

**RANCANG BANGUN APLIKASI DIGITAL ARSIP
SURAT MASUK DAN SURAT KELUAR
BERBASIS WEB
(Studi Kasus Dinas Perhubungan Provinsi Jawa Barat)**

BAB I

Saat ini arsip digital merupakan suatu basis data digital, yang dibangun internet, budaya situs web dan teknik informasi.

Tujuan untuk pengarsipan elektronik tersebut merupakan untuk menangkap, mengklasifikasikan, menyimpan, mengambil, serta menemukan kembali arsip elektronik. Arsip elektronik dikembangkan karena pada saat ini dunia akan selalu berkaitan dengan dunia digital, sebagai antisipasi untuk dokumen dalam intensitas besar, mengurangi biaya kertas, sangat mengurangi waktu pencarian.

Perkembangan teknologi informasi memberi sangat pengaruh besar dalam hal manajemen perkantoran terutama dalam hal penanganan surat. Surat merupakan sarana komunikasi untuk menyampaikan informasi tertulis oleh suatu pihak kepada pihak lain. Jenis surat berdasarkan prosedur pengurusannya terdiri dua prosedur yaitu surat masuk dan surat keluar. Surat masuk merupakan surat yang di terima dari instansi lain, dan surat keluar merupakan surat yang dikirim kepada instansi lain. Arsip (*record*) merupakan suatu catatan yang tertulis, baik dalam bentuk gambar ataupun bagan yang dapat memuat keterangan-keterangan mengenai sesuatu subyek (pokok persoalan) ataupun peristiwa-peristiwa yang dibuat orang untuk membantu daya ingatan orang. Adapun arsip merupakan suatu kumpulan dokumen yang disimpan secara sistematis yang memiliki suatu kegunaan agar setiap kali diperlukan dapat secara cepat ditemukan kembali.

Arsip merupakan berkas berharga bagi setiap Instansi ataupun suatu Perusahaan, sehingga secara tidak langsung dalam proses perencanaan, pelaksanaan, penyelenggaraan instansi membuat sering kali timbul masalah dibidang kearsipan. Adapun pengarsipan surat menyurat pada suatu perusahaan telah terkomputerisasi dengan menggunakan suatu *tools* atau alat seperti *microsoft excel*, akan tetapi yang terkomputerisasi hanya bagian penyimpanan pengarsipan surat ada beberapa tahapan yang perlu melakukan proses manualisasi dalam beberapa hal antara lain yaitu dalam hal agenda surat, dan penyampaian (Disposisi) surat masuk. Buku Agenda yaitu pencatatan dengan buku agenda dilakukan oleh instansi yang belum menerapkan sistem kartu kendali, dan Kartu Kendali yaitu helai tipis berukuran 10 x 15 cm berisi kolomkolom untuk mencatat surat masuk dan surat keluar serta untuk mengendalikan surat tersebut. Ketika ada surat masuk pada suatu perusahaan surat tersebut akan di terima di bagian Umum dimana surat tersebut akan di data pada agenda surat terlebih dahulu sebagai tanda ada nya surat masuk setelah itu surat akan di disposisikan kepada atasan yang mana atasan berhak menunjukan kepada bagian apa saja surat tersebut di peruntukan.

Maka dengan terdapatnya suatu permasalahan tersebut, Tujuan peneliti membahas permasalahan ini adalah untuk membantu pengelola arsip pada suatu perusahaan dalam meningkatkan akurasi data dengan merancang dan membuat

aplikasi perangkat lunak yang berbasis *website*, Dengan adanya perangkat lunak yang berbasis *website* dalam melakukan pengolahan surat masuk dan keluar pada perusahaan dapat mempercepat dan mempermudah penyampaian informasi. diperlukanlah sebuah solusi yang dapat mengatasi problematika yang dihadapi. Sehingga dibuatlah suatu peancang Aplikasi Digital Arsip Surat Masuk Dan Surat Keluar Berbasis Web.

Untuk pembuatan suatu aplikasi maka diperlukan nya suatu rancangan yang diperlukan untuk mendukung realisasi dari aplikasi tersebut, seperti *tools* yang digunakan, bahasa pemrograman yang digunakan, *text editor* yang digunakan, dan lain-lain.

BAB 2

Menyambung pembahasan dari bab sebelumnya, dalam hal pembuatan Aplikasi ini menggunakan bahasa Pemrograman PHP dimana bahasa pemrograman PHP merupakan bahasa pemrograman dasar yang dikenal para pembuat *website*. Berikut merupakan sejarah dari bahasa pemrograman PHP :

Pertama kali PHP dikenal dengan kependekan dari (*Personal Home Page*) atau Situs Personal yang dibuat oleh Rasmus Lerdorf pada Tahun 1995. Saat pertama kali dibuat PHP masih bernama *Form Interpreted* (FI) yang berupa sekumpulan skrip untuk pengolahan data formulir dari web.



Logo PHP Pertama

Kemudian Rasmus merilis kode sumber tersebut pada bulan November tahun 1997 untuk umum dengan memberikan nama

PHP/FI. Dengan merilis kode tersebut menjadi kode sumber terbuka sehingga banyak pemrograman yang tertarik untuk mengembangkan PHP tersebut.

Di tahun yang sama, sebuah perusahaan bernama Zend menulis kembali interpreter PHP menjadi lebih cepat, rapih dan lebih baik. Sehingga pada bulan Juni tahun 1998, perusahaan tersebut merilis kembali interpreter baru untuk PHP dan meresmikan perilisan tersebut dengan nama PHP 3.0 serta mengubah nama menjadi akronim berulang yang diberi nama PHP (*Hypertext Preprocessing*).

Pada pertengahan tahun 1999, perusahaan Zend merilis kembali interpreter PHP dengan nama PHP 4.0 pada versi 4.0 ini banyak digunakan pada awal abad ke-21. Karena versi ini memiliki kemampuan untuk membangun aplikasi web kompleks tetapi tetap memiliki stabilitas dan kecepatan yang tinggi.

Setelah 5 tahun berlalu pada akhirnya Zend merilis kembali PHP dengan versi 5.0 pada bulan Juni tahun 2004. Pada versi ini, PHP memasukkan model pemrograman berorientasi objek untuk mengembangkan bahasa pemograman ke arah paradigm berorientasi objek.



Gambar logo PHP 5

Di tanggal 18 Desember pada tahun 2014 Zend merilis bahasa pemrograman PHP menjadi versi 5.6.4 yang diperuntukan agar mempermudah pengembang menjalankan kode PHP tanpa menginstall *software server* terlebih dahulu. Tidak lama kemudian pada tanggal 17 Februari 2017 versi 5.6.4 di kembangkan kembali menjadi versi terbaru dan stabil pada bahasa pemrograman PHP saat ini menjadi versi 7.0.16 dan 7.1.2.



Gambar Logo PHP 7

Mengenal PHP

PHP merupakan bahasa pemrograman scripting karena bahasa pemrograman PHP diproses pada web server, sehingga semua kode-kode yang ditulis menggunakan PHP akan dijalankan dan disimpan di web server maka PHP menyatu dengan HTML (Kode dasar website) dan dijalankan pada server side. Artinya, semua sintaks PHP yang diberikan akan sepenuhnya dijalankan pada server, sedangkan yang dikirimkan ke browser hanya hasilnya saja.

PHP digunakan untuk membuat website dinamis dan interaktif. Dinamis yaitu website tersebut dapat berubah-ubah tampilan dan kontennya sesuai kondisi tertentu.

Terdapat dua cara yang dapat dilakukan untuk menulis kode-kode PHP untuk membangun sebuah aplikasi dinamis, seperti menggunakan PHP native dan framework.

a. PHP Native

PHP Native merupakan metode penulisan kode-kode bahasa pemrograman PHP yang benar-benar dimulai dari awal atau dari nol.

Pada PHP native ini programmer akan membuat kerangka kerjanya sendiri untuk mengembangkan website dinamis. Biasanya pemula akan disarankan untuk mempelajari PHP Native.

b. Framework PHP

Untuk memudahkan pengembangan pada aplikasi web dengan menggunakan bahasa pemrograman PHP maka gunakanlah framework PHP, karena terdapat beberapa keuntungan yang bisa didapatkan ketika membuat website menggunakan framework, diantaranya :

- 1) Tidak perlu melakukan pengulangan coding
- 2) Aplikasi web yang dibangun dengan framework akan lebih stabil
- 3) Mengembangkan aplikasi web dapat lebih cepat
- 4) Membantu pemula dalam membuat aplikasi

Adapun beberapa framework PHP yang sering didengar oleh para developer web, yaitu :

1) CodeIgniter



Gambar Logo CodeIgniter

2) Laravel



Gambar Logo Laravel

3) Symfony



Gambar Logo Symfony

4) Zend Framework



Gambar logo Zend Framework

5) CakePHP



Gambar Logo Cake PHP

6) Seagull



Gambar Logo Framework Seagull

PHP dimulai dengan tulisan antara tag :

- a. `<? Dan ?>`
- b. `<?php dan ?>`
- c. `<script language="php">` dan `</script>`
- d. `<% dan %>`
- e. Setiap *statement* atau perintah yang digunakan biasanya diakhiri dengan titik-koma (;)
- f. *Case Sensitive* untuk nama identifier yang dibuat user yaitu berupa variable, konstanta, fungsi, dan lain-lain. Namun tidak *Case Sensitive* untuk identifier built-in dari PHP. Maka :
 - a) `$nama ≠ $Nama ≠ NAMA`

b) `hitungLuas() ≠ HitungLuas()`

c) `echo = ECHO`

d) `while = WHILE`

g. Mudah dipahami

Variabel

Variabel merupakan suatu lokasi penyimpanan yang berisikan nilai atau informasi yang nilainya tidak diketahui maupun telah diketahui.

Nilai dari variabel dapat di isi dengan informasi yang diinginkan dan dapat diubah nilainya pada saat kode program sedang berjalan. Sebuah variabel memiliki nama yang digunakan untuk mengakses nilai dari variabel itu. Jika anda memiliki pengetahuan dasar tentang bahasa pemrograman, tentunya tidak asing dengan istilah variabel.

- a. Variabel digunakan untuk menyimpan sebuah value, informasi atau data
- b. Penamaan variable diawali dengan tanda \$
- c. Panjang yang digunakan tidak ada batas maksimum penggunaan
- d. Setelah menggunakan tanda \$ diawali oleh huruf atau *under-score* (`_`). Maka karakter yang berikutnya bias terdiri dari angka, huruf dan karakter tertentu yang diperbolehkan.
- e. Case yang digunakan bersifat case-sensitive

PHP membedakan variabel yang ditulis dengan huruf besar dan kecil (bersifat case sensitive), contoh \$nilai_kuadrat tidak sama dengan \$NilaiKuadrat dan \$NILAIKUADRAT ketiganya akan dianggap sebagai variabel yang berbeda.

- f. Tidak boleh mengandung spasi
- g. Tidak perlu di deklarasikan.

Tipe Data

Pada PHP tipe data variable secara otomatis ditentukan oleh interpreter PHP. Tetapi, PHP mendukung 8 buah tipe data primitive, seperti :

- a. Integer

Integer merupakan semua angka bulat baik itu positif ataupun negative bukan pecahan, seperti : 1,2,3,5,dst

- b. Float

Float atau floating point atau yang disebut dengan nomor pecahan atau juga bilangan real, seperti : 0.1,0.25,0.2, 0.314E1,dst.

- c. Boolean

Boolean merupakan tipe data standar, Boolean hanya menyatakan kebenaran dengan kata kunci perintah TRUE (benar) atau FALSE (salah).

- d. String

String merupakan rangkaian karakter, sehingga karakter tersebut sama seperti byte. String dapat didefinisikan dengan cara yang

paling mudah yaitu menggunakan tanda satu petik (‘’) terdapat 265 karakter yang dapat dijadikan string.

e. Array

Array merupakan jenis tipe data yang unik, berbeda dengan tipe data lainnya, tipe data array dapat menyimpan satu atau lebih data dalam variabel tunggal. Tipe data array berguna untuk menyimpan banyak data dalam satu variabel.

f. Resource

Resource merupakan jenis tipe data yang menyimpan sebuah referensi external resource seperti bagian koneksi database.

g. Object

Object merupakan keluaran dari class. Object dapat menampilkan atau mengelola isi class.

h. Null

Null mendeklarasikan variabel tanpa value. Sebuah variabel dapat menjadi null jika :

- a) Variabel belum diset ke nilai apapun
- b) Menerima unset()
- c) Dialokasikan konstanta NULL

Konstanta

Dalam bahasa pemrograman, Konstanta atau *constant* merupakan suatu lokasi penyimpanan yang berisikan nilai yang memiliki sifat tetap dan tidak dapat berubah sepanjang program berjalan.

Konstanta memiliki fungsi yang sama seperti variabel namun nilainya statis/konstan dan tidak bisa berubah.

Aturan Penulisan Konstanta PHP

1. Cara Pendefinisikan Konstanta dalam PHP

Jika variabel pada PHP dibuat dengan menambahkan “tanda dollar”, seperti : `$nilai_kuadrat`. Untuk membuat konstanta PHP memiliki 2 cara :

- a. Menggunakan kata kunci (*Keyword*) `const`
- b. Menggunakan fungsi `define`

2. Konstanta PHP bersifat case sensitive

Sama seperti variabel, konstanta dalam PHP bersifat case sensitif, sehingga perbedaan huruf besar dan kecil dianggap berbeda. `GAJI`, `GaJi`, dan `gaji` merupakan 3 konstanta yang berbeda.

3. Nilai konstanta PHP tidak dapat diubah
4. Konstanta hanya dapat berisi tipe data tertentu

Konstanta dalam PHP hanya dapat berisi tipe data sederhana, yaitu :

- a. Boolean
- b. Integer
- c. Float
- d. String

Hal ini berbeda dengan variabel yang dapat berisi tipe data turunan seperti array, objek atau resources.

5. Konstanta Sistem PHP (Predefined Constant)

Sama seperti variabel, PHP juga telah membuat beberapa konstanta yang telah didefinisikan dan tidak bisa di ubah nilainya. Namun karena banyaknya modul yang dapat ditambahkan kedalam PHP, Predefined Constant dalam PHP akan bertambah tergantung modul yang ada. Namun sebagai contoh, berikut adalah Predefined Constant dalam sistem inti PHP:

PHP_VERSION, PHP_MAJOR_VERSION,
PHP_MINOR_VERSION,
PHP_RELEASE_VERSION,PHP_VERSION_ID,PHP_EXTRA_
VERSION,PHP_ZTS, PHP_DEBUG, PHP_MAXPATHLEN,
PHP_OS, PHP_SAPI, PHP_EOL, PHP_INT_MAX,
PHP_INT_SIZE,
DEFAULT_INCLUDE_PATH,PEAR_INSTALL_DIR,PEAR_E
XTENSION_DIR, PHP_EXTENSION_DIR, PHP_PREFIX,
PHP_BINDIR, PHP_BINARY, PHP_MANDIR, PHP_LIBDIR,
PHP_DATADIR, __LINE__ , __FILE__ , __DIR__ ,
__FUNCTION__ , __CLASS__ , __TRAIT__ , __METHOD__,
__NAMESPACE__,

Mengenal Operator

Operator merupakan simbol yang berfungsi untuk melakukan aksi/operasi tertentu pada nilai operand yang umumnya pada hasil operasi tersebut menghasilkan nilai baru. Operator terdiri dari 4 macam yaitu :

a. Operator Assignment

Operator ini untuk mengalokasikan nilai tertentu dengan tanda sama dengan (=). Seperti :

Operator	Contoh	Artinya
=	$X=y$	$X=y$
$+=$	$X+=y$	$X=x+y$
$-=$	$X-=y$	$X=x-y$
$*=$	$X*=y$	$X=x*y$
$/=$	$x/=y$	$X=x/y$
$.=$	$x.=y$	$X=x.y$
$\%=$	$X\%=y$	$X=x\%y$

b. Operator Aritmeka

Operator ini akan melakukan perhitungan matematika. Seperti :

Operator	Nama	Contoh	Hasil
+	Penambahan	$1 + 4$	5
-	Pengurangan	$1 - 4$	-3
/	Pembagian	$1 / 4$	0.25

*	Perkalian	1 * 4	4
%	Sisa pembagian (Modulus)	5 % 2	1
++	Inkremen	X = 5 x ++	X = 6
--	Dekremen	X = 5 X --	X = 4

Operator Perbandingan

Operator perbandingan ini digunakan untuk menghasilkan 2 nilai yang hasil akhirnya merupakan nilai Boolean True atau False.

Seperti :

Operator	Nama	Contoh	Hasil
==	Sama dengan	6 == 6	True
!=	Tidak sama dengan	3 != 3	False
>	Lebih besar	1 > 5	False
>=	Lebih besar sama dengan	3 >= 4	False

<	Lebih kecil	$2 < 4$	True
< =	Lebih kecil sama dengan	$5 < = 4$	False

Operator Logical

Operator logical ini berfungsi untuk mengoperasikan secara logical menggunakan and, or, atau not. Seperti :

Operator	Description	Example
∧	And	$X = 5$ $y = 2$ $(x < 10 \wedge y > 1)$ hasilnya true
	Or	$X = 4$ $y = 6$ $(x == 5 y == 6)$ hasilnya true
!	Not	$X = 5$ $y = 3$ $!(x == y)$ hasilnya true

Kelebihan PHP :

Sebagai salahsatu bahasa pemrograman PHP memiliki banyak kelebihan diantaranya :

a. Komunitas yang besar

Komunitas PHP sangat besar dan tidak dapat dipungkiri bahwa komunitas ini tersebar diseluruh dunia. Tidak hanya di social media seperti Facebook bahkan komunitas PHP ini berada di Telegram, WhatsApp, dll.

b. Mudah dipelajari

PHP merupakan bahasa pemrograman semua orang yang hampir semua orang yang bergelut dengan dunia Web Development pernah menggunakannya atau mencobanya.

c. Resouces yang melimpah

Dengan banyaknya komunitas yang besar maka akan berdampak pada kemudahan mencari resources yang berhubungan dengan PHP baik itu permasalahan yang terjadi di library, software, CMS hingga framework PHP.

d. Simpel

PHP memiliki syntax yang sangat sederhana dan mudah untuk dipelajari. Untuk memulai belajar PHP tida perlu melakukan pengaturan apapun cukup hanya install XAMPP ataupun WAMP maka dapat memulai langsung belajar PHP.

e. Maintenance mudah

Sekali web yang menggunakan PHP berjalan, programmer dapat dengan mudah melakukan update dari software PHP dengan mudah jika memang diperlukan. karena sifat PHP yang merupakan interpreter. Aplikasi web yang dibuat dengan menggunakan PHP dapat dengan mudah diupgrade versi PHP tanpa harus melakukan kompilasi ulang source code. berbeda sekali dengan bahasa pemrograman lain yang membutuhkan kompilasi ulang jika melakukan upgrade versi dari bahasa pemrograman. PHP juga dapat berjalan pada berbagai macam web server seperti apache, nginx, dan IIS.

f. Open Source

PHP merupakan sebuah projek Open source dengan license yang dikeluarkan oleh PHP group yaitu PHP license V3.01. Inti dari license ini adalah setiap pengguna program PHP bebas menggunakan PHP secara gratis tanpa harus memberikan royalty apapun ke PHP group namun tetap wajib mencantumkan licensi atas PHP yang dimiliki PHP Group. Dengan kata lain selama pemakai program PHP tidak mengakui produk PHP adalah buaatannya maka perjual belian program yang menggunakan PHP diperbolehkan tanpa harus membayar licensi apapun.

g. Perkembangan pesat

Karena sifat PHP yang open source, banyak sekali bermunculan proyek proyek open source besar yang menggunakan PHP seperti Prestashop, WordPress, Drupal, dan lain lain. Hal ini menjadi keunggulan yang sangat besar bagi orang yang menguasai pemrograman PHP. Dengan sangat luasnya perkembangan PHP, maka kesempatan untuk bisnis ataupun kerja pada bidang pemrograman PHP sangatlah luas

Kekurangan PHP

Kekurangan utama PHP yaitu PHP merupakan bahasa yang weak type dimana variabel tidak memiliki tipe data sehingga menyulitkan ketika melakukan debugging. Pada weak type ini menyebabkan terjadinya *juggling* dimana variable yang terjadi berisi nilai integer misalnya dapat berubah menjadi nilai string atau tipe data lainnya.

Berikut merupakan kekurangan dalam bahasa pemrograman PHP :

a. Banyak kompetisi

Komunitas yang banyak tentu membawa kompetisi yang ketat. Para web developer yang menguasai PHP tiap hari semakin bertambah. Namun kekurangan ini seharusnya menjadi pemacu bagi para pebisnis yang menginginkan produk IT untuk menggunakan PHP sebagai bahasa pemrograman yang digunakan untuk mengembangkan aplikasi bisnisnya karena terbukanya para

programmer PHP yang sangat kompetitif dan tiap hari semakin banyak

b. Mudah di bajak

Karena sifat PHP yang merupakan interpreter, source code dari aplikasi php dapat dengan mudah di modifikasi dan diubah fungsinya. hal ini membuat PHP tidak cocok untuk digunakan mengembangkan aplikasi jika pemilik aplikasi memiliki source code yang ingin dijaga kerahasiaannya.

Meskipun ada cara untuk mengamankan source code yang menggunakan bahasa pemrograman PHP, namun dibutuhkan sebuah ekstensi yang berbayar yang dikeluarkan oleh Zend sebuah korporasi di bidang pemrograman PHP.

c. Terkesan kurang prestigious

Entry level yang berada pada tingkat pemula, yakni mudah dipelajari oleh programmer pemula membuat bahasa pemrograman PHP terkesan kurang prestisius jika dibandingkan dengan bahasa pemrograman web lain yang terkesan lebih sulit untuk digunakan. Aplikasi web yang dihasilkan dari penggunaan bahasa pemrograman PHP terkesan kurang aman dan memiliki celah. Namun sebenarnya hal ini disebabkan karena faktor pengembang yang mungkin belum mempelajari secara penuh bagaimana standar dan cara membuat aplikasi yang benar dengan menggunakan PHP. Dari sisi performa, pemrograman PHP dan pemrograman lainnya

jika digunakan dengan standar dan penerapan yang benar, akan menghasilkan sebuah aplikasi web yang berkualitas.

d. Tidak adanya *type* data pada PHP

PHP tidak memiliki *type* data. hal ini membuat kadang pada saat menggunakan bahasa pemrograman PHP muncul bug bug yang tidak diinginkan berkaitan dengan tidak adanya standar *type* data pada php. contohnya adalah data “1000” dan “1e3” jika dibandingkan akan memiliki *type* data yang sama karena secara implisit data tersebut dirubah menjadi floating point. namun kekurangan ini berkaitan sekali dengan pengalaman dari developer yang menggunakan bahasa pemrograman PHP. developer yang sudah ahli tentunya sudah paham betul bagaimana mengatasi permasalahan ini.

Untuk Basisdata yang digunakan yaitu menggunakan MySQL, MySQL yaitu :

Data merupakan fakta atau observasi mental yang biasanya mengenai fenomena fisik atau transaksi bisnis. Data pun merupakan suatu ukuran objektif dari atribut (karakteristik) dari entitas.

Dalam pemrograman tempat penyimpanan data disebut juga dengan basis data.

Basis data merupakan suatu focus utama pada aplikasi, pada awal tahun 1960 Charles Bachman di perusahaan General electric mendesain generasi pertama DBMS yang disebut dengan

penyimpanan data terintegrasi (Integrated Data Store) karena hampir semua data computer disimpan pada pita magnetic, namun karena pita hanya dapat di proses secara berurut, maka data harus disimpan dalam bentuk daftar (atau biasa disebut file sekuensial).

Database terdiri dari kelompok table-tabel yang berhubungan. Pada kebanyakan kasus, masing-masing table berisi data dengan satu tema. Hubungan antara table-tabel direpresentasikan dengan cara memberikan masing-masing baris ID yang unik dan menggunakan ID tersebut untuk menghubungkan sebuah baris dalam satu table dengan sebuah baris dalam table baris kedua. Key pada database merupakan kolom yang mengidentifikasi baris yang unik. Kemudian foreign key merupakan sebuah kolom dalam satu table yang merupakan identifier unik pada table kedua. Foreign key digunakan untuk memperlihatkan hubungan diantara table. Table dapat dikombinasikan dengan menggunakan SQL yang merupakan bahasa standar industry untuk pemrosesan table.

Sebuah sistem database terdiri dari empat unsur dasar yaitu :

- a. User
- b. Aplikasi database
- c. DBMS
- d. Database

User menggunakan sistem database untuk melakukan pekerjaannya. Aplikasi hasilkan form, queries dan report, mengeksekusi logika aplikasi dan mengendalikan pemrosesan DBMS menciptakan, memproses, dan mengadministrasikan database.

Sebuah database adalah sekumpulan records terintegrasi yang menggambarkan dirinya sendiri. Database terdiri dari data user, metadata, indeks, prosedur tersimpan (*stored procedur*), trigger, dan metadata aplikasi database.

Trigger merupakan prosedur yang aktif saat tindakan tertentu terjadi. Teknologi database dapat digunakan dalam aplikasi yang luas. Sebagian database digunakan oleh satu individu, sementara sebagian lagi digunakan oleh workgroup, dan yang lainnya digunakan oleh organisasi besar.

Seperti sistem informasi, sistem database dapat dikembangkan dengan menggunakan sebuah proses yang mencakup tiga fase, yaitu :

- a. Requirement
- b. Desain
- c. Implementasi

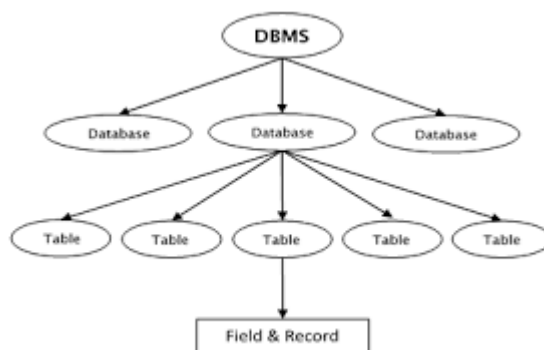
Selama fase requirement, sebuah model data atau representasi logis dari struktur database, dikembangkan.

Model data merupakan bagian penting karena desain database dan aplikasinya diturunkan dari model data. Diagram entity-

relationship merupakan sebuah tool yang digunakan untuk merepresentasikan model data.

Model data ditransformasikan menjadi table dan hubungan selama fase desain. Indeks, batasan, prosedur tersimpan, dan trigger juga dirancang dalam fase ini. Diagram struktur data kadang-kadang digunakan untuk mendokumentasikan table dan hubungannya.

Model Database Pertama, dengan suksesnya media penyimpanan disk pada tahun 1960an, maka kita dapat memiliki akses non-sekuensial atau langsung ke records (Perekaman). Dalam hal ini, database dirancang untuk menghilangkan masalah pemrosesan file sekuensial. Terdapat dua arsitektur atau model yang awalnya sukses. IBM mengembangkan dan mempromosikan DL/I atau *Data language one* yang membuat model data database dalam bentuk pohon hirarki.

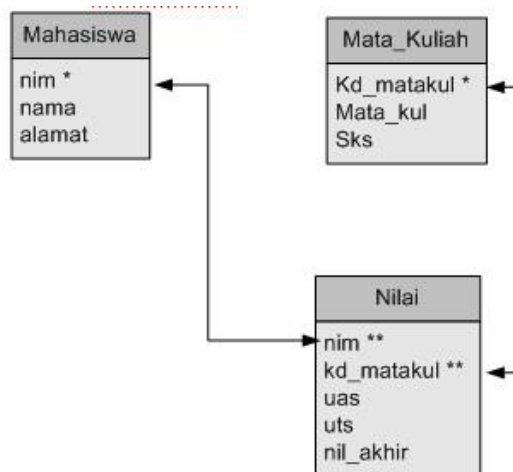


Gambar DBMS Hirarki

Model Relasional

Model database relasional pertama kali diajukan oleh E.F. Codd pada tahun 1970. Codd bekerja untuk IBM dan setelah 10 tahun melakukan penelitian, pengembangan dan lobi, dia dan temannya yang lain berhasil meyakinkan IBM untuk mengembangkan produk DBMS berdasarkan model relasional, model yang paling terkenal dalam produk ini yaitu DB2, yaitu DBMS yang masih aktif digunakan hingga saat ini.

Kemudian perusahaan seperti Oracle, Ingres, Sybase dan Informix mengembangkan juga produk DBMS berdasarkan model relasional.



Gambar Database Relasional

Dasar untuk model data jaringan terbentuk lalu di standarisasi oleh CODASYL (Conference on Data System Languages).

Pada tahun 1973 Bachman menerima penghargaan CM Turing Award yaitu nobel pada ilmu computer.

Menurut sejarah, system pemrosesan basis data terbentuk setelah masa system pemrosesan manual dan system pemrosesan berkas. System pemrosesan manual (Berbasis kertas) merupakan bentuk pemrosesan yang menggunakan dasar berupa setumpuk rekaman yang disimpan pada rak-rak berkas. Jika berkas-berkas tersebut diperlukan, berkas tersebut harus dicari pada rak-rak tersebut. System pemrosesan berkas merupakan system komputer, dimana sekelompok rekaman disimpan pada sejumlah berkas secara terpisah. Perancangan system ini didasarkan pada kebutuhan individual pengguna, bukan kebutuhan sejumlah pengguna. Sehingga setiap aplikasi menuliskan data sendiri, alhasil ada kemungkinan data yang sama terdapat pada berkas-berkas lain yang digunakan oleh program aplikasi lain.

Sejarah DBMS (*Database Management System*), generasi pertama DBMS didesain oleh Charles Bachman di perusahaan General Electric pada awal tahun 1960, disebut sebagai penyimpanan data terintegrasi (*Integrated Data Store*). Dibentuk dasar untuk model data jaringan yang kemudian distandardisasi oleh CODASYL (*Conference on Data System Languages*).

Pada akhir 1960, IBM mengembangkan system manajemen informasi (*Information Management System*) DBMS. IMS dibentuk dari representasi data pada kerangka kerja yang disebut

dengan model data hirarki. Dalam waktu yang sama, dikembangkan system SABRE sebagai hasil kerjasama antara IBM dengan perusahaan penerbangan Amerika. *System* ini memungkinkan user untuk mengakses data yang sama pada jaringan komputer.

Pada tahun 1985, Microsoft dan IBM mengumumkan perjanjian kerjasama jangka panjang untuk mengembangkan system operasi dan produk-produk perangkat lunak lainnya. Pengumuman ini adalah permulaan dimulainya OS/2, sebuah system operasi setelah masa kejayaan MS-DOS. OS/2 ini akan lebih hebat dan lebih canggih daripada MS-DOS, ia akan mampu menangani multitasking application dengan memanfaatkan kemampuan processor Intel yang terbaru yaitu 80286. OS/2 secara resmi diumumkan pada bulan april 1987, dan dijanjikan akan tersedia buat end-user pada akhir tahun tersebut.

Tetapi dalam waktu yang sangat singkat setelah pengumuman perjanjian kerjasama tersebut, IBM mengumumkan peluncuran sebuah versi spesial OS/2 yang disebut OS/2 Extended Edition. Versi lebih powerfull ini akan menyertakan sebuah Database SQL yang disebut OS/2 Database Manager, OS/2 Database Manager akan sangat berguna bagi pengembangan aplikasi yang sederhana dan kompatibel dengan DB/2 , sebuah Database server milik IBM yang beroperasi pada mainframe. OS/2 Database manager juga akan menyertakan SNA (System Network Architecture)

communication service, yang disebut OS/2 Communication Manager. Sebagai bagian dari SSA (System Application Architecture)-nya. IBM menjanjikan semua produk-produk tersebut dapat saling bekerjasama pada masa yang akan datang.

Hal inilah yang membuat Microsoft segera mencari solusi sendiri. Pada tahun 1986, Microsoft sudah menghasilkan 197 JutaUS\$ per tahun, dengan 1153 pegawai. (Sepuluh tahun kemudian, Microsoft telah mendapatkan 6 Milyar US\$ dari bisnis softwarena, dengan hampir sekitar 18.000 pegawai). Produk-produk Microsoft hampir semuanya terfokus pada aplikasi desktop dengan produk utamanya adalah MS-DOS. Komputasi Client/Server pada saat itu belum menjadi fokus utama Microsoft dan industri komputer. Manajemen data pada sebuah PC hanyalah menjadi sebuah fantasi pada masa itu, User biasanya hanya menggunakan LOTUS 1-2-3 untuk menyimpan data. Produk dBASE buatan Ashton Tate segera menjadi sangat populer setelah diluncurkan, Selanjutnya Anca Software merilis Paradox dan Micro Rim dengan produk RBase-nya. Pada tahun 1986 tersebut, Microsoft belum mempunyai produk manajemen Database sendiri. (Tetapi pada tahun 1992, Microsoft mendapatkan sukses yang luar biasa dari produk manajemen database desktop-nya dengan Microsoft Access dan Microsoft FoxPro).

Tetapi IBM Database Manager sangatlah berbeda dari dBASE, Paradox atau RBase. Produk IBM ini walaupun tidak begitu user-

friendly, tetapi ia mempunyai SQL query processor-nya sendiri dan bekerja berdasarkan transactions, hampir sama dengan Database Server yang berjalan diatas Minicomputer dan mainframe, seperti DB/2, Oracle ataupun Informix. Microsoft membutuhkan produk DBMS (Database Management System) sekaliber ini dan membutuhkannya segera!.

Microsoft berpaling kepada Sybase, Inc. Sebuah perusahaan pembuat software DBMS yang sedang 'Naik daun', yang merilis produk Data Server-nya pada bulan mei 1987 untuk Sun Microsystem yang berjalan diatas UNIX. Data Server mendapatkan reputasi berkat inovasi dan kreatifitasnya dengan Stored Procedure dan Trigger serta paradigma baru dalam dunia komputasi yaitu: Client/Server.

Basis data merupakan kumpulan dari elemen data logis yang saling berhubungan. Basis data mengonsolidasi banyak catatan yang sebelumnya disimpan dalam file terpisah. Basis data pun merupakan suatu kumpulan data yang berhubungan secara logis dan deskripsi data tersebut, yang dirancang untuk memenuhi informasi yang dibutuhkan oleh organisasi. Maka basis data merupakan tempat penyimpanan data yang besar dimana dapat digunakan oleh banyak pengguna. Seluruh item basis data tidak dimiliki oleh satu departemen, melainkan menjadi sumber daya perusahaan yang dapat digunakan bersama.

Object Oriented DBMS (OODBMS)

Pemrograman berorientasi objek (*object oriented*) mulai digunakan pada tahun 1980 dan akan dikembangkan menjadi produk DBMS berorientasi objek. Tujuan produk ini untuk menyimpan objek pemrograman berorientasi objek seperti C++ atau java, dalam sebuah database tanpa harus mentransformasikannya ke format relasional.

MySQL

MySQL merupakan sebuah perangkat lunak yang mengatur basis data relasional (RDBMS) yang di distribusikan gratis dengan lisensi GPL (General Public License). Pengguna dapat menggunakan MySQL secara bebas namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan bersifat komersial.

MySQL adalah server yang melayani database. Untuk membuat dan mengolah database, pada dMySQL ini dapat mempelajari pemrograman khusus yang disebut *query* (perintah) SQL. Database diperlukan jika akan menginput data dari user menggunakan form HTML untuk diolah PHP agar dapat disimpan kedalam database MySQL.

MySQL sebenarnya merupakan turunan salah satu konsep utama dalam basisdata yang telah ada sebelumnya. SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoperasian basisdata, terutama untuk pemilihan atau seleksi

dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.

MySQL mendukung operasi basisdata transaksional maupun operasi basisdata non-transaksional. Pada modul operasi non-transaksional, MySQL dapat dikatakan unggul dalam hal unjuk kerja dibandingkan perangkat lunak peladen basisdata kompetitor lainnya. Namun pada modul non-transaksional tidak ada jaminan atas reliabilitas terhadap data yang tersimpan, karenanya modul non-transaksional hanya cocok untuk jenis aplikasi yang tidak membutuhkan reliabilitas data seperti aplikasi blogging berbasis web (wordpress), CMS, dan sejenisnya. Untuk kebutuhan sistem yang ditujukan untuk bisnis sangat disarankan untuk menggunakan modul basisdata transaksional, hanya saja sebagai konsekuensinya unjuk kerja MySQL pada modul transaksional tidak secepat unjuk kerja pada modul non-transaksional.



Gambar Logo MySQL

Sejarah MySQL

MySQL pada awalnya diciptakan pada tahun 1979, oleh Michael "Monty" Widenius, seorang programmer komputer asal Swedia. Monty mengembangkan sebuah sistem database sederhana yang dinamakan UNIREG yang menggunakan koneksi low-level ISAM database engine dengan indexing. Pada saat itu Monty bekerja pada perusahaan bernama TcX di Swedia. TcX pada tahun 1994 mulai mengembangkan aplikasi berbasis web, dan berencana menggunakan UNIREG sebagai sistem database. Namun sayangnya, UNIREG dianggap tidak cocok untuk database yang dinamis seperti web. TcX kemudian mencoba mencari alternatif sistem database lainnya, salah satunya adalah mSQL (miniSQL). Namun mSQL versi 1 ini juga memiliki kekurangan, yaitu tidak mendukung indexing, sehingga performanya tidak terlalu bagus.

Dengan tujuan memperbaiki performa mSQL, Monty mencoba menghubungi David Hughes (programmer yang mengembangkan mSQL) untuk menanyakan apakah ia tertarik mengembangkan sebuah konektor di mSQL yang dapat dihubungkan dengan UNIREG ISAM sehingga mendukung indexing. Namun saat itu Hughes menolak, dengan alasan sedang mengembangkan teknologi indexing yang independen untuk mSQL versi 2.

Dikarenakan penolakan tersebut, David Hughes, TcX (dan juga Monty) akhirnya memutuskan untuk merancang dan mengembangkan sendiri konsep sistem database baru. Sistem ini merupakan gabungan dari UNIREG dan mSQL (yang source codenya dapat bebas digunakan). Sehingga pada May 1995, sebuah RDBMS baru, yang dinamakan MySQL dirilis. David Axmark dari Detron HB, rekanan TcX mengusulkan agar MySQL di ‘jual’ dengan model bisnis baru. Ia mengusulkan agar MySQL dikembangkan dan dirilis dengan gratis. Pendapatan perusahaan selanjutnya di dapat dari menjual jasa “support” untuk perusahaan yang ingin mengimplementasikan MySQL. Konsep bisnis ini sekarang dikenal dengan istilah Open Source.

Pada tahun 1995 itu juga, TcX berubah nama menjadi MySQL AB, dengan Michael Widenius, David Axmark dan Allan Larsson sebagai pendirinya. Titel “AB” di belakang MySQL, adalah singkatan dari “Aktiebolag”, istilah PT (Perseroan Terbatas) bagi perusahaan Swedia.

Keistimewaan MySQL :

MySQL memiliki beberapa keistimewaan, antara lain:

1. Portabilitas. MySQL dapat berjalan stabil pada berbagai sistem operasi seperti Windows, Linux, FreeBSD, Mac Os X Server, Solaris, Amiga, dan masih banyak lagi.

2. Perangkat lunak sumber terbuka. MySQL didistribusikan sebagai perangkat lunak sumber terbuka, di bawah lisensi GPL sehingga dapat digunakan secara gratis.
3. Multi-user. MySQL dapat digunakan oleh beberapa pengguna dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.
4. 'Performance tuning', MySQL memiliki kecepatan yang menakjubkan dalam menangani query sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.
5. Ragam tipe data. MySQL memiliki ragam tipe data yang sangat kaya, seperti signed / unsigned integer, float, double, char, text, date, timestamp, dan lain-lain.
6. Perintah dan Fungsi. MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah Select dan Where dalam perintah (query).
7. Keamanan. MySQL memiliki beberapa lapisan keamanan seperti level subnetmask, nama host, dan izin akses user dengan sistem perizinan yang mendetail serta sandi terenkripsi.
8. Skalabilitas dan Pembatasan. MySQL mampu menangani basis data dalam skala besar, dengan jumlah rekaman (records) lebih dari 50 juta dan 60 ribu tabel serta 5 miliar

baris. Selain itu batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.

9. Konektivitas. MySQL dapat melakukan koneksi dengan klien menggunakan protokol TCP/IP, Unix soket (UNIX), atau Named Pipes (NT).
10. Lokalisasi. MySQL dapat mendeteksi pesan kesalahan pada klien dengan menggunakan lebih dari dua puluh bahasa. Meski pun demikian, bahasa Indonesia belum termasuk di dalamnya.
11. Antar Muka. MySQL memiliki antar muka (interface) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (Application Programming Interface).
12. Klien dan Peralatan. MySQL dilengkapi dengan berbagai peralatan (tool) yang dapat digunakan untuk administrasi basis data, dan pada setiap peralatan yang ada disertakan petunjuk online.
13. Struktur tabel. MySQL memiliki struktur tabel yang lebih fleksibel dalam menangani ALTER TABLE, dibandingkan basis data lainnya semacam PostgreSQL ataupun Oracle.

Kelebihan MySQL dibandingkan dengan RDBMS lainnya :

1. Berlisensi GPL dan Multi Platform.

2. Dapat diintegrasikan dengan beberapa bahasa Pemrograman seperti .Net, Java, Python, Perl yang merupakan bahasa pemrograman yang paling dominan di kalangan programmer.
3. Mendukung ODBC untuk sistem operasi Windows sehingga bisa digunakan aplikasi yang berjalan di windows.
4. Bisa dijalankan pada spesifikasi hardware yang rendah karena lebih hemat resource memory (dibandingkan database lain) sehingga mudah digunakan untuk bahan pembelajaran.
5. MySQL dapat mendeteksi pesan kesalahan pada klien dengan menggunakan lebih dari 20 bahasa meskipun bahasa indonesia belum termasuk didalamnya.
6. MySQL dapat diintegrasikan dengan Hosting.

Kekurangan MySQL dengan RDBMS lainnya :

1. Banyak mengklaim kurang support terhadap pemrograman Visual/Desktop, sehingga sedikit yang menggunakan untuk aplikasi visual.
2. Karena berlisensi GPL sehingga sulit mendapatkan update untuk problem yang urgent, sehingga perusahaan skala menengah keatas lebih memilih RDBMS berlisensi dan disupport seperti Oracle dan MS SQL Server.

3. Sangat diragukan dalam menangani data skala besar, karena ada beberapa opini yang pro dan kontra terhadap kemampuan MySQL terhadap pengolahan data yang besar.

Kemampuan MySQL

Berikut merupakan fitur yang terdapat pada MySQL :

1. Unjuk kerja yang tinggi dalam memproses query sederhana, dalam arti dapat memproses lebih banyak SQL per satuan waktu.
2. Memiliki lebih banyak tipe data seperti: signed/unsigned integer yang memiliki panjang data sebesar 1,2,3,4 dan 8 byte, FLOAT, DOUBLE, CHAR, VARCHAR, TEXT, BLOB, DATE, TIME, DATETIME, TIMESTAMP, YEAR, SET dan tipe ENUM.
3. Mendukung field yang dijadikan Index, dengan maksimal 32 index dalam satu tabel. *
4. MYSQL memiliki beberapa lapisan keamanan, seperti subnetmask, nama host, dan izin akses user dengan sistem perijinan yang mendetail serta sandi/password terenkripsi.
5. Konektivitas , MySQL dapat melakukan koneksi dengan klien menggunakan protokol TCP/IP, Unix socket (UNIX), atau Named Pipes(NT).
6. Multi-user. MySQL dapat digunakan oleh beberapa pengguna dalam waktu yang bersamaan tanpa mengalami masalah atau konflik

7. Command and function, MySQL memiliki fungsi dan operator secara penuh yang mendukung perintah select dan where dalam query.
8. Structure Table, MySQL memiliki struktur tabel yang lebih fleksibel dalam menangani ALTER TABLE dibandingkan DBMS lainnya.
9. Mendukung penuh terhadap kalimat SQL GROUP BY dan ORDER BY. Mendukung terhadap fungsi penuh (COUNT(),COUNT(), DISTINCT() AVG(), STD(), SUM(), MAX() dan MIN())

Bahasa Pemrograman pada MySQL :

Terdapat beberapa API (Application Programming Interface) tersedia yang memungkinkan aplikasi-aplikasi komputer yang ditulis dalam berbagai bahasa pemrograman untuk dapat mengakses basis data MySQL antara lain: bahasa pemrograman C, C++, C#, bahasa pemrograman Eiffel, bahasa pemrograman Smalltalk, bahasa pemrograman Java, bahasa pemrograman Lisp, Perl, PHP, bahasa pemrograman Python, Ruby, REALbasic dan Tcl. Sebuah antarmuka ODBC memanggil MyODBC yang memungkinkan setiap bahasa pemrograman yang mendukung ODBC untuk berkomunikasi dengan basis data MySQL. Kebanyakan kode sumber MySQL dalam ANSI C.

XAMPP



Gambar Logo XAMPP

XAMPP merupakan server yang banyak digunakan untuk keperluan belajar PHP secara mandiri, terutama bagi programmer pemula. Selain gratis, fiturnya tergolong lengkap dan gampang sehingga banyak digunakan oleh programmer tingkat awal.

Xampp juga merupakan salah satu aplikasi web server apache yang terintegrasi dengan mysql dan phpmyadmin. XAMPP adalah singkatan dari X, Apache Server, MySQL, PHPMyadmin, dan Python. Huruf X di depan menandakan XAMPP dapat diinstal pada berbagai operating system. XAMPP dapat diinstal pada Windows, Linux, MacOS, dan Solaris.

Sampai saat ini, XAMPP masih didapatkan secara gratis, bebas di download dan digunakan tanpa harus membayar.


Xampp merupakan buatan dari team apache friends. Xampp dapat di download langsung dari situs resminya.

Xampp 1.8.3 memiliki beberapa fitur seperti :

- a. Apache 2.4.6
- b. MySQL 5.6.14
- c. PHP 5.5.6
- d. phpMyAdmin 4.0.9
- e. FileZilla FTP Server 0.9.41
- f. Tomcat 7.0.42
- g. Strawberry Perl 5.16.3.1 Portable
- h. XAMPP Control Panel 3.2.1

A. Penjelasan Setiap Tampilan Pada Aplikasi

1. Halaman *Login*



Welcome

Username*


Password*

LOGIN


Halaman *Login*

Halaman *Login* seperti ini adalah tampilan paling awal dari Aplikasi Digital Arsip Surat Masuk dan Surat Keluar. Sebelum menjalankan fungsi yang ada di dalam aplikasi ini maka para pegawai Arsip Surat setiap bagian di Dinas Perhubungan Provinsi Jawa Barat harus melakukan *Login* terlebih dahulu.

2. Tampilan *required Username*



Welcome


 Please fill out this field.

LOGIN

Tampilan *required username*

Tampilan *required username* akan muncul ketika menekan tombol *Login* tanpa memasukan *Username*.

3. Tampilan *required Password*



Welcome

admin

Password*

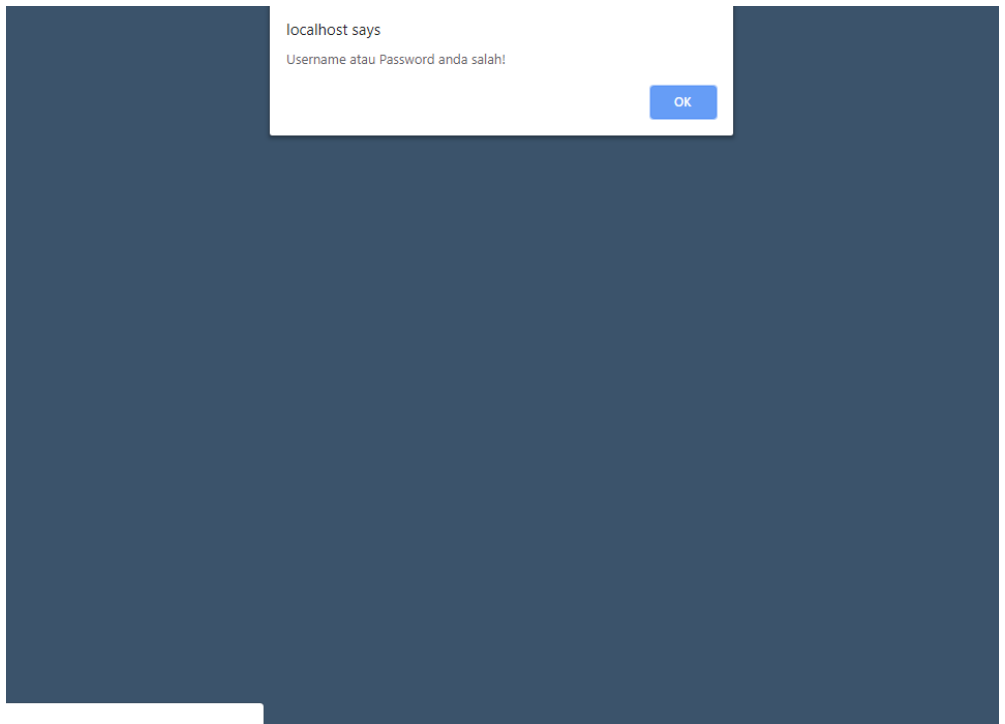
! Please fill out this field.

LOGIN

Tampilan *required password*

Tampilan *required password* akan muncul ketika menekan tombol *Login* tanpa memasukan *password*.

4. Tampilan *Username* atau *Password* salah



Tampilan *Username* atau *Password* salah

Tampilan *Username* atau *Password* salah akan muncul apabila *user* memasukan *Username* atau *Password* yang salah ataupun *Username* dan *Password* yang belum terdaftar.

5. Halaman *Dashboard*



Halaman *Dashboard*

Halaman *Dashboard* Adalah halaman awal setelah *Login*. Di halaman ini menampilkan grafik yang menunjukkan jumlah surat pada tiap bulannya. Grafiknya terdiri dari dua grafik yaitu grafik surat masuk dan grafik surat keluar.

6. Halaman Tabel *User*

The screenshot shows the 'Tabel User' page in the 'ArsipDishub' application. The page features a table with 7 rows of user data. Each row contains a unique ID, a full name, a username, a password, a position, and two action buttons (delete and edit). The table is titled 'Tabel User' and includes a search bar and export options (Excel, PDF). The sidebar on the left shows the main navigation menu with options like Dashboard, User, Surat Masuk, and Surat Keluar. The top header displays the application name 'ArsipDishub' and a user profile for 'Dadi' who is 'Online'.

No	Nama Lengkap	Username	Pass	Jabatan	Aksi
1	Ryan	Ryan	Ryan	Sekretaris Dinas	
2	Dadi	Dadi	Dadi	Subag Kepegawaian dan Umum	
3	Asep	Asep	Asep	Subag Perencanaan dan Laporan	
4	acep	acep	acep	Bidang Transportasi Darat	
5	aji	aji	aji	Kepala Dinas	
6	ari afrianto	ari	ari	Subag Kepegawaian dan Umum	
7	adi	adi	adi	UPTD PPP Pelabuhan Laut	

Halaman Tabel *User*

Halaman tabel *user* menampilkan data dari seluruh *user* yang dapat melakukan *Logini* pada aplikasi ini. Tampilan ini hanya dapat di akses oleh admin(Subag Kepegawaian dan Umum), admin dapat menambahkan ,mengubah, dan menghapus data *user*.

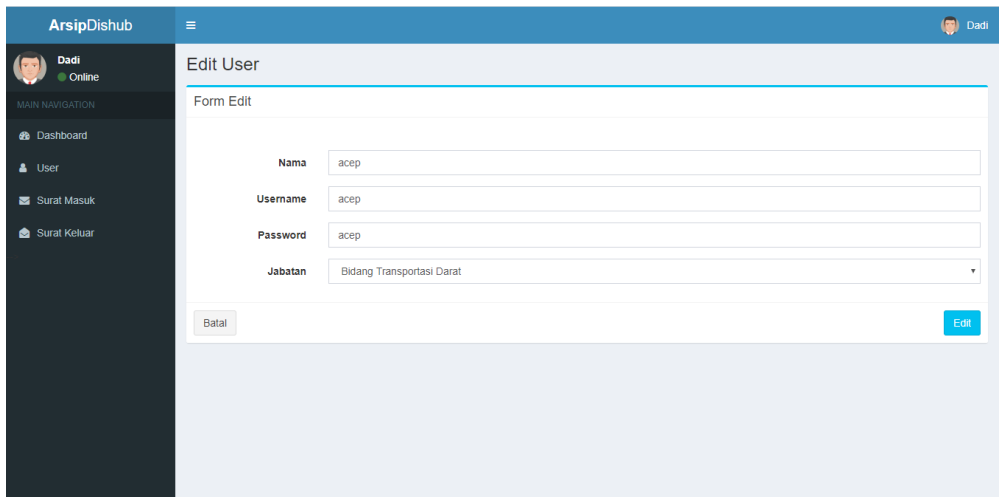
7. Halaman Tambah *User*

The screenshot shows a web application interface for 'ArsipDishub'. On the left is a dark sidebar with a user profile 'Dadi' (Online) and a 'MAIN NAVIGATION' menu containing 'Dashboard', 'User', 'Surat Masuk', and 'Surat Keluar'. The main content area is titled 'Tambah User' and contains a 'Form Tambah' section. This form has four input fields: 'Nama' (Name), 'Username', 'Password', and 'Jabatan' (Job Position) which is a dropdown menu currently showing '-- Pilih Jabatan --'. At the bottom of the form are two buttons: 'Batal' (Cancel) on the left and 'Tambah' (Add) on the right.

Halaman Tambah *User*

Halaman tambah *User* menampilkan *form* tambah user dimana tampilan ini hanya di akses oleh admin(Subag Kepegawaian dan Umum), admin akan memasukan nama, *username*, *password*, dan jabatan pegawai yang akan di masukan.

8. Halaman *Edit User*



The screenshot displays the 'Edit User' interface within the 'ArsipDishub' application. On the left, a dark sidebar contains the user profile 'Dadi' (Online) and a 'MAIN NAVIGATION' menu with options: Dashboard, User, Surat Masuk, and Surat Keluar. The main content area is titled 'Edit User' and contains a 'Form Edit' section. This form includes four input fields: 'Nama' (containing 'acep'), 'Username' (containing 'acep'), 'Password' (containing 'acep'), and 'Jabatan' (a dropdown menu currently showing 'Bidang Transportasi Darat'). At the bottom of the form, there are two buttons: a grey 'Batal' button on the left and a blue 'Edit' button on the right.

Halaman *Edit User*

Halaman *edit user* menampilkan *form edit* user dimana tampilan ini hanya di akses oleh admin(Subag Kepegawaian dan Umum), admin akan mengkoreksi nama, *username*, *password*, dan jabatan pegawai apabila ada kesalahan dalam memasukan data.

9. Halaman Tabel Surat Masuk Admin

ArsipDishub

Dadi

Online

MAIN NAVIGATION

Dashboard

User

Surat Masuk

Surat Keluar

Surat Masuk

+

Show 10 entries

Excel

PDF

Print

CSV

















Search:

No	No Urut	Isi Ringkasan	Surat Dari	Tanggal Surat	Tanggal Masuk	No Naskah Dinas	File Surat	Aksi
1	1317	penyampayan daftar uraian piket dari UPTD P3 LLAJ WIL II	UPTD P3 LLAJ WIL II	2019-09-16	2019-09-17	071/071/159/uptd ilaj		
2	456	fcvghdgtcv	fcgvbn	2019-11-05	2019-11-06	12/rttcgvhb		
3	4635	we	rewf	2019-11-04	2019-11-05	5324/fwet3r		
4	3	sd	dta	2019-11-05	2019-11-21	s/tes		
5	677	gvh	777	2019-12-02	2019-12-16	rht/7hn		
6	123	dwwkkkm	jxnsaj3r	2020-01-14	2020-01-15	qwsed/cdsk		
7	78	t78t89y	aasds	2020-01-15	2020-01-17	/071/159/uptd ilaj		

Halaman Tabel Surat Masuk Admin

Halaman Tabel Surat Masuk Menampilkan daftar surat masuk yang telah di tambahkan tampilan ini di akses oleh admin(Subag Kepegawaian dan Umum). Dimana admin dapat menambahkan, mengubah dan menghapus data surat masuk.

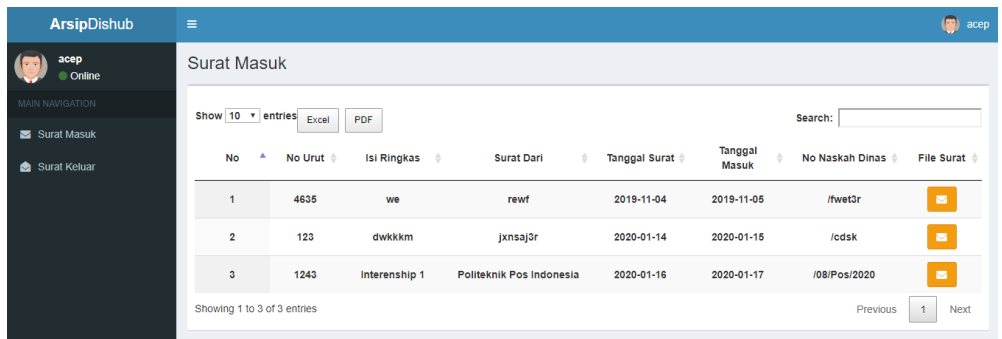
10. Halaman Tabel Surat Masuk Pimpinan




No	No Urut	Isi Ringkas	Surat Dari	Tanggal Surat	Tanggal Masuk	No Naskah Dinas	File Surat
1	1317	penyampayan daftar uraian piket dari UPTD P3 LLAJ WIL II	UPTD P3 LLAJ WIL II	2019-09-16	2019-09-17	071/071/159/uptd ilaj	 
2	456	fcvghdgcfcv	fcgvbn	2019-11-05	2019-11-06	12/rfcgvhb	 
3	4635	we	rewf	2019-11-04	2019-11-05	5324/hwet3r	 
4	3	sd	dfa	2019-11-05	2019-11-21	s/fes	 
5	677	gvh	777	2019-12-02	2019-12-16	rht7hn	 
6	123	dwkkkm	jxsaj3r	2020-01-14	2020-01-15	qwsed/cdsk	 
7	78	t78t89y	aasds	2020-01-15	2020-01-17	/071/159/uptd ilaj	 
8	1243	Interenship 1	Politeknik Pos Indonesia	2020-01-16	2020-01-17	/08/Pos/2020	 

Halaman Tabel Surat Masuk Pimpinan

Halaman tabel surat masuk pimpinan menampilkan seluruh data surat masuk yang di masukan oleh admin, pimpinan tidak bisa memnambah data surat mengubah data surat ataupun menghapus data surat. Yang dapat di lakukan oleh pimpinan hanya melihat isi surat dan mendisposisikan.

11. Halaman Tabel Surat Masuk User



No	No Urut	Isi Ringkas	Surat Dari	Tanggal Surat	Tanggal Masuk	No Naskah Dinas	File Surat
1	4635	we	rewf	2019-11-04	2019-11-05	/fwet3r	
2	123	dwkkkm	jxnsaj3r	2020-01-14	2020-01-15	/cdsk	
3	1243	Interenship 1	Politeknik Pos Indonesia	2020-01-16	2020-01-17	/08/Pos/2020	

Halaman Surat Masuk User

Halaman surat masuk user menampilkan data surat masuk yang telah didisposisikan oleh pimpinan dimana user tidak dapat mengakses seluruh data surat masuk yang ada di Dinas Perhubungan Provinsi Jawa Barat selain yang didisposisikan oleh pimpinan.

12. Halaman Tambah Surat Masuk

The screenshot shows the 'Tambah Surat Masuk' (Add Incoming Letter) form in the ArsipDishub application. The form is titled 'Form Tambah' and contains the following fields:

- No Urut :** Agenda Surat
- Isi Ringkasan :** Prihal
- Surat Dari :** Surat Dari
- Nomor Surat :** Nomor Surat
- Tanggal Surat :** mm/dd/yyyy
- Tanggal Masuk :** mm/dd/yyyy
- File Surat :** Choose File No file chosen

At the bottom of the form, there are two buttons: 'Batal' (Cancel) and 'Tambah' (Add).

Halaman Tambah Surat Masuk

Halaman Tambah Surat Masuk menampilkan *form* tambah dimana admin akan memasukkan data-data surat seperti nomor surat, prihal, surat dari mana dan beserta *scan* surat yang berbentuk *fdp*. Fungsi tambah surat masuk ini hanya dapat diakses oleh admin.

13. Halaman *Edit* Surat Masuk

The screenshot shows the 'Edit Surat Masuk' form in the ArsipDishub application. The interface includes a top navigation bar with the application name 'ArsipDishub' and a user profile 'Dadi' with an 'Online' status. A left sidebar contains a 'MAIN NAVIGATION' menu with options: 'Dashboard', 'User', 'Surat Masuk', and 'Surat Keluar'. The main content area is titled 'Edit Surat Masuk' and contains a 'Form Edit' section with the following fields:

- No Urut :** 1317
- Isi Ringkasan :** penyampayan daftar uraian piket dari UPTD P3 LLAJ WIL II
- Surat Dari :** UPTD P3 LLAJ WIL II
- Nomor Surat :** 071/159/upld ilaj
- Tanggal Surat :** 09/16/2019
- Tanggal Masuk :** 09/17/2019
- File Surat :** Choose File No file chosen

At the bottom of the form, there are two buttons: 'Batal' (Cancel) and 'Edit'.

Halaman *Edit* Surat Masuk

Halaman *edit* surat masuk Menampilkan data surat dalam bentuk *form* dimana data tersebut dapat di ubah dan di sesuaikan dengan semestinya apabila ada kesalahan dalam tahapan memasukan data pada proses tambah data surat masuk.

14. Halaman Tabel Surat Keluar

The screenshot displays the 'Surat Keluar' (Outgoing Letter) management interface. The interface includes a sidebar with navigation options (Dashboard, User, Surat Masuk, Surat Keluar) and a top header with the user's name 'Dadi'. The main content area shows a table with 4 entries, each containing details like ID, subject, recipient, and dates. The table is titled 'Surat Keluar' and includes a search bar and export options (Excel, PDF).

No	No Urut	Isi Ringkasan	Surat Untuk	Tanggal Surat	Tanggal Keluar	Nomer Surat	File Surat	Aksi
1	1318	penyampayan daftar uraian piket untuk UPTD P3 LLAJ WIL II	UPTD P3 LLAJ WIL II	2019-09-16	2019-09-16	071/160/uptd IlaJ		
2	1317	penyampayan daftar uraian piket dari UPTD P3 LLAJ WIL II	qwer	2019-11-05	2019-11-06	071/159/uptd IlaJ		
3	78	t78t89y	yfduyftgu	2020-01-14	2020-01-15	gvf6yd6657fgj		
4	130	UBIU	UIBJKB	2020-01-16	2020-01-17	IJBIUB		

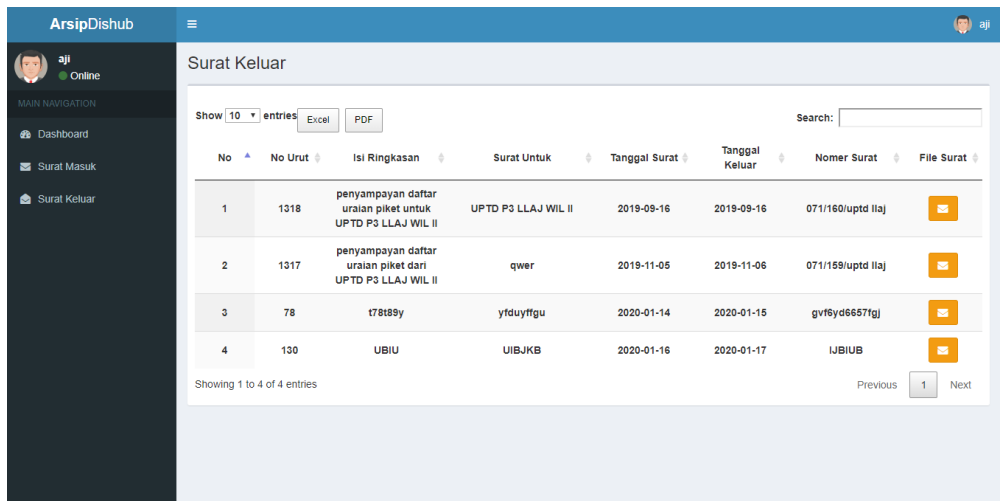
Showing 1 to 4 of 4 entries





Previous 1 Next

Halaman Tabel Surat Keluar

Halaman Tabel Surat Keluar Menampilkan daftar surat keluar yang telah di tambahkan tampilan ini di akses oleh admin(Subag Kepegawaian dan Umum). Dimana admin dapat menambahkan, mengubah dan menghapus data surat keluar.

15. Halaman Tabel Surat Keluar Pimpinan



No	No Urut	Isi Ringkasan	Surat Untuk	Tanggal Surat	Tanggal Keluar	Nomer Surat	File Surat
1	1318	penyampayan daftar uraian piket untuk UPTD P3 LLAJ WIL II	UPTD P3 LLAJ WIL II	2019-09-16	2019-09-16	071/160/uptd ilaj	
2	1317	penyampayan daftar uraian piket dari UPTD P3 LLAJ WIL II	qwer	2019-11-05	2019-11-06	071/159/uptd ilaj	
3	78	t78t89y	ytduyffgu	2020-01-14	2020-01-15	gvf6yd6657fgj	
4	130	UBIU	UIBJKB	2020-01-16	2020-01-17	IJBUIB	

Halaman Tabel Surat Keluar Pimpinan

Halaman tabel surat keluar pimpinan menampilkan seluruh data surat yang di keluarkan oleh Dinas Perhubungan Provinsi Jawa Barat dari berbagai bagian atau jabatan.

16. Halaman Tabel Surat Keluar User

ArsipDishub

acep Online

MAIN NAVIGATION

- Surat Masuk
- Surat Keluar

Surat Keluar

Show 10 entries Excel PDF Search:

No	No Urut	Isi Ringkasan	Surat Untuk	Tanggal Surat	Tanggal Keluar	Nomer Surat	File Surat
1	1318	penyampayan daftar uraian piket untuk UPTD P3 LLAJ WIL II	UPTD P3 LLAJ WIL II	2019-09-16	2019-09-16	071/160/uptd ilaj	
2	1317	penyampayan daftar uraian piket dari UPTD P3 LLAJ WIL II	qwer	2019-11-05	2019-11-06	071/159/uptd ilaj	
3	78	t78t89y	yfdyffgu	2020-01-14	2020-01-15	gvf6yd6657fgj	
4	130	UBIU	UIBJKB	2020-01-16	2020-01-17	IUBIUB	

Showing 1 to 4 of 4 entries

Previous 1 Next

Halaman Tabel Surat Keluar User

Halaman tabel surat keluar user menampilkan data surat keluar yang di keluar kan oleh setiap bagian atau setiap jabatannya.

17. Halaman Tambah Surat Keluar

The screenshot shows a web application interface for 'ArsipDishub'. On the left is a dark sidebar with a user profile 'Dadi' (Online) and a 'MAIN NAVIGATION' menu containing 'Dashboard', 'User', 'Surat Masuk', and 'Surat Keluar'. The main content area is titled 'Tambah Surat Keluar' and contains a 'Form Tambah' section. This form has several input fields: 'No Urut' with the value 'Agenda Surat', 'Isi Ringkasan' with 'Prihal', 'Surat Untuk' with 'Surat Untuk', 'Nomor Surat' with 'No Naskah Dinas', 'Tanggal Surat' with a date format 'mm/dd/yyyy', 'Tanggal Keluar' with a date format 'mm/dd/yyyy', and 'File Surat' with a 'Choose File' button and the text 'No file chosen'. At the bottom of the form are two buttons: 'Batal' (Cancel) and 'Tambah' (Add).

Halaman Tambah Surat Keluar

Halaman Tambah Surat Keluar seperti Gambar 13 menampilkan *form* tambah dimana admin akan memasukan data-data surat seperti nomor surat, prihal, surat dari mana dan beserta *scan* surat yang berbentuk *fdp*. Fungsi tambah surat keluar ini hanya dapat diakses oleh admin.

18. Halaman *Edit* Surat Keluar

The screenshot displays the 'Edit Surat Keluar' interface. On the left is a sidebar with the user profile 'Dadi' (Online) and a navigation menu including 'Dashboard', 'User', 'Surat Masuk', and 'Surat Keluar'. The main content area is titled 'Edit Surat Keluar' and contains a 'Form Edit' section. The form fields are as follows:

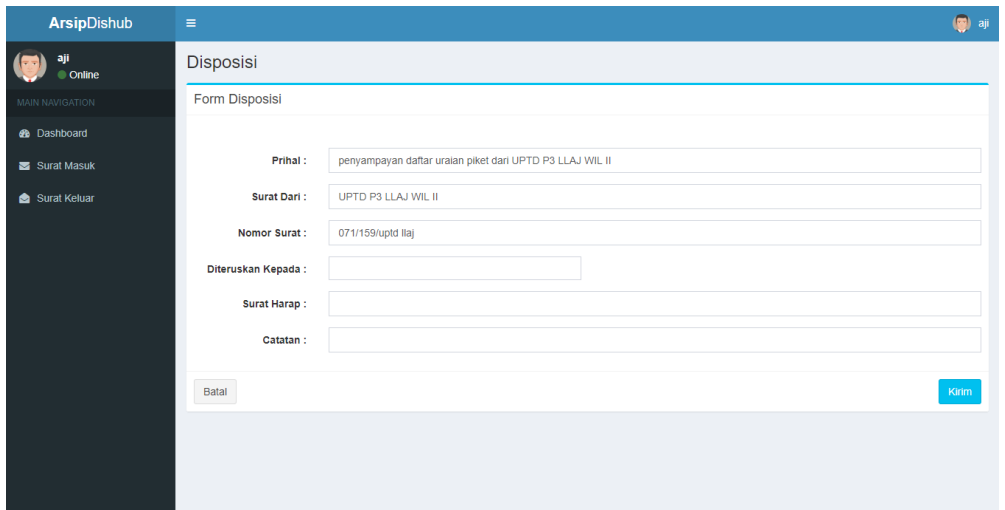
Field Label	Value
No Urut :	1318
Isi Ringkasan :	penyampayan daftar uraian piket untuk UPTD P3 LLAJ WIL II
Surat Untuk :	UPTD P3 LLAJ WIL II
Nomor Surat :	071/160/uptd ilaj
Tanggal Surat :	09/16/2019
Tanggal Keluar :	2019-09-16
File Surat :	<input type="button" value="Choose File"/> No file chosen

At the bottom of the form are two buttons: 'Batal' (Cancel) and 'Edit'.

Halaman *Edit* Surat Keluar

Halaman *edit* surat keluar Menampilkan data surat dalam bentuk *form* dimana data tersebut dapat di ubah dan di sesuaikan dengan semestinya apabila ada kesalahan dalam tahapan memasukan data pada proses tambah data surat keluar.

19. Halaman Disposisi



The screenshot shows the 'Disposisi' page in the 'ArsipDishub' application. The interface includes a top navigation bar with the application name and a user profile. A left sidebar contains a 'MAIN NAVIGATION' menu with options like 'Dashboard', 'Surat Masuk', and 'Surat Keluar'. The main content area is titled 'Disposisi' and contains a 'Form Disposisi'. This form has several input fields: 'Prihal' (containing 'penyampayan daftar uraian piket dari UPTD P3 LLAJ WIL II'), 'Surat Dari' (containing 'UPTD P3 LLAJ WIL II'), 'Nomor Surat' (containing '071/159/upld ilaj'), 'Diteruskan Kepada' (empty), 'Surat Harap' (empty), and 'Catatan' (empty). At the bottom of the form are two buttons: 'Batal' (Cancel) and 'Kirim' (Send).

Halaman Disposisi

Halaman disposisi mempunyai fungsi mendisposisikan surat atau menyampaikan surat kepada bagian yang akan melaksanakan tugas dalam surat tersebut. Halaman ini hanya dapat diakses oleh pimpinan.

ENCRYPTION SYSTEM FOR DIGITAL ARCHIVING OF CULTURAL PROPERTIES

Hiroki Ujiie¹, Shoko Imaizumi¹, Takeshi Ogasawara¹, Kazuma Motegi¹, Seiji Shirono², Mai Sarai²

¹ Graduate School of Science and Engineering, Chiba University

1-33 Yayoicho, Inage-ku, Chiba-shi, Chiba, Japan

² Tokyo National Research Institute for Cultural Properties

13-43 Uenokoen, Taitou-ku, Tokyo, Japan

ABSTRACT

Recently, digital archiving of cultural properties has been developed. On another front, copyright protection for archived images has been a serious concern. This paper proposes an encryption system to protect copyrights of the archived images. Our encryption system has two features, that is, reversible data hiding and encryption. By using the proposed system, the image owner provides the protected images with copyright information to the clients. While the clients can decrypt the received images, they cannot extract the copyright information from the decrypted images.

Index Terms— Reversible data hiding, encryption, copyright protection, digital archive, cultural property

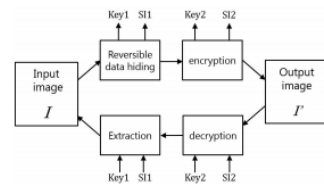


Fig. 1: Proposed encryption system

I. INTRODUCTION

Due to the rapid development of information and communi-

Another technique to protect the copyrights is image encryption. Encryption then Compression (hereafter, ETC)

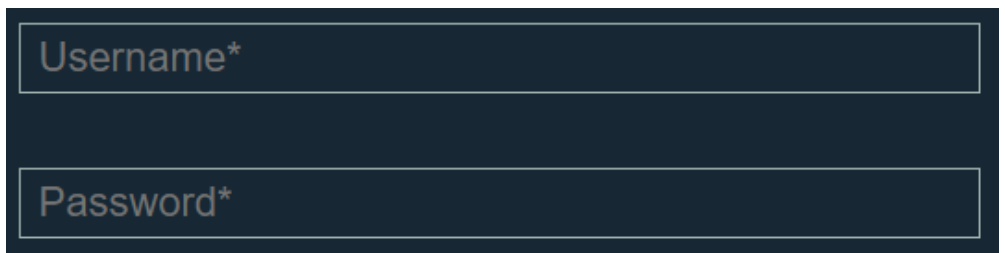
Halaman Surat

Halaman surat mempunyai fungsi untuk melihat isi surat dimana dalam browsernya langsung menampilkan lembar surat dengan format pdf akantetapi ada beberapa browser yang otomatis mendownload.

B. Penjelasan Fungsi Element HTML Disetiap Halaman

1. Halaman Login

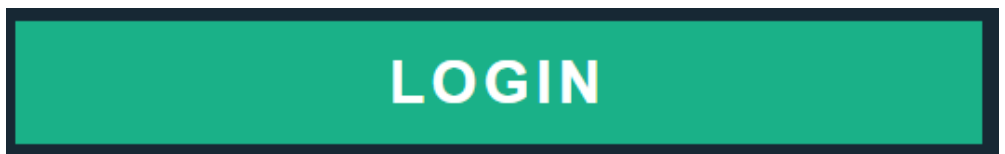
a. *Textbox username dan password*

A screenshot of a login form. It consists of two dark blue rectangular textboxes stacked vertically. The top textbox is labeled 'Username*' in a light gray font. The bottom textbox is labeled 'Password*' in a light gray font. The entire form is set against a dark blue background.

Textbox username dan password

Textbox pada gambar diatas adalah *textbox* yang akan diisikan oleh pegawai arsip surat setiap bagian di Dinas Perhubungan Provinsi Jawa Barat dan sekaligus menampung *username* dan *password* yang akan digunakan sebagai kunci masuk kedalam aplikasi.

b. *Button Sign In*

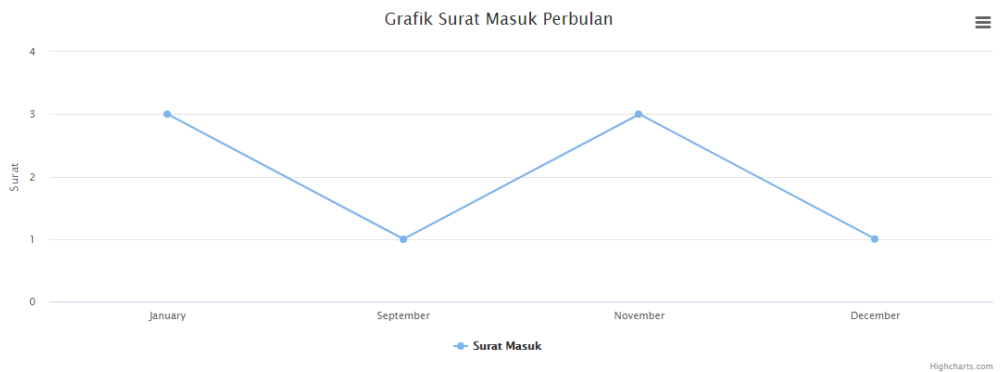


Button Sign In

Button pada gambar diatas adalah sebagai akses permintaan masuk pada program yang nanti akan mengirimkan hasil yang sudah diisikan oleh pegawai arsip surat di *textbox username* dan *password* kedalam aplikasi, apabila *username* dan *password* sesuai maka pegawai arsip surat dapat menjalankan seluruh fungsi yang terdapat didalam aplikasi sesuai hak aksesnya.

2. Halaman Dashboard

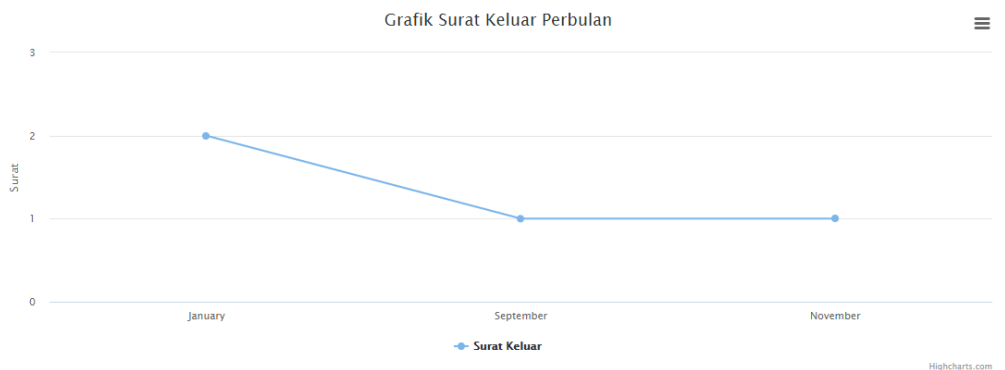
a. Grafik Surat Masuk



Grafik Surat Masuk

Grafik surat masuk menandakan jumlah surat masuk dalam satu tahun dengan berdasarkan bulan.

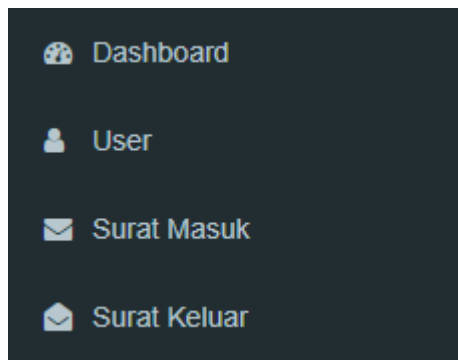
b. Grafik Surat Keluar



Grafik Surat Keluar

Grafik surat keluar menandakan jumlah surat keluar dalam satu tahun dengan berdasarkan bulan.

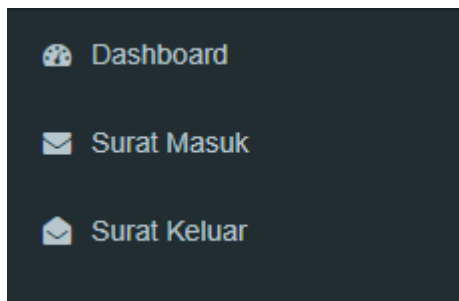
c. Menu Pada Admin



Menu Pada Admin

Menu Pada Admin dimana menu ini hanya dapat diakses oleh admin dengan rincian menu *Dashboard*, menu *User*, menu *Surat Masuk*, menu *Surat Keluar*.

d. Menu Pada User dan Pimpinan



Menu Pada User dan Pimpinan

Menu Pada User dan Pimpinan dimana menu ini dapat diakses oleh seluruh user beserta pimpinan dengan rincian menu *Dashboard*, menu *Surat Masuk*, menu *Surat Keluar*.

3. Halaman Tabel User

a. Tabel User

No	Nama Lengkap	Username	Pass	Jabatan
1	Ryan	Ryan	Ryan	Sekretaris Dinas
2	Dadi	Dadi	Dadi	Subag Kepegawaian dan Umum
3	Asep	Asep	Asep	Subag Perencanaan dan Laporan
4	acep	acep	acep	Bidang Transportasi Darat
5	aji	aji	aji	Kepala Dinas
6	ari afrianto	ari	ari	Subag Kepegawaian dan Umum
7	adi	adi	adi	UPTD PPP Pelabuhan Laut

Tabel User

Tabel User menampilkan seluruh data-data user yang dapat melakukan login kepada aplikasi tersebut.

b. Button Tambah User



Button Tambah User

Button tambah user mempunyai fungsi dimana ketika button tambah user ditekan maka

tampilan akan berubah ke tampilan tambah user yang berisi form tambah user.

c. Show Entris

Show 10 ▼ entries

Show Entris

Show entris mempunyai fungsi menampilkan jumlah isi pada tabel berdasarkan jumlah yang dipilih seperti pada gambar menampilkan jumlah sepuluh maka isi tabel tersebut berjumlah sepuluh baris.

d. *Button Export PDF*

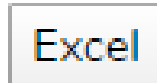


Button Export PDF

Button Export PDF memiliki fungsi untuk melakukan *export* seluruh data yang ditampilkan

pada tabel *user* didalam halaman tabel *user* atau pada Tampilan Tabel *User* menjadi format PDF.

e. Button Export Excel



Button Export Excel

Button Export Excel memiliki fungsi untuk melakukan *export* seluruh data yang ditampilkan pada tabel *user* didalam halaman tabel *user* atau pada Tampilan Tabel *User* menjadi format *excel*.

f. Button Export CSV



Button Export CSV

Button Export CSV memiliki fungsi untuk melakukan *export* seluruh data yang ditampilkan pada tabel *user* didalam halaman tabel *user* atau pada Tampilan Tabel *User* menjadi format CSV.

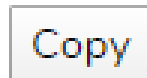
g. Button Print



Button Print

Button Print memiliki fungsi untuk mencetak secara langsung seluruh data yang ditampilkan pada tabel *user* didalam halaman tabel *user* atau pada Tampilan Tabel *User*.

h. Button Copy



Button Copy

Button Copy memiliki fungsi untuk menyalin seluruh data yang ditampilkan pada tabel *user* didalam halaman tabel *user* atau pada Tampilan Tabel *User*.

i. Textbox Search



Textbox Search

Textbox Search memiliki fungsi untuk melakukan pencarian data secara menyeluruh kepada seluruh data didalam tabel *user* sesuai dengan yang diisikan pada *textbox* tersebut.

j. Button Hapus



Button Hapus

Button hapus mempunyai fungsi untuk menghapus data user berdasarkan data pada baris dimana button ditekan.

k. Button Edit



Button Edit

Button edit mempunyai fungsi untuk merubah data user berdasarkan data pada baris dimana button ditekan maka tampilan akan berubah pada tampilan edit user yang berisi form edit user.

l. Button Previous, Number, dan Next



Button Previous, Number, dan Next

Seluruh *button Previous, Number, dan Next* umumnya memiliki fungsi yang sama yaitu sebagai navigasi menampilkan data pegawai pada tabel *user*. Untuk lebih spesifik *button previous* digunakan untuk memundurkan halaman dari tabel di *user*, kemudian *button number* digunakan untuk memilih halaman dari tabel *user* sesuai yang diinginkan pegawai keuangan, dan terakhir *button*

next digunakan untuk memajukan halaman dari tabel *user*.

4. Halaman Tambah User

a. Form Tambah User

Form Tambah

Nama :	<input type="text" value="Nama"/>
Username :	<input type="text" value="Username"/>
Password :	<input type="password" value="Password"/>
Jabatan :	<input type="text" value="-- Pilih Jabatan --"/>

Form Tambah User

Form Tambah User menampilkan text box yang harus di isi oleh admin(subag umum dan kepegawaian), dimana isinya adalah data-data user yang akan di masukan atau di daftar kan pada aplikasi ini.

b. Combo Box Jabatan

-- Pilih Jabatan --

-- Pilih Jabatan --

Kepala Dinas
Sekertaris Dinas
Subag Perencanaan dan Laporan
Subag Keuangan dan Asset
Subag Kepegawaian dan Umum
Bidang Transportasi Darat
Bidang Transportasi Laut dan ASDP
Bidang Transportasi Udara
Bidang Transportasi Perkeretaapian
UPTD PPP LLAJ Wil.I
UPTD PPP LLAJ Wil.II
UPTD PPP LLAJ Wil.III
UPTD PPP LLAJ Wil.IV
UPTD PPP LLASDP
UPTD Pelayanan Jasa Kebandarudaraan Nusawiru
UPTD PPP Pelabuhan Laut

Combo Box Jabatan

Combo box jabatan berisi seluruh jabatan-jabatan yang ada di dinas perhubungan provinsi jawa barat. Dimana jabatan ini akan di pilih berdasarkan jabatan user atau pegawai oleh admun(subag umum dan kepegawaian).

c. Button Batal



Button Batal

Button Batal Berfungsi untuk membatalkan proses tambah user, maka tampilan dari halaman tambah user akan kembali ke halaman tabel user.

d. Button Tambah



Button Tambah

Button Tambah mempunyai fungsi dimana admin setelah memasukan data pegawai dan memilih jabatan pada combo box di halaman tambah user maka untuk menyimpan data tersebut admin harus menekan tombol tambah dan setelah berhasil tampilan akan kembalike halaman tabel user.

5. Halaman Edit User

a. Form Edit User

Form Edit

Nama	<input type="text" value="Ryan"/>
Username	<input type="text" value="Ryan"/>
Password	<input type="text" value="Ryan"/>
Jabatan	<input type="text" value="Sekertaris Dinas"/>

Form Edit User

Form Edit User menampilkan label dan text box dimana text box tersebut mempunyai isi data pegawai atau user yang akan diedit.

b. Combo Box Jabatan

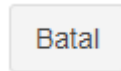
Sekertaris Dinas	▼
-- Pilih Jabatan--	
Kepala Dinas	
Sekertaris Dinas	
Subag Perencanaan dan Laporan	
Subag Keuangan dan Asset	
Subag Kepegawaian dan Umum	
Bidang Transportasi Darat	
Bidang Transportasi Laut dan ASDP	
Bidang Transportasi Udara	
Bidang Transportasi Perkeretaapian	
UPTD PPP LLAJ WII.I	
UPTD PPP LLAJ WII.II	
UPTD PPP LLAJ WII.III	
UPTD PPP LLAJ WII.IV	
UPTD PPP LLASDP	
UPTD Pelayanan Jasa Kebandarudaraan Nusawiru	
UPTD PPP Pelabuhan Laut	

Combo Box Jabatan

Combo Box Jabatan pada halaman edit mempunyai perbedaan dengan combo box jabatan pada form tambah user. Dimana pada halaman edit

user combo box tersebut aktif pada jabatan masing-masing user atau pegawai.

c. Button Batal



Button Batal

Button Batal Berfungsi untuk membatalkan proses tambah user, maka tampilan dari halaman tambah user akan kembali ke halaman tabel user.

d. Button Edit



Button Edit

Button Edit mempunyai fungsi dimana admin setelah mengubah data pegawai dan memilih jabatan terbaru pada combo box atau tidak mengubah jabatan di halaman edit user maka untuk menyimpan data tersebut admin harus menekan tombol edit dan setelah berhasil tampilan akan kembalike halaman tabel user.

6. Halaman Tabel Surat Masuk

a. Tabel Surat Masuk

No	No Urut	Isi Ringkasan	Surat Dari	Tanggal Surat	Tanggal Masuk	No Naskah Dinas
1	1317	penyampayan daftar uraian piket dari UPTD P3 LLAJ WIL II	UPTD P3 LLAJ WIL II	2019-09-16	2019-09-17	071/071/159/uptd Ilaj
2	456	fcvghdgfcv	fcgvbn	2019-11-05	2019-11-06	12/rtfcgvhb
3	4635	we	rewf	2019-11-04	2019-11-05	5324/fwet3r
4	3	sd	dfa	2019-11-05	2019-11-21	s/fes
5	677	gvh	777	2019-12-02	2019-12-16	rht/7hn
6	123	dwkkkm	jxnsaj3r	2020-01-14	2020-01-15	qwsed/cdsk

Tabel Surat Masuk

Tabel Surat Masuk menampilkan Data Surat Masuk yang telah di masukan pada sistem. Pada tabel surat masuk mempunyai perbedaan dimana tampilan surat masuk pada admin dan pimpinan

menampilkan seluruh surat yang di inputkan akan tetapi pada user data surat masuk akan muncul berdasarkan disposisi dari pimpinan.

b. Button Tambah Surat Masuk



Button Tambah Surat Masuk

Button Tambah Surat Masuk berfungsi untuk menambahkan data baru diaman ketika tombol ini di klik maka tampilan akan berubah ke halaman tambah surat masuk. Tombol tambah surat masuk ini hanya ada pada tampilan tabel surat masuk admin(subag umum dan kepegawaian).

c. Show Entris



Show Entris

Show entris mempunyai fungsi menampilkan jumlah isi pada tabel berdasarkan jumlah yang dipilih seperti pada gambar menampilkan jumlah

sepuluh maka isi tabel tersebut berjumlah sepuluh baris.

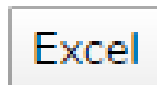
d. Button Export PDF



Button Export PDF

Button Export PDF memiliki fungsi untuk melakukan *export* seluruh data yang ditampilkan pada tabel surat masuk didalam halaman tabel surat masuk menjadi format PDF.

e. Button Export Excel



Button Export Excel

Button Export Excel memiliki fungsi untuk melakukan *export* seluruh data yang ditampilkan pada tabel surat masuk didalam halaman tabel surat masuk menjadi format *excel*.

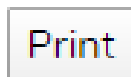
f. *Button Export CSV*



Button Export CSV

Button Export CSV memiliki fungsi untuk melakukan *export* seluruh data yang ditampilkan pada tabel surat masuk didalam halaman tabel surat masuk menjadi format CSV.

g. *Button Print*



Button Print

Button Print memiliki fungsi untuk mencetak secara langsung pada aplikasi seluruh data surat masuk di Dinas Perhubungan Provinsi Jawa Barat pada tabel surat masuk.

h. Button Copy



Button Copy

Button Copy memiliki fungsi untuk menyalin seluruh data surat masuk di Dinas Perhubungan Provinsi Jawa Barat pada tabel surat masuk.

i. Textbox Search



Textbox Search

Textbox Search memiliki fungsi untuk melakukan pencarian data secara menyeluruh kepada seluruh data didalam tabel surat masuk sesuai dengan yang diisikan pada *textbox* tersebut.

j. Button Surat



Button Surat

Button Surat Memiliki fungsi dimana ketika tombol surat di klik maka tampilan akan berubah menjadi tampilan surat dalam bentuk pdf. Akan tetapi dalam beberapa browser surat tersebut akan secara otomatis di download dengan format pdf.

k. Button Disposisi



Button Disposisi

Button disposisi hanya ada di tampilan halaman tabel surat masuk pada pimpinan saja karena setiap surat baru masuk pimpinan akan mendisposisikan surat tersebut kepada bagaian.

l. Button Hapus



Button Hapus

Button hapus mempunyai fungsi untuk menghapus data surat masuk berdasarkan data pada baris dimana button ditekan.

m. Button Edit



Button Edit

Button edit mempunyai fungsi untuk merubah data surat masuk berdasarkan data pada baris dimana button ditekan maka tampilan akan berubah pada tampilan edit surat masuk yang berisi form edit surat masuk.

n. Button Previous, Number, dan Next



Button Previous, Number, dan Next

Seluruh *button previous*, *number*, dan *next* umumnya memiliki fungsi yang sama yaitu sebagai

navigasi menampilkan data pegawai pada tabel surat masuk. Untuk lebih spesifik *button previous* digunakan untuk memundurkan halaman dari tabel surat masuk, kemudian *button number* digunakan untuk memilih halaman dari tabel surat masuk sesuai yang diinginkan pegawai keuangan, dan terakhir *button next* digunakan untuk memajukan halaman dari tabel surat masuk.

7. Halaman Tambah Surat Masuk

a. Form Tambah Surat Masuk

No Urut :	<input type="text" value="Agenda Surat"/>
Isi Ringkasan :	<input type="text" value="Prihal"/>
Surat Dari :	<input type="text" value="Surat Dari"/>
Nomor Surat :	<input type="text" value="Nomor Surat"/>
Tanggal Surat :	<input type="text" value="mm/dd/yyyy"/>
Tanggal Masuk :	<input type="text" value="mm/dd/yyyy"/>

Form Tambah Surat Masuk

Form Tambah Surat Masuk menampilkan Label dan Text box kosong dimana text box akan di isi data-data surat masuk oleh admin(subag umum dan kepegawaian) berdasar label.

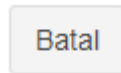
b. Choose File

File Surat :	<input type="button" value="Choose File"/> No file chosen
---------------------	---

Choose File

Choose file memiliki fungsi untuk mengupload file surat masuk dalam bentuk pdf

c. Button Batal



Button Batal

Button Batal Berfungsi untuk membatalkan proses tambah surat masuk, maka tampilan dari halaman tambah surat masuk akan kembali ke halaman tabel surat masuk.

d. Button Tambah



Button Tambah

Button Tambah mempunyai fungsi dimana admin setelah memasukan data surat masuk dan mengupload file surat masuk pada choose file di halaman tambah surat masuk maka untuk menyimpan data tersebut admin harus menekan tombol tambah dan setelah berhasil tampilan akan kembalike halaman tabel surat masuk.

8. Halaman Edit Surat Masuk

a. Form Edit Surat Masuk

No Urut :	1317
Isi Ringkasan :	penyampayan daftar uraian piket dari UPTD P3 LLAJ WIL II
Surat Dari :	UPTD P3 LLAJ WIL II
Nomor Surat :	071/159/uptd llaj
Tanggal Surat :	09/16/2019
Tanggal Masuk :	09/17/2019

Form Edit Surat Masuk

Form edit surat masuk menampilkan Label dan Text box yang berisi data surat masuk dimana text box akan di ubah data-data surat masuk oleh admin(subag umum dan kepegawaian) berdasar label.

b. Choose File

File Surat :

Choose File

No file chosen

Choose File

Choose file memiliki fungsi untuk mengupload file surat masuk dalam bentuk pdf

c. Button Batal

Batal

Button Batal

Button Batal Berfungsi untuk membatalkan proses edit surat masuk, maka tampilan dari halaman edit surat masuk akan kembali ke halaman tabel surat masuk.

d. Button Edit








Edit

Button Edit

Button edit mempunyai fungsi dimana admin setelah mengubah data surat masuk dan mengupload file surat masuk pada choose file di halaman edit surat masuk maka untuk menyimpan data tersebut admin harus menekan tombol edit dan setelah berhasil tampilan akan kembalike halaman tabel surat masuk.

9. Halaman Tabel Surat Keluar

a. Tabel Surat Keluar

No 	No Urut 	Isi Ringkasan 	Surat Untuk 	Tanggal Surat 	Tanggal Keluar 	Nomer Surat 
1	1318	penyampayan daftar uraian piket untuk UPTD P3 LLAJ WIL II	UPTD P3 LLAJ WIL II	2019-09-16	2019-09-16	071/160/uptd Ilaj
2	1317	penyampayan daftar uraian piket dari UPTD P3 LLAJ WIL II	qwer	2019-11-05	2019-11-06	071/159/uptd Ilaj
3	78	t78t89y	yfduyffgu	2020-01-14	2020-01-15	gvf6yd6657fgj
4	130	UBIU	UIBJKB	2020-01-16	2020-01-17	IJBIUB

Showing 1 to 4 of 4 entries

Tabel Surat Keluar

Tabel surat keluar menampilkan data surat keluar yang telah di masukan pada sistem. Pada tabel surat keluar mempunyai perbedaan dimana tampilan surat keluar pada admin dan pimpinan menampilkan seluruh surat yang di inputkan akantetatpi pada user data surat keluar akan muncul berdasarkan surat yang dikeluarkan dari setiap bidang user tersebut.

b. Button Tambah Surat Keluar



Button Tambah Surat Keluar

Button tambah surat keluar berfungsi untuk menambahkan data baru diaman ketika tombol ini di klik maka tampilan akan berubah ke halaman tambah surat keluar. Tombol tambah surat keluar ini hanya ada pada tampilan tabel surat keluar admin(subag umum dan kepegawaian).

c. Show Entris

Show 10 ▼ entries

Show Entris

Show entris mempunyai fungsi menampilkan jumlah isi pada tabel berdasarkan jumlah yang dipilih seperti pada gambar menampilkan jumlah sepuluh maka isi tabel tersebut berjumlah sepuluh baris.

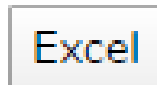
d. Button Export PDF



Button Export PDF

Button Export PDF memiliki fungsi untuk melakukan *export* seluruh data yang ditampilkan pada tabel surat keluar didalam halaman tabel surat keluar menjadi format PDF.

e. Button Export Excel



Button Export Excel

Button Export Excel memiliki fungsi untuk melakukan *export* seluruh data yang ditampilkan pada tabel surat keluar didalam halaman tabel surat keluar menjadi format *excel*.

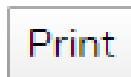
f. Button Export CSV



Button Export CSV

Button Export CSV memiliki fungsi untuk melakukan *export* seluruh data yang ditampilkan pada tabel surat keluar didalam halaman tabel surat keluar menjadi format CSV.

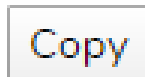
g. *Button Print*



Button Print

Button Print memiliki fungsi untuk mencetak secara langsung pada aplikasi seluruh data surat keluar di Dinas Perhubungan Provinsi Jawa Barat.

h. *Button Copy*



Button Copy

Button Copy memiliki fungsi untuk menyalin seluruh data surat keluar di Dinas Perhubungan Provinsi Jawa Barat.

i. *Textbox Search*



Textbox Search

Textbox Search memiliki fungsi untuk melakukan pencarian data secara menyeluruh kepada seluruh data didalam tabel surat keluar sesuai dengan yang diisikan pada *textbox* tersebut.

j. Button Surat



Button Surat

Button Surat Memiliki fungsi dimana ketika tombol surat di klik maka tampilan akan berubah menjadi tampilan surat dalam bentuk pdf. Akan tetapi dalam beberapa browser surat tersebut akan secara otomatis di download dengan format pdf.

k. Button Hapus



Button Hapus

Button hapus mempunyai fungsi untuk menghapus data surat keluar berdasarkan data pada baris dimana button ditekan.

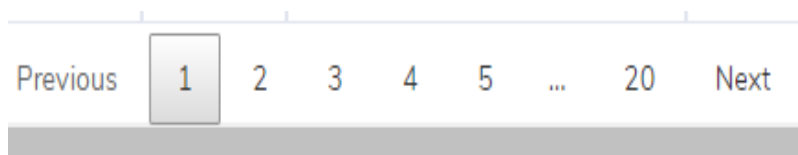
1. Button Edit



Button Edit

Button edit mempunyai fungsi untuk merubah data surat keluar berdasarkan data pada baris dimana button ditekan maka tampilan akan berubah pada tampilan edit surat keluar yang berisi form edit surat keluar.

m. Button Previous, Number, dan Next



Button Previous, Number, dan Next

Seluruh *button Previous*, *Number*, dan *Next* umumnya memiliki fungsi yang sama yaitu sebagai navigasi menampilkan data pegawai pada tabel surat keluar. Untuk lebih spesifik *button previous* digunakan untuk memundurkan halaman dari tabel surat keluar, kemudian *button number* digunakan untuk memilih halaman dari tabel surat keluar sesuai yang diinginkan user, dan terakhir *button next* digunakan untuk memajukan halaman dari tabel surat keluar.

10. Halaman Tambah Surat Keluar

a. Form Tambah Surat Keluar

No Urut :	<input type="text" value="Agenda Surat"/>
Isi Ringkasan :	<input type="text" value="Prihal"/>
Surat Dari :	<input type="text" value="Surat Dari"/>
Nomor Surat :	<input type="text" value="Nomor Surat"/>
Tanggal Surat :	<input type="text" value="mm/dd/yyyy"/>
Tanggal Masuk :	<input type="text" value="mm/dd/yyyy"/>

Form Tambah Surat Keluar

Form tambah surat keluar menampilkan label dan text box kosong dimana text box akan di isi

data-data surat keluar oleh admin(subag umum dan kepegawaian) berdasrkan label.

b. Choose File

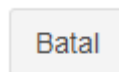
File Surat :

A file upload interface consisting of a rectangular container. Inside, on the left, is a button with the text 'Choose File'. To the right of the button is the text 'No file chosen'.

Choose File

Choose file memiliki fungsi untuk mengupload file surat keluar dalam bentuk pdf.

c. Button Batal

A rectangular button with a light gray background and a thin border. The button contains the text 'Batal' in a dark gray font.

Button Batal

Button Batal Berfungsi untuk membatalkan proses tambah surat masuk, maka tampilan dari halaman tambah surat masuk akan kembali ke halaman tabel surat masuk.

d. Button Tambah

A rectangular button with a solid blue background. The button contains the text 'Tambah' in white font.

Button Tambah

Button Tambah mempunyai fungsi dimana admin setelah memasukan data surat keluar dan mengupload file surat keluar pada choose file di halaman tambah surat keluar maka untuk menyimpan data tersebut admin harus menekan tombol tambah dan setelah berhasil tampilan akan kembalike halaman tabel surat keluar.

11. Halamana Edit Surat Keluar

a. Form Edit Surat Keluar

No Urut :	1317
Isi Ringkasan :	penyampayan daftar uraian piket dari UPTD P3 LLAJ WIL II
Surat Dari :	UPTD P3 LLAJ WIL II
Nomor Surat :	071/159/uptd llaj
Tanggal Surat :	09/16/2019
Tanggal Masuk :	09/17/2019

Form Edit Surat Keluar

Form edit surat keluar menampilkan Label dan Text box yang berisi data surat keluar dimana text box akan di ubah data-data surat keluar oleh admin(subag umum dan kepegawaian) berdasar label.

b. Choose File

File Surat : No file chosen

Choose File

Choose file memiliki fungsi untuk mengupload file surat keluar dalam bentuk pdf

c. Button Batal

Button Batal

Button Batal Berfungsi untuk membatalkan proses edit surat masuk, maka tampilan dari halaman edit surat masuk akan kembali ke halaman tabel surat masuk.

d. Button Edit



Button Edit

Button edit mempunyai fungsi dimana admin setelah mengubah data surat keluar dan mengupload file surat keluar pada choose file di halaman edit surat keluar maka untuk menyimpan data tersebut admin harus menekan tombol edit dan setelah berhasil tampilan akan kembalike halaman tabel surat keluar.

12. Halaman Disposisi

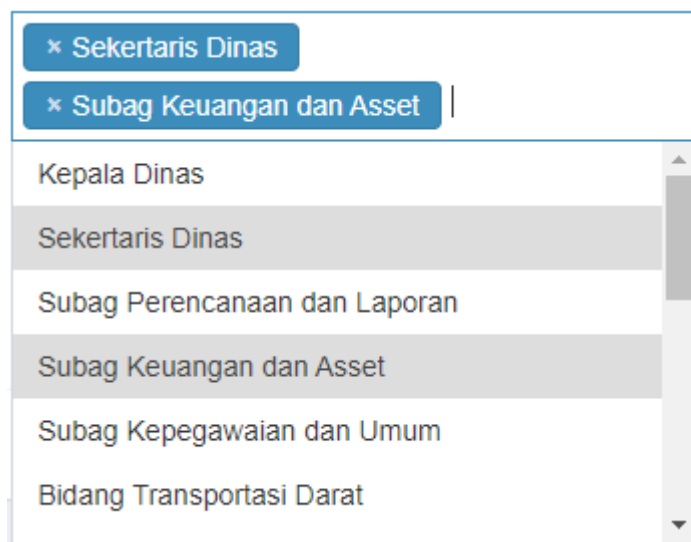
a. Form Disposisi

Prihal :	<input type="text" value="penyampayan daftar uraian piket dari UPTD P3 LLAJ WIL II"/>
Surat Dari :	<input type="text" value="UPTD P3 LLAJ WIL II"/>
Nomor Surat :	<input type="text" value="071/159/uptd llaj"/>
Diteruskan Kepada :	<input type="text"/>
Surat Harap :	<input type="text"/>
Catatan :	<input type="text"/>

Form Disposisi

Form disposisi menampilkan label dan text box dimana text box data surat telah terisi, dan jang di isi oleh pimpinan text box yang masih kosong dengan menyesuaikan isinya dengan label yang sudah ada.

b. Multiselect Combo box

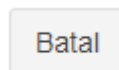


Multiselect Combo box

Multiselection combo box mempunyai fungsi dimana combo box di sini berisikan jabatan-jabatan dan pimpinan tinggal memilih siapa saja yang akan di beri disposisi berdasarkan surat masuk tersebut. Dan fungsi dari multiselectionnya supaya bisa memilih lebih dari satu jabatan yang akan di berikan

disposisi oleh pimpinan berdasar surat masuk tersebut.

c. Button Batal



Button Batal

Button batal mempunyai fungsi dimana ketika pimpinan memijit tombol disposisi surat masuk yang sudah didisposisikan ketika tombol batal dipijit maka tampilan akan kembali ke halaman tabel surat masuk.

d. Button Disposisi



Button Disposisi

Button Disposisi Mempunya fungsi dimana ketika pimpinan telah mengisi form disposisi maka pimpinan memijit tombol disposisi untuk meneruskan surat kepada jabatan-jabatan atau bagian-bagian yang berkaitan dengan surat masuk

tersebut. Maka setiap user akan memndapatkan akses surat masuk setelah diberi disposisi oleh pimpinan.

C. Penjelasan kodingan *Controller* Perbaris

1. *Controller Login*

a. *Controller Class login*

```
1  <?php
2  defined('BASEPATH') OR exit('No direct script access allowed');
3
4  class Login extends CI_Controller {
5
6      public function __construct()
7      {
8          parent::__construct();
9          $this->load->helper("url_helper");
10         $this->load->model('Mlogin');
11     }
```

Controller Class Login

- Baris 1

Pada baris 1 adalah membuka bahasa php pada halaman tersebut dimulai dari baris 1 yang kemudian baris-baris selanjutnya dapat juga menggunakan seluruh elemen dari bahasa php.

- Baris 2

Digunakan untuk memastikan bahwa permintaan telah melalui indeks. php di direktori root Anda. Ini untuk alasan seperti

memastikan bahwa semua kelas dasar Codeigniter sedang dimuat dan memastikan vars tertentu telah ditetapkan dll.

- Baris 3

Pada Baris 3 berisi 1 enter untuk membuat baris koding menjadi lebih rapi.

- Baris 4

Pada baris 4 berisi pendeklarasian *class* yang bernama *Login* dan hasil *extends* dari *CI_Controller*, yang berarti *class Login* ini mendapatkan sifat-sifat/atribut dari class induk yaitu *CI_Controller*.

- Baris 5

Pada Baris 5 berisi 1 enter untuk membuat baris koding menjadi lebih rapi.

- Baris 6

Pada baris 6 ini berisi *function construct* yang berarti setiap ada *link* yang mengarah kepada

controller class Login ini yang dijalankan pertama kali adalah beberapa fungsi tertentu yang sudah disiapkan sesuai kebutuhan *controller class Login* yang dipanggil didalam *function construct*.

- Baris 7

Pada baris 7 ini berisi pembuka menggunakan kurung buka kurawal dari seluruh fungsi yang akan dijalankan pada *function construct*.

- Baris 8

Pada baris 8 ini sebenarnya hampir sama seperti pada baris 6 namun perbedaannya adalah yaitu setiap ada *link* yang mengarah kepada *controller class Login* ini *construct parent* akan menjalankan seluruh *function construct* yang terdapat pada *CI_Controller* karena *class Login* ini adalah hasil *extends* dari *class CI_Controller*.

- Baris 9

- Baris 10

Pada Baris 10 ini berisi pemanggilan model bernama *MLogin* yang sebelumnya sudah dibuat terlebih dahulu dan kemudian model *MLogin* ini dapat dipakai oleh setiap fungsi di *controller class Login*.

- Baris 11

Pada baris 11 ini berisi penutup menggunakan kurung tutup kurawal dari seluruh fungsi yang akan dijalankan pada *function construct*.

b. *Function Index*

```
13 | public function index()  
14 | {  
15 |     $this->load->view("login");  
16 | }
```

Function Index

- Baris 13

Pada Baris 13 ini berisi pendeklarasian fungsi yang bernama *index* yang dapat

dipanggil oleh setiap *view* apabila *view* tersebut mengarahkannya pada fungsi ini.

- Baris 14

Pada baris 14 ini berisi pembuka menggunakan kurung buka kurawal dari seluruh isi yang akan dijalankan pada *function index*.

- Baris 15

Pada baris 15 ini adalah isi dari *function index* yang berisi pemanggilan *view* bernama *login*.

- Baris 16

Pada baris 16 ini berisi penutup menggunakan kurung tutup kurawal dari seluruh isi yang akan dijalankan pada *function index*.

c. *Function* proses

```
18 public function proses()
19 {
20     $data = array(
21         'username' => $this->input->post('username'),
22         'password' => $this->input->post('password')
23     );
24
25     $hasil = $this->Mlogin->cek_login($data);
26     if ($hasil->num_rows() == 1) {
27         foreach ($hasil->result() as $sess) {
28             $sess_data['id_user'] = $sess->id_user;
29             $sess_data['nama'] = $sess->nama;
30             $sess_data['username'] = $sess->username;
31             $sess_data['password'] = $sess->password;
32             $sess_data['id_jabatan'] = $sess->id_jabatan;
33             $sess_data['jabatan'] = $sess->jabatan;
34             $sess_data['status_login'] = 'login';
35             $sess_data['pw'] = $this->input->post('password');
36             $this->session->set_userdata($sess_data);
37         }
38         // print_r($sess); die;
39
40         if ($sess_data['id_jabatan'] == '5'){
41             redirect('Csubag_umum/index');
42         }elseif ($sess_data['id_jabatan'] == '1'){
43             redirect('Cpimpinan/index');
44         }
45         redirect('Cbagian_bagian/masuk');
46     }
47     else
48     echo"<script>alert('Username atau Password anda salah!');
49     window.location = '../login/index'</script>";
50 }
51 }
```

Function proses

- Baris 18

Pada Baris 18 ini berisi pendeklarasian fungsi yang bernama proses yang dapat dipanggil setiap *view* tertentu apabila *view* tersebut mengarahkannya pada fungsi ini.

- Baris 19

Pada baris 19 ini berisi pembuka menggunakan kurung buka kurawal dari seluruh isi yang akan dijalankan pada *function* proses.

- Baris 20

Pada baris 20 Membuat variabel array dengan nama \$data.

- Baris 21

Pada baris 21 berisikan isi dari variabel array yang memiliki isi hasil dari elemen html pada *view* yang sudah diberi nama *username*.

- Baris 22

Pada baris 22 berisikan isi dari variabel array yang memiliki isi hasil dari elemen html pada *view* yang sudah diberi nama *password*.

- Baris 23

Pada baris 23 ini berisi penutup menggunakan kurung tutup dari *variable* \$data.

- Baris 24

Pada Baris 24 berisi 1 enter untuk membuat baris koding menjadi lebih rapi.

- Baris 25

Pada baris 25 ini berisi pendeklarasian *variable* dengan nama \$hasil yang memiliki isi pemanggilan fungsi *cek_Login* pada model *MLogin* dengan membawa parameter \$data yang sudah dibuat pada baris 20.

- Baris 26

Pada baris 26 ini berisi jika *variable* \$hasil pada baris 25 memiliki nomor baris sama dengan 1 atau bisa diartikan data yang dilempar pada *variable* \$data baris 20 itu ada yang sama seperti pada tabel database.

- Baris 27

Pada baris 27 ini adalah pemanggilan seluruh data yang didapat dari *variable* \$hasil yang di inisialkan menjadi \$sess.

- Baris 28

Pada baris 28 ini adalah pembuatan *variable* dengan nama \$sess_data['id_user'] yang berisi \$sess->id_user dimana ini adalah id yang didapat dari hasil pemanggilan *variable* \$hasil pada baris 27.

- Baris 29

Pada baris 29 ini adalah pembuatan *variable* dengan nama \$sess_data['nama'] yang berisi \$sess->nama dimana ini adalah id yang didapat dari hasil pemanggilan *variable* \$hasil pada baris 27.

- Baris 30

Pada baris 30 ini adalah pembuatan *variable* dengan nama \$sess_data['username'] yang berisi \$sess->username dimana ini adalah id yang didapat dari hasil pemanggilan *variable* \$hasil pada baris 27.

- Baris 31

Pada baris 30 ini adalah pembuatan *variable* dengan nama `$sess_data['password']` yang berisi `$sess->password` dimana ini adalah id yang didapat dari hasil pemanggilan *variable* `$hasil` pada baris 27.

- Baris 32

Pada baris 30 ini adalah pembuatan *variable* dengan nama `$sess_data['id_jabatan']` yang berisi `$sess->id_jabatan` dimana ini adalah id yang didapat dari hasil pemanggilan *variable* `$hasil` pada baris 27.

- Baris 33

Pada baris 30 ini adalah pembuatan *variable* dengan nama `$sess_data['jabatan']` yang berisi `$sess->jabatan` dimana ini adalah id yang didapat dari hasil pemanggilan *variable* `$hasil` pada baris 27.

- Baris 34

Pada baris 34 ini adalah pembuatan *variable* dengan `$sess_data['status_login']` yang berisi

login dimana isinya dibuat secara bebas sesuai kebutuhan.

- Baris 35

Pada baris 35 ini adalah pembuatan *variable* dengan `$sess_data['pw']` yang berisi password yang di kirim dari *view* seperti pada baris 22 dimana isinya dibuat secara bebas sesuai kebutuhan.

- Baris 36

Pada baris 36 ini semua data yang sudah dimasukan ke dalam *variable* masing-masing mulai dari baris 28 sampai baris 35 akan dimasukan kedalam *session* sebagai wadah penyimpanan sementara data tersebut.

- Baris 37

Pada baris 37 ini berisi penutup menggunakan kurung tutup kurawal dari seluruh isi *foreach* yang ada pada baris 27 sebelumnya.

- Baris 38

Pada baris 38 berisi komen untuk mengecek isi dari variabel `$sess`.

- Baris 39

Pada Baris 24 berisi 1 enter untuk membuat baris koding menjadi lebih rapi.

- Baris 40

Pada baris 40 ini berisi jika `id_jabatan` didalam `session` dari `user` yang sudah melakukan `login` adalah 5, maka yang terjadi adalah

- Baris 41

Pada baris 41 ini berisi tujuan fungsi dari `controller Csubag_umum`, fungsi yang akan diambil adalah fungsi `index` karena tidak ada keterangan fungsi yang lain maka yang diambil adalah fungsi `index`.

- Baris 42

Pada baris 42 ini berisi penutup menggunakan kurung tutup kurawal dari seluruh isi `if` yang ada pada baris 40 sebelumnya. Selain itu jika `id_jabatan` didalam `session` dari `user` yang sudah melakukan `login` adalah 1, maka yang terjadi adalah.

- Baris 43

Pada baris 43 ini berisi tujuan fungsi dari *controller Cpinpinan*, fungsi yang akan diambil adalah fungsi *index* karena tidak ada keterangan fungsi yang lain maka yang diambil adalah fungsi *index*.

- Baris 44

Pada baris 44 ini berisi penutup menggunakan kurung tutup kurawal dari seluruh isi *if* yang ada pada baris 42 sebelumnya.

- Baris 45

Pada baris 45 ini berisi tujuan fungsi dari *controller Cbagian_bagian*, fungsi yang akan diambil adalah fungsi masuk baris ini akan berjalan apabila user yang melakukan *login* *id_jabatan* didalam *session* dari *user* selain 1 dan 5.

- Baris 46

- Baris 47

Pada baris 47 ini berisi penutup menggunakan kurung tutup kurawal dari seluruh isi *if* yang ada pada baris 26 sebelumnya.

- Baris 48

Pada baris 48 berisikan *else* yang artinya apabila variable data tidak sesuai atau tidak di temukan pada database maka

- Baris 49

Pada baris 49 akan memunculkan *pop-up* yang bertuliskan Username atau Password anda salam!

- Baris 50

Pada baris 50 ini akan menjalankan fungsi index pada controller login apabila pada *pop-up* di baris 49 menekan tombol ok.

- Baris 51

Pada baris 51 ini berisi penutup menggunakan kurung tutup kurawal dari seluruh isi yang akan dijalankan pada *function* proses.

d. *Function logout*

```

53 | public function logout()
54 | {
55 |     $this->session->sess_destroy();
56 |     redirect('login/index');
57 | }
58 |
59 | }

```

Function logout

- Baris 53

Pada Baris 53 ini berisi pendeklarasian fungsi yang bernama *logout* yang dapat dipanggil setiap *view* tertentu apabila *view* tersebut mengarahkannya pada fungsi ini.

- Baris 54

Pada baris 54 ini berisi pembuka menggunakan kurung buka kurawal dari seluruh isi yang akan dijalankan pada *function logout*.

- Baris 55

Pada baris 55 ini adalah menghapus seluruh data yang terdapat pada *session* atau penyimpanan sementara yang sebelumnya

sudah dideklarasikan pada baris 28 sampai baris 36.

- Baris 56

Pada baris 56 ini berisi tujuan fungsi dari *controller Login*, fungsi yang akan diambil adalah fungsi *index* karena tidak ada keterangan fungsi yang lain maka yang diambil adalah fungsi *index*.

- Baris 57

Pada baris 57 ini berisi penutup menggunakan kurung tutup kurawal dari seluruh isi yang akan dijalankan pada *function logout*.

- Baris 58

Pada Baris 58 berisi 1 enter untuk membuat baris koding menjadi lebih rapi.

- Baris 59

Pada baris 59 ini berisi penutup menggunakan kurung tutup kurawal dari seluruh isi *controller class Login* berisi fungsi-fungsi yang akan dijalankan.

2. *Controller* Csubag_umum

a. *Controller class* Csubag_umum

```
1  <?php
2  defined('BASEPATH') OR exit('No direct script access allowed');
3
4  class Csubag_umum extends CI_Controller {
5
6      public function __construct(){
7          parent::__construct();
8          $this->load->model('Msubag_umum');
9          $this->load->helper('url_helper');
10         if($this->session->userdata('status_login') != "login"){
11             redirect(base_url("login/index"));
12         }
13     }
14 }
```

Controller class Csubag_umum

- Baris 1

Pada baris 1 adalah membuka bahasa php pada halaman tersebut dimulai dari baris 1 yang kemudian baris-baris selanjutnya dapat juga menggunakan seluruh elemen dari bahasa php.

- Baris 2

Digunakan untuk memastikan bahwa permintaan telah melalui indeks. php di direktori root Anda. Ini untuk alasan seperti

memastikan bahwa semua kelas dasar Codeigniter sedang dimuat dan memastikan vars tertentu telah ditetapkan dll.

- Baris 3

Pada Baris 3 berisi 1 enter untuk membuat baris koding menjadi lebih rapi.

- Baris 4

Pada baris 4 berisi pendeklarasian *class* yang bernama *Csubag_umum* dan hasil *extends* dari *CI_Controller*, yang berarti *class* *Csubag_umum* ini mendapatkan sifat-sifat/atribut dari class induk yaitu *CI_Controller*.

- Baris 5

Pada Baris 5 berisi 1 enter untuk membuat baris koding menjadi lebih rapi.

- Baris 6

Pada baris 6 ini berisi *function construct* yang berarti setiap ada *link* yang mengarah kepada *controller class* *Csubag_umum* ini yang dijalankan pertama kali adalah beberapa

fungsi tertentu yang sudah disiapkan sesuai kebutuhan *controller class* Csubag_umum yang dipanggil didalam *function construct*.

- Baris 7

Pada baris 7 ini sebenarnya hampir sama seperti pada baris 6 namun perbedaannya adalah yaitu setiap ada *link* yang mengarah kepada *controller class Login* ini *construct parent* akan menjalankan seluruh *function construct* yang terdapat pada *CI_Controller* karena *class Login* ini adalah hasil *extends* dari *class CI_Controller*.

- Baris 8

Pada Baris 8 ini berisi pemanggilan model bernama Msubag_umum yang sebelumnya sudah dibuat terlebih dahulu dan kemudian model Msubag_umum ini dapat dipakai oleh setiap fungsi di *controller class* Csubag_umum.

- Baris 9

- Baris 10

Pada baris 10 ini memiliki arti jika session dari usertdata atau session dari data user yang login memiliki 'status_login' tidak sama dengan 'login' atau lebih sederhananya setatus dari session tersebut tidak login maka.

- Baris 11

Pada baris 11 melanjutkan baris 10 dimana apa bila status nya tidak login maka menjalankan fungsi index pada controller login.

- Baris 12

Pada baris 12 memnjadi penutup kondisi 'if' atau jika dengan menggunakan tanda tutup kurung kurawal yang menjlankan fungsi jika dari baris ke 10.

- Baris 13

Pada baris 13 ini berisi penutup menggunakan kurung tutup kurawal dari seluruh fungsi yang akan dijalankan pada *function construct*.

b. *Function index*

```
15 public function index()  
16 {  
17     $title['title'] = "Dashboard";  
18     $data['grafik'] = $this->Msubag_umum->grafik();  
19     $data['grafik1'] = $this->Msubag_umum->grafik_keluar();  
20     // print_r($data); die;  
21     $this->load->view('componen_umum/head',$title);  
22     $this->load->view('componen_umum/side');  
23     $this->load->view('subag_umum/index',$data);  
24     $this->load->view('componen_umum/foot');  
25 }
```

Function index

- Baris 15

Pada Baris 15 ini berisi pendeklarasian fungsi yang bernama *index* yang dapat dipanggil oleh setiap *view* apabila *view* tersebut mengarahkannya pada fungsi ini.

- Baris 16

Pada baris 16 ini berisi pembuka menggunakan kurung buka kurawal dari seluruh fungsi yang akan dijalankan pada *function index*.

- Baris 17

Pada baris 17 ini berisi variable yang bernama \$title yang mana variable ini berfungsi untuk mengubah title dari halaman web tersebut dan isi \$title yaitu 'Dashboard'.

- Baris 18

Pada baris 18 ini berisikan variable yang bernama \$data diaman isi dari variable ini yaitu memanggil fungsi grafik pada model Msubag_umum.

- Baris 19

Pada baris 19 ini berisikan variable yang bernama \$data diaman isi dari variable ini yaitu memanggil fungsi grafik_keluar pada model Msubag_umum.

- Baris 20

Pada baris 20 berisi komen dimana untuk menampilkan isi yang ada pada variabel data(\$data).

- Baris 21

Pada baris 21 memanggil view dengan nama head pada folder componen_umum di bagian

view, dan membawa variable title untuk view head.

- Baris 22

Pada baris 22 memanggil view dengan nama side pada folder componen_umum di bagian view.

- Baris 23

Pada baris 23 memanggil view dengan nama index pada folder subag_umum di bagian view, dan membawa variable data untuk view index.

- Baris 24

Pada baris 24 memanggil view dengan nama foot pada folder componen_umum di bagian view.

- Baris 25

Pada baris 25 ini berisi penutup menggunakan kurung tutup kurawal dari seluruh fungsi yang akan dijalankan pada *function index*.

c. *Function user*

```
26 public function user()  
27 {  
28     $title['title'] = "Tabel User";  
29     $data['user'] = $this->Msubag_umum->get_user();  
30     // print_r($data); die;  
31     $this->load->view('componen_umum/head',$title);  
32     $this->load->view('componen_umum/side');  
33     $this->load->view('subag_umum/user',$data);  
34     $this->load->view('componen_umum/foot');  
35 }
```

Function user

- Baris 26

Pada Baris 26 ini berisi pendeklarasian fungsi yang bernama *user* yang dapat dipanggil oleh setiap *view* apabila *view* tersebut mengarahkannya pada fungsi ini.

- Baris 27

Pada baris 27 ini berisi pembuka menggunakan kurung buka kurawal dari seluruh fungsi yang akan dijalankan pada *function user*.

- Baris 28

Pada baris 28 ini berisi variable yang bernama \$title yang mana variable ini berfungsi untuk mengubah title dari halaman web tersebut dan isi \$title yaitu 'Tabel User'.

- Baris 29

Pada baris 29 ini berisikan variable yang bernama \$data diaman isi dari variable ini yaitu memanggil fungsi get_user pada model Msubag_umum.

- Baris 30

Pada baris 30 berisi komen dimana untuk menampilkan isi yang ada pada variabel data(\$data).

- Baris 31

Pada baris 31 memanggil view dengan nama head pada folder componen_umum di bagian view, dan membawa variable title untuk view head.

- Baris 32

Pada baris 32 memanggil view dengan nama side pada folder componen_umum di bagian view.

- Baris 33

Pada baris 23 memanggil view dengan nama user pada folder subag_umum di bagian view, dan membawa variable data untuk view user.

- Baris 34

Pada baris 34 memanggil view dengan nama foot pada folder componen_umum di bagian view.

- Baris 35

Pada baris 35 ini berisi penutup menggunakan kurung tutup kurawal dari seluruh fungsi yang akan dijalankan pada *function user*.

d. *Function user*Tambah


```

36     public function userTambah()
37     {
38         $title['title'] = "Tambah User";
39         $data['jabatan'] = $this->Msubag_umum->get_jabatan();
40         $this->load->view('componen_umum/head',$title);
41         $this->load->view('componen_umum/side');
42         $this->load->view('subag_umum/userTambah',$data);
43         $this->load->view('componen_umum/foot');
44     }
45

```

Function userTambah

- Baris 36

Pada Baris 36 ini berisi pendeklarasian fungsi yang bernama `userTambah` yang dapat dipanggil oleh setiap *view* apabila *view* tersebut mengarahkannya pada fungsi ini.

- Baris 37

Pada baris 37 ini berisi pembuka menggunakan kurung buka kurawal dari seluruh fungsi yang akan dijalankan pada *function userTambah*.

- Baris 38

Pada baris 38 ini berisi variable yang bernama \$title yang mana variable ini berfungsi untuk mengubah title dari halaman web tersebut dan isi \$title yaitu 'Tambah User'.

- Baris 39

Pada baris 39 ini berisikan variable yang bernama \$data diaman isi dari variable ini yaitu memanggil fungsi get_jabatan pada model Msubag_umum.

- Baris 40

Pada baris 40 memanggil view dengan nama head pada folder componen_umum di bagian view, dan membawa variable title untuk view head.

- Baris 41

Pada baris 41 memanggil view dengan nama side pada folder componen_umum di bagian view.

- Baris 42

Pada baris 42 memanggil view dengan nama userTambah pada folder subag_umum di

bagian view, dan membawa variable data untuk view userTambah.

- Baris 43

Pada baris 43 memanggil view dengan nama foot pada folder componen_umum di bagian view.

- Baris 44

Pada baris 44 ini berisi penutup menggunakan kurung tutup kurawal dari seluruh fungsi yang akan dijalankan pada *function userTambah*.

e. Function userEdit

```

46     public function userEdit($id_user)
47     {
48         $title['title'] = "Edit User";
49         $data['u'] = $this->Msubag_umum->show_user($id_user);
50         $data['jabatan'] = $this->Msubag_umum->get_jabatan();
51         // print_r($data); die;
52         $this->load->view('componen_umum/head',$title);
53         $this->load->view('componen_umum/side');
54         $this->load->view('subag_umum/userEdit',$data);
55         $this->load->view('componen_umum/foot');
56     }
57

```

Function userEdit

- Baris 46

Pada Baris 46 ini berisi pendeklarasian fungsi yang bernama userEdit yang dapat dipanggil oleh setiap *view* apabila *view* tersebut mengarahkannya pada fungsi ini.

- Baris 47

Pada baris 47 ini berisi pembuka menggunakan kurung buka kurawal dari seluruh fungsi yang akan dijalankan pada *function userEdit*.

- Baris 48

Pada baris 48 ini berisi variable yang bernama \$title yang mana variable ini berfungsi untuk mengubah title dari halaman web tersebut dan isi \$title yaitu 'Edit User'.

- Baris 49

Pada baris 49 ini berisikan variable yang bernama \$data diaman isi dari variable ini yaitu memanggil fungsi show_user pada model Msubag_umum.

- Baris 50

Pada baris 50 ini berisikan variable yang bernama \$data diaman isi dari variable ini yaitu memanggil fungsi get_jabatan pada model Msubag_umum.

- Baris 51

Pada baris 51 berisikan koment dimana mempunyai fungsi untuk mengeahui is dari variable data(\$data).

- Baris 52

Pada baris 52 memanggil view dengan nama head pada folder componen_umum di bagian

view, dan membawa variable title untuk view head.

- Baris 53

Pada baris 53 memanggil view dengan nama side pada folder componen_umum di bagian view.

- Baris 54

Pada baris 54 memanggil view dengan nama userEdit pada folder subag_umum di bagian view, dan membawa variable data untuk view userEdit.

- Baris 55

Pada baris 55 memanggil view dengan nama foot pada folder componen_umum di bagian view.

- Baris 56

Pada baris 56 ini berisi penutup menggunakan kurung tutup kurawal dari seluruh fungsi yang akan dijalankan pada *function userEdit*.

f. Function insertuser

```
58 public function insertUser()  
59 {  
60     $this->Msubag_umum->insert_user();  
61     redirect('Csubag_umum/user');  
62 }  
63  
64
```

Function insertuser

- Baris 58

Pada Baris 58 ini berisi pendeklarasian fungsi yang bernama insertuser yang dapat dipanggil oleh setiap *view* apabila *view* tersebut mengarahkannya pada fungsi ini.

- Baris 59

Pada baris 59 ini berisi pembuka menggunakan kurung buka kurawal dari seluruh fungsi yang akan dijalankan pada *function insertuser*.

- Baris 60

Pada baris 60 ini mempunyai fungsi memanggil atau menjalankan fungsi

insert_user yang ada pada model yang bernama Msubag_umum.

- Baris 61

Pada baris 61 melanjutkan dari baris 60 setelah selesai pada baris ini memanggil atau menjalankan fungsi user pada controler yang bernama Csubag_umum

- Baris 62

- Baris 63

Pada baris 63 ini berisi penutup menggunakan kurung tutup kurawal dari seluruh fungsi yang akan dijalankan pada *function insertuser*.

g. *Function updateuser*

```
65 public function updateUser()  
66 {  
67     // print_r($id); die;  
68     $id = $this->input->post('id_user');  
69     $this->Msubag_umum->update_user($id);  
70     redirect('Csubag_umum/user');  
71 }  
72
```

Function updateuser

- Baris 65

Pada Baris 65 ini berisi pendeklarasian fungsi yang bernama `updateuser` yang dapat dipanggil oleh setiap *view* apabila *view* tersebut mengarahkannya pada fungsi ini.

- Baris 66

Pada baris 66 ini berisi pembuka menggunakan kurung buka kurawal dari seluruh fungsi yang akan dijalankan pada *function updateuser*.

- Baris 67

- Baris 68

Pada baris 68 ini membuat variable bernama `id($id)` dimana isi variabel tersebut di peroleh dari *view* yang di kirim(post) dengan nama `id_user` untuk fungsi `updateuser` ini.

- Baris 69

Pada barais 69 ini mempunyai fungsi memanggil atau menjalankan fungsi `insert_user` yang ada pada model yang

bernama `Msubag_umum`, berdasarkan id yang didapat.

- Baris 70

Pada baris 70 melanjutkan dari baris 69 setelah selesai pada baris ini memanggil atau menjalankan fungsi user pada controller yang bernama `Csubag_umum`

- Baris 71

Pada baris 71 ini berisi penutup menggunakan kurung tutup kurawal dari seluruh fungsi yang akan dijalankan pada *function insertuser*.

h. Function deleteuser

```
73 public function deleteUser($id)
74 {
75     $this->Msubag_umum->delete_user($id);
76     redirect('Csubag_umum/user');
77 }
78
```

Function deleteuser

- Baris 73

Pada Baris 73 ini berisi pendeklarasian fungsi yang bernama `deleteuser` yang dapat

dipanggil oleh setiap *view* apabila *view* tersebut mengarahkannya pada fungsi ini.

- Baris 74

Pada baris 74 ini berisi pembuka menggunakan kurung buka kurawal dari seluruh fungsi yang akan dijalankan pada *function deleteuser*.

- Baris 75

Pada baris 75 ini mempunyai fungsi memanggil atau menjalankan fungsi `delete_user` yang ada pada model yang bernama `Msubag_umum`, berdasarkan id yang di dapat.

- Baris 76

Pada baris 76 melanjutkan dari baris 75 setelah selesai, pada baris ini memanggil atau menjalankan fungsi user pada controler yang bernama `Csubag_umum`

- Baris 77

Pada baris 77 ini berisi penutup menggunakan kurung tutup kurawal dari

seluruh fungsi yang akan dijalankan pada *function deleteuser*.

i. *Function* masuk

```
79 public function masuk()  
80 {  
81     $title['title'] = "Daftar Surat Masuk";  
82     $data['masuk'] = $this->Msubag_umum->get_masuk();  
83     $data['acara'] = $this->Msubag_umum->get_acara();  
84     // print_r($data); die;  
85     $this->load->view('componen_umum/head',$title);  
86     $this->load->view('componen_umum/side');  
87     $this->load->view('subag_umum/masuk/masuk',$data);  
88     $this->load->view('componen_umum/foot');  
89 }
```

Function masuk

- Baris 79

Pada Baris 79 ini berisi pendeklarasian fungsi yang bernama masuk yang dapat dipanggil oleh setiap *view* apabila *view* tersebut mengarahkannya pada fungsi ini.

- Baris 80

Pada baris 80 ini berisi pembuka menggunakan kurung buka kurawal dari seluruh fungsi yang akan dijalankan pada *function* masuk.

- Baris 81

Pada baris 81 ini berisi variable yang bernama \$title yang mana variable ini berfungsi untuk mengubah title dari halaman web tersebut dan isi \$title yaitu 'Dafar Surat Masuk'.

- Baris 82

Pada baris 82 ini berisikan variable yang bernama \$data diaman isi dari variable ini yaitu memanggil fungsi get_masuk pada model Msubag_umum.

- Baris 83

Pada baris 83 ini berisikan variable yang bernama \$data diaman isi dari variable ini yaitu memanggil fungsi get_acara pada model Msubag_umum.

- Baris 84

Pada baris 84 berisikan koment dimana mempunyai fungsi untuk mengeahui is dari variable data(\$data).

- Baris 85

Pada baris 85 memanggil view dengan nama head pada folder componen_umum di bagian view, dan membawa variable title untuk view head.

- Baris 86

Pada baris 86 memanggil view dengan nama side pada folder componen_umum di bagian view.

- Baris 87

Pada baris 87 memanggil view dengan nama masuk pada folder masuk dalam folder subag_umum di bagian view, dan membawa variable data untuk view masuk.

- Baris 88

Pada baris 88 memanggil view dengan nama foot pada folder componen_umum di bagian view.

- Baris 89

Pada baris 89 ini berisi penutup menggunakan kurung tutup kurawal dari

seluruh fungsi yang akan dijalankan pada *function* masuk.

j. *Function* masukTambah

```
90 public function masukTambah()  
91 {  
92     $title['title'] = "Tambah Surat Masuk";  
93     // $data['masuk'] = $this->Msubag_umum->get_masuk();  
94     // print_r($data); die;  
95     $this->load->view('componen_umum/head',$title);  
96     $this->load->view('componen_umum/side');  
97     $this->load->view('subag_umum/masuk/masuk_tambah');  
98     $this->load->view('componen_umum/foot');  
99 }  
100
```

Function masukTambah

- Baris 90

Pada Baris 90 ini berisi pendeklarasian fungsi yang bernama *masukTambah* yang dapat dipanggil oleh setiap *view* apabila *view* tersebut mengarahkannya pada fungsi ini.

- Baris 91

Pada baris 91 ini berisi pembuka menggunakan kurung buka kurawal dari seluruh fungsi yang akan dijalankan pada *function* *masukTambah*.

- Baris 92

Pada baris 92 ini berisi variable yang bernama \$title yang mana variable ini berfungsi untuk mengubah title dari halaman web tersebut dan isi \$title yaitu 'Tambah Surat Masuk'.

- Baris 93

- Baris 94

- Baris 95

Pada baris 95 memanggil view dengan nama head pada folder componen_umum di bagian view, dan membawa variable title untuk view head.

- Baris 96

Pada baris 96 memanggil view dengan nama side pada folder componen_umum di bagian view.

- Baris 97

Pada baris 97 memanggil view dengan nama `masuk_tambah` pada folder `masuk` dalam folder `subag_umum` di bagian `view`, dan membawa `variable data` untuk `view` `masuk_tambah`.

- Baris 98

Pada baris 98 memanggil view dengan nama `foot` pada folder `componen_umum` di bagian `view`.

- Baris 99

Pada baris 99 ini berisi penutup menggunakan kurung tutup kurawal dari seluruh fungsi yang akan dijalankan pada *function* `masukTambah`.

k. *Function insertmasuk*

```

101 public function insertMasuk()
102 {
103     $config['upload_path'] = './assets/file/';
104     $config['allowed_types'] = 'doc|docx|xls|xlsx|pdf';
105     $config['max_size'] = '2000';
106
107     $this->load->library('upload', $config);
108
109     if (!$this->upload->do_upload('file_surat')) {
110         $error = $this->upload->display_errors();
111         // menampilkan pesan error
112         print_r($error);
113     } else {
114         $result = $this->upload->data();
115         $name = $result['file_name'];
116         $this->Msubag_umum->insert_masuk($name);
117         echo"<script>alert('Data Berhasil di Insert');
118         window.location = '../Csubag_umum/masuk/masuk'</script>";
119     }
120 }
121

```

Function insertmasuk

l. *Function masukEdit*

```

123 public function masukEdit($id_surat)
124 {
125     $title['title'] = "Edit Surat Masuk";
126     $data['m'] = $this->Msubag_umum->show_masuk($id_surat);
127
128     $this->load->view('componen_umum/head',$title);
129     $this->load->view('componen_umum/side');
130     $this->load->view('subag_umum/masuk/masukEdit',$data);
131     $this->load->view('componen_umum/foot');
132 }
133

```

Function masukEdit

- Baris 123

Pada Baris 123 ini berisi pendeklarasian fungsi yang bernama `masukEdit` yang dapat dipanggil oleh setiap *view* apabila *view* tersebut mengarahkannya pada fungsi ini.

- Baris 124

Pada baris 124 ini berisi pembuka menggunakan kurung buka kurawal dari seluruh fungsi yang akan dijalankan pada *function* `masukEdit`.

- Baris 125

Pada baris 125 ini berisi variable yang bernama `$title` yang mana variable ini berfungsi untuk mengubah title dari halaman web tersebut dan isi `$title` yaitu 'Edit Surat Masuk'.

- Baris 126

Pada baris 126 ini berisikan variable yang bernama `$data` diaman isi dari variable ini yaitu memanggil fungsi `show_masuk` pada model `Msubag_umum`, berdasarkan id yang di bawa.

- Baris 127

- Baris 128

Pada baris 128 memanggil view dengan nama head pada folder componen_umum di bagian view, dan membawa variable title untuk view head.

- Baris 129

Pada baris 129 memanggil view dengan nama side pada folder componen_umum di bagian view.

- Baris 130

Pada baris 130 memanggil view dengan nama masukEdit pada folder masuk dalam folder subag_umum di bagian view, dan membawa variable data untuk view masukEdit.

- Baris 131

Pada baris 131 memanggil view dengan nama foot pada folder componen_umum di bagian view.

- Baris 132

Pada baris 132 ini berisi penutup menggunakan kurung tutup kurawal dari seluruh fungsi yang akan dijalankan pada *function* masukEdit.

m. Function updatemasuk

```

134 public function updateMasuk()
135 {
136     $id = $this->input->post('id_surat');
137     $config['upload_path'] = './assets/file/';
138     $config['allowed_types'] = 'doc|docx|xls|xlsx|pdf';
139     $config['max_size'] = '2000';
140
141     $this->load->library('upload', $config);
142
143     if (!$this->upload->do_upload('file_surat')) {
144         $error = $this->upload->display_errors();
145         // menampilkan pesan error
146         print_r($error);
147     } else {
148         $result = $this->upload->data();
149         $name = $result['file_name'];
150         // print_r($id); die;
151         $this->Msubag_umum->update_masuk($id, $name);
152         echo"<script>alert('Data Berhasil di Update');
153         window.location = '../Csubag_umum/masuk/masuk'</script>";
154     }
155 }
156
157

```

Function updatemasuk

n. Function deletemasuk

```
158     public function deleteMasuk($id)
159     {
160         $this->Msubag_umum->delete_masuk($id);
161         redirect('Csubag_umum/masuk/masuk');
162     }
163
```

Function deletemasuk

- Baris 158

Pada Baris 158 ini berisi pendeklarasian fungsi yang bernama `delete_masuk` yang dapat dipanggil oleh setiap *view* apabila *view* tersebut mengarahkannya pada fungsi ini.

- Baris 159

Pada baris 159 ini berisi pembuka menggunakan kurung buka kurawal dari seluruh fungsi yang akan dijalankan pada *function deletemasuk*.

- Baris 160

Pada barais 160 ini mempunyai fungsi memanggil atau menjalankan fungsi `delete_masuk` yang ada pada model yang bernama `Msubag_umum`, berdasarkan id yang di dapat.

- Baris 161

Pada baris 161 melanjutkan dari baris 160 setelah selesai, pada baris ini memanggil atau menjalankan fungsi masuk pada controler yang bernama Csubag_umum

- Baris 162

Pada baris 162 ini berisi penutup menggunakan kurung tutup kurawal dari seluruh fungsi yang akan dijalankan pada *function deletemasuk*.

o. Function keluar

```
164 public function keluar()  
165 {  
166     $title['title'] = "Daftar Surat Keluar";  
167     $data['keluar'] = $this->Msubag_umum->get_keluar();  
168     // print_r($data); die;  
169     $this->load->view('componen_umum/head',$title);  
170     $this->load->view('componen_umum/side');  
171     $this->load->view('subag_umum/keluar/keluar',$data);  
172     $this->load->view('componen_umum/foot');  
173 }  
174
```

Function keluar

- Baris 79

Pada Baris 79 ini berisi pendeklarasian fungsi yang bernama `masuk` yang dapat dipanggil oleh setiap *view* apabila *view* tersebut mengarahkannya pada fungsi ini.

- Baris 80

Pada baris 80 ini berisi pembuka menggunakan kurung buka kurawal dari seluruh fungsi yang akan dijalankan pada *function* `masuk`.

- Baris 81

Pada baris 81 ini berisi variable yang bernama `$title` yang mana variable ini berfungsi untuk mengubah title dari halaman web tersebut dan isi `$title` yaitu 'Dafar Surat Masuk'.

- Baris 82

Pada baris 82 ini berisikan variable yang bernama `$data` diaman isi dari variable ini yaitu memanggil fungsi `get_masuk` pada model `Msubag_umum`.

- Baris 83

Pada baris 83 ini berisikan variable yang bernama \$data diaman isi dari variable ini yaitu memanggil fungsi get_acara pada model Msubag_umum.

- Baris 84

Pada baris 84 berisikan koment dimana mempunyai fungsi untuk mengeahui is dari variable data(\$data).

- Baris 85

Pada baris 85 memanggil view dengan nama head pada folder componen_umum di bagian view, dan membawa variable title untuk view head.

- Baris 86

Pada baris 86 memanggil view dengan nama side pada folder componen_umum di bagian view.

- Baris 87

Pada baris 87 memanggil view dengan nama masuk pada folder masuk dalam folder

subag_umum di bagian view, dan membawa variable data untuk view masuk.

- Baris 88

Pada baris 88 memanggil view dengan nama foot pada folder componen_umum di bagian view.

- Baris 89

Pada baris 89 ini berisi penutup menggunakan kurung tutup kurawal dari seluruh fungsi yang akan dijalankan pada *function* masuk.

p. Function keluarTambah

```
175 public function keluarTambah()  
176 {  
177     $title['title'] = "Tambah Surat Keluar";  
178     $this->load->view('componen_umum/head',$title);  
179     $this->load->view('componen_umum/side');  
180     $this->load->view('subag_umum/keluar/keluar_tambah');  
181     $this->load->view('componen_umum/foot');  
182 }  
183
```

Function keluarTambah

- Baris 90

Pada Baris 90 ini berisi pendeklarasian fungsi yang bernama `masukTambah` yang dapat dipanggil oleh setiap *view* apabila *view* tersebut mengarahkannya pada fungsi ini.

- Baris 91

Pada baris 91 ini berisi pembuka menggunakan kurung buka kurawal dari seluruh fungsi yang akan dijalankan pada *function* `masukTambah`.

- Baris 92

Pada baris 92 ini berisi variable yang bernama `$title` yang mana variable ini berfungsi untuk mengubah title dari halaman web tersebut dan isi `$title` yaitu ‘Tambah Surat Masuk’.

- Baris 93

- Baris 94

- Baris 95

Pada baris 95 memanggil view dengan nama head pada folder componen_umum di bagian view, dan membawa variable title untuk view head.

- Baris 96

Pada baris 96 memanggil view dengan nama side pada folder componen_umum di bagian view.

- Baris 97

Pada baris 97 memanggil view dengan nama masuk_tambah pada folder masuk dalam folder subag_umum di bagian view, dan membawa variable data untuk view masuk_tambah.

- Baris 98

Pada baris 98 memanggil view dengan nama foot pada folder componen_umum di bagian view.

- Baris 99

Pada baris 99 ini berisi penutup menggunakan kurung tutup kurawal dari

seluruh fungsi yang akan dijalankan pada *function* masukTambah.

q. Function insertkeluar

```
184 public function insertkeluar()  
185 {  
186     $config['upload_path'] = './assets/file/';  
187     $config['allowed_types'] = 'doc|docx|xls|xlsx|pdf';  
188     $config['max_size'] = '2000';  
189  
190     $this->load->library('upload', $config);  
191  
192     if (!$this->upload->do_upload('file_surat')) {  
193         $error = $this->upload->display_errors();  
194         // menampilkan pesan error  
195         print_r($error);  
196     } else {  
197         $result = $this->upload->data();  
198         $name = $result['file_name'];  
199         $this->Msubag_umum->insert_keluar($name);  
200         echo"<script>alert('Data Berhasil di Insert');  
201         window.location = '../Csubag_umum/keluar/keluar'</script>";  
202     }  
203 }  
204  
205
```

Function insertkeluar

r. Function keluarEdit

```

206 public function keluarEdit($id_keluar)
207 {
208     $title['title'] = "Edit Surat Keluar";
209     $data['m'] = $this->Msubag_umum->show_keluar($id_keluar);
210
211     $this->load->view('componen_umum/head',$title);
212     $this->load->view('componen_umum/side');
213     $this->load->view('subag_umum/keluar/keluarEdit',$data);
214     $this->load->view('componen_umum/foot');
215 }
216

```

Function keluarEdit

- Baris 123

Pada Baris 123 ini berisi pendeklarasian fungsi yang bernama masukEdit yang dapat dipanggil oleh setiap *view* apabila *view* tersebut mengarahkannya pada fungsi ini.

- Baris 124

Pada baris 124 ini berisi pembuka menggunakan kurung buka kurawal dari seluruh fungsi yang akan dijalankan pada *function* masukEdit.

- Baris 125

Pada baris 125 ini berisi variable yang bernama \$title yang mana variable ini berfungsi untuk mengubah title dari halaman

web tersebut dan isi \$title yaitu 'Edit Surat Masuk'.

- Baris 126

Pada baris 126 ini berisikan variable yang bernama \$data diaman isi dari variable ini yaitu memanggil fungsi show_masuk pada model Msubag_umum, berdasrakan id yang di bawa.

- Baris 127

- Baris 128

Pada baris 128 memanggil view dengan nama head pada folder componen_umum di bagian view, dan membawa variable title untuk view head.

- Baris 129

Pada baris 129 memanggil view dengan nama side pada folder componen_umum di bagian view.

- Baris 130

Pada baris 130 memanggil view dengan nama `masukEdit` pada folder `masuk` dalam folder `subag_umum` di bagian `view`, dan membawa variable data untuk view `masukEdit`.

- Baris 131

Pada baris 131 memanggil view dengan nama `foot` pada folder `componen_umum` di bagian `view`.

- Baris 132

Pada baris 132 ini berisi penutup menggunakan kurung tutup kurawal dari seluruh fungsi yang akan dijalankan pada *function* `masukEdit`.

s. *Function updatekeluar*


```

217 public function updatekeluar()
218 {
219     $id = $this->input->post('id_surat');
220     $config['upload_path'] = './assets/file/';
221     $config['allowed_types'] = 'doc|docx|xls|xlsx|pdf';
222     $config['max_size'] = '2000';
223
224     $this->load->library('upload', $config);
225
226     if (!$this->upload->do_upload('file_surat')) {
227         $error = $this->upload->display_errors();
228         // menampilkan pesan error
229         print_r($error);
230     } else {
231         $result = $this->upload->data();
232         $name = $result['file_name'];
233         // print_r($id); die;
234         $this->Msubag_umum->update_keluar($id, $name);
235         echo"<script>alert('Data Berhasil di Update');
236         window.location = '../Csubag_umum/keluar/keluar'</script>";
237     }
238 }

```

Function updatekeluar

t. *Function deletekeluar*

```

239
240 public function deletekeluar($id)
241 {
242     $this->Msubag_umum->delete_keluar($id);
243     redirect('Csubag_umum/keluar/keluar');
244 }
245
246 }

```

Function deletekeluar

- Baris 240

Pada Baris 240 ini berisi pendeklarasian fungsi yang bernama `deletekeluar` yang dapat dipanggil oleh setiap *view* apabila *view* tersebut mengarahkannya pada fungsi ini.

- Baris 241

Pada baris 241 ini berisi pembuka menggunakan kurung buka kurawal dari seluruh fungsi yang akan dijalankan pada *function deletekeluar*.

- Baris 242

Pada baris 242 ini mempunyai fungsi memanggil atau menjalankan fungsi `delete_keluar` yang ada pada model yang bernama `Msubag_umum`, berdasarkan id yang di dapat.

- Baris 243

Pada baris 243 melanjutkan dari baris 242 setelah selesai, pada baris ini memanggil atau menjalankan fungsi keluar pada controler yang bernama `Csubag_umum`

- Baris 244

Pada baris 244 ini berisi penutup menggunakan kurung tutup kurawal dari seluruh fungsi yang akan dijalankan pada *function deleteuser*.

- Baris 246

Pada baris 246 ini berisi penutup menggunakan kurung tutup kurawal dari seluruh isi *controller class Csubag_umum* berisi fungsi-fungsi yang akan dijalankan.

3. *Controller Cpimpinan*

a. *Controller class Cpimpinan*

```
1  <?php
2  defined('BASEPATH') OR exit('No direct script access allowed');
3
4  class Cpimpinan extends CI_Controller {
5
6      public function __construct(){
7          parent::__construct();
8          $this->load->model('Mpimpinan');
9          $this->load->helper('url_helper');
10         if($this->session->userdata('status_login') != "login"){
11             redirect(base_url("login/index"));
12         }
13     }
14 }
```

b. *Function index*

```

15     public function index()
16     {
17         $title['title'] = "Dashboard";
18         $data['grafik'] = $this->Mpimpinan->grafik();
19         $data['grafik1'] = $this->Mpimpinan->grafik_keluar();
20         // print_r($data); die;
21         $this->load->view('componen_pimpinan/head',$title);
22         $this->load->view('componen_pimpinan/side');
23         $this->load->view('pimpinan/index',$data);
24         $this->load->view('componen_pimpinan/foot');
25     }
26

```

c. *Function* masuk

```

27     public function masuk()
28     {
29         $title['title'] = "Daftar Surat Masuk";
30         $data['masuk'] = $this->Mpimpinan->get_masuk();
31         // print_r($data); die;
32         $this->load->view('componen_pimpinan/head',$title);
33         $this->load->view('componen_pimpinan/side');
34         $this->load->view('pimpinan/masuk/masuk',$data);
35         $this->load->view('componen_pimpinan/foot');
36     }
37

```

d. *Function* keluar

```

38     public function keluar()
39     {
40         $title['title'] = "Daftar Surat Keluar";
41         $data['keluar'] = $this->Mpimpinan->get_keluar();
42         // print_r($data); die;
43         $this->load->view('componen_pimpinan/head',$title);
44         $this->load->view('componen_pimpinan/side');
45         $this->load->view('pimpinan/keluar/keluar',$data);
46         $this->load->view('componen_pimpinan/foot');
47     }
48

```

e. *Function dispos*

```

49     public function dispos($id_surat)
50     {
51         $title['title'] = "Form Disposisi";
52         $data['m'] = $this->Mpimpinan->show_masuk($id_surat);
53         $data['jabatan'] = $this->Mpimpinan->get_jabatan();
54         // print_r($data); die;
55         $this->load->view('componen_pimpinan/head',$title);
56         $this->load->view('componen_pimpinan/side');
57         $this->load->view('pimpinan/disposisi',$data);
58         $this->load->view('componen_pimpinan/foot');
59     }
60

```

f. *Function disposisi*

```

61     public function disposisi()
62     {
63         $this->Mpimpinan->insert_dispos();
64         redirect('Cpimpinan/masuk');
65     }
66 }
67
68 }

```

4. *Controller Cbagian_bagian*

a. *Controller Class Cbagian_bagian*

```

1  <?php
2  defined('BASEPATH') OR exit('No direct script access allowed');
3
4  class Cbagian_bagian extends CI_Controller {
5
6      public function __construct(){
7          parent::__construct();
8          $this->load->model('Mbagian_bagian');
9          $this->load->helper('url_helper');
10         if($this->session->userdata('status_login') != "login"){
11             redirect(base_url("login/index"));
12         }
13     }
14 }

```

b. *Function index*

```

15 | public function index()
16 | {
17 |     $title['title'] = "Dashboard";
18 |     $data['grafik'] = $this->Mbagian_bagian->grafik();
19 |     $data['grafik1'] = $this->Mbagian_bagian->grafik_keluar();
20 |     // print_r($data); die;
21 |     $this->load->view('componen_bagian/head',$title);
22 |     $this->load->view('componen_bagian/side');
23 |     $this->load->view('bagian_bagian/index',$data);
24 |     $this->load->view('componen_bagian/foot');
25 | }
26 |

```

c. *Function masuk*

```

27 | public function masuk()
28 | {
29 |     $title['title'] = "Daftar Surat Masuk";
30 |     $data['masuk'] = $this->Mbagian_bagian->get_masuk();
31 |     // print_r($data); die;
32 |     $this->load->view('componen_bagian/head',$title);
33 |     $this->load->view('componen_bagian/side');
34 |     $this->load->view('bagian_bagian/masuk/masuk',$data);
35 |     $this->load->view('componen_bagian/foot');
36 | }
37 |

```

d. *Function keluar*

```

38 public function keluar()
39 {
40     $title['title'] = "Daftar Surat Keluar";
41     $data['keluar'] = $this->Mbagian_bagian->get_keluar();
42     // print_r($data); die;
43     $this->load->view('componen_bagian/head',$title);
44     $this->load->view('componen_bagian/side');
45     $this->load->view('bagian_bagian/keluar/keluar',$data);
46     $this->load->view('componen_bagian/foot');
47 }
48
49 }

```

D. Penjelasan kodingan model perbaris

1. Mlogin
 - a. Model Class Mlogin
 - b. Function cek_login
2. Msubag_umum
 - a. Model class Msubag_umum
 - b.
3. Mpimpinan
 - a. Model class Mpimpinan
4. Mbagian_bagian
 - a. Model class Mbagian_bagian

