

SISTEM INFORMASI KEARSIPAN BERBASIS WEB PADA FAKULTAS EKONOMI DAN BISNIS UNIKA ATMA JAYA JAKARTA

Nuris Avrilia

12057201

Sistem Informasi Universitas Suryadarma

Abstrak

Arsip merupakan kumpulan warkat yang disimpan secara sistematis karena mempunyai kegunaan agar setiap kali diperlukan dapat secara cepat ditemukan kembali. Arsip mempunyai peranan yang penting dalam sebuah kantor, apabila arsip yang dimiliki kurang baik pengelolaannya dapat mengakibatkan sulitnya menemukan informasi yang telah disimpan dan akhirnya dapat menghambat tahapan proses pekerjaan selanjutnya. Oleh karena itu sebaiknya arsip dikelola menggunakan sistem pengelolaan arsip yang baik dan benar. Permasalahan pada penelitian ini adalah bagaimana pelaksanaan sistem pengelolaan arsip serta kendala-kendala yang dihadapi dalam pelaksanaan sistem pengelolaan arsip di Sekretariat Fakultas Ekonomi Dan Bisnis Unika Atma Jaya Jakarta. Tujuan dari penelitian ini adalah untuk mengetahui bagaimana pelaksanaan dan kendala-kendala yang dihadapi dalam pelaksanaan sistem pengelolaan arsip di Sekretariat Fakultas Ekonomi Dan Bisnis Unika Atma Jaya Jakarta. Metode penelitian yang penulis gunakan adalah: Metode pengumpulan data yang digunakan adalah wawancara dan elisitasi. Dalam penelitian ini penulis menggunakan analisis data secara deskriptif, yaitu memaparkan tentang sesuatu obyek sebagaimana adanya pada suatu waktu. Sistem pengelolaan arsip di Sekretariat Fakultas Ekonomi Dan Bisnis Unika Atma Jaya Jakarta meliputi: Sistem penyimpanan arsip, penemuan kembali arsip, pemeliharaan dan pengamanan arsip. Kendala-kendala yang dihadapi dalam pengelolaan arsip meliputi : terbatasnya fasilitas kearsipan, sumber daya manusia yang mengelola arsip dan sistem pengelolaan. Berdasarkan penelitian yang dilakukan dapat disimpulkan bahwa dalam pengelolaan arsip di Sekretariat Fakultas Ekonomi Dan Bisnis Unika Atma Jaya Jakarta belum dapat berjalan secara maksimal. Hal ini dibuktikan dengan belum diberlakukannya pedoman dalam kearsipan, serta belum diberlakukannya pemusnahan arsip secara teratur.

Kata Kunci: Arsip.

Abstract

Archive is a collection of slips to be recorded systematically as have utility in order whenever required can be quickly recovered. Archive has an important role in an office, if the archive owned by poor management can lead to difficulties in finding the information that has been stored and can ultimately hinder the further work process stages. Therefore, we recommend using the managed archival records management system is good and right. The problem in this research is how the implementation of records management systems and constraints faced in the implementation of records management systems in the Secretariat of the Faculty of Economics and Business Atma Jaya Catholic University, Jakarta. The purpose of this study was to determine how the implementation and the constraints faced in the implementation of records management systems in the Secretariat of the Faculty of Economics and Business Atma Jaya Catholic University, Jakarta. The research method that I use is: Data collection methods used were interviews and elicitation. In this study the author uses descriptive data analysis, which describes something as their object at a time. Archive management system at the Secretariat of the Faculty of Economics and Business Atma Jaya Jakarta include: archival storage systems, the rediscovery of the archive, maintenance and security archives. Constraints faced in the management of archives include: lack of archival facilities, human resources and managing archives management system. Based on this research can be concluded that in records management at the Secretariat of the Faculty of Economics and Business Atma Jaya Jakarta can not run optimally. This is evidenced by yet implemented the guidelines in the archives, as well as the destruction of records has not yet implemented regularly.

Keywords: Archive.

PENDAHULUAN

Sejak jaman dahulu arsip adalah hal terpenting dalam menjaga eksistensi suatu dokumen, data, gambar ataupun tulisan. Dengan mengarsipkan dokumen ataupun tulisan maka dokumen atau tulisan tersebut dapat terus terjaga dan dapat diakses saat dibutuhkan kembali.

Dengan berkembangnya teknologi komputerisasi dan internet saat ini kita dapat mengakses berbagai macam informasi serta saling berbagikan semua informasi yang ada disekitar kita kapan saja dan dimana saja tanpa harus terbatas ruang dan waktu.

Universitas Atma Jaya yaitu yayasan yang bergerak pada bidang pendidikan. Pada Yayasan Unika Atma Jaya ini pengarsipan dokumen masih sangat primitif dan manual, yaitu dokumen yang telah diprint/cetak dan ditanda- tangani lalu diphotocopy kemudian disimpan pada lemari penyimpanan. Hal ini memungkinkan dokumen yang sangat penting tersebut dapat rusak dan hilang dikarenakan kecelakaan, kelalaian manusia, dan dimakan usia, serta juga membutuhkan waktu dalam pencarian kembali dokumen pada saat dibutuhkan dengan tepat misalnya sebagai bukti kepada pihak pemeriksa.

Selain itu dengan semakin banyaknya dokumen-dokumen penting yang dimiliki perusahaan menyebabkan ruangan penyimpanan semakin berkurang, sehingga perusahaan dipaksa untuk menambah ruangan hanya untuk menjadi tempat penyimpanan arsip dokumen.

Dikarenakan hal tersebut maka dalam tugas akhir ini penulis melakukan perancangan Aplikasi Sistem Informasi Pengarsipan Surat Menyurat Berbasis Web pada Fakultas Ekonomi dan Bisnis Unika Atma Jaya Jakarta.

Seiring dengan kemajuan teknologi di berbagai bidang, seakan-akan tidak pernah ada matinya untuk dipelajari. Dari hari ke hari kemajuan teknologi terus berkembang, salah satunya adalah komputer. Komputer merupakan sarana untuk menyelesaikan pekerjaan dengan tepat dan efisien, aplikasi komputer di bidang kearsipan surat menyurat salah satunya adalah pendataan surat masuk dan surat keluar. Pendataan surat masuk dan surat keluar setiap bulannya mengalami perubahan seiring dengan banyak nya surat masuk dan surat keluar. Oleh karena itu, diperlukan suatu pengarsipan yang lebih baik.

Kearsipan merupakan sarana yang digunakan setiap lembaga khususnya lembaga pendidikan seperti Fakultas Ekonomi Universitas Atma Jaya Jakarta untuk penyimpanan, pengelompokan, penataan, pengendalian dan juga pemeliharaan berbagai arsip yang ada khususnya arsip persuratan. Menurut Budiharto Sanusi (Meditek, 1996) Istilah arsip sendiri berasal dari bahasa Yunani yaitu Archeion berarti kantor. Schelenbergr memberikan definisi arsip sebagai berkas pranata umum ataupun swasta yang dinilai perlu disimpan secara permanen untuk tujuan acuan dan penelitian dan telah disimpan atau dipilah untuk disimpan pada sebuah lembaga kearsipan.

Pengelolaan Sistem Kearsipan pada Fakultas Ekonomi Universitas Atma Jaya saat ini mengelola data surat masuk dan jenis jenis surat keluar seperti Surat Perintah Perjalanan Dinas (Perdin), Surat Keputusan, Surat Undangan, Surat Keterangan Mahasiswa, Surat Pengantar dan Surat Tugas untuk dosen dan mahasiswa. Prosedur pengarsipan surat masuk dan surat keluar tersebut masih bersifat manual belum menggunakan aplikasi khusus untuk mengagendakan kearsipan sehingga sering terjadinya penumpukan dokumen surat masuk dan surat keluar, sering terjadinya kesalahan pencarian data dan sering terjadinya kehilangan data.

Rumusan Masalah

Berdasarkan latar belakang masalah diatas yang telah diuraikan, maka perumusan

masalah yang diteliti adalah:

- a. Sistem kearsipan yang berjalan, untuk proses penyimpanan berkas masih menggunakan cara konvensional (hanya menyimpan berkas secara fisik) tidak dengan penyimpanan digital, sehingga diperlukan banyak tempat penyimpanan.
- b. Sistem arsip (klasifikasi arsip) tidak dilakukan pembagian jenis arsip yang penting dan tidak penting, menimbulkan penumpukan berkas arsip yang kian bertambah untuk setiap periodenya.
- c. Pengelolaan arsip tidak dilakukan secara selektif, tidak dilakukan pemusnahan arsip yang sudah usang/tidak bernilai pada periode tertentu.
- d. Penanganan arsip yang tidak terpusat pada satu induk yang menyebabkan kesulitan dalam pencarian arsip yang diperlukan.

Batasan Masalah

Supaya pembahasan masalah yang dilakukan dapat terarah dengan baik dan tidak menyimpang dari pokok permasalahan, maka penulis membatasi permasalahan yang akan dibahas, yakni:

Lingkup sistem informasi dimulai dari pendataan arsip (surat), pembuatan kode berdasarkan klasifikasi untuk disimpan kedalam sistem *database* dan sampai pada laporan arsip.

LANDASAN TEORI

Sistem Informasi

Pengertian Sistem

Definisi sistem dalam kamus *Webster's Unbringed* adalah elemen-elemen yang saling berhubungan dan membentuk satu kesatuan atau organisasi menurut Hanif Al Fatta, 2007.

Menurut Murdick, R. G (1991:27), Sistem adalah seperangkat elemen yang membentuk kumpulan atau prosedur-prosedur atau bagan-bagan pengolahan yang mencari suatu tujuan bagian atau tujuan bersama dengan mengoperasikan data dan/atau barang pada waktu rujukan tertentu untuk menghasilkan informasi dan/atau energi dan/atau barang.

Dari definisi tersebut maka dapat disimpulkan bahwa sistem dapat didefinisikan sebagai suatu kumpulan elemen dari subsistem yang saling berhubungan dan bekerja sama untuk mencapai suatu tujuan.

Komponen atau Karakteristik Sistem

Komponen atau Karakteristik sistem adalah bagian yang membentuk sebuah sistem, diantaranya:

- **Objek** merupakan bagian elemen atau variabel. Ia dapat berupa benda fisik, abstrak atau keduanya.
- **Atribut** merupakan penentu kualitas atau sifat kepemilikan sistem dan objeknya.
- **Hubungan internal** merupakan penghubungan diantara objek-objek yang terdapat dalam sebuah sistem.
- **Lingkungan** merupakan tempat dimana sistem berada.
- **Tujuan** setiap sistem memiliki tujuan dan tujuan inilah yang menjadi motivasi yang mengarahkan sistem. Tanpa tujuan sistem menjadi tidak terkendali. Tentu tujuan antara satu sistem dengan sistem yang lain berbeda.
- **Masukan** adalah sesuatu yang masuk ke dalam sistem dan selanjutnya menjadi bahan untuk diproses. Masukan tersebut dapat berupa hal-hal yang tampak fisik (bahan mentah) atau yang tidak tampak (jasa).

- **Proses** adalah bagian yang melakukan perubahan dari masukan menjadi keluaran yang berguna dan lebih bernilai (informasi) atau yang tidak berguna (limbah)
- **Keluaran** adalah hasil dari proses. Pada sistem informasi berupa informasi atau laporan, dsb
- **Batas** adalah pemisah antara sistem dan daerah luar sistem. Batas disini menentukan konfigurasi, ruang lingkup atau kemampuan sistem. Batas juga dapat diubah atau dimodifikasi sehingga dapat merubah perilaku sistem.
- **Mekanisme** pengendalian dan umpan balik, digunakan untuk mengendalikan masukan atau proses. Tujuannya untuk mengatur agar sistem berjalan sesuai dengan tujuan.

Pengertian Informasi

Definisi sistem informasi yang lain merupakan satu kesatuan data olahan yang terintegrasi dan saling melengkapi yang menghasilkan output baik dalam bentuk gambar, suara maupun tulisan menurut Al-Bahra Bin Ladjamudin, *Sistem Informasi*, 2005.

Pengertian Sistem Informasi

Menurut Robert A. Leitch dan K. Roscoe Davis dalam buku Jogiyanto HM, (1999:11) 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.

Menurut T Notohadiprawiro - Universitas Gajah Mada, 2006: Istilah sistem informasi menyiratkan suatu pengumpulan data yang terorganisasi beserta tatacara penggunaannya yang mencakup jauh daripada sekedar penyajiannya. Istilah tersebut menyiratkan suatu maksud yang ingin dicapai dengan jalan memilih dan mengatur data serta menyusun tatacara penggunaannya.

Pengertian Sistem Basis Data

Menurut Linda Marlinda, 2004:1 Sistem basis data adalah suatu sistem menyusun dan mengelola *–record* menggunakan komputer untuk menyimpan atau merekam serta memelihara data operasional lengkap sebuah organisasi atau perusahaan sehingga mampu menyediakan informasi yang optimal yang di perlukan pemakai untuk proses pengambilan keputusan. Sistem basis data dapat juga diartikan sebuah tatanan (keterpaduan) yang terdiri atas sejumlah komponen fungsional (dengan satuan fungsi/tugas khusus) yang saling berhubungan dan secara bersama-sama bertujuan untuk memenuhi suatu proses/pekerjaan tertentu.

Priyadi (2014:3) juga menjelaskan beberapa elemen basis data, diantaranya adalah sebagai berikut:

1. **Tabel**
Pada suatu basis data, tabel di representasikan menjadi suatu bentuk segiempat berupa matriks, yang terdiri dari kolom dan baris.
2. **Field**
Pada tabel basis data, kolom merupakan representasi untuk sebuah nama *field* yang pembacaan datanya dilakukan secara vertikal.
3. **Record**
Pada tabel basis data, baris merupakan suatu representasi untuk sebuah *record* yang pembacaan datanya dilakukan secara horizontal. Satu baris pada sebuah tabel merupakan data yang dimiliki oleh satu *record*. Nilai-nilai yang dimiliki oleh sebuah

record merupakan gabungan dari semua *field* yang terdapat dalam tabel tersebut.

4. Kardinalitas

Kardinalitas merupakan batasan dari banyaknya hubungan, yang dapat dilakukan oleh himpunan entitas dalam melakukan relasi dengan himpunan entitas lainnya. Variasi kemungkinan untuk melakukan relasi yang dimiliki oleh kardinalitas terdiri dari empat macam, yaitu:

- a. Satu ke satu (1:1).
- b. Satu ke banyak (1:N).
- c. Banyak ke satu (N:1).
- d. Banyak ke banyak (N:N).

Pengertian Adobe *Macromedia Dreamweaver 8*

Sebuah HTML editor profesional untuk mendesain secara visual dan mengelola situs *web* maupun halaman *web* (Awaludin, 2018). Bilamana kita menyukai untuk berurusan dengan kode-kode HTML secara manual atau lebih menyukai bekerja dengan lingkungan secara visual dalam melakukan editing, *Dreamweaver* membuatnya menjadi lebih mudah dengan menyediakan tool-tool yang sangat berguna dalam peningkatan kemampuan dan pengalaman kita dalam mendesain *web*.

Dreamweaver MX dalam hal ini digunakan untuk *web* desain. *Dreamweaver MX* mengikutsertakan banyak tool untuk kode-kode dalam halaman web beserta fasilitas-fasilitasnya, antara lain : Referensi HTML, CSS dan Javascript, Javascript debugger, dan editor kode (tampilan kode dan Code inspector) yang mengizinkan kita mengedit kode Javascript, XML, dan dokumen teks lain secara langsung dalam *Dreamweaver*. Teknologi *Dreamweaver Roundtrip HTML* mampu mengimpor dokumen HTML tanpa perlu memformat ulang kode tersebut dan kita dapat menggunakan *Dreamweaver* pula untuk membersihkan dan memformat ulang HTML bila kita menginginkannya.

Selain itu *Dreamweaver* juga dilengkapi kemampuan manajemen situs, yang memudahkan kita mengelola keseluruhan elemen yang ada dalam situs. Kita juga dapat melakukan evaluasi situs dengan melakukan pengecekan broken link, kompatibilitas browser, maupun perkiraan waktu download halaman web. Versi ini bukan hanya *software* untuk desain *web* tetapi juga untuk menyunting kode serta pembuatan aplikasi *web* antara lain *JSP*, *PHP*, *ASP*, *XML*, dan *ColdFusion*.

Ruang Kerja *Macromedia Dreamweaver 8*

Berikut ini adalah keterangan untuk bagian-bagian dari ruang kerjaruang kerja *Adobe Macromedia Dreamweaver 8* adalah:

- **Title Bar** merupakan baris yang menampilkan nama program dan nama dokumen yang sedang berjalan.
- **Menu Bar** merupakan perintah-perintah untuk merancang dan mengolah dokumen website.
- **Insert Bar** merupakan tombol-tombol untuk memasukkan berbagai objek ke dalam dokumen website yang sedang kita buat agar lebih cepat dan praktis.
- **Document Toolbar** merupakan tool-tool untuk menentukan model tampilan dan judul halaman website.
- **Document Windows** merupakan ruang kerja *Dreamweaver* sebagai tempat untuk merancang sebuah website.
- **Panel Groups** merupakan panel-panel dari berbagai fungsi *Dreamweaver* seperti: Design, Code, Application, Tag Inspector, Files, Frames, dan History.

- **Tag Selector** merupakan tampilan informasi tag-tag yang sedang aktif seperti: *<title>*, *<head>*, *<body>* dan lain-lain.
- **Property Inspector** merupakan pengaturan-pengaturan yang diterapkan pada suatu objek, sehingga isi dari property inspector ini akan selalu berubah sesuai dengan objek yang sedang aktif.
- **Files Panel** merupakan direktori untuk membuka, maupun menyimpan file-file website.

Pengertian Komputer

Menurut Asropudin (2013:19) menyatakan, “Komputer adalah alat bantu pemrosesan data secara elektronik dan cara pemrosesan datanya berdasarkan urutan instruksi atau program yang tersimpan dalam memori masing-masing komputer.

Menurut Siallagan (2009:1) menyatakan, “Komputer adalah sekumpulan alat elektronik yang saling bekerja sama, dapat menerima data (input), mengolah data (process), memberikan informasi (output), dan terkoordinasi di bawah kontrol program yang tersimpan dalam memorinya.”

Dari kedua pengertian tersebut, dapat disimpulkan bahwa komputer adalah suatu alat yang dipakai untuk mengolah data menurut prosedur yang telah dirumuskan.

Pengertian Internet

Menurut Ahmadi dan Hermawan (2013:2) menyatakan, “Internet adalah sistem informasi global berbasis komputer”.

Internet (kependekan dari interconnection-networking) adalah seluruh jaringan komputer yang saling terhubung menggunakan standar sistem global Transmission Control Protocol / Internet Protocol Suite (TCP/IP) sebagai protokol pertukaran paket (packet switching communication protocol) untuk melayani miliaran pengguna di seluruh dunia.

Dari definisi-definisi di atas, dapat disimpulkan bahwa Internet adalah sistem global dari seluruh jaringan komputer yang saling terhubung sehingga antara komputer dapat saling mengakses informasi dan bertukar data.

Komponen Sistem Informasi

1. **Hardware** (perangkat keras), terdiri dari:
 - ✓ Komputer
Satu untuk sistem yang *stand-alone*, lebih dari satu untuk sistem jaringan.
 - ✓ Memori sekunder yang *on-line* (*Hardisk*)
 - ✓ Memori sekunder yang *off-line* (*Tape* atau *Removable Disk*) untuk keperluan *backup* data.
 - ✓ Media/perangkat komunikasi (untuk sistem jaringan)
2. **Software** kumpulan perintah yang ditulis dengan aturan untuk memerintah komputer melaksanakan tugas tertentu, terdiri dari:
 - a. Untuk kelas sederhana: dBase III+, dBaseIV, FoxBase, RBase, MS-Access, Borland- Paradox, dll.
 - b. Untuk kelas kompleks/berat: Borland-Interbase, MS-SQL Server, MySQL, CA-Open Ingres, Oracle, Informix, Sybase, dll.
3. **Data** merupakan komponen dasar dari informasi yang akan diproses lebih lanjut untuk menghasilkan informasi.
4. **Manusia** yang terlibat dalam komponen manusia seperti:
 - a. *Programmer* Aplikasi.

- b. User Mahir (*Casual User*).
 - c. User Umum (*End User Naive User*).
 - d. User Khusus (*Specialized User*).
5. **Prosedur** dokumentasi proses sistem, buku penuntun operasional (aplikasi) dan teknis. (Leman, 1998).

Teknologi

Menurut (Miarso, 2007), teknologi adalah proses yang meningkatkan nilai tambah, proses tersebut menggunakan atau menghasilkan suatu produk, produk yang dihasilkan tidak terpisah dari produk lain yang telah ada dan karena itu menjadi bagian integral dari suatu sistem.

Web

Menurut HD Erinawati-Speed-Sentra Penelitian Engineering dan Edukasi, 2013 - ijns.org Pengertian WWW atau *world wide web* atau *web* saja merupakan sebuah sistem yang saling terkait dalam sebuah dokumen yang berformat hypertext yang berisi beragam informasi, baik tulisan, gambar, suara, video, dan informasi multimedia lainnya dan dapat diakses melalui sebuah perangkat yang disebut *web browser*. Untuk menterjemahkan dokumen dalam bentuk *hypertext* ke dalam bentuk dokumen yang bisa dipahami, maka *web browser* melalui *web client* akan membaca halaman *web* yang tersimpan di sebuah *web server* melalui protocol yang biasa disebut http atau *Hypertext Transfer Protocol*. Untuk menterjemahkan dokumen dalam bentuk *hypertext* ke dalam bentuk dokumen yang bisa dipahami, maka *web browser* melalui *web client* akan membaca halaman *web* yang tersimpan di sebuah *web server* melalui protokol yang biasa disebut http atau *Hypertext Transfer Protocol*.

Web Server

Web server merupakan *software* yang memberikan layanan data yang berfungsi yang menerima permintaan http ataupun https (*hypertext transfer protocol security*) dari *client* melalui *web browser* dan mengirimkan kembali hasilnya dalam bentuk halaman *web* yang umumnya berbentuk dokumen dalam format HTML.

Xampp

Menurut Alan Nur Aditya, 2011:16 XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai server yang berdiri sendiri (localhost), yang terdiri atas program Apache HTTP Server, *MySQL database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMP merupakan singkatan dari X (empat sistem operasi apapun), Apache, *MySQL*, PHP dan Perl. Program ini tersedia dalam GNU *General Public License* dan bebas, merupakan *web server* yang mudah digunakan yang dapat melayani tampilan halaman *web* yang dinamis.

PHP

PHP adalah singkatan dari "PHP: *Hypertext Preprocessor*", yang merupakan sebuah Bahasa *scripting* yang terpasang pada *HyperText Markup Language* (HTML). Sebagian besar sintaks mirip dengan bahasa C, Java dan Perl, ditambah beberapa fungsi PHP yang spesifik. Tujuan utama penggunaan bahasa ini adalah untuk memungkinkan perancang *web* menulis halaman *web* dinamik dengan cepat. (Dodit Supriyanto, 2008).

Konsep Kerja PHP

Saputra (2012) menjelaskan tentang konsep kerja PHP adalah sebagai berikut:

1. *Server* membaca permintaan dari *client/browser*.
2. Kemudian dilanjutkan untuk mencari halaman/*page* pada *server*.
3. *Server* melakukan instruksi yang diberikan oleh php untuk melakukan modifikasi pada halaman/*page*.
4. Selanjutnya hasil modifikasi tersebut akan dikembangkan kepada

Tipe Data PHP

MADCOMS (2011:221) menjelaskan tentang tipe data PHP, yaitu:

1. Tipe Data *Integer*

Tipe data ini berisikan data semua bilangan bulat yang besarnya range sama dengan data pada bahasa C, yaitu antara -2, 147, 483, 648 sampai +2, 147, 483, 647 pada platform 32bit. Apabila data di luar kisaran tersebut maka PHP secara otomatis mengkonversikan data tersebut dari tipe *Integer* menjadi tipe *Floating Point*.

2. Tipe Data *Floating*

Point Tipe data ini berisikan bilangan pecahan atau bilangan desimal. Kisaran data floating adalah antara 1.7E-308 sampai 1.7E+308. Bentuk dari data ini adalah desimal ataupun dalam bentuk pangkat.

3. Variabel

Dalam program PHP, variabel digunakan untuk menyimpan data sementara baik jenis *string*, *integer*, maupun *array*. Variabel dinyatakan dengan tanda \$ di depan nama variabel. Nama variabel dapat berupa huruf, angka maupun garis bawah. Namun dalam penulisannya, variabel harus diawali dengan huruf atau garis bawah (_), kemudian diikuti huruf atau angka. Penulisan variabel yang diawali angka tidak dibenarkan.

MySQL

MySQL sebenarnya merupakan turunan salah satu konsep utama dalam basis data yang telah ada sebelumnya; *SQL (Structured Query Language)*. *SQL* adalah sebuah konsep pengopeasian basisdata, terutama untuk pemilihan atau seleksi dan pemasukan data yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis (Alan Nur Aditya, 2011:61). Untuk melakukan administrasi dalam basis data *MySQL*, dapat menggunakan modul yang sudah termasuk yaitu *command-line* (perintah: *mysql* dan *mysqladmin*). Juga dapat di unduh dari situs *MySQL* yaitu sebuah modul berbasis grafik (GUI), *MySQL Administrator* dan *MySQL Query Browser*. Selain itu terdapat juga sebuah perangkat lunak gratis untuk administrasi basis data *MySQL* berbasis *web* yang sangat populer yaitu *phpMyAdmin*. Untuk perangkat lunak untuk administrasi basis data *MySQL* yang dijual secara komersial antara lain: *MySQL front*, *Navicat* dan *EMS SQL Manager for MySQL*.

Web Server Apache

Web Server Apache menurut Firdaus (2007 : 5) *Web server* merupakan sebuah bentuk *server* yang khusus digunakan untuk menyimpan halaman *website* atau *homepage*. Apache merupakan turunan dari web server yang dikeluarkan oleh NSCA yaitu NSCA HTTPd sekitar tahun 1995-an.

Pada dasarnya, apache adalah “APatCHy” (patch) dan pengganti dari NCSA HTTPd. Apache web server merupakan tulang punggung permintaan dari client yang menggunakan browser, seperti Netscape Navigator, Internet Explorer, Mozilla, lynx dan lain-lain. Web Server dalam berkomunikasi dengan kliennya menggunakan protokol HTTP (*Hyper Text Transfer Protocol*).

Apache berada di bawah GNU, *General Public License* yang bersifat *free* sehingga Apache dapat didownload gratis pada alamat <http://www.apache.org>. Adapun pertimbangan dalam memilih Apache adalah:

1. Apache termasuk dalam kategori *free software* (software gratis).
2. Instalasi apache sangat mudah.
3. Mampu beroperasi pada banyak *platform* sistem operasi, seperti Linux, Windows dan lain-lain.

Apache Web Server merupakan *web server* yang bersifat *open source* dan mempunyai *performance* yang sangat bagus, fleksibel dan mendukung berbagai macam platform sistem operasi seperti Windows NT/9x, UNIX, Netware 5x, OS/2 dan berbagai macam sistem operasi lainnya. Apache sangat cepat sekali mengeluarkan *update* terbarunya, sehingga mengurangi munculnya *bugs* dan kelemahan program.

UML

Pengertian UML - *Unified Modeling Language* (UML) adalah tujuan umum, perkembangan, bahasa pemodelan di bidang rekayasa perangkat lunak, yang dimaksudkan untuk menyediakan cara standar untuk memvisualisasikan desain sistem.

UML awalnya termotivasi oleh keinginan untuk membakukan sistem notasi yang berbeda dan pendekatan untuk desain perangkat lunak yang dikembangkan oleh Grady Booch, Ivar Jacobson dan James Rumbaugh di Rational Software di 1994-1995, dengan pengembangan lebih lanjut yang dipimpin oleh mereka melalui tahun 1996.

Pada tahun 1997 UML diadopsi sebagai standar oleh Object Management Group (OMG), dan telah dikelola oleh organisasi ini sejak. Pada tahun 2005 UML juga diterbitkan oleh International Organization for Standardization (ISO) sebagai standar ISO disetujui. Sejak itu telah periodik direvisi untuk menutupi revisi terbaru dari UML.

Menurut *Booch* (2005:7) UML adalah bahasa standar untuk membuat rancangan *software*. UML biasanya digunakan untuk menggambarkan dan membangun, dokumen artifak dari *software-intensive system*.

Langkah-langkah penggunaan UML

Batasan sistem harus ditentukan terlebih dahulu, tujuannya agar pemakai mengetahui dengan lingkungan mana saja sistem mereka berhubungan. Untuk itu setiap komponen *actor*, (sumber atau tujuan) ini harus diberi nama sesuai dengan lingkungan luar yang mempengaruhi sistem ini.

Menurut Henderi (2008:6), langkah-langkah penggunaan *Unified Modeling Language* (UML) sebagai berikut:

- Buatlah daftar *business process* dari level tertinggi untuk mendefinisikan aktivitas dan proses yang mungkin muncul.
- Petakan use case untuk setiap *business process* untuk mendefinisikan dengan tepat fungsional yang harus disediakan oleh sistem, kemudian perhalus *use case diagram* dan lengkapi dengan *requirement*, *constraints* dan catatan-catatan lain.
- Buatlah *deployment diagram* secara kasar untuk mendefinisikan arsitektur fisik sistem.
- Definisikan *requirement* lain non fungsional, *security* dan sebagainya yang juga harus

disediakan oleh sistem.

- Berdasarkan *use case* diagram, mulailah membuat *activity* diagram.
- Definisikan obyek - obyek level atas package atau domain dan buatlah *sequence* dan/atau *collaboration* untuk tiap alur pekerjaan, jika sebuah *use case* memiliki kemungkinan alur normal dan *error*, buat lagi satu diagram untuk masing-masing alur.
- Buatlah rancangan user *interface* model yang menyediakan antar muka bagi pengguna untuk menjalankan skenario *use case*.
- Berdasarkan model-model yang sudah ada, buatlah class diagram. Setiap package atau domain dipecah menjadi hirarki class lengkap dengan atribut dan metodenya. Akan lebih baik jika untuk setiap class dibuat unit test untuk menguji fungsionalitas *class* dan interaksi dengan *class* lain.
- Setelah *class* diagram dibuat, kita dapat melihat kemungkinan pengelompokkan *class* menjadi komponen-komponen karena itu buatlah *component* diagram pada tahap ini. Juga didefinisikan test *integrasi* untuk setiap komponen meyakinkan ia bereaksi dengan baik.
- Perluas *deployment* diagram yang sudah dibuat. Detilkan kemampuan dan *requirement* piranti lunak, sistem operasi, jaringan dan sebagainya. Petakan komponen ke dalam node.
- Mulailah membangun sistem. Ada dua pendekatan yang tepat digunakan:
 - 1) Pendekatan *use case* dengan mengassign setiap *use case* kepada tim pengembang tertentu untuk mengembangkan unit kode yang lengkap dengan test.
 - 2) Pendekatan komponen yaitu mengassign setiap komponen kepada tim pengembang tertentu.

Model Diagram UML

Menurut Widodo (2011:10), “Beberapa *literature* menyebutkan bahwa UML menyediakan sembilan jenis diagram, yang lain menyebutkan delapan karena ada beberapa diagram yang digabung, misalnya diagram komunikasi, diagram urutan dan diagram pewaktuan digabung menjadi diagram interaksi”. Namun demikian model-model itu dapat dikelompokkan berdasarkan sifatnya yaitu statis atau dinamis. Jenis diagram itu antara lain:

1. **Diagram kelas (Class Diagram)**
Bersifat statis, Diagram ini memperlihatkan himpunan kelas-kelas, antarmuka-antarmuka, kolaborasi-kolaborasi, serta relasi-relasi. Diagram ini umum dijumpai pada pemodelan sistem berorientasi objek. Meskipun bersifat statis, sering pula diagram kelas memuat kelas-kelas aktif.
2. **Diagram paket (Package Diagram)**
Bersifat statis. Diagram ini memperlihatkan kumpulan kelas-kelas, merupakan bagian dari diagram komponen.
3. **Diagram use-case (Usecase Diagram)**
Bersifat statis. Diagram ini memperlihatkan himpunan *use-case* dan aktor- aktor (suatu jenis khusus dari kelas). Diagram ini terutama sangat penting untuk mengorganisasi dan memodelkan perilaku suatu sistem yang dibutuhkan serta diharapkan pengguna.
4. **Diagram interaksi dan sequence (Sequence Diagram)**
Bersifat dinamis. Diagram urutan adalah iterasi yang menekankan pada pengiriman pesan dalam suatu waktu tertentu.
5. **Diagram komunikasi (Communication Diagram)**
Bersifat dinamis. Diagram sebagai pengganti diagram kolaborasi UML yang

menekankan organisasi struktural dari objek-objek yang menerima serta mengirim pesan.

6. **Diagram *statechart* (*Statechart Diagram*)**

Bersifat dinamis. Diagram status memperlihatkan keadaan-keadaan pada sistem, memuat status (*state*), transisi, kejadian serta aktivitas.

7. **Diagram aktivitas (*Activity Diagram*)**

Bersifat dinamis. Diagram aktivitas adalah tipe khusus dari diagram status yang memperlihatkan aliran dari suatu aktivitas ke aktivitas lainnya dalam suatu sistem. Diagram ini terutama penting dalam pemodelan fungsi-fungsi suatu sistem dan memberi tekanan pada aliran kendali antar objek.

8. **Diagram komponen (*Component Diagram*)**

Bersifat statis. Diagram komponen ini memperlihatkan organisasi serta kebergantungan sistem/perangkat lunak pada komponen-komponen yang telah ada sebelumnya.

9. **Diagram *deployment* (*deployment diagram*)**

Bersifat statis. Diagram ini memperlihatkan konfigurasi saat aplikasi dijalankan (*run-time*). Memuat simpul-simpul beserta komponen-komponen yang di dalamnya. Kesembilan diagram ini tidak mutlak harus digunakan dalam pengembangan perangkat lunak, semuanya dibuat sesuai kebutuhan. Pada UML dimungkinkan kita menggunakan diagram-diagram lainnya misalnya data *flow* diagram, *entity relationship* diagram, dan sebagainya.

Bangunan dasar Metodologi *Unified Modeling Language* (UML)

Menurut Nugroho (2009:117). Bangunan dasar metodologi *Unified Modeling Language* (UML) menggunakan tiga bangunan dasar untuk mendeskripsikan sistem/perangkat lunak yang akan dikembangkan, yaitu:

1. **Sesuatu (*things*)**

Ada 4 (empat) *things* dalam *Unified Modeling Language* (UML), yaitu:

➤ ***Structural things***

Merupakan bagian yang relatif statis dalam model *Unified Modeling Language* (UML). Bagian yang relatif statis dapat berupa elemen-elemen yang bersifat fisik maupun konseptual.

➤ ***Behavioral things***

Merupakan bagian yang dinamis pada model *Unified Modeling Language* (UML), biasanya merupakan kata kerja dari model *Unified Modeling Language* (UML), yang mencerminkan perilaku sepanjang ruang dan waktu.

➤ ***Grouping things***

Merupakan bagian pengorganisasi dalam *Unified Modeling Language* (UML). Dalam penggambaran model yang rumit kadang diperlukan penggambaran paket yang menyederhanakan model. Paket-paket ini kemudian dapat didekomposisi lebih lanjut. Paket berguna bagi pengelompokkan sesuatu, misalnya model-model dan subsistem-subsistem.

➤ ***Annotational things***

Merupakan bagian yang memperjelas model *Unified Modeling Language* (UML) dan dapat berupa komentar-komentar yang menjelaskan fungsi serta ciri-ciri setiap elemen dalam model *Unified Modeling Language* (UML).

2. Relasi (Relationship)

Ada 4 (empat) macam *relationship* dalam *Unified Modeling Language* (UML), yaitu:

- Kebergantungan
Merupakan hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (*independent*) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (*dependent*).
- Asosiasi
Merupakan apa yang menghubungkan antara objek satu dengan objek lainnya, bagaimana hubungan suatu objek dengan objek lainnya. Suatu bentuk asosiasi adalah agregasi yang menampilkan hubungan suatu objek dengan bagian-bagiannya.
- Generalisasi
Merupakan hubungan dimana objek anak (*descendent*) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (*ancestor*). Arah dari atas ke bawah dari objek induk ke objek anak dinamakan spesialisasi, sedangkan arah berlawanan sebaliknya dari arah bawah ke atas dinamakan generalisasi.
- Realisasi
Merupakan operasi yang benar-benar dilakukan oleh suatu objek.

3. Diagram-diagram UML.

Pemetaan (*mapping*) *Unified Modeling Language* (UML) bersifat dua arah yaitu:

- Generasi kode bahasa pemrograman tertentu dari *Unified Modeling Language* (UML) *forward engineering*.
- Generasi kode belum sesuai dengan kebutuhan dan harapan pengguna, pengembang dapat melakukan langkah balik bersifat *iterative* dari implementasi ke *Unified Modeling Language* (UML) hingga didapat sistem/peranti lunak yang sesuai dengan harapan pengguna dan pengembang.

Use Case Diagram

Menurut Maimunah (2011), “*Use case* adalah deksripsi fungsi dari sebuah sistem dari perspektif pengguna”. *Use case* bekerja dengan cara mendeskripsikan tipikal interaksi antara pengguna sebuah sistem (aktor) dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai.

Use case diagram secara grafis menggambarkan interaksi antara sistem, sistem eksternal dan pengguna. Dengan kata lain *use case* diagram secara grafis mendeskripsikan siapa yang akan menggunakan sistem dan dalam cara apa pengguna (*user*) mengharapkan interaksi dengan sistem itu. *Use case* secara naratif digunakan untuk secara tekstual menggambarkan sekuensi langkah-langkah dari setiap interaksi.

Penjelasan bagian bagian *use case* diagram

1. System

Menyatakan batasan sistem dalam relasi dengan aktor-aktor yang menggunakannya (di luar sistem) dan fitur-fitur yang harus disediakan (dalam sistem). Digambarkan dengan segi empat yang membatasi semua *use case* dalam sistem terhadap pihak mana sistem akan berinteraksi. Sistem disertai label yang menyebutkan nama dari sistem, tapi umumnya tidak digambarkan karena tidak terlalu memberi arti tambahan pada diagram.

2. **Actor**

Aktor adalah segala hal diluar sistem yang akan menggunakan sistem tersebut untuk melakukan sesuatu. Bisa merupakan manusia, sistem, atau *device* yang memiliki peranan dalam keberhasilan operasi dari sistem. Cara mudah untuk menemukan aktor adalah dengan bertanya hal-hal berikut: SIAPA yang akan menggunakan sistem? APAKAH sistem tersebut akan memberikan NILAI bagi aktor?

3. **Use case**

Mengidentifikasi fitur kunci dari sistem. Tanpa fitur ini, sistem tidak akan memenuhi permintaan *user/actor*. Setiap *use case* mengekspresikan goal dari sistem yang harus dicapai. Diberi nama sesuai dengan goal-nya dan digambarkan dengan elips dengan nama di dalamnya. Fokus tetap pada goal bukan bagaimana mengimplementasikannya walaupun *use case* berimplikasi pada prosesnya nanti. Setiap *use case* biasanya memiliki *trigger*/pemicu yang menyebabkan *use case* memulai (misalnya, pasien mendaftar dan membuat janji baru atau meminta untuk membatalkan atau mengubah janji yang sudah ada) ada 2 *trigger* pertama *trigger eksternal*, seperti pelanggan memesan atau alarm kebakaran berbunyi, kedua *trigger* temporal, seperti tanggal pengembalian buku terlewat di perpustakaan atau keterlambatan bayar sewa.

4. **Assosiation**

Mengidentifikasi interaksi antara setiap actor tertentu dengan setiap *use case* tertentu. Digambarkan sebagai garis antara *actor* terhadap *use case* yang bersangkutan. Asosiasi bisa berarah (garis dengan anak panah) jika komunikasi satu arah, namun umumnya terjadi kedua arah (tanpa anak panah) karena selalu diperlukan demikian.

5. **Dependency**

Dependensi <<include>>

- Mengidentifikasi hubungan antar dua *use case* di mana yang satu memanggil yang lain.
- Jika pada beberapa *use case* terdapat bagian yang memiliki aktivitas yang sama maka bagian aktivitas tersebut biasanya dijadikan *use case* tersendiri dengan relasi dependensi setiap *use case* semula ke *use case* yang baru ini sehingga memudahkan pemeliharaan.
- Digambarkan dengan garis putus-putus bermata panah dengan notasi <<include>> pada garis.
- Arah mata panah sesuai dengan arah pemanggilan. Dependensi <<extend>>
- Jika pemanggilan memerlukan adanya kondisi tertentu maka berlaku dependensi <<extend>>.
- Note: konsep “extend” ini berbeda dengan “extend” dalam Java!
- Digambarkan serupa dengan dependensi <<include>> kecuali arah panah berlawanan.

6. **Generalization**

Mendefinisikan relasi antara dua *actor* atau dua *use case* yang mana salah satunya meng-inherit dan menambahkan atau *override* sifat dari yang lainnya. Penggambaran menggunakan garis bermata panah kosong dari yang meng-inherit mengarah ke yang di-*inherit*.

Class Diagram

Class diagram adalah “*Class* yang menggambarkan keadaan (atribut/property) suatu sistem, sekaligus menawarkan pelayanan untuk memanipulasi keadaan tersebut (metode/fungsi)”.

Menggambarkan struktur object sistem. Diagram ini menunjukkan *class* object yang menyusun sistem dan juga hubungan antara *class* object tersebut.

Class memiliki tiga area pokok:

- a. Nama (dan *stereotype*)
- b. Atribut
- c. Metoda

Sequence Diagram

Secara grafis menggambarkan bagaimana objek berinteraksi dengan satu sama lain melalui pesan pada sekuensi sebuah *use case* atau operasi. *Sequence* diagram (diagram urutan) adalah suatu diagram yang memperlihatkan atau menampilkan interaksi-interaksi antar objek di dalam sistem yang disusun pada sebuah urutan atau rangkaian waktu. Interaksi antar objek tersebut termasuk pengguna, *display*, dan sebagainya berupa pesan/*message*. *Sequence* Diagram digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai sebuah respon dari suatu kejadian/event untuk menghasilkan output tertentu. *Sequence* Diagram diawali dari apa yang me-*trigger* aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal dan output apa yang dihasilkan. Diagram ini secara khusus berasosiasi dengan *use case* diagram. *Sequence* diagram juga memperlihatkan tahap demi tahap apa yang seharusnya terjadi untuk menghasilkan sesuatu didalam *use case*. *Sequence* diagram juga dapat merubah atribut atas *method* pada *class* yang telah dibentuk oleh *class* diagram, bahkan menciptakan sebuah *class* baru. *Sequence* diagram memodelkan aliran logika dalam sebuah sistem dalam cara yang *visual*. *Sequence* diagram biasanya digunakan untuk tujuan analisa dan desain, memfokuskan pada identifikasi *method* didalam sebuah sistem. *Sequence* diagram biasanya dipakai untuk memodelkan:

- Deskripsi tentang sistem yang ada pada sebuah / beberapa *use case* pada *use case* diagram, yang menggambarkan hubungan antara *actor* dan *use case* diagram.
- Logika dari *method* (operation, *function* atau *procedure*).
- Logika dari service (*high level method*).

State Chart Diagram

Digunakan untuk memodelkan *behaviour* objek khusus yang dinamis. Diagram ini mengilustrasikan siklus hidup objek berbagai keadaan yang dapat diasumsikan oleh objek dan event-event (kejadian) yang menyebabkan objek beralih dari satu *state* ke *state* yang lain.

Activity Diagram

Secara grafis digunakan untuk menggambarkan rangkaian aliran aktivitas baik proses bisnis maupun *use case*. *Activity* diagram dapat juga digunakan untuk memodelkan *action* yang akan dilakukan saat sebuah operasi dieksekusi, dan memodelkan hasil dari *action* tersebut.

Activity diagram merupakan state diagram khusus, di mana sebagian besar *state* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya state sebelumnya (*internal processing*). Oleh karena itu *activity* diagram tidak menggambarkan *behaviour* internal sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atassecara umum. Menggambarkan proses bisnis dan urutan aktivitas dalam sebuah proses. Dipakai pada

business modeling untuk memperlihatkan urutan aktifitas proses bisnis.

Struktur diagram ini mirip *flowchart* atau *Data Flow Diagram* pada perancangan terstruktur. Sangat bermanfaat apabila kita membuat diagram ini terlebih dahulu dalam memodelkan sebuah proses untuk membantu memahami proses secara keseluruhan. *Activity* diagram dibuat berdasarkan sebuah atau beberapa *use case* pada *use case* diagram.

Pengertian Data

Data adalah representasi dari suatu fakta, yang dimodifikasi dalam bentuk gambar, kata, dan/atau angka. Manfaat data adalah sebagai satuan representasi yang dapat diingat, direkam, dan dapat diolah menjadi informasi. Karakteristiknya, data bukanlah fakta, namun representasi dari fakta. Kata sederhananya, data adalah catatan tentang fakta, atau data merupakan rekaman catatan tentang fakta. Data yang baik, adalah yang sesuai dengan faktanya. (Witarto, 2004).

Pengertian Arsip

Menurut *The Liang Gie* (2000:20) arsip sebagai kumpulan warkat-warkat yang disimpan secara teratur, berencana karena mempunyai sesuatu kegunaan agar setiap kali diperlukan dapat cepat ditemukan kembali. Arsip dapat diartikan sebagai segala jenis kertas naskah, buku, foto, film, microfilm, rekaman suara, gambar peta, bagan, atau dokumen-dokumen lain dalam segala macam bentuk dan sifatnya, aslinya, atau salinannya, serta dengan segala cara penciptaannya, dan yang dihasilkan atau diterima suatu badan sebagai bukti atas tujuan organisasi fungsi kerja-pekerjaan, kebijaksanaan-kebijaksanaan, keputusan-keputusan, prosedur-prosedur atau keputusan-keputusan suatu organisasi atau karena pentingnya informasi yang terkandung didalamnya (Suparjati, 2000).

Drs. E. Martono (2008:18) dalam kajian “kearsipan adalah pengaturan dan penyimpanan warkat/*record* atas dasar sistem tertentu serta dengan prosedur tertentu yang sistematis sehingga sewaktu-waktu diperlukan dapat ditemukan kembali dalam waktu singkat”. Dari pendapat tersebut diatas dapat disimpulkan bahwa kearsipan adalah suatu kegiatan atau proses pengaturan, penyimpanan arsip dengan menggunakan sistem tertentu, sehingga apabila arsip tersebut diperlukan dapat ditemukan kembali secara tepat dalam waktu yang singkat. Untuk dokumentasi arsip menggunakan mengisi form keterangan surat masuk dan juga surat keluar mulai dari kode surat, nomor surat, ringkasan, tujuan/asal surat, disposisi/penyaluran surat dan juga tanggal surat. Sedangkan penggunaan *Microsoft Excel* sebagai catatan *record* transaksi surat, menurut peneliti terkait arsip merupakan kumpulan naskah-naskah atau dokumen dalam corak apapun (cd, peta, perangko) yang di dalamnya memberikan keterangan-keterangan atau bukti tentang suatu kejadian, sehingga pada saat di perlukan dapat dengan mudah ditemukan. Arsip adalah naskah-naskah dinas yang dibuat dan diterima oleh semua satuan organisasi dalam lingkungan Departemen dalam negeri dalam bentuk corak apapun, baik dalam keadaan tunggal maupun berkelompok dalam pelaksanaan tugas.

Definisi Arsip Digital

Arsip Elektronik atau sering disebut juga arsip digital merupakan arsip yang sudah mengalami perubahan bentuk fisik dari lembaran kertas menjadi lembaran elektronik. Proses konversi arsip dari lembaran kertas menjadi lembaran elektronik disebut alih media. Proses alih media menggunakan perangkat komputer yang dibantu dengan perangkat

scanner kecepatan tinggi.

Hasil alih media arsip disimpan dalam bentuk *file-file* yang secara fisik direkam dalam *media* elektronik seperti *Harddisk*, CD, DVD dan lain-lain. Penyimpanan file-file ini dilengkapi dengan *Database* yang akan membentuk suatu sistem arsip *elektronik* yang meliputi fasilitas pengaturan, pengelompokan dan penamaan *file-file* hasil alih media.

Sistem arsip *elektronik* merupakan otomasi dari *sistem* arsip manual. Maka *sistem* arsip elektronik sangat tergantung dengan sistem arsip manual, dengan kata lain sistem arsip elektronik tidak akan terbentuk tanpa ada sistem arsip manual.

Gambaran Umum Institusi

Sejarah Unika Atma Jaya

Unika Atma Jaya merupakan buah gagasan yang dibahas pada rapat para Uskup se-Jawa pada Juni 1952. Dalam pertemuan itu diutarakan kemungkinan pembentukan suatu perguruan tinggi Katolik di Indonesia. Di Jakarta gagasan itu terwujud sejak didirikannya Yayasan Atma Jaya oleh sekelompok cendekiawan muda Katolik pada tanggal 1 Juni 1960. Yayasan inilah yang kemudian mendirikan sebuah perguruan tinggi Katolik dengan nama Universitas Katolik Indonesia Atma Jaya. Mereka antara lain: Ir. J.P. Cho, Ir. Lo Siang Hien-Ginting, Drs. Goei Tjong Tik, I.J. Kasimo, J.B. Legiman, S.H., Drs. F.X. Seda, Pang Lay Kim, Tan Bian Seng, Anton M. Moeliono, St. Munadjat Danusaputro, J.E. Tan, Ben Mang-Reng Say.

Pada tahun-tahun awal, Unika Atma Jaya dibantu oleh para suster Ursulin, dengan menyediakan ruang kuliah di kompleks persekolahan Ursulin, di Jalan Lapangan Banteng Utara dan kompleks Santa Theresia, Menteng. Sejak tahun 1967, Atma Jaya berangsur-angsur menempati kampus di Jalan Sudirman yang terkenal dengan nama kampus Semanggi. Selanjutnya menempati kampus Pluit, di Jakarta Utara untuk Fakultas Kedokteran (FK), Rumah Sakit Atma Jaya (RSA), dan Rumah Duka Atma Jaya.

Atma Jaya berarti Rohlah yang jaya. Roh yang jaya memberi semangat untuk selalu meningkatkan mutu pendidikan. Keunggulan akademis dan lulusan professional adalah orientasi utama.

Kini, Unika Atma Jaya telah memiliki delapan fakultas dengan tujuh belas program studi untuk program sarjana (S1) dan Program Pasca Sarjana dengan tujuh program magister: Magister Manajemen (MM) dan Magister Linguistik Terapan Bahasa Inggris (LTBI) pada tahun 1992, Magister Profesi Psikologi pada 2005, Magister Bioteknologi pada 2011, Magister Sains Psikologi, Magister Ilmu Hukum pada 2012 dan Magister Teknik Mesin pada 2013 serta satu program doktor Linguistik Terapan Bahasa Inggris (LTBI) pada tahun 2002. Untuk program sarjana (S1): Fakultas Ekonomi (FE) dan Fakultas Ilmu Administrasi Bisnis dan Ilmu Komunikasi (FIABIKOM) berdiri pada tahun 1960, Fakultas Keguruan dan Ilmu Pendidikan (FKIP) dan Fakultas Teknik (FT) tahun 1961, Fakultas Hukum (FH) tahun 1965, Fakultas Kedokteran (FK) tahun 1967, Fakultas Psikologi (FP) tahun 1992 dan Program Magister Profesi Psikologi tahun 2005, serta Fakultas Teknobiologi (FTb) tahun 2002.

Prosedur Sistem Yang Berjalan

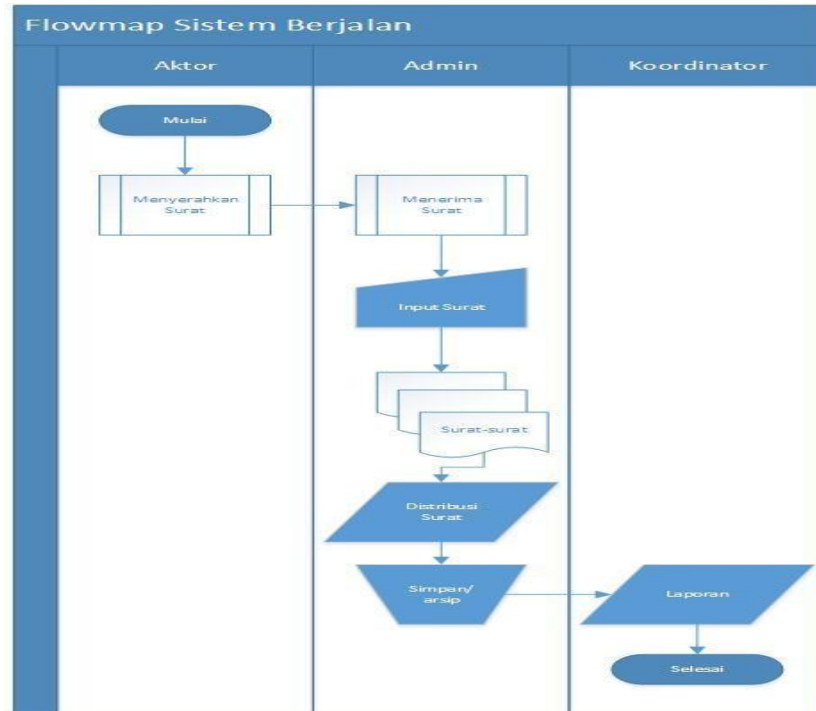
Didalam sistem lama untuk proses pengarsipan masih menggunakan cara manual, yaitu dengan mencatatkan no surat, tanggal surat, perihal surat, dan lain-lainnya, sehingga petugas harus melakukan penginputan ulang kedalam komputer.

Proses pengarsipan yang terjadi saat ini tidak diklasifikasikan menurut kelompok surat yang seharusnya dan tidak sesuai dengan standar kearsipan. Tidak adanya pemisahan

antara surat biasa, surat penting, surat segera, surat rahasia dan surat-surat lainnya.

Sistem Informasi Kearsipan yang berjalan pada Fakultas Ekonomi Dan Bisnis unika Atma Jaya Jakarta akan dijelaskan melalui diagram UML berikut:

Flowmap Sistem Berjalan

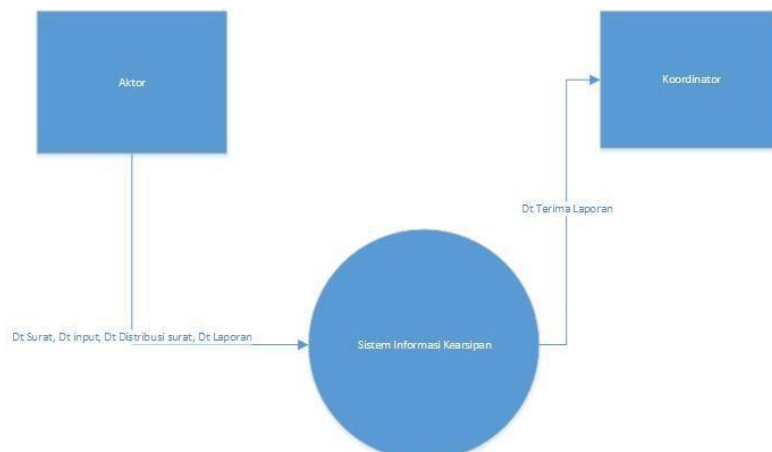


Gambar Flowmap Sistem Berjalan

Pada Flowmap di atas, masing-masing fungsionalitas sebagai berikut:

1. *Aktor* menyerahkan surat.
2. *Admin*: menerima surat dari aktor, menginput surat, distribusi surat, simpan/arsip surat.
3. *Koordinator/pimpinan*: menerima laporan arsip.

Diagram Konteks Sistem Berjalan



Gambar Diagram Konteks Sistem Berjalan

Pada Diagram Konteks di atas, masing-masing fungsionalitas sebagai berikut:

1. Aktor: bertugas mendata surat, input surat, distribusi surat, buat laporan
2. Sistem Informasi Kearsipan: mengolah semua data untuk di proses
3. Koordinator: menerima laporan

Permasalahan yang Dihadapi

Pada sistem informasi yang berjalan saat ini terdapat beberapa masalah yang apabila tidak diatasi dengan baik dapat mengganggu kegiatan kearsipan pada Fakultas Ekonomi Dan Bisnis, dan dapat berdampak pada kualitas pengarsipan surat. Masalah yang dihadapi stakeholder kesulitan dalam mengetahui seberapa banyak pengarsipan surat-surat yang terjadi dalam sebulan. Masalah yang kedua adalah kesulitan mengetahui dalam hal pengarsipan karena letak penyimpanan surat yang berubah-ubah dan masih menggunakan ordner atau box file serta memerlukan tempat yang sendiri untuk penyimpanannya.

Alternatif Pemecahan Masalah

Berdasarkan hasil observasi dan pengamatan yang dilakukan pada bagian Pengarsipan Surat, penulis menemukan beberapa kendala dan masalah yang terjadi. Oleh karena itu, penulis memberikan alternatif yang dapat digunakan untuk pemecahan masalah tersebut, diantaranya:

1. Dibangun sistem yang dibutuhkan oleh *user* dengan *database* yang terintegrasi agar mempermudah pekerjaan misalnya dengan menggunakan program aplikasi *visual basic*.
2. Dibangun suatu sistem yang berbasis *web*, sehingga memungkinkan user atau Pimpinan dapat mengakses data kapan saja.

Berdasarkan beberapa alternatif pemecahan masalah di atas penulis melakukan suatu kajian terhadap permasalahan yang ada, maka penulis memutuskan perlu dibangun aplikasi sistem yang berbasis *web* karena banyak keuntungan yang diperoleh antara lain:

1. Perkembangan penggunaan *device* berbasis *web* yang sangat pesat dan hampir setiap orang familiar menggunakan internet
2. Dapat diakses kapan saja
3. Memberikan informasi yang tepat dan akurat bagi Pimpinan untuk mendukung pengambilan keputusan

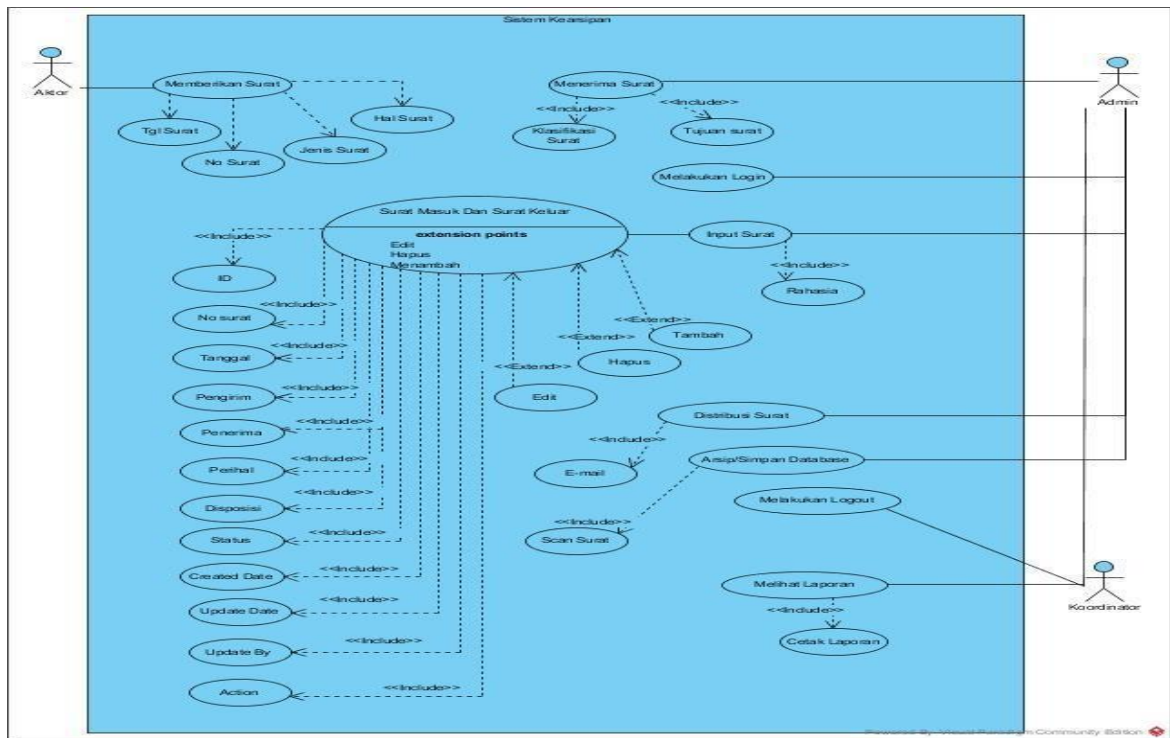
Diagram Rancangan Sistem

Rancangan Fungsional (*Functional Design*)

Rancangan fungsional bertujuan untuk mengkomunikasikan rancangan sistem kepada *user*, dengan menggunakan *Use Case Diagram* dan *Activity Diagram*, rancangan-rancangan tersebut dibuat seperti pada gambar dibawah ini;

Use Case Diagram

Dalam sistem informasi kearsipan dapat digambarkan dalam struktur dibawah



Gambar Use Case Diagram yang diusulkan

Pada Use Case Diagram terdapat:

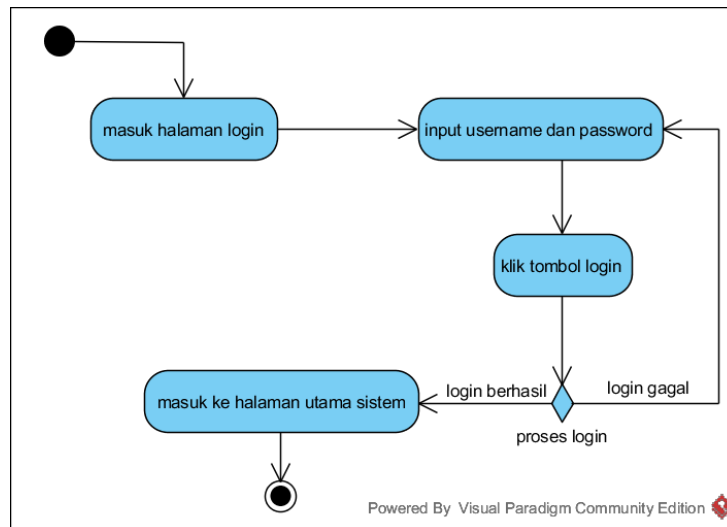
- a. 1 (satu) sistem, yaitu sistem informasi kearsipan
- b. 33 (tiga puluh tiga) usecase, meliputi usecase:
 1. Memberikan surat
 2. Tanggal surat
 3. No surat
 4. Jenis surat
 5. Perihal surat
 6. Menerima surat
 7. Klasifikasi surat
 8. Tujuan surat
 9. Melakukan login
 10. Input surat
 11. Surat masuk dan surat keluar
 12. Id surat
 13. No surat
 14. Tanggal surat
 15. Pengirim
 16. Penerima
 17. Perihal surat
 18. Disposisi
 19. Status
 20. Created date
 21. Update date
 22. Update by
 23. Action

24. Edit
 25. Hapus
 26. Menambah
 27. Distribusi surat
 28. E-mail
 29. Arsip/simpan database
 30. Scan surat
 31. Melakukan logout
 32. Melihat laporan
 33. Cetak laporan
- c. 3 (tiga) *actor*, yaitu *aktor*, admin dan koordinator
1. Aktor: menyerahkan surat
 2. Admin: mengelola surat masuk dan surat keluar sampai proses pengarsipan
 3. Koordinator: mengecek dan melihat laporan surat

Activity Diagram

Activity Diagram menggambarkan berbagai alur aktivitas dalam sistem yang dirancang, bagaimana masing-masing alur berawal, *decision* yang mungkin terjadi, dan bagaimana alur itu berakhir. *Activity Diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi.

a. Activity Diagram Login

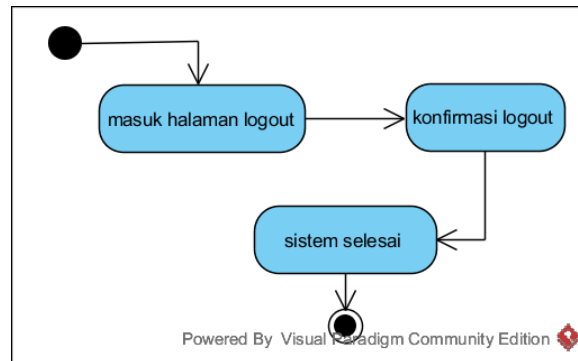


Gambar Activity Diagram Login Sistem yang diusulkan

Pada *Activity Diagram Login* gambar 4.2.a, terdapat:

1. 1 (satu) *Initial Node*
2. 4 (empat) *Action*, meliputi:
 - a. Masuk halaman login
 - b. Input username dan password
 - c. Klik tombol login
 - d. Masuk kehalaman utama sistem
3. 1 (satu) *Decision Node*
4. 1 (satu) *Activity Final Node*

b. Activity Diagram Logout Sistem

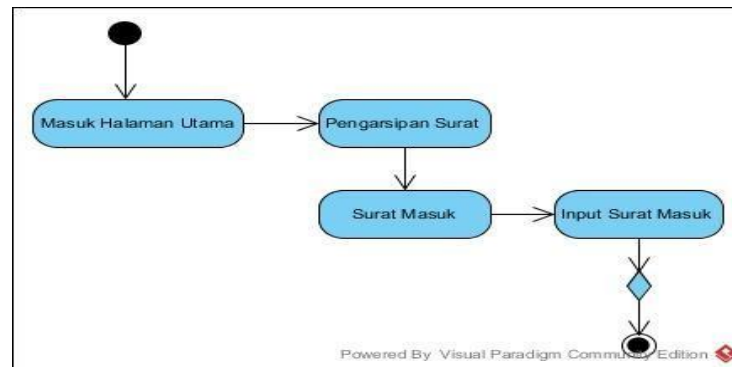


Gambar Activity Diagram Logout Sistem

Pada *Activity Diagram Logout* gambar 4.2.b, terdapat:

1. 1 (satu) *Initial Node*
2. 3 (tiga) *Action*, meliputi:
 - a. Masuk halaman *logout*
 - b. Konfirmasi *logout*
 - c. Sistem selesai
3. 1 (satu) *Activity Final Node*

c. Activity Diagram Surat Masuk

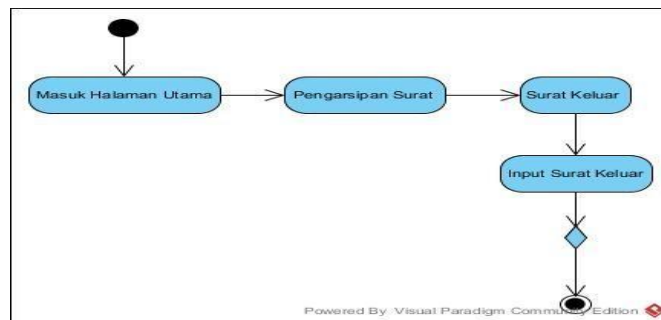


Gambar Activity Diagram Surat Masuk

Pada *Activity Diagram Surat Masuk* gambar 4.2.c, terdapat:

1. 1 (satu) *Initial Node*
2. 4 (empat) *Action*, meliputi:
 - a. Masuk Halaman Utama *User*
 - b. Pilih Pengarsipan Surat
 - c. Pilih Surat Masuk
 - d. Input Surat Masuk
3. 1 (satu) *Decision Node*
4. 1 (satu) *Activity Final Node*

d. Activity Diagram Surat Keluar

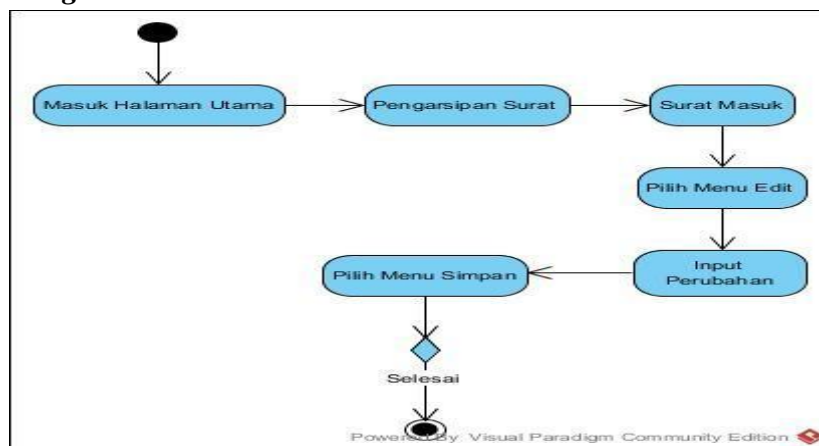


Gambar Activity Diagram Surat Keluar

Pada Activity Diagram Surat Masuk gambar 4.2.c, terdapat:

1. 1 (satu) Initial Node
2. 4 (empat) Action, meliputi:
 - a. Masuk Halaman Utama User
 - b. Pilih Pengarsipan Surat
 - c. Pilih Surat Keluar
 - d. Input Surat Keluar
3. 1 (satu) Decision Node
4. 1 (satu) Activity Final Node

e. Activity Diagram Ubah Surat Masuk



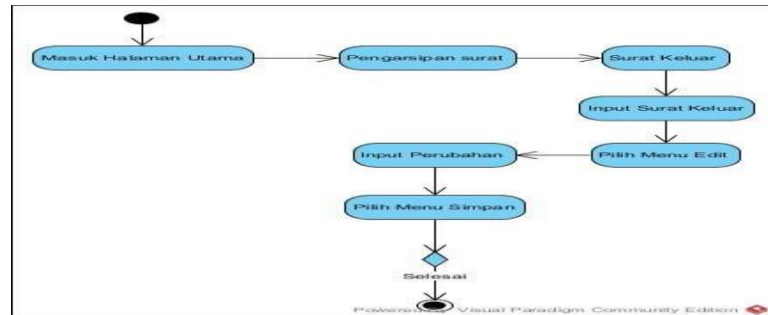
Gambar Activity Diagram Ubah Surat Masuk

Pada Activity Diagram Ubah Surat Masuk gambar 4.2.e, terdapat:

1. 1 (satu) Initial Node
2. 6 (enam) Action, meliputi:
 - a. Masuk halaman utama
 - b. Pilih pengarsipan surat
 - c. Pilih surat masuk
 - d. Pilih menu edit
 - e. Input perubahan
 - f. Pilih menu simpan
3. 1 (satu) Decision Node

4. 1 (satu) *Activity Final Node*

f. Activity Diagram Ubah Surat Keluar

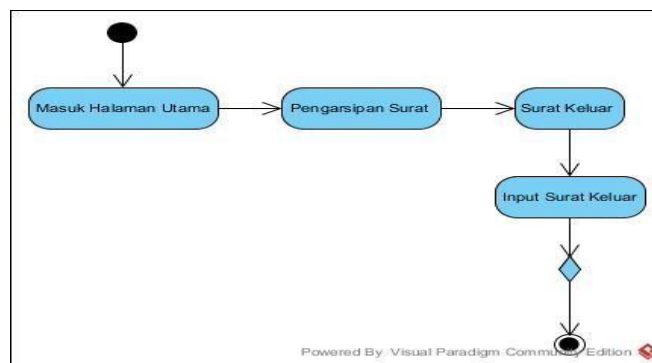


Gambar Activity Diagram Ubah Surat Keluar

Pada Activity Diagram Ubah Surat Keluar gambar 4.2.f, terdapat:

1. 1 (satu) *Initial Node*
2. 6 (enam) *Action*, meliputi:
 - a. Masuk halaman utama
 - b. Pilih pengarsipan surat
 - c. Pilih surat keluar
 - d. Pilih menu edit
 - e. Input perubahan
 - f. Pilih menu simpan
3. 1 (satu) *Decision Node*
4. 1 (satu) *Activity Final Node*

g. Activity Diagram Menambah Surat Masuk



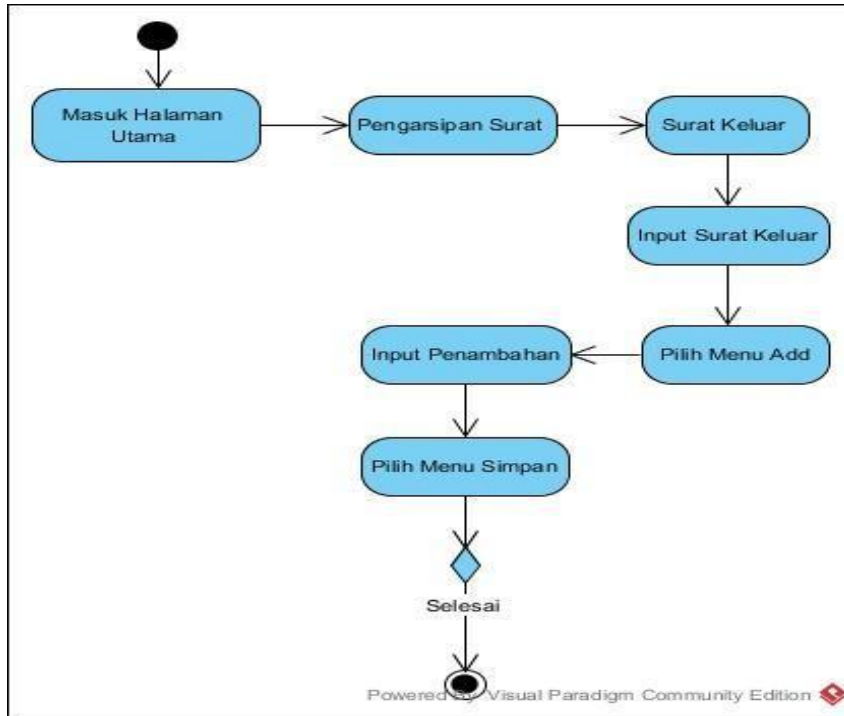
Gambar Activity Diagram Tambah Surat Masuk

Pada Activity Diagram Menambah Surat Masuk gambar 4.2.f, terdapat:

1. 1 (satu) *Initial Node*
2. 6 (enam) *Action*, meliputi:
 - a. Masuk halaman utama
 - b. Pilih pengarsipan surat
 - c. Pilih surat masuk
 - d. Pilih menu add
 - e. Input penambahan
 - f. Pilih menu simpan

3. 1 (satu) *Decision Node*
4. 1 (satu) *Activity Final Node*

h. Activity Diagram Menambah Surat Keluar



Gambar Activity Diagram Tambah Surat Keluar

Pada Activity Diagram Menambah Surat Keluar gambar 4.2.h, terdapat:

1. 1 (satu) *Initial Node*
2. 6 (enam) *Action*, meliputi:
 - a. Masuk halaman utama
 - b. Pilih pengarsipan surat
 - c. Pilih surat keluar
 - d. Pilih menu add
 - e. Input penambahan
 - f. Pilih menu simpan
3. 1 (satu) *Decision Node*
4. 1 (satu) *Activity Final Node*

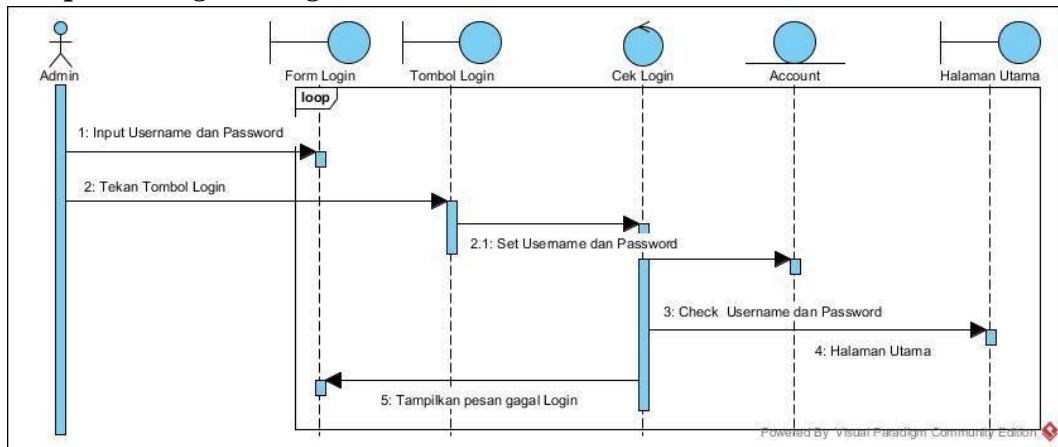
Behavioral Design

Behavioral Design merupakan rancangan yang berhubungan dengan interaksi *user* dengan komputer. Rancangan ini menggunakan *Sequence Diagram* dan *Statechart Diagram* untuk menggambarkan bagaimana proses kerja sistem aplikasi yang dibuat.

Sequence Diagram

Sequence Diagram digunakan untuk menggambarkan perilaku pada sebuah skenario. Diagram ini menunjukkan sejumlah contoh objek dan *message* yang diletakkan diantara obyek-obyek di dalam *use case*.

a. Sequence Diagram Login Sistem

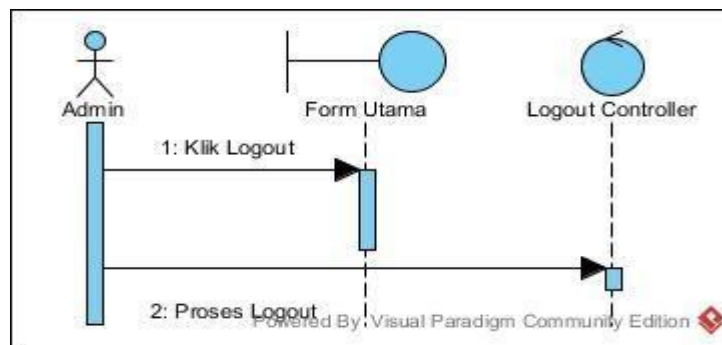


Gambar Sequence Diagram Login Sistem

Pada *Sequence Diagram* gambar 4.2.2.1.a, terdapat 1 (satu) aktor (admin), 3 (tiga) *Boundary Lifeline* (form login, tombol login, halaman utama), 1 (satu) *Control Lifeline* (cek login), 1 (satu) *Entity Lifeline* (account), 1 (satu) *Loop Combined Fragment*. Kegiatan atau *message* yang terjadi dalam diagram tersebut, antara lain:

1. *Input username dan password*
2. *Tekan tombol login*
3. *Set username dan password*
4. *Check username dan password*
5. *Ke halaman utama*
6. *Tampilkan pesan gagal login*

b. Sequence Diagram Logout Sistem

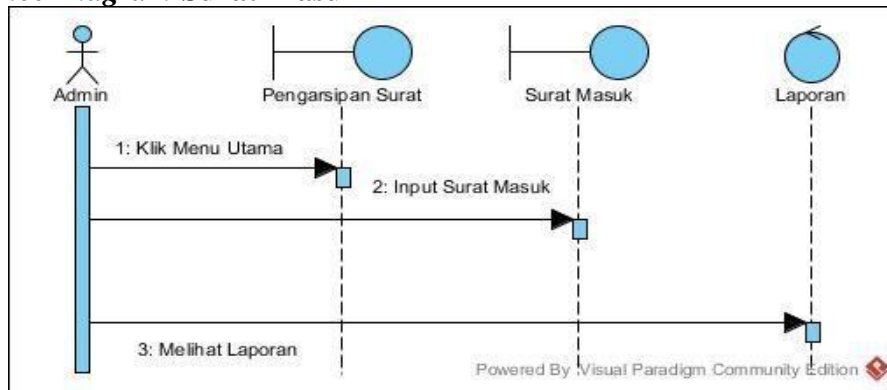


Gambar Sequence Diagram Logout Sistem

Pada *Sequence Diagram* gambar 4.2.2.1.b, terdapat 1 (satu) aktor (*Admin*), 1 (satu) *Boundary Lifeline* (form utama) 1 (satu) *Control Lifeline* (logout). Kegiatan atau *message* yang terjadi dalam diagram tersebut, antara lain:

1. *Menekan tombol logout*
2. *Ke form utama*
3. *Proses logout*

c. **Sequence Diagram Surat Masuk**



Gambar *Sequence Diagram* Surat Masuk

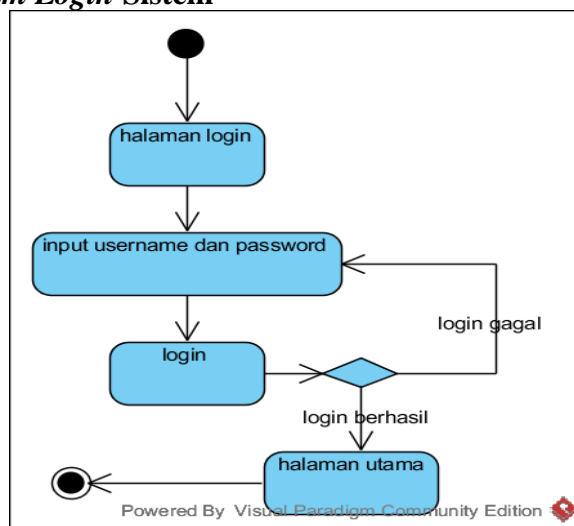
Pada *Sequence Diagram* gambar 4.2.2.1.c, terdapat 1 (satu) aktor (admin), 2(dua) *Boundary Lifeline* (pengarsipan surat, tombol surat masuk), 1(satu) *Control Lifeline* (laporan). Kegiatan atau *message* yang terjadi dalam diagram tersebut, antara lain:

1. AsKlik tombol pengarsipan surat
2. Klik tombol pengasrsipan surat
3. Input surat masuk
4. Klik tombol surat masuk
5. Laporan

Statechart Diagram

Statechart diagram adalah suatu diagram yang menggambarkan daur hidup (*behavior pattern*) dari sebuah objek dari awal *object* tersebut diinisialisasi hingga *object* tersebut didestroy. Sebuah *statetechart* terdiri dari satu initial *state*, nol atau lebih final *state*, *state*, dan *event*. Berikut ini *statechart* diagram pada sistem informasi Kearsipan Fakultas Ekonomi Dan Bisnis Unika Atma Jaya Jakarta.

a. **Statechart Diagram Login Sistem**

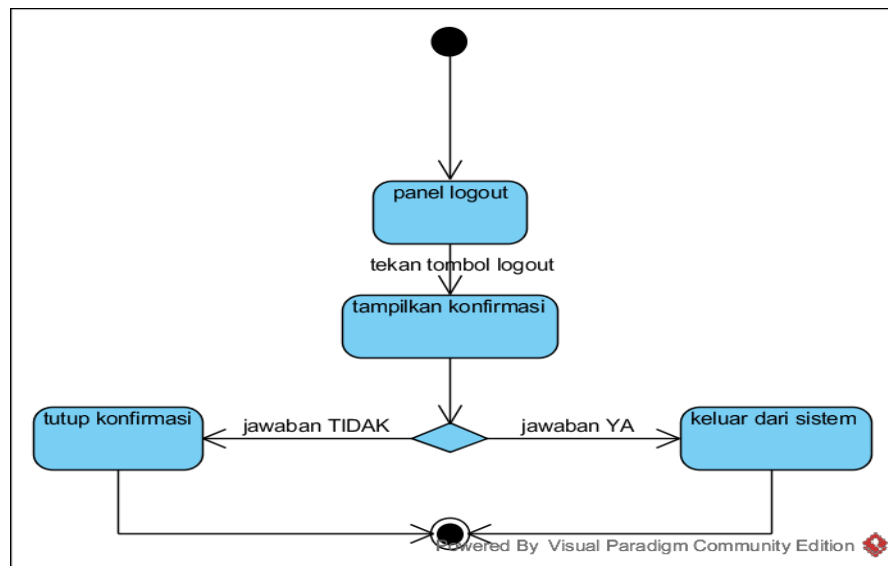


Gambar *Statechart Diagram* Login Sistem

Pada *Statechart diagram* Login Sistem Informasi Kearsipan pada Fakultas Ekonomi Dan Bisnis Unika Atma Jaya Jakarta gambar terdapat:

1. 1 (satu) initial *pseudo state*
2. 1 (satu) final *state*
3. 1 (satu) *choice*
4. 4 (empat) *state*, meliputi:
 - a. Halaman *login*
 - b. Input *username* dan *password*
 - c. *Login*
 - d. Halaman utama

b. Statechart Diagram Log out Sistem



Gambar Statechart Diagram Logout Sistem

Pada Statechart diagram Logout Sistem Informasi Kearsipan pada Fakultas Ekonomi Dan Bisnis Unika Atma Jaya Jakarta gambar terdapat:

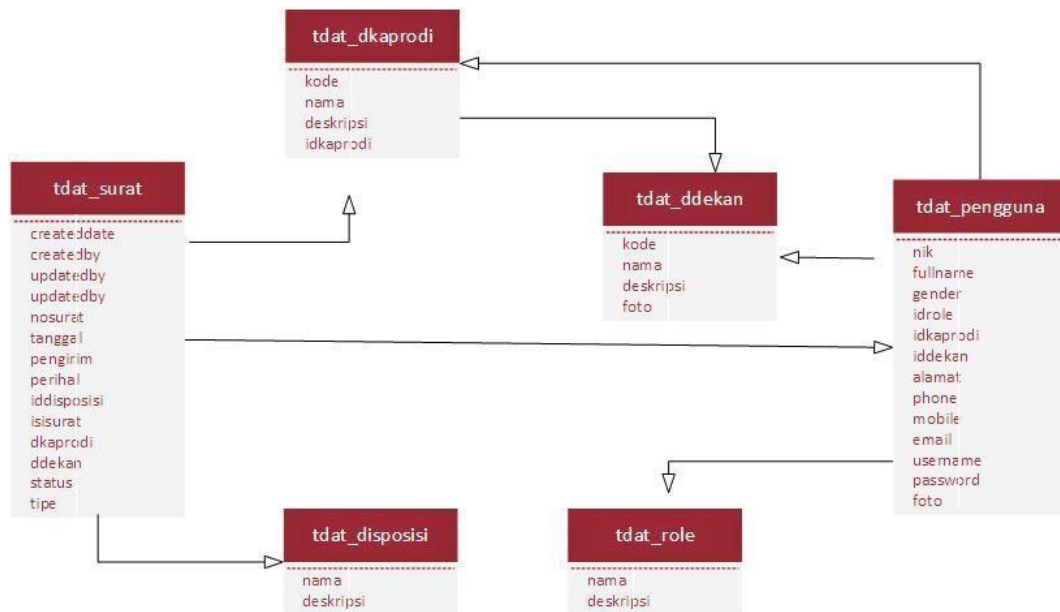
1. 1 (satu) initial *pseudo state*
2. 1 (satu) final *state*
3. 1 (satu) *choice*
4. 4 (empat) *state*, meliputi:
 - a. Panel *logout*
 - b. Tampilkan konfirmasi
 - c. Tutup konfirmasi
 - d. Keluar dari sistem

Rancangan Struktural (Structural Design)

Rancangan struktural menggunakan *Class Diagram* untuk menggambarkan model data.

Class Diagram

Class Diagram merupakan sebuah spesifikasi yang jika diintansiasi akan menghasilkan sebuah obyek dan merupakan inti dari pengembangan dan desain berorientasi obyek. *Class* menggambarkan keadaan suatu sistem sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut.



Gambar *Class Diagram* Sistem Informasi Kearsipan

Pada *Class Diagram* Sistem Informasi Kearsipan Fakultas Ekonomi Dan Bisnis Unika Atma Jaya Jakarta gambar 4.2.3.1 terdapat 6 (enam) *class*, meliputi:

1. *Class* surat
2. *Class* disposisi
3. *Class* dkaprodi
4. *Class* ddekan
5. *Class* role
6. *Class* pengguna

Rancangan Basis Data/Database

Spesifikasi *database* merupakan desain basis data yang dianggap telah normal. Desain *database* menjelaskan media penyimpanan yang digunakan, isi yang disimpan, *primary key*, panjang *record*, dan struktur. Spesifikasi *database* yang digunakan dalam sistem yang akan dibangun adalah sebagai berikut:

1. Nama File : Tabel Login
- Akronim : Login.MYD
- Fungsi : Untuk masuk kehalaman utama
- Tipe File : File Master
- Organisasi File : *Indexed Sequential*
- Akses File : Random
- Media : *Harddisk*
- Panjang Record : 224 Byte
- Kunci Field : Id
- Software : MySQL

Tabel Struktur Tabel Login

No	Elemen data	Nama Field	Tipe	Size	Keterangan
1	Id	Id	Int	4	<i>Primary Key</i>
2	Nama	Nama	Varchar	60	
3	User_id	Userid	Varchar	60	
4	Password	Password	Varchar	100	

2. Nama File : Tabel Pengarsipan Surat
 Akronim : Surat.MYD
 Fungsi : Untuk menginput surat.
 Tipe File : File Master
 Organisasi File : *Indexed Sequential*
 Akses File : Random
 Media : *Harddisk*
 Panjang Record : 5610 Byte
 Kunci Field : tdat_surat
 Software : MySQL

Tabel Struktur Tabel Pengarsipan Surat

No	Elemen data	Nama Field	Tipe	Size	Keterangan
1	Data Surat	tdat_surat	Char	4	<i>Primary Key</i>
2	Id	id	Int	11	
3	No Surat	no surat	Varchar	100	
4	Tanggal	tanggal	Date		
5	Pengirim	pengirim	Varchar	200	
6	Perihal	perihal	Varchar	200	
7	Disposisi	iddisposisi	Int	11	
8	Isi Surat	isi surat	Varchar	5000	
9	Status	status	Varchar	20	
10	Created Date	created date	Date time		
11	Created By	created by	Int	11	
12	Update Date	update date	Date time		

13	Updated By	updated by	Int	11	
14	Tipe	tipe	Varchar	20	
15	Kaprodi	idkaprodi	Int	11	
16	Dekan	iddekan	Int	11	

3. Nama File : Tabel Pengguna
 Akronim : Surat.MYD
 Fungsi : Untuk pengguna.
 Tipe File : File Master
 Organisasi File : *Indexed Sequential*
 Akses File : Random
 Media : *Harddisk*
 Panjang Record : 7264 Byte
 Kunci Field : tdat_pengguna
 Software : MySQL

Tabel Struktur Tabel Pengguna

No	Elemen data	Nama Field	Tipe	Size	Keterangan
1	Pengguna	tdat_pengguna	Char	50	<i>Primary Key</i>
2	Id	id	Int	11	
3	NIK	nik	Varchar	10	
4	Full Name	fullname	Varchar	200	
5	Gender	gender	Varchar	20	
6	Birth Date	birthdate	Date		
7	Role	idrole	Int	11	
8	Kabupaten/kaprodi	idkabupaten	Int	11	
9	Provinsi/dekan	idprovinsi	Int	11	
10	Alamat	alamat	Varchar	600	
11	Phone	phone	Varchar	20	
12	Mobile	mobile	Varchar	20	
13	Email	email	Varchar	100	

14	Username	username	Varchar	100	
15	Password	password	Varchar	100	
16	Foto	foto	Varchar	6000	

Rancangan *Prototype* atau Tampilan

Tahap ini merupakan gambaran yang jelas mengenai rancang bangun yang lengkap, juga sebagai pemenuhan kebutuhan dari para pengguna sistem. Berikut merupakan *prototype* dari perancangan sistem kearsipan yang akan dibuat.

a. Halaman Tampilan Login

Gambar Halaman tampilan login

Sistem informasi kearsipan ini dapat diakses setelah halaman *login* diisi *username* dan *password* dengan benar sesuai dengan sistem yang dipilih. Halaman ini berfungsi sebagai pengatur hak akses terhadap *user* sehingga keamanan pada sistem ini dapat terjaga dengan baik.

b. Halaman Tampilan Utama

Gambar Halaman tampilan utama

c. Halaman Tampilan Surat Masuk



Gambar Halaman tampilan surat masuk

d. Halaman Tampilan Surat Keluar



Gambar Halaman tampilan surat keluar

Implementasi Sistem yang diusulkan Spesifikasi *Hardware*, *Software* dan *Brainware*

Untuk mengimplementasikan system ini dengan baik, penulis mengklasifikasikannya sebagai berikut:

a. Perangkat Keras (*Hardware*)

Perangkat keras yang digunakan adalah sebagai berikut:

- 1) Processor Pentium Dual Core cpu e 5700 @3.00 Ghz
- 2) Memory 1 GB RAM DDR2
- 3) Harddisk 60 GB
- 4) Monitor Standart
- 5) Comic USB Keyboard
- 6) Monitor Comic Optical Mouse USB
- 7) Printer HP DESKJET/LasetJet

b. Perangkat Lunak (*Software*)

Perangkat ini digunakan untuk membuat dan menjalankan aplikasi ini adalah sebagai berikut:

- 1) Sistem Operasi *Windows XP*
- 2) *Adobe Reader*
- 3) *XAMPP*
- 4) *Adobe Dreamweaver CS 8*
- 5) *Google Chrome*

c. Hak Akses (*Brainware*)

Sistem yang diusulkan ini mengklasifikasikan hak akses *user* sebagai berikut:

- 1) Administrator: *user* yang mempunyai hak akses dalam pengaturan *account* yang diperbolehkan untuk masuk dalam sistem ini.
- 2) Koordinator: *user* yang mempunyai hak akses dalam melihat hasil dari proses manajemen kearsipan yang telah dilakukan. Hak akses tersebut terbatas pada kegiatan, melihat statistik menu dan laporan kearsipan.

Pengujian Unit

Pengujian terhadap program yang dibuat menggunakan *blackbox testing* yang fokus terhadap proses masukan (*input*) dan keluaran (*output*) program.

Masukan (*Input*) merupakan awal dimulainya proses informasi, bahan mentah dari informasi adalah data yang terjadi dari transaksi-transaksi yang dilakukan oleh organisasi (Jogiyanto, 1997).

Keluaran (*Output*) adalah produk dari sistem informasi, alasan utama suatu sistem informasi yang dibentuk adalah karena dibutuhkannya keluaran ini, isi dari keluaran merupakan informasi yang berguna bagi para pemakainya, keluaran dapat dihasilkan oleh *soft copy device* maupun oleh *hard copy device* (Jogiyanto, 1997). Berikut ini hasil pengujian unit yang telah dilakukan:

1. Pengujian Terhadap *Form Login Admin*.

Tabel Pengujian *Form Login Admin*

No	Skenario Pengujian	Test Case	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
1.	<i>Username</i> dan <i>password</i> tidak diisi kemudian klik tombol login	<i>Username</i> : (kosong) <i>Password</i> : (kosong)	Sistem akan menolak akses user dan menampilkan “ <i>Username</i> dan <i>password</i> tidak boleh kosong”	Sesuai harapan	Valid
2.	Mengetikan <i>Username</i> dan <i>password</i> tidak diisi atau kosong kemudian klik tombol login	<i>Username</i> : admin <i>Password</i> : (kosong)	Sistem akan menolak akses user dan menampilkan “ <i>password</i> tidak boleh kosong”	Sesuai harapan	Valid

3.	Username tidak diisi (kosong) dan password diisi kemudian klik tombol login	Username : kosong Password : admin	Sistem akan menolak akses user dan menampilkan "Username tidak boleh kosong!"	Sesuai harapan	Valid
4.	Mengetik salah satu kondisi salah pada Username atau password kemudian klik tombol login	Username : admin (benar) Password : caca (salah)	Sistem akan menolak akses user dan menampilkan "Login Gagal"	Sesuai harapan	Valid
5.	Mengetik Username dan password dengan data yang benar kemudian klik tombol login	Username : admin (benar) Password : admin (benar)	Sistem menerima akses login dan kemudian langsung menampilkan menu utama login	Sesuai harapan	Valid

2. Pengujian Terhadap Menu Tambah Kearsipan

Tabel Pengujian Form Menu Tambah Kearsipan

No	Skenario Pengujian	Test Case	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
1.	Materi tidak diisi kemudian klik tombol simpan	Materi: (kosong)	Sistem akan menolak akses user dan menampilkan "materi tidak masih kosong"	Sesuai harapan	Valid

3. Pengujian Terhadap Menu Edit Kearsipan

Tabel Pengujian Form Menu Edit Kearsipan

No	Skenario Pengujian	Test Case	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
1.	Materi tidak diisi kemudian klik tombol simpan	Materi: (kosong)	Sistem akan menolak akses user dan menampilkan "materi tidak masih kosong"	Sesuai harapan	Valid

4. Pengujian Terhadap Form Logout Admin.

Tabel Pengujian *Form Logout Admin*

No	Skenario Pengujian	Test Case	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
1.	Ketika memilih menu logout, akan keluar dari aplikasi	Menu Logout	Memilih menu logout, sistem akan keluar dari aplikasi	Sesuai harapan	Valid

5. Pengujian Terhadap Tombol *Delete*.

Tabel Pengujian Tombol *Delete*

No	Skenario Pengujian	Test Case	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
1.	Ketika memilih tombol delete, akan menghapus data dari aplikasi	Tombol Delete	Memilih menu delete, sistem akan menghapus dari aplikasi	Sesuai harapan	Valid

PENUTUP

Berdasarkan analisa yang telah dilakukan terhadap sistem informasi kearsipan pada Kearsipan Fakultas Ekonomi Dan Bisnis Unika Atma Jaya Jakarta, maka penulis dapat menarik kesimpulan sebagai berikut:

1. Sistem Informasi tersebut memudahkan penyimpanan surat-surat karena tidak memerlukan tempat penyimpanan khusus.
2. Sistem Informasi tersebut memudahkan proses penginputan surat karena sudah diklasifikasikan berdasarkan pentingnya surat sehingga tidak terjadi penumpukan surat.
3. Sistem Informasi tersebut jika sudah usah dapat langsung dimusnahkan.
4. Sistem Informasi tersebut terpusat pada database yang dengan mudah dapat dilakukan jika memerlukan surat yang sudah di arsip.

DAFTAR PUSTAKA

- Abdul Kadir. (2008). Tuntunan Praktis Belajar *Database* Menggunakan *MySQL*, C.V Andi Offset. Yogyakarta.
- Adi Nugroho. (2005). Analisis dan Perancangan Sistem Informasi Dengan Metodologi Berorientasi Objek. Informatika. Bandung.
- Agus Saputra. (2012). Membuat Aplikasi Absensi Dan Kuesioner untuk Panduan Skripsi. PT. Elex Media Komputindo. Jakarta
- Ahmadi , Hermawan, 2013. *E-Business & E-Commerse*. Andi: Denpasar

- Alan Nur Aditya. (2011). *Jago PHP & MySQL*, Bekasi – Jawa Barat: Dunia Komputer.
- Al-Bahra bin Ladjamudin. (2005). *Analisis dan Desain Sistem Informasi*. Yogyakarta: Graha Ilmu.
- Amsyah, Zulkifli. 2003. *Manajemen Kearsipan*. Edisi Kesepuluh. PT. Gramedia Utama. Jakarta.
- Asropudin, Pipin. (2013). *Teknologi Informasi Komunikasi*. Bandung : Titian Ilmu. *Booch, G. James, R. Ivar, J.* (2005). *The Unified Modeling Language User Guide Second Edition*. United State: Addison Wesley Professional.
- Budiharto Sanusi (Meditek, 1996), *Administrasi Arsip*
Burket, L.W. (1977). *White Lesions. Dalam. Burket's Oral Medicine. Diagnosis and treatment. Malcolm A Lynch* (editor). Ed. Ke-7. JB. Lippincott. Philadelphia
- Conny R. Semiawan. 2010. *Metode Penelitian Kualitatif*. Jakarta:Grasindo.
- Cook ,dkk. (1997). *Management and organizational* . USA: McGraw Hill. Firdaus. (2007). *7 Jam Belajar Interaktif PHP & MySQL dengan Dreamwever*. Palembang:Maxikom.
- Gie, The Liang 2000. *Administrasi Perkantoran*. Yogyakarta : *Modern Liberty*
- H Al Fatta - Yogyakarta: Andi Offset, (2007)
- HD Erinawati - Speed-Sentra Penelitian *Engineering* dan Edukasi, 2013. Henderi,Maimunah, Randy Andrian. (2011). "Desain aplikasi *E-Learning* Sebagai Media Pembelajaran Artificial Informatics".
- Henderi. (2008). *Unified Modelling Language*. Tangerang: Raharja Enrichment Centre (REC).
- Herlawati, Prabowo Pudjo Widodo. (2011). *Menggunakan UML*. Bandung: Informatika.
- Hidayati, (2007) <http://informatika.web.id/elisitasi.html>
- I.G Wursanto, (1989), *Manajemen Kepegawaian*. Kanisius, Yogyakarta. Jogyanto, HM., (1999), *Pengenalan Komputer*, Penerbit Andi Offset, Yogyakarta. Leman (1998). *Metodologi Pengembangan Sistem Informasi*. Jakarta, PT Elex
- Awaludin, M. (2018). Penerapan Sistem Piranti Lunak Personal Finance Berbasis Android untuk Peningkatkan Kualitas Ekonomi Individu. *Jurnal Sistem Informasi Universitas Suryadarma*, 3(2), 107–114. <https://doi.org/10.35968/jsi.v3i2.67>
- Media Komputindo. ISBN 979-20-0467-X.

- Linda Marlinda. (2004). *Sistem Basis Data*. Yogyakarta: Andi Offset.
- Madcoms, Litbang. (2011). *Aplikasi Web Database dengan Dreamweaver dan php-MySQL*. Yogyakarta: Andi.
- Martono Aris, Padeli, Fitria Dina Murad. (2009). *Pengembangan Sistem Database Penempatan Tenaga Kerja Berbasis Web*.
- Martono dan Agus Harjito, (2008), *Manajemen Keuangan*, edisi pertama cetakan ketujuh, penerbit : Ekonesia, Yogyakarta.
- Miarso, Yusufhadi. (2007). *Menyemai Benih Teknologi Pendidikan*. Prenada Media Group. Jakarta.
- Notohadiprawiro, T. (2006). *Metode Penelitian dan Penulisan Ilmiah*. Yogyakarta: Universitas Gajah Mada.
- Priyadi (2004). *Kolaborasi SQL & ERD Dalam Implementasi Database*.
- Robert G Murdick, dkk, Sistem *Informasi Untuk Manajemen Modern*, Jakarta: Erlangga, (1991).
- Robert K. Leitch dan K. Roscoe Davis, (1983), *Accounting Information Systems*, Prentice-Hall, New Jersey.
- Siallagan, Sariadin. (2009). *Pemrograman Java*. Yogyakarta: Andi Yogyakarta. Suparjati, 2000, *Tata Usaha dan Kearsipan (Seri Administrasi Perkantoran)*, Yogyakarta: Kanisius.
- Supriyanto, Dodit. (2008). *Buku Pintar Pemrograman PHP*. oase media. Bandung.
- Witarto. (2004). "Memahami Sistem Informasi". *Studi Dalam: Pendekatan Praktis Rekayasa Sistem Informasi Melalui Kasus-kasus Sistem Informasi di Sekitar Kita*. Edisi Pertama. Bandung: Informatika.
- Yakub, 2012. *Pengantar sistem Informasi*, Yogyakarta: Graha Ilmu
Yudhim.blogspot.com/2008/02/sejarah-internet/ 5 Juni 2009.
- Yuniarti. Evi, dkk. 2012. *Kinerja Laporan Keuangan Untuk Pengambilan Keputusan Pemberian Kredit Modal Kerja*. Lampung: Politeknik Negeri Lampung.

