16. Tampilan Sub Menu Surat-Surat

Gambar diatas merupakan rancangan tampilan dari surat-surat, didalam sub surat-surat terdapat surat masuk dan surat keluar, ketika admin memilih salah satu dari surat masuk dan surat dikeluarkan akan ada form yang harus diinput dalam surat masuk atau surat keluar.

- Tampilan sub menu cari arsip



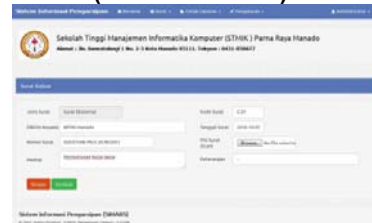
Gambar 17. Tampilan sub menu Cari Arsip

Tampilan rancangan pencarian arsip, dalam pencarian arsip pengguna hanya akan menginput data arsip yang akan dicari dan klik tombol cari, sesudah itu data yang

Page | 34

Tampilan *input* data surat keluar internal, dalam form tersebut admin harus menginput data yang lengkap.

5. Pengujian tampilan input surat keluar (surat eksternal)



(Sumber, perancangan & implementasi)
Gambar 23. Tampilan input surat keluar eksternal

6. Pengujian tampilan input surat keluar (surat umum)



(Sumber, perancangan & implementasi)
Gambar 24. Tampilan Input surat keluar umum

7. Pengujian tampilan surat masuk



(Sumber, perancangan & implementasi)
Gambar 25. Tampilan surat masuk

Tampilan data surat masuk, setelah data-data diinput akan tersimpan di *database* sistem informasi pengarsipan jika ada penginputan yang salah admin tinggal merubah data surat dengan memilih (aksi) Edt. Dan setelah mengubah data yang salah akan simpan data tersebut dan akan terlihat seperti gambar diatas. Dan untuk menghapus data surat yang sudah tidak perlu admin tinggal menghapus data dengan memilih (aksi) Del, untuk menghapus data yang sudah tidak terpakai.

8. Pengujian tampilan surat keluar



(Sumber, perancangan & implementasi)
Gambar 26. Tampilan surat keluar

Setelah menginput data surat keluar maka data tersebut akan tersimpan dalam *database* sistem informasi pengarsipan, seperti yang terlihat dalam tampilan diatas. Ketika ada penginputan data surat keluar yang salah maka admin tinggal memilih (aksi) Edt untuk mengubah data yang salah. Jika ada data surat keluar yang sudah tidak terpakai maka admin akan memilih (aksi) Del untuk menghapus data yang sudah tidak terpakai.

9. Pengujian tampilan cetak laporan surat keluar (Internal, eksternal, umum)



(Sumber, perancangan & implementasi)
Gambar 27. Tampilan cetak laporan surat keluar (internal, eksternal, umum)

Tampilan gambar 35 untuk mencetak agenda laporan surat keluar internal, eksternal, umum. Dan juga ada pilihan jika admin hanya perlu mencetak salah satu surat Internal, eksternal atau surat umum apabila admin akan mencetak semua laporan surat admin memilih "semua tipe surat" untuk mencetak semua agenda surat keluar sesuai tanggal

yang akan dibutuhkan untuk mencetak agenda laporan surat.

10. Pengujian tampilan cetak laporan surat keluar



(Sumber, perancangan & implementasi)
Gambar 28. Tampilan cetak laporan surat keluar

11. Pengujian tampilan surat masuk



(Sumber, perancangan & implementasi)
Gambar 29. Tampilan cetak laporan surat masuk



(Sumber, perancangan & implementasi)
Gambar 30. Tampilan cetak agenda laporan surat masuk

Tampilan cetak laporan surat masuk, tampilan tersebut adalah tampilan *preView* sebelum mencetak. Jika ingin mencetak pilihan OK pada form Print akan mencetak laporan surat masuk.

Tampilan pengaturan (Instansi pengguna) admin bisa mengubah tampilan yang ada dalam sistem informasi pengarsipan.

12. Pengguna sistem informasi pengarsipan ini bisa lebih dari satu orang, dan tampilan diatas adalah untuk membuat ID bagi pengguna selain dari admin.

13. Pengujian tampilan mengubah *password* pengguna



(Sumber, perancangan & implementasi)
Gambar 31. Tampilan mengubah *password* pengguna.

Tampilan ini untuk mengubah *password* pengguna, jika pengguna sudah tidak ingin menggunakan *password* yang lama pengguna hanya meinput data yang ada pada tampilan gambar diatas.

E. Analisis Sistem

Analisis dari sistem merupakan tahap terakhir penulis lakukan setelah mengumpulkan data pada objek kampus STMIK Parna Raya Manado, berupa data surat masuk, surat keluar.

Penulis lakukan selama penelitian yaitu menganalisa serta mengidentifikasi masalah-masalah pada sistem pengarsipan kampus STMIK Parna Raya Manado yang masih manual dan memperlambat dalam pencarian data arsip. Kemudian penulis merancang sistem informasi pengarsipan untuk membantu dalam pengarsipan surat masuk dan surat keluar pada STMIK Parna Raya Manado.

V. PENUTUP

A. Kesimpulan

1. Sistem informasi pengarsipan ini bertujuan untuk mempermudah dan mempercepat dalam pengarsipan yang ada dalam kampus STMIK Parna Raya Manado.



2. Data-data yang diinput akan tersimpan dalam *database* sistem informasi pengarsipan. Agar data yang disimpan tidak rusak (terkena virus atau *komputer* rusak) maka penulis menyimpan data tersebut dalam *WEB* www.arsip.parnaraya.ac.id.
3. Aplikasi ini hanya untuk dipergunakan oleh kampus STMIK Parna Raya Manado.
4. Sistem informasi pengarsipan menggunakan *Model View Controller* ini memiliki beberapa keunggulan, antara lain :
 - a. Pengarsipan lebih cepat dari sebelumnya yang masih manual
 - b. Data yang diarsip tersimpan dalam *WEB*. www.arsip.parnaraya.ac.id
 - c. Dengan menggunakan pemrograman *Model View Controller*, dalam pengembangan aplikasi program, *update* program akan lebih mempermudah pengguna.
5. Penelitian ini dilaksanakan mulai bulan akhir Januari 2014 sampai Juni 2014.

B. Saran

1. Penggunaan dari pada sistem inf pengarsipan ini kiranya dapat dikembangkan sehingga bisa dikembangkan menjadi lebih baik sesuai dengan kebutuhan kampus STMIK Parna Raya Manado.
2. Karena program ini terkoneksi dengan jaringan internet maka data-data yang ada dalam harus dijaga dengan sebaik-baiknya.
3. Penulis harapkan sistem yang telah dibuat ini dapat dijaga penggunaanya serta dipelajari sehingga mampu bertahan dan dikembangkan sesuai yang dibutuhkan.

DAFTAR PUSTAKA

- Abdul Kadir. 2008.** Dasar Pemrograman *WEB* Dinamis menggunakan *PHP*. Andi offset. Yogyakarta
- Abdul Kadir. 2010.** Mudah Mempelajari *Database* MySQL . Andi offset. Yogyakarta
- Agus Mulyono. 2009.** Sistem Informasi Konsep dan Aplikasi. Pustaka Belajar. Yogyakarta
- Agus S. 2011.** Panduan Praktis Menguasai *Database* Server MySQL. Media Komputindo. Jakarta
- Anonymous. 2014.** <http://ebookservicekomputer.blogspot.com/2013/06/printer.html>. diakses 26 Agustus 2014
- Anonymous. 2014.** <http://tip-mega.blogspot.com/2013/01/pengertian-scanner.html>. diakses 26 Agustus 2014
- Arifin. Zaenal. 2008.** Bahasa Indonesia Bagi Sekretaris. PT. Grasindo. Jakarta
- Basuki. Awan P. 2010.** Membangun *WEB* berbasis *PHP* dengan *Framework CodeIgniter*. Lokomedia. Yogyakarta
- Betha. Sidik. 2012.** *Framework* Code Igniter. BI-Obses. Bandung
- Blanco, Jose A. and Upton, David. 2009.** *CodeIgniter 1.7*. Packt Publishing. Birmingham
- Budi Raharjo. 2011.** Belajar Otodidak Membuat *Database* menggunakan MySQL. Informatika. Bandung
- Christopher L. 2011.** Refrensi Ringkas *HTML*. Media Komputindo. Jakarta
- Griffiths, Adam. 2010.** *CodeIgniter 1.7 Profesional Development*. Packt Publishing
- Jack Febrian, Farida Andayani. 2008.** Kamus komputer dan istilah Teknologi Informasi. Informatika Bandung. Bandung
- Jogiyanto. 2009.** Analisis Dan Desain Sistem Informasi. Andi. Yogyakarta
- Jubilee E. 2011.** Tips dan Trik *HTML*. Jubilee Enterprise. Jakarta
- Kanisius. 2008.** Manajemen administrasi perkantoran. Kanisius offset. Yogyakarta
- Myer, Thomas. 2008.** *Profesional CodeIgniter*. Wiley Publishing. Indianapolis
- Nugroho. Adi. 2009.** Rekayasa perangkat lunak menggunakan UML dan Java. Andi offset. Yogyakarta
- Nugroho. B. 2010.** *Sistem Control via WEB, PHP dan ajax*. Elex Media Komputindo. Jakarta
- Soeherman. Bonnie dkk. 2010.** Membangun Sistem Informasi UMKM dagang dengan MS. Access. PT Elex Media Komputindo. Jakarta
- Soeherman.et.all. 2008.** Membangun Sistem Dengan MS.Access. Elex Media Komputindo. Jakarta



- Sony, ak. 2008.** Algoritma dan Flowchart. Graha ilmu. Yogyakarta
- STMIK Parna Raya Manado. 2010.** *Panduan penyusunan skripsi dan penelitian mahasiswa.* STMIK Parna Raya. Manado.
- Upton, David. 2007.** *CodeIgniter for Rapid PHP Application Development.* Packt Publishing. Birmingham
- Wardana, S.hut. M.Si. 2010.** Menjadi Master PHP dengan Framework CodeIgniter. Elex Media. Jakarta
- YM Kusuma. Ardhana. 2013.** Pemrograman PHP Code Igniter Black Box. Jasacom. Purwokerto
- Zanianotto, Francios and Potencier, Fabien. 2010.** *Simfony , A Gentle Introduction to Symfony.* Sensio SA. Clichy France
- Zuchdi. Darmiyanti. 2010.** Menjadi master PHP dengan Framework Dodelgniter. PT. Media Komputindo, Jakarta