**PERANCANGAN SISTEM INFORMASI DISPOSISI SURAT**

**BERBASIS WEB MENGGUNAKAN TWITTER BOOTSTRAP PHP DAN MYSQL**



**Untuk memenuhi persyaratan**

**Mata Kuliah Berbasis Kompetensi**

**Perancangan Web**

**Program Studi Manajemen Informatika**

**Diajukan oleh:**

**12142498 Maulana Ramdhan**

**12153504 Maulana Nur Ramadhani**

**12154046 Enggar Riska Winaryatin**

**12154012 Kemala Chintya Putri**

**12154250 Novita Triastuti**

**Akademi Manajemen Informatika dan Komputer**

**Bina Sarana Informatika**

**JAKARTA**

**2017**

**HALAMAN PERNYATAAN**

Kami yang bertanda tangan di bawah ini memberikan pernyataan bahwa laporan yang telah kami buat dengan judul “**Perancangangan Sistem Informasi Disposisi Surat Berbasis Web menggunakan Twitter Bootstrap dan MySQL”** telah selesai tepat waktu dan laporan ini adalah hasil dari pemikiran dan diskusi kami selama belajar.

Demikian surat pernyataan kami buat sebagai bukti atas selesainya laporan ini agar dapat menjadi bahan diskusi dalam presentasi dan pembelajaran.

**Ketua**

**Maulana Nur Ramadhani**

**12153504**

**Anggota Anggota**

**Maulana Ramdhan Enggar Riska Winaryatin**

**12142498 12154046**

**Anggota Anggota**

**Kemala Chintya Putri Novita Triastuti**

**12154012 12154250**

**HALAMAN PENGESAHAN**

**PERANCANGAN SISTEM INFORMASI DISPOSISI SURATBERBASIS WEB MENGGUNAKAN TWITTER BOOTSRAP PHP DAN MYSQL**

Telah dipresentasikan di depan Dosen Mata

Kuliah Perancangan Web Kelas 12.5A.37

pada tanggal 20 November 2017

**(Petrus Christo, M.Kom)**

KATA PENGANTAR

Segala puji bagi Allah SWT Pencipta dan Pemelihara alam semesta yang telah memberikan kami kemudahan menyelesaikan makalah ini. Tanpa pertolongan-Nya mungkin penyusun tidak akan sanggup menyelesaikannya dengan baik. Shalawat serta salam semoga terlimpah curahkan kepada baginda tercinta kita yakni Nabibesar Muhammad SAW.

Makalah ini di susun agar pembaca dapat memperluas ilmu tentang “Disposisi Surat”, yang kami sajikan berdasarkan pengamatan dari berbagai sumber baik datang dari kami maupun dari luar. Makalah ini sesungguhnya bukanlah sebuah kerja individual dan akan sulit terlaksana tanpa bantuan banyak pihak yang tak mungkin penyusun sebutkan satu persatu, namun dengan segala kerendahan hati, kami mengucapkan terima kasih kepada:

1. Bapak Petrus Christo, M.Kom, selaku dosen mata kuliah Perancangan Web
2. Teman – teman kelas 12.5A.37
3. Dan seluruh pihak yang terkait baik secara langsung maupun tidak langsung yang tidak dapat disebutkan satu per satu.

Akhirnya, Penulis berharap semoga makalah ini memberikan pengetahuan yang lebih luas kepada pembaca dan menjadi sumbangsih yang bermanfaat bagi dunia Ilmu teknologi dan Komputer di Indonesia, khususnya disiplin keilmuan yang Penulis dalami.

Jakarta, 20 November 2017

Penyusun

DAFTAR ISI

Cover i

**Halaman Pernyataan ii**

**Halaman Pengesahan iii**

**Kata Pengantar iv**

**Daftar Isi v**

**Daftar Tabel vii**

**Daftar Gambar viii**

**Abstrak ix**

**Abstract x**

[BAB I PENDAHULUAN 1](#_Toc498902079)

[1.1 Latar Belakang 1](#_Toc498902080)

[1.2 Rumusan Masalah 2](#_Toc498902081)

[1.3 Batasan Masalah 2](#_Toc498902082)

[1.4 Penjadwalan 3](#_Toc498902083)

[BAB II LANDASAN TEORI 4](#_Toc498902084)

[2.1 Teori Pendukung 4](#_Toc498902085)

[2.1.1 Bahasa Pemrograman 4](#_Toc498902086)

[2.1.2 Permodelan Berbasis Objek (UML) 5](#_Toc498902087)

[2.1.3 Pemrograman Web 11](#_Toc498902088)

[2.2 Perangkat bantu 16](#_Toc498902089)

[2.2.1 XAMPP 16](#_Toc498902090)

[2.2.2 PHP 17](#_Toc498902091)

[2.2.3 MySQL 18](#_Toc498902092)

[2.2.4 Materialize 18](#_Toc498902093)

[2.2.5 Bootstrap 18](#_Toc498902094)

[2.2.6 Adobe Dreamweaver 19](#_Toc498902095)

[BAB III PEMBAHASAN 20](#_Toc498902096)

[3.1 Spesifikasi Rancangan Program 20](#_Toc498902097)

[3.1.1 Rancangan Masukan (*Input*) 20](#_Toc498902098)

[3.1.2 Rancangan Keluaran (*output*) 21](#_Toc498902113)

[3.1.3 Use Case Diagram 23](#_Toc498902150)

[3.1.4 Activity Diagram 24](#_Toc498902151)

[3.1.5 ERD 25](#_Toc498902152)

[3.1.6 LRS 26](#_Toc498902153)

[3.1.7 Class Diagram 26](#_Toc498902154)

[3.1.8 Sequence Diagram 27](#_Toc498902155)

[3.1.9 Struktur Navigasi 28](#_Toc498902156)

[3.2 Spesifikasi Basis Data 29](#_Toc498902157)

[3.3 Rancangan Antar Muka 31](#_Toc498902158)

[3.4 Hasil Pengujian 36](#_Toc498902159)

[3.4.1 Hasil pengujian dengan Desk Checking 36](#_Toc498902160)

[BAB IV PENUTUP 37](#_Toc498902161)

[4.1 Kesimpulan 37](#_Toc498902162)

[4.2 Saran 38](#_Toc498902163)

**Daftar Pustaka 39**

DAFTAR TABEL

[Tabel I.1. Penjadwalan 3](#_Toc498897863)

Tabel II.1 Use Case Diagram 7

Tabel II.2 Simbol-simbol Activity Diagram 8

Tabel II.3 Simbol-simbol Sequence 9

Tabel II.4 Class Diagram 10

Tabel III.1 Tabel Admin 29

Tabel III.2 Tabel Disposisi 29

Tabel III.3 Tabel Surat Masuk 30

Tabel III.4 Tabel Surat Keluar 30

Tabel III.5 Tabel Klarifikasi 31

Tabel III.6 Tabel Sett 31

DAFTAR GAMBAR

1. Gambar II.1 Struktur Navigasi Linier 11

2. Gambar II.2 Struktur Navigasi Hirarki 12

3. Gambar II.3 Struktur Navigasi Non Linier 13

4. Gambar II.4 Stuktur Navigasi Campuran 13

5. Gambar III.1 Use Case Diagram 23

6. Gambar III.2 Activity Diagram 24

7. Gambar III.3 ERD 25

8. Gambar III.4 LRS 26

9. Gambar III.5 Class Diagram 26

10. Gambar III.6 Sequence Diagram 27

11. Gambar III.7 Struktur Navigasi 28

12. Gambar III.8 Rancangan Halaman Home 32

13. Gambar III.9 Rancangan Transaksi Surat Masuk 32

14. Gambar III.10 Rancangan Transaksi Surat Keluar 32

15. Gambar III.11 Rancangan Agenda Surat Masuk 33

16. Gambar III.12 Rancangan Agenda Surat Keluar 34

17. Gambar III.13 Rancangan Galeri File Surat Masuk 34

18. Gambar III.14 Rancangan Galeri File Surat Keluar 35

19. Gambar IV.1 Desk Checking Transaksi Data 36

20. Gambar IV.2 Desk Checking Laporan 36

ABSTRAK

Penulisan tugas ini bertujuan untuk membangun suatu aplikasi yang dapat digunakan untuk proses surat menyurat. Aplikasi ini dirancang untuk memudahkan pekerjaan dalam pengelolaan surat menyurat suatu instansi/perusahaan khususnya dalam bidang akademik. Sistem ini dirancang dengan menggunakan perangkat lunak *Adobe Dreamweaver Cs5* (Program) dan MySQL *(Database).* SMP Taman Harapan membutuhkan sekali adanya suatu program sistem informasi yang memberikan efisiensi terhadap kegiatan surat menyurat bagi para guru dan staff yang berada di naungan yayasan sekolah. Sistem yang ada di SMP Taman Harapan saat ini masih dilakukan secara manual mulai dari adanya penambahan surat masuk yang kurang efektif dan penyimpanan surat yang masih tidak teratur,sehingga memungkinkan pada saat proses surat menyurat berlangsung terjadi kesalahan dalam penerimaan surat masuk,kurang akurat nya laporan yang dibuat dan keterlambatan dalam pencarian data-data yang diperlukan.Dari sistem yang sudah diterapkan terlebih dahulu, yaitu sistem yang manual, maka kini akan lebih akurat karena sistem yang diterapkan adalah sistem yang terkomputerisasi.

**Kata Kunci : Sistem, Informasi, Sistem informasi, perancangan sistem informasi, disposisi surat**

*ABSTRACT*

*Writing this task aims to build an application that can be used for the process of correspondence. This application is designed to facilitate the work in the management of correspondence of an agency / company, especially in the academic field. The system is designed using Adobe Dreamweaver Cs5 (Program) and MySQL (Database) software. Taman Harapan Junior High School is in need of an information system program that provides efficiency to the correspondence activities for teachers and staff who are in the shade of the school foundation. The existing system in Taman Harapan Junior High School is still done manually starting from the addition of incoming letters that are less effective and the storage of the letter is still irregular, making it possible at the time of the correspondence process going wrong error in receipt of incoming mail, its less accurate report made and delays in searching the necessary data. From the system that has been applied in advance, the manual system, it will now be more accurate because the system applied is a computerized system.*

***Keywords:*** *System, Information, Information system, information system design, letter disposition*

# 

PENDAHULUAN

## Latar Belakang

Surat adalah suatu alat penyampaian berita secara tertulis yang berisi pemberitahuan, pernyataan permintaan, dan lain – lain kepada pihak lain. Sedangkan surat menyurat adalah kegitan pengendalian arus berita, baik tertulis maupun lisan yang timbul karna adanya pencatatan, laporan perencanaan atau suatu program. Kegiatan surat menyurat merupakan suatu media komunikasi yang sangat penting disebuah instansi, perusahan maupun organisasi, baik untuk berkomunikasi dengan pihak- pihak diluar organisasi (eksternal) maupun didalam organisasi (internal). Setiap hal yang berkaitan dengan kegiatan organisasi yang sifatnya resmi ataupun tidak resmi selalu dikomunikasikan dalam bentuk surat baik surat edaran, surat keputusan, surat pernyataan permintaan, surat perintah tugas, surat pendelegasian wewenang, surat undangan rapat, surat permohonan dan berbagai jenis surat lainnya yang berkaitan dengan kegiatan administrasi organisasi tersebut.

Salah satu kegiatan administrasi surat menyurat adalah disposisi surat. Pengertian disposisi surat adalah petunjuk singkat tentang tindak lanjut terhadap suatu urusan atau surat masuk. Disposisi dibuat oleh pimpinan untuk staf atau bawahannya sesuai dengan kewenangannya. Tujuan pembuatan disposisi itu sendiri adalah agar staf dapat menindaklanjuti atau menyelesaikan suatu urusan yang telah diberikan oleh pimpinan. Tindak lanjut tersebut dapat berupa surat balasan, tindakan – tindakan lain dalam rangka menyelesaikan urusan tersebut. Secara umum, permasalahan yang dihadapi oleh berbagai perusahaan saat ini dalam pelaksanaan disposisi suratadalah waktu. Dalam hal waktu, kegiatan disposisi surat membutuhkan beberapa tahapan proses untuk sampai kepada penerima disposisi surat. Proses tersebut mengharuskan adanya seseorang yang mengantar surat ke beberapa bagian perusahaan dan akan memakan waktu lebih banyak. Hal ini membuat pelaksanaan disposisi surat dalam perusahaan menjadi tidak efektif dan efisien.

Maka dari itu, kami berusaha untuk membuat sistem disposisi surat yang memanfaatkan teknologi berbasis *web*. Sistem ini akan membantu perusahaan khususnya pimpinan dan staf perusahaan dalam melaksanakan kegiatan disposisi surat dengan efektif. Semua kegiatan disposisi surat perusahaan sudah diatur oleh sistem *web* agar lebih mudah dan hemat waktu.Berdasarkan uraian diatas, maka penulis bermaksud membahasnya lebih lanjut dalam laporan tugas ini yang berjudul **“Perancangan Sistem Informasi Disposisi Surat Berbasis Web Dengan Menggunakan Twitter Bootstrap PHP Dan MySQL.”**

## Rumusan Masalah

Berdasarkan latar belakang yang sudah dijelaskan diatas, maka penulis merumuskan beberapa masalah sebagai berikut :

1. Bagaimana merancang dan membangun suatu sistem dengan baik dalam mendukung pelaksanaan disposisi surat antar bagian perusahaan.
2. Bagaimana membuat format surat yang sesuai dengan ketentuan.

## Batasan Masalah

Supaya penelitian dalam makalah ini lebih terarah dan memudahkan dalam pembahasan, maka perlu adanya pembatasan masalah, yaitu :

1. Perancangan sistem yang dibahas adalah sistem informasi yang berhubungan dengan disposisi surat.
2. Pembangunan sistem ini menggunakan teknologi berbasis web, menggunakan bahasa pemrograman PHP dan desain *web* CMS *(Content Management System)* Materialize.
3. *Server* menggunakan *Apache*.

## Penjadwalan

Tabel I.1. Penjadwalan

|  |  |  |
| --- | --- | --- |
| No. | Tanggal | Kegiatan |
|  | 17 Oktober 2017 | Membuat Rancangan Program |
|  | 17 Oktober 2017 | Menganalisa Kebutuhan sistem |
|  | 21 Oktober 2017 | Mendesain perangkat lunak atau sistem |
|  | 28-29 Oktober 2017 | Implementasi perangkat lunak |
|  | 30 Oktober 2017 | Pengujian sistem |
|  | 4 November 2017 | Membuat makalah |
|  | 18 November 2017 | Observasi data pengertian disposisi surat |
|  | 19 November 2017 | Membuat *Power Point* |
|  | 20 November 2017 | Presentasi hasil penelitian |

# 

LANDASAN TEORI

## Teori Pendukung

### Bahasa Pemrograman

1. HTML

Menurut Alexander F. K. Sibero (2011:19) “ *Hypertext Markup Language* atau HTML adalah bahasa yang digunakan pada dokumen *web* sebagai bahasa untuk pertukaran dokumen *web*”. Dokumen HTML terdiri dari komponen yaitu *tag*, elemen, dan atribut.

Menurut Alexander F. K. Sibero (2011:19) “ Tag adalah tanda awal < dan tanda akhir > yang digunakan sebagai pengapit suatu elemen. Tag pada elemen pembuka diawali dengan < dan diakhiri dengan tanda >. Sedangkan untuk elemen penutup diawalin dengan tanda < dan / kemudian diakhiri dengan tanda >.”

1. CSS *(Cascade Style Sheet)*

Menurut Jasakom (2012:107) “ *Cascading Style Sheet* (CSS) merupakan salah satu bahasa pemrograman *web* untuk mengendalikan beberapa komponen dalam sebuah *web* sehingga akan lebih terstruktur dan seragam”. Sama halnya *style* dalam aplikasi pengolahan kata seperti Microsoft Word yang dapat mengatur beberapa *style* misalnya *heading, subbab, bodytext, footer, images,* dan *style* lainnya untuk dapat digunakan bersama – sama dalam beberapa berkas *(file)*. Pada umumnya CSS dipakai untuk memformat tampilan halaman *web* yang dibuat dengan bahasa HTML dan XHTML. Nama CSS sendiri didapat dari fakta bahwa setiap deklarasi *style* yang berbeda dapat diletakkan secara berurutan, yang kemudian membentuk hubungan ayah-anak (*parent-child*) pada setiap *style*. CSS merupakan sebuah teknologi internet yang direkomendasikan oleh *World Wide Web Consortium* atau W3C pada tahun 1996. Setelah CSS distandarisasikan, Internet Explorer dan Netscape melepas *browser* terbaru mereka yang telah sesuai atau paling tidak hampir mendekati standar CSS.

1. JavaScript

Menurut Alexander F. K. Siberto (2011:151) “*Javascript* adalah suatu bahasa pemograman yang dikembangkan untuk dapat berjalan pada *web browser*”. Pada awalnya *Javascript* dikembangkan pada *web browser Netscape* oleh Brendan Eich dengan nama *Mocha,* kemudian berubah menjadi *Live-Script* dan yang akhirnya sampai sekarang ini menjadi Javascript. Javascript yang pada awalnya dikembangkan pada *web browser Netscape* kemudian menjadi populer dikalangan pengguna dan pengembangan *web.* Perkembangan Javascript menarik perhatian produsen *web browser* seperti Microsoft yang kemudian mengembangkan bahasa skripnya dengan nama Jscript. Dikarenakan masing-masing perkembangan memiliki aturan dan standar yang berbeda serta akibat dari persaingan pasar, maka pada tahun 1996 Netscape menyerahkan Javascript pada ECMA (*European Computer Manufacture Association*) Intenasional yaitu suatu organisasi internasional nirlaba untuk dipertimbangkan sebagai standar industri, yang kemudian oleh ECMA Internasional mengeluarkan Javascript standar dengan nama ECMA Script.

1. jQuery

Menurut Munigmini (2012:2) “jQuery adalah *Javascript Library* atau kumpulan kode fungsi Javascript siap pakai, sehingga mempermudah kita untuk membuat kode Javascript. Dalam kesimpulannya jQuery menyederhanakan kode Javascript”.

### Permodelan Berbasis Objek (UML)

*Unified Modeling Language* (UML) adalah notasi yang lengkap untuk membuat visualisasi model suatu sistem. Sistem berisi informasi dan fungsi, tetapi secara normal digunakan untuk pemodelan sistem komputer. Didalam pemodelan objek guna menyajikan sistem yang berorientasi objek pada orang lain, akan sangat sulit dilakukan jika pemodelan tersebut dilakukan dalam bentuk bahasa pemrograman. Kesulitan yang muncul adalah timbulnya ketidakjelasan dan salah interpretasi di dalam pembacaan kode pemrograman untuk pemodelan objek tersebut.

Menurut Widodo (2011:6), “ UML (*Unified Modeling Language*) adalah bahasa pemodelan standar yang memiliki sintak dan semantik”. Pemodelan sesungguhnya digunakan untuk penyederhanaan masalah-masalah yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami.

Untuk mendapatkan banyak pandangan terhadap sistem informasi yang akan dibangun, UML menyediakan beberapa diagram visual yang menunjukkan berbagai aspek dalam sistem. Ada beberapa diagram yang disediakan dalam UML antara lain :

1. *Use Case* Diagram

*Use Case* Diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem dan bukan “bagaimana”. Sebuah *use case* mempresentasikan interaksi antar actor dengan sistem. Seorang / sebuah actor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu. *Use Case Diagram* dapat sangat membantu bila kita sedang menyusun*requirement* sebuah sistem, mengkomunikasikan rancangan dengan klien, dan merancang *test case* untuk semua *feature* yang ada pada sistem. Sebuah *use case* dapat meng*include*fungsionalitas *use case* lain sebagian bagian dari proses dalam dirinya. Secara umum diasumsikan bahwa *use case* yang di *include* akan dipanggil setiap kali *use case* yang di *include* dieksekusikan secara normal. Sebuah *use case* dapat di-include oleh lebih dari satu *use case* lain, sehingga duplikasi *fungsionalitas* dapat dihindari dengan cara menarik keluar *fungsionalitas* yang *common.*

**Tabel II.1 *Use Case* Diagram**

|  |  |
| --- | --- |
| Simbol | Keterangan |
|  | Aktor : Seseorang atau sesuatu yang berinteraksi dengan sistem yang sedang dikembangkan. |
|  | *Use case* : Peringkat tertinggi dari fungsionalitas yang dimiliki sistem. |
|  | *Assosiation* : adalah relasi antara aktor dan *use case.* |
|  | Generalisasi : Untuk memperlihatkan struktur pewaris yang terjadi |

1. *Activity* Diagram

*Activity* diagram menggambarkan berbagai aliran, digunakan untuk mendeskripsikan aktifitas yang dibentuk dalam operasi sehingga dapat juga digunakan untuk aktifitas lainnya seperti *use case* dan interaksi. *Activity* diagram berupa *flow chart* yang digunakan untuk memperlihatkan aliran kerja dari sistem.

Tabel II.2 Simbol – simbol Acivity Diagram

|  |  |
| --- | --- |
| Simbol | Keterangan |
|  | Titik awal |
|  | Titik Akhir |
|  | *Activity* |
|  | Pilihan untuk pengambilan keputusan |
|  | *Fork* : Untuk menunjukkan kegiatan yang dilakukan secara pararel |

1. *Sequence* Diagram

*Sequence* diagram menggambarkan kolaborasi dinamis antara antara sejumlah objek dan untuk menunjukkan rangkaian pesan yang dikirim antar objek juga interaksi antar objek, sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem. *Sequence* diagram menjelaskan interaksi objek yang disusun berdasarkan urutan waktu. Secara mudahnya *sequence* diagram adalah gambaran tahap demi tahap yang seharusnya dilakukan untuk menghasilkan sesuatu sesuai dengan *use case* diagram. *Sequence* diagram berhubungan erat dengan use case diagram dimana 1 *use case* akan menjadi 1 *sequence* diagram. Dalam *sequence* diagram terdapat simbol *Actor* untuk menggambarkan pengguna sistem dan *Lifeline* untuk menggambarkan kelas objek.

Tabel . II.3Simbol – simbol *Sequence* diagram

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Gambar | Nama | Keterangan |
| 1 |  | *Lifeline* | Objek *entity,* antarmuka yang saling berinteraksi. |
| 2 |  | *Message* | Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi. |
| 3 |  | *Message* | Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi |

1. *Class* Diagram

*Class* diagram menggambarkan struktur dan deskripsi *class*, *package,* dan objek beserta hubungan satu sama lain seperti *containment,* pewaris, asosiasi dan lain-lain. *Class* diagram berfungsi menjelaskan tipe dari objek sistem dan hubungannya dengan objek yang lain. Objek adalah nilai tertentu dari setiap atribut kelas *entity. Class* adalah sebuah spesifikasi yang diinstansikan akan menghasilkan sebuah objek dan merupakan inti dari pengembangan berorientasi objek. *Class* menggambarkan keadaan (*attribute/property*) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metode/fungsi).

Tabel II.4 *Class Diagram*

|  |  |  |  |
| --- | --- | --- | --- |
| No | Gambar | Nama | Keterangan |
| 1 |  | *Generalitation* | Hubungan dimana objek anak (*descendent*) berbagi perilaku dan struktur data dari objek yang ada diatasnya objek induk (*ancestor*). |
| 2 |  | *Nary Association* | Upaya untuk menghindari asosiasi dengan lebih dari 2 objek. |
| 3 |  | *Class* | Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama. |
| 4 |  | *Collaboration* | Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor. |
| 5 |  | *Realization* | Operasi yang benar-benar dilakukan oleh suatu objek. |
| 6 |  | *Dependency* | Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (*independent*) akan mempengaruhi elemen yang bergantung padanya elemen yang mandiri. |
| 7 |  | *Association* | Apa yang menghubungkan antara objek satu dengan objek lainnya. |

### Pemrograman Web

1. Struktur Navigasi

Menurut Mahendra (2011:1), mendefinisikan bahwa “Struktur Navigasi adalah alur yang digunakan dalam aplikasi yang dibuat. Sebelum menusun aplikasi multimedia kedalam sebuah software, kita harus menentukan terlebih dahulu alur apa yang akan digunakan dalam aplikasi yang dibuat”. Struktur navigasi terbagi menjadi 4 macam, yaitu :

1. Struktur Navigasi Linier

Struktur navigasi linier hanya mempunyai rangkaian cerita berurut, yang menampilkan satu demi satu tampilan layar secara berurut menurut urutannya. Tampilan yang dapat ditampilkan pada struktur jenis ini adalah satu halaman sebelumnya atau satu halaman sesudahnya, tidak dapat dua halaman sebelumnya atau halaman sesudahnya.

Sumber : Prihatna (2011:269)

Gambar II.1

Struktur Navigasi Linier

1. Struktur Navigasi Hirarki

Struktur navigasi hirarki disebut struktur bercabang, merupakan suatu struktur yang mengandalkan percabangan untuk menampilkan data berdasarakan kriteria tertentu. Tampilan pada menu pertama akan disebut sebagai *Master Page* (halaman utama pertama), halaman utama ini mempunyai halaman percabangan yang disebut *Slave Page* (halaman pendukung). Jika salah satu halaman pendukung dipilih atau diaktifkan, maka tampilan tersebut akan bernama *Master Page* (halaman kedua), dan seterusnya. Pada struktur navigasi ini tidak diperkenankan adanya tampilan secara linier.

Sumber : Binanto (2011:269)

Gambar II.2

Struktur Navigasi Hirarki

1. Struktur Navigasi Non-Linier

Struktur Navigasi non-linier atau struktur tidak berurut merupakan pengembangan dari struktur navigasi-linear. Pada struktur ini diperkenankan membuat navigasi bercabang. Percabangan yang dibuat pada struktur non-linear ini berbeda dengan percabangan pada struktur hirarki, karena pada percabangan non-linear ini walaupun terdapat percabangan, tetapi tiap-tiap tampilan mempunyai kedudukan yang sama yaitu tidak ada *Master Page* dan *Slave Page*.

Sumber : Binanto (2011:270)

Gambar II.3

Struktur Navigas Non-linierr

1. Struktur Navigasi Campuran

Struktur Navigasi campuran merupakan gabungan dari ketiga struktur sebelumnya yaitu linear, non-linear, dan hirarki. Struktur navigasi ini juga biasa disebut dengan struktur navigasi bebas. Struktur navigasi ini banyak digunakan dalam pembuatan website sehingga dapat memberikan ke-interaksian yang lebih tinggi.

Sumber : Binanto (2011:270)

Gambar II.4 Struktur Navigasi Campuran

1. ERD (*Entity Relationship Diagram*)

Pemodelan awal basis data yang paling banyak digunakan adalah menggunakan *Entity Relationship Diagram* (ERD). Menurut Rosa A. S dan M. Shalahudin (2014 : 53) “ERD adalah bentuk paling awal dalam melakukan perancangan basis data relasional”. ERD dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data relasional. Sehingga jika penyimpanan basis data tidak perlu menggunakan ERD. ERD memiliki beberapa aliran notasi seperti notasi Chen (dikembangkan oleh Peter Chen), Barker (Dikembangkan oleh Richard Barker, Ian Palmer, Harry Ellis), notasi Crow’s Foot, dan beberapa notasi lain. Namun yang banyak digunakan adalah notasi dari Chen. ERD memiliki simbol – simbol yang dalam penggunaannya menggunakan notasi Chen, yaitu:

1. Entitas / *Entity*

Entitas merupakan data inti yang akan disimpan, bakal tabel pada data, benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer, penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel.

1. Atribut

Atribut adalah Field atau kolom data yang butuh disimpan dalam suatu entitas.

1. Atribut Kunci Primer

Field atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses record yang diinginkan, biasanya berupa id, kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama).

1. Atribut multinilai / multivalue

Field atau kolom dalam yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu.

1. Relasi

Relasi yang menghubungkan antar entitas, biasanya diawali dengan kata kerja.

1. Asosiasi / association

Penghubung antara relasi dan entitas relasi dari entitas dimana dikedua ujungnya memiliki multiplicity kemungkinan jumlah pemakaian.

1. *Logical Relational Structure* (LRS)

Menurut Hasugian dan Shidiq (2012:608) memberikan batasan bahwa LRS adalah “sebuah model sistem yang digambarkan dengan sebuah diagram-ER akan mengikuti pola atau aturan pemodelan tertentu dalam kaitannya dengan konvensi ke LRS”. Perubahan yang terjadi yaitu mengikuti aturan-aturan sebagai berikut :

1. Setiap entitas akan diubah kebentuk kotak.
2. Sebuah atribut relasi disatukan dalam sebuah kotak bersama entitas jika hubungan yang terjadi pada *diagram*-ER 1:M (relasi bersatu dengan *cardinality* M) atau tingkat hubungan 1:1 (relasi bersatu dengan *cardinality* yang paling membutuhkan referensi).
3. Sebuah relasi dipisah dalam sebuah kotak tersendiri (menjadi entitas baru) jika tingkat hubungannya M:M (*many to many*) dan memiliki *foreign key* sebagai *primary key* yang diambil dari kedua entitas yang sebelumnya saling berhubungan.
4. Pengujian Web (*Black Box Testing*)

Menurut Agustiar Budiman (2012:4), berpendapat bahwa “Pengujian black box merupakan metode perancangan data uji yang didasarkan pada spesifikasi perangkat lunak. Data uji dibangkitkan, dieksekusi pada perangkat lunak dan kemudian keluaran dari perangkat lunak diuji apakah telah sesuai dengan yang diharapkan.” Pengujian *Black Box* dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai dengan spesifikasi yang dibutuhkan. Pengujian *Black Box* berusaha menemukan kesalahan dalam kategori sebagai berikut :

1. Fungsi-fungsi yang tidak benar atau hilang
2. Kesalahan *Interface*
3. Kesalahan dalam struktur data atau akses *database eksternal*
4. Kesalah kinerja
5. Inisialisasi dan kesalahan terminasi

## Perangkat bantu

### XAMPP

Menurut Puspitasari (2011:1), berpendapat bahwa “XAMPP adalah sebuah perangkat lunak yang didalamnya sudah tersedia database server MySQL, webserver Apache dan mendukung program bahasa php (php4 dan php5). Xampp merupakan software gratis yang mudah digunakan gratis dan mendukung instalasi di linux dan windows.”

### PHP

1. Definisi PHP

Menurut Agus Saputra (2012:2), berpendapat bahwa “PHP memiliki kepanjangan PHP Hypertext Prepocessor merupakan suatu bahasa pemrograman yang difungsikan untuk membangun suatu website dinamis. PHP menyatu dengan kode HTML, maksudnya adalah beda kondisi, HTML digunakan sebagai pembangun atau pondasi dari kerangka layout web, sedangkan PHP difungsikan sebagai prosesnya, sehingga dengan adanya PHP tersebut, sebuah web akan mudah dimaintenance”.

1. Sejarah PHP

Menurut Sibero (2011:49), berpendapat “Padat tahun 1994 seorang programmer bernama Rasmus Lerdorf awalnya membuat sebuah halaman website pribadi, tujuannya adalah untuk mempertahankan halaman website pribadi tersebut sekaligus membangun halaman web yang dinamis. PHP pada awalnya diperkenalkan sebagai singkatan dari PHP Hypertext Preprocessor pertama ditulis menggunakan bahasa Perl (Perl Script), kemudian ditulis ulang menggunakan bahasa pemrograman C CGI-BIN (Common Gateway Interface-Binary) yang ditujukan untuk mengembangkan halaman website yang mendukung formulir dan penyimpanan data. Pada tahun 1995 PHP Tool 1.0 dirilis untuk umum, kemudian pengembangannya dilanjutkan oleh Andi Gutmans dan Zeev Suraski”. Perusahaan bernama Zend kemudian melanjutkan pengembangan PHP dan merilis PHP versi terakhir pada saat ini. Aplikasi bahasa PHP dapat dipergunakan untuk :

1. PHP digunakan sebagai landasan operasi pada pemrograman jaringan berbasis web.
2. PHP digunakan juga untuk pemrograman database.
3. PHP digunakan untuk membuat aplikasi web.

### MySQL

Menurut Alexander F. K. Sibero (2011:97), berpendapat bahwa “MySQL atau di baca “my sekuel” adalah suatu RDBMS (Relational Data-Base Management System) yaitu aplikasi sistem yang menjalankan fungsi pengolahan data”. Structured Query Language (SQL) adalah bahasa standar yang digunakan untuk mengakses database relasional. Bahasa ini mendukung perintah untuk membuat database, menciptakan tabel, manambahkan data, merubah data, menghapus data, mengambil data dan bahkan mengatur wewenangan mengakses data.

### Materialize

Materialize CSS adalah sebuah framework css yang dibuat dan didesain oleh Google, Material Desain adalah sebuah bahasa desain yang mengkombinasikan prinsip klasik dari desain sukses dengan inovasi dan teknologi. Tujuan Google adalah mengembangkan sebuah sistem desain yang dapat mengizinkan menggabungkan pengalaman pengguna dari seluruh produk mereka pada platform manapun.

### Bootstrap

*Bootstap* merupakan *framework* ataupun *tools* untuk memudahkan aplikasi *web* ataupun situs *web responsive* secara cepat, mudah dan gratis hal ini berdasarkan kutipan Husein Alatas (2013: 2). *Bootsrap* sendiri terdiri dari CSS dan HTML untuk menghasilkan *grid, layout, typography, table, form, navigation* dan lain-lain. Di dalam *bootsrap* juga sudah terdapat *jquery* plugin untuk menghasilkan komponen UI yang cantik seperti *transitions, modal, dropdown, scrollspy, tooltip, tab, popover, alert, button, carousel* dan lain-lain.

Dengan bantuan bootstrap kita dapat membuat responsive website dengan mudah dan dapat berjalan sempurna pada browser-browser populer seperti *chrome, firefox, safari*, dan opera serta *internet explorer*.

*Bootsrap* sendiri diciptakan oleh dua orang programmer twitter, yaitu Mark Otto dan Jacob Thorton pada tahun 2011. Pada saat itu programmer twitter menggunakan berbagai macam tool dan library yang mereka kenal dan suka untuk melaksanakan pekerjaan mereka, sehingga tidak ada standarisasi dan akibatnya sulit untuk dikelola sehingga Mark Otto dan Jacob Thorton tergerak untuk menciptakan satu tool atau *framework* yang dapat digunakan bersama dilingkungan internal twitter. *Bootsrap* sendiri diciptakan oleh dua orang programmer twitter, yaitu Mark Otto dan Jacob Thorton pada tahun 2011. Pada saat itu programmer twitter menggunakan berbagai macam tool dan library yang mereka kenal dan suka untuk melaksanakan pekerjaan mereka, sehingga tidak ada standarisasi dan akibatnya sulit untuk dikelola sehingga Mark Otto dan Jacob Thorton tergerak untuk menciptakan satu *tool* atau *framework* yang dapat digunakan bersama dilingkungan internal twitter.

### Adobe Dreamweaver

Menurut Madcoms (2013:2) “Adobe dreamweaver merupakan aplikasi yang digunakan sebagai HTML editor professional untuk mendesain web secara visual”. Aplikasi ini biasa dikenal dengan istilah WYSIWYG (*What You See Is What You Get*). Dengan kemampuan fasilitas yang optimal dalam jendela design akan memberikan kemudahan untuk mendesain web meskipun untuk para web designer pemula sekalipun. Kemampuan Adobe Dreamweaver untuk berinteraksi dengan beberapa bahasa pemrograman seperti PHP, ASP, JavaScript, dan yang lainnya juga memberikan fasilitas maksimal kepada desainer web dengan menyertakan bahasa pemrograman.

# 

PEMBAHASAN

## Spesifikasi Rancangan Program

Spesifikasi rancangan program yang penulis gunakan dalam pembuatan perancangan web di makalah ini terdiri dari :

### Rancangan Masukan (*Input*)

Menguraikan secara rinci setiap rancangan dokumen masukan yang digunakan dalam sistem dengan uraian parameter-parameter dokumen tersebut dengan parameter-parameter dokumennya sebagai berikut:

1. Nama Dokumen : Surat Masuk

Fungsi : Untuk menambah dan melihat surat masuk

Sumber : Pegawai

Tujuan : Admin

Media : Kertas

Jumlah : 1 Lembar

Frekuensi : Setiap terjadi penambahan surat masuk

1. Nama Dokumen : Surat Keluar

Fungsi : Untuk membuat surat Keluar

Sumber : Admin

Tujuan : Pihak kedua

Media : Kertas

Jumlah : 1 Lembar

Frekuensi : Setiap terjadi pembuatan surat keluar

### Rancangan Keluaran (*output*)

Menguraikan Rancangan DiagramMenguraikan secara rinci setiap rancangan dokumen keluaran yang digunakan dalam system dengan uraian parameter-parameter dokumen tersebut dengan parameter-parameter dokumennya sebagai berikut :

1. Nama Dokumen : Surat Masuk

Fungsi : Sebagai bukti balasan disposisi

Sumber : Admin

Tujuan :Pihak Kedua

Media : Kertas

Jumlah : 1 Lembar

Frekuensi : Setiap terjadi balasan disposisi

1. Nama Dokumen : Cetak Agenda Surat Masuk

Fungsi : Sebagai bukti laporan surat masuk

Sumber : Admin

Tujuan : Kepala sekolah

Media : Kertas

Jumlah : 1 Lembar

Frekuensi : Setiap terjadi penambahan surat masuk

1. Nama Dokumen : Cetak Agenda Surat Keluar

Fungsi : Sebagai bukti laporan surat keluar

Sumber : Admin

Tujuan : Kepala Sekolah

Media : Kertas

Jumlah : 1 Lembar

Frekuensi : Setiap terjadi penambahan surat keluar

1. Nama Dokumen : Galeri File Surat Masuk

Fungsi : Sebagai bukti laporan file surat masuk

Sumber : Admin

Tujuan : Kepala Sekolah

Media : Monitor

Jumlah : -

Frekuensi : Setiap ada file galeri surat masuk yang diupload

1. Nama Dokumen : Galeri File Surat Keluar

Fungsi : Sebagai bukti laporan file surat keluar

Sumber : Admin

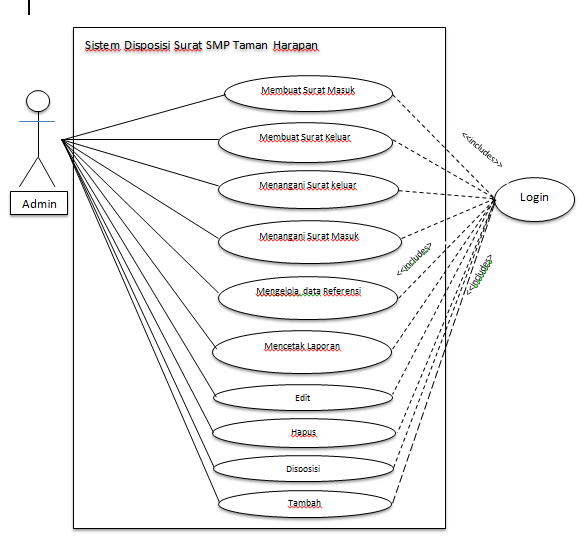
Tujuan : Kepala Sekolah

Media : Monitor

Jumlah : -

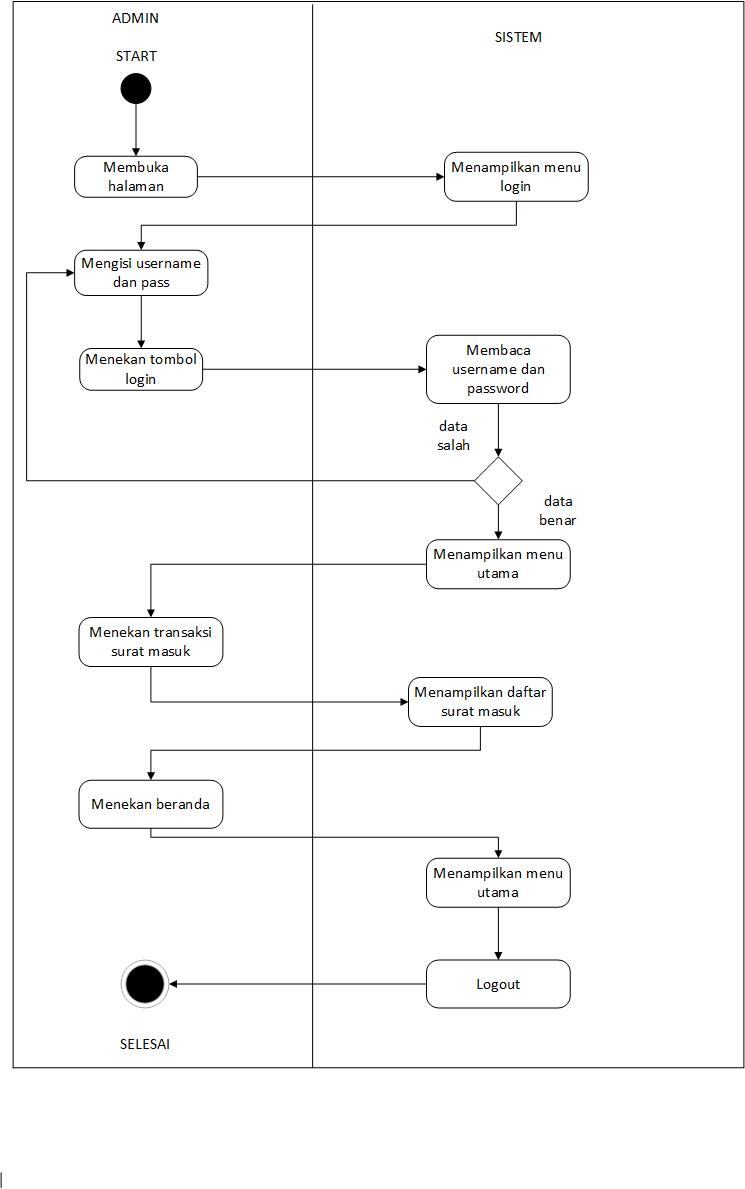
Frekuensi : Setiap ada file galeri surat keluar yang diupload

### Use Case Diagram



Gambar III.1 Use Case Diagram

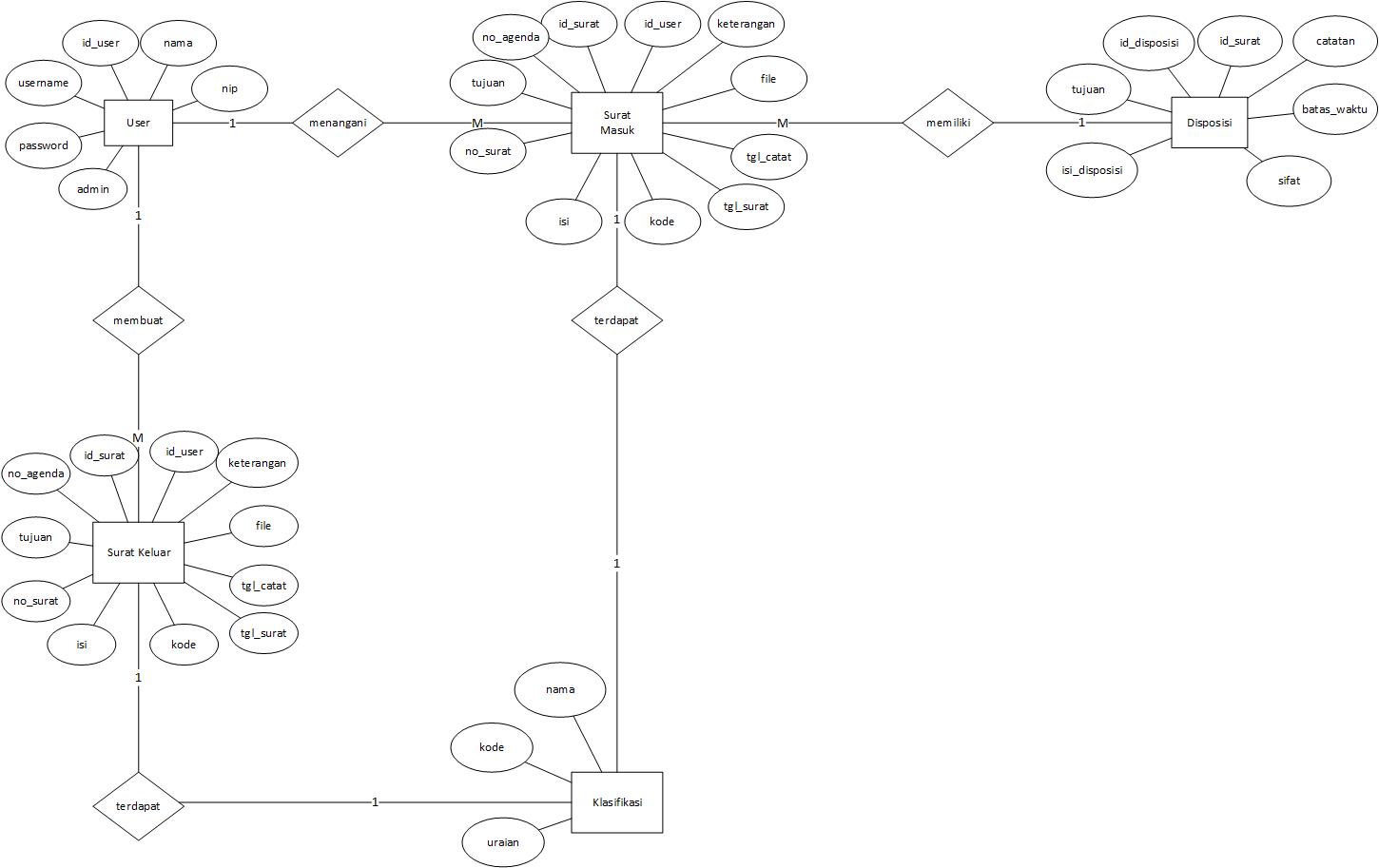
### Activity Diagram



Gambar III.2 Activity Diagram

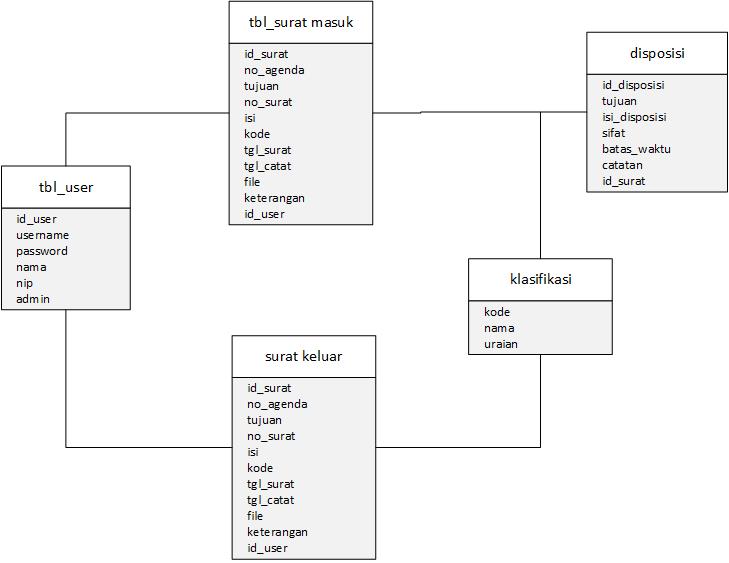
### ERD

*Entity Relationship Diagram* digunakan untuk pemodelan data pada perancangan data logika. Adapun daftar seluruh entitas, relasi dan atribut yang terdapat dalam sistem aplikasi ini yang telah dianalisis dapat dilihat lebih jelas pada gambar.



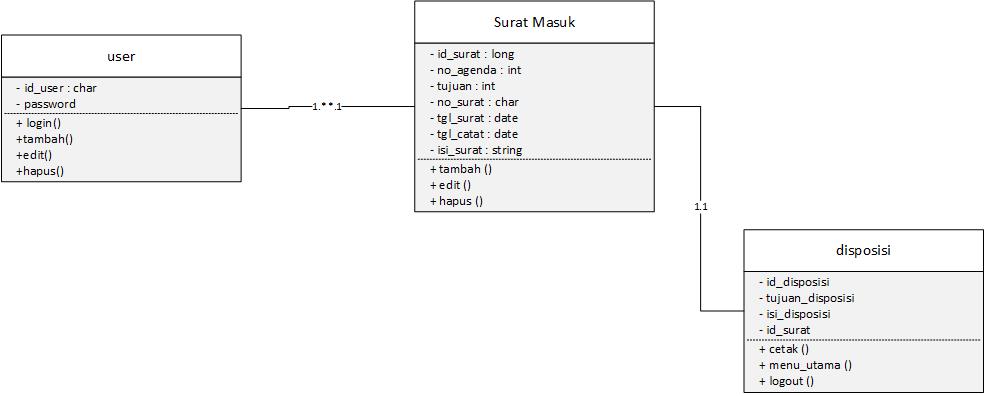
Gambar III.3 ERD

### LRS



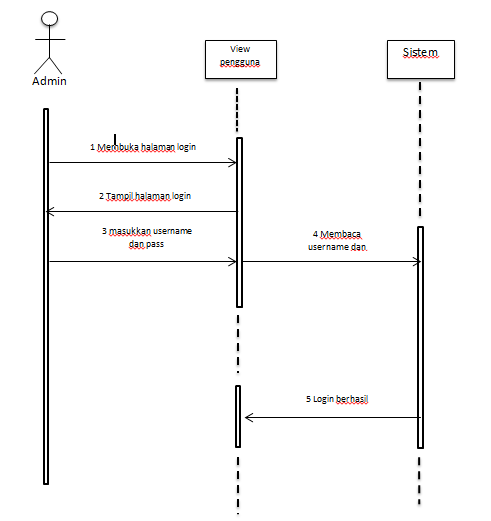
Gambar III.4 LRS

### Class Diagram



Gambar III.5 Class Diagram

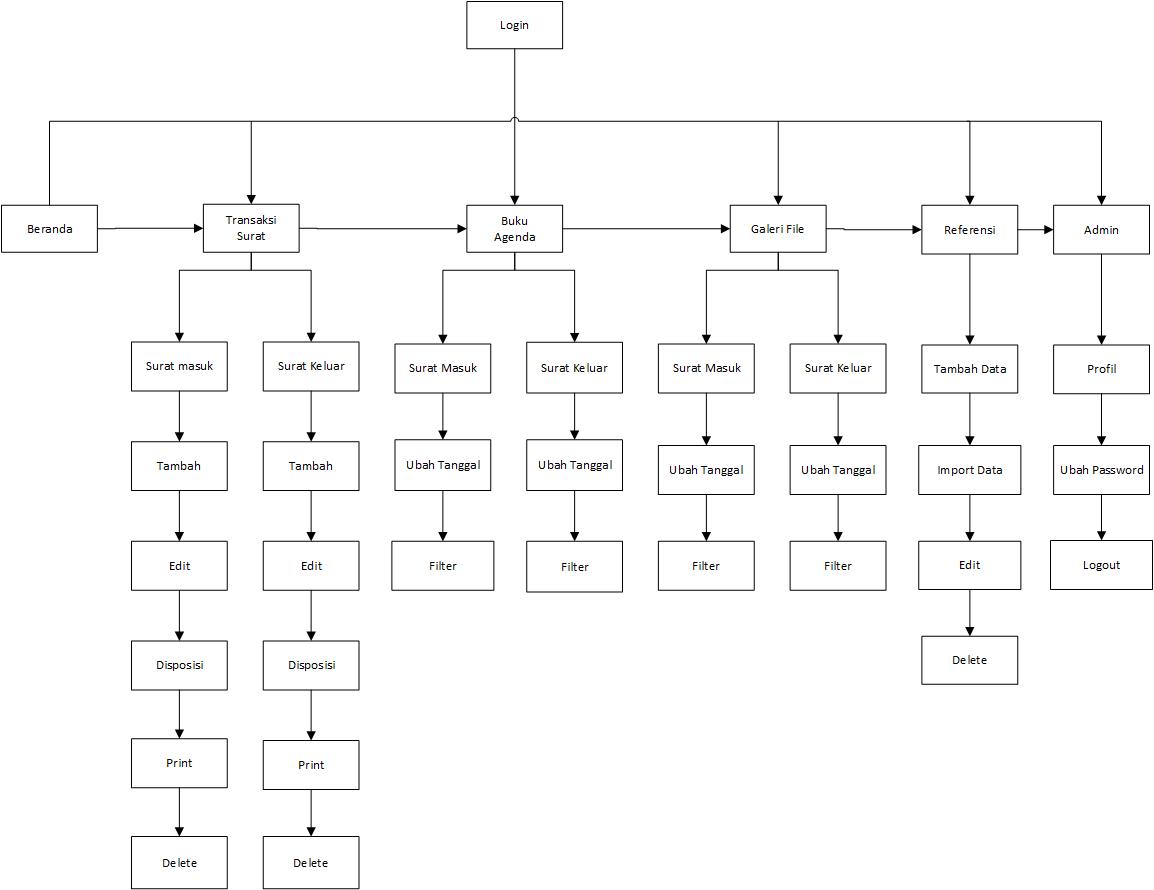
### Sequence Diagram



Gambar III.6 Sequence Diagram

### Struktur Navigasi

Dalam pembahasan struktur navigasi dijelaskan cara kerja web agar lebih mudah memahaminya. Serta melihat bagaimana prosedur web dijalankan sesuai dengan ketentuannya. Adapun penjelasan mengenai struktur navigasi tersebut sebagai berikut:



Gambar III.7 Struktur Navigasi

## Spesifikasi Basis Data

Spesifikasi file yang digunakan dalam perancangan program disposisi surat ini terdiri satu database dan enam tabel, yaitu :

1. Table Admin

Tabel III.1 Tabel Admin

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data(panjang)** | **Null** | **Keterangan** |
| id\_user | tinyint(2) | Yes | **Primary Key** |
| username | varchar(30) | Yes |  |
| password | varchar(35) | Yes |  |
| nama | varchar(50) | Yes |  |
| nip | varchar(25) | Yes |  |
| admin | tinyint(1) | Yes |  |

1. Tabel Disposisi

Tabel III.2 Tabel Disposisi

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data(panjang)** | **Null** | **Keterangan** |
| id\_disposisi | int(10) | Yes | **Primary Key** |
| tujuan | varchar(250) | Yes |  |
| isi\_disposisi | mediumtext | Yes |  |
| sifat | varchar(100) | Yes |  |
| batas\_waktu | date | Yes |  |
| catatan | varchar(250) | Yes |  |
| id\_surat | int(10) | Yes |  |
| id\_user | tinyint(2) | Yes |  |

1. Tabel Surat Masuk

Tabel III.3 Tabel Surat Masuk

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data(panjang)** | **Null** | **Keterangan** |
| id\_surat | int(5) | Yes | **Primary Key** |
| no\_agenda | int(5) | Yes |  |
| no\_surat | varchar(25) | Yes |  |
| asal\_surat | varchar(25) | Yes |  |
| isi | mediumtext | Yes |  |
| kode | varchar(30) | Yes |  |
| indeks | varchar(30) | Yes |  |
| tgl\_surat | Date | Yes |  |
| tgl\_diterima | Date | Yes |  |
| file | varchar(25) | Yes |  |
| keterangan | text | Yes |  |
| id\_user | tinyint(2) | Yes |  |

1. Tabel Surat Keluar

Table III.4 Tabel Surat Keluar

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data(panjang)** | **Null** | **Keterangan** |
| id\_surat | int(5) | Yes | **Primary Key** |
| no\_agenda | int(5) | Yes |  |
| tujuan | varchar(25) | Yes |  |
| no\_surat | varchar(25) | Yes |  |
| isi | Mediumtext | Yes |  |
| kode | varchar(3) | Yes |  |
| tgl\_surat | Date | Yes |  |
| tgl\_catat | Date | Yes |  |
| file | varchar(25) | Yes |  |
| keterangan | text | Yes |  |
| id\_user | tinyint(2) | Yes |  |

1. Tabel Klasifikasi

Tabel III.5 Tabel Klarifikasi

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data(panjang)** | **Null** | **Keterangan** |
| id\_klasifikasi | int(5) | Yes | **Primary Key** |
| Kode | varchar(30) | Yes |  |
| Nama | varchar(250) | Yes |  |
| Uraian | Mediumtext | Yes |  |
| id\_user | tinyint(2) | Yes |  |

1. Tabel Sett

Tabel III.6 Tabel Sett

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Tipe Data(panjang)** | **Null** | **Keterangan** |
| id\_sett | tinyint(1) | Yes | **Primary Key** |
| surat\_masuk | tinyint(2) | Yes |  |
| surat\_keluar | tinyint(2) | Yes |  |
| Referensi | tinyint(2) | Yes |  |
| id\_user | tinyint(2) | Yes |  |

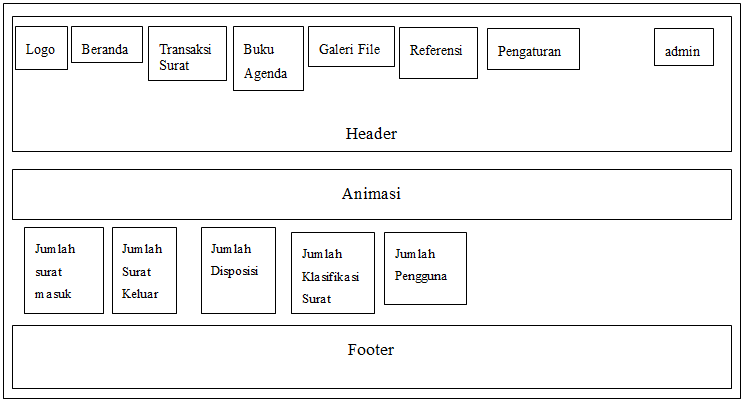
## Rancangan Antar Muka

Pada rancangan antar muka (*interface*) ini penulis akan menggambarkan semua rancangan tampilan yang ada pada website ini :

1. Halaman Utama / Beranda

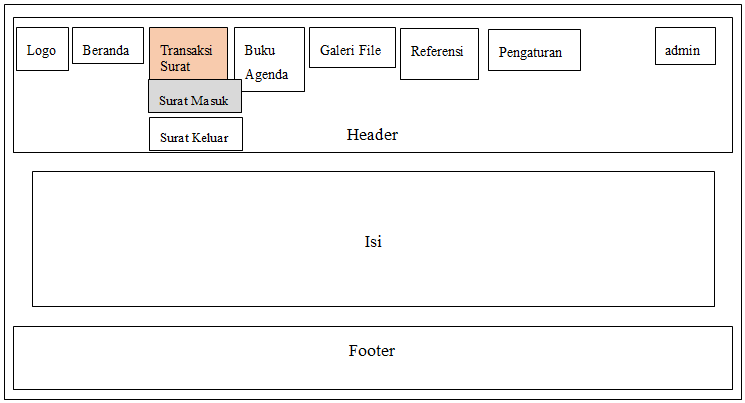
Halaman utama lebih difungsikan sebagai penjelasan dan pengenalan terhadap jumlah surat masuk, jumlah surat keluar, jumlah disposisi, jumlah klasifikasi surat, dan jumlah pengguna.

1. Rancangan Halaman Home



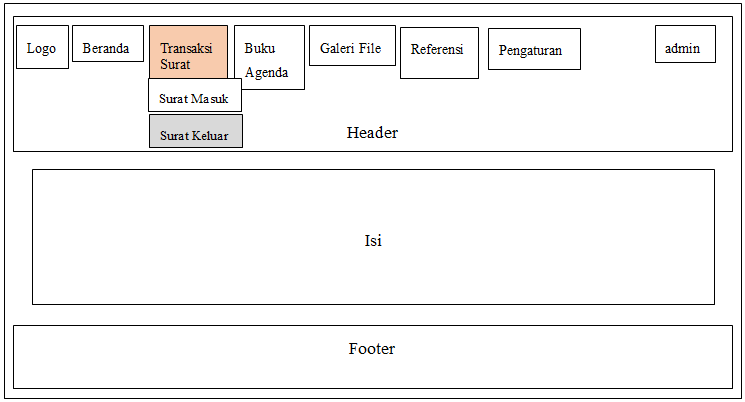
Gambar III.8 Rancangan Halaman Home

1. Transaksi Surat Masuk



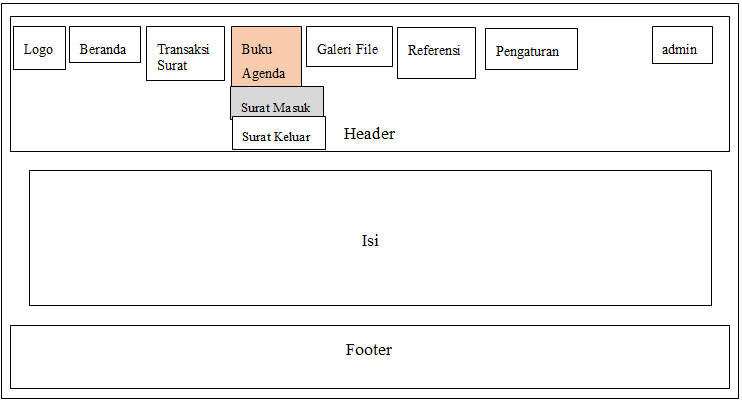
Gambar III.8 Rancangan Transaksi Surat Masuk

1. Transaksi Surat Keluar



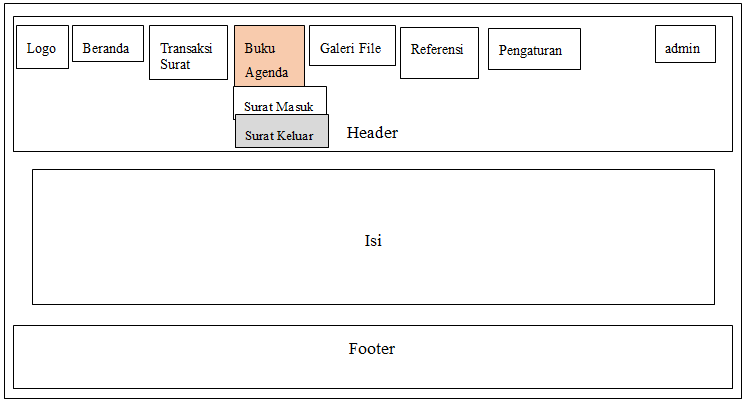
Gambar III.10 Rancangan Transaksi Surat Keluar

1. Agenda Surat Masuk



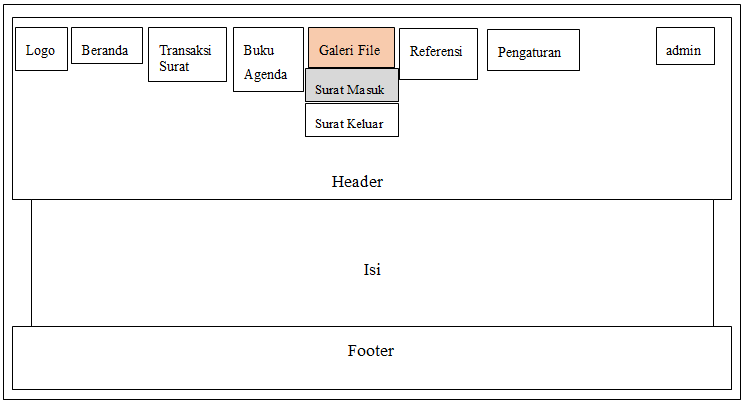
Gambar III.11 Rancangan Agenda Surat Masuk

1. Agenda Surat Keluar



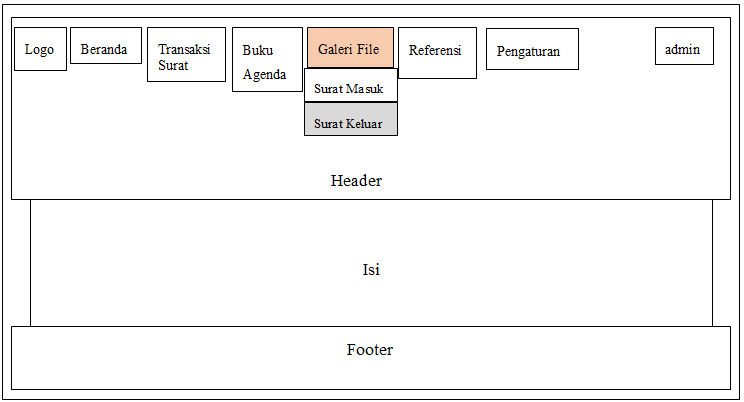
Gambar III.12 Rancangan Agenda Surat Keluar

1. Galeri File Surat Masuk



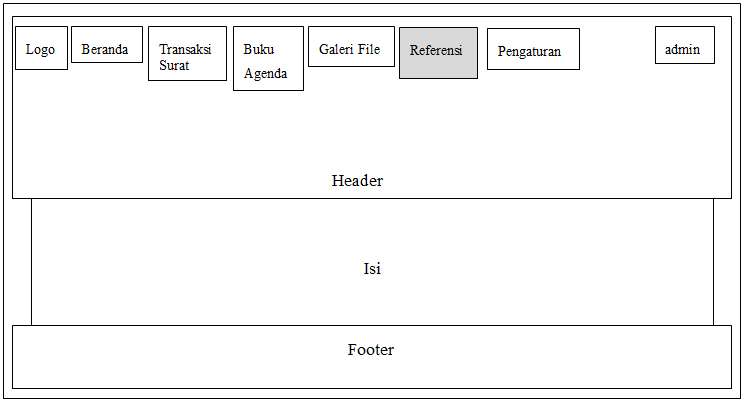
Gambar III.13 Rancangan Galeri File Surat Masuk

1. Galeri File Surat Keluar



Gambar III.14 Rancangan Galeri File Surat Keluar

1. Referensi



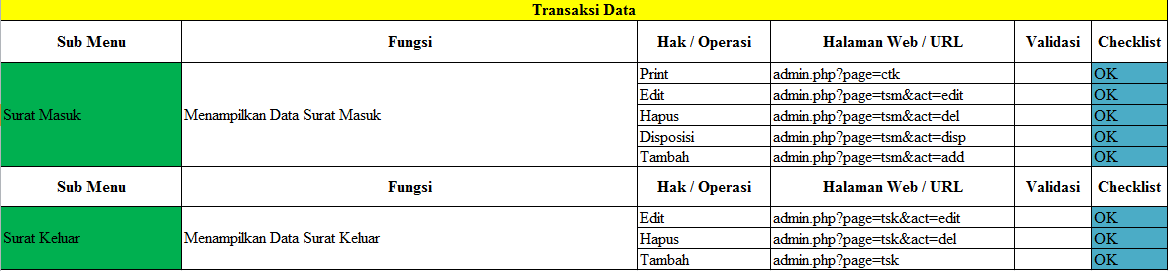
Gambar III.15 Rancangan Referensi

HASIL PEMBAHASAN

## Hasil Pengujian

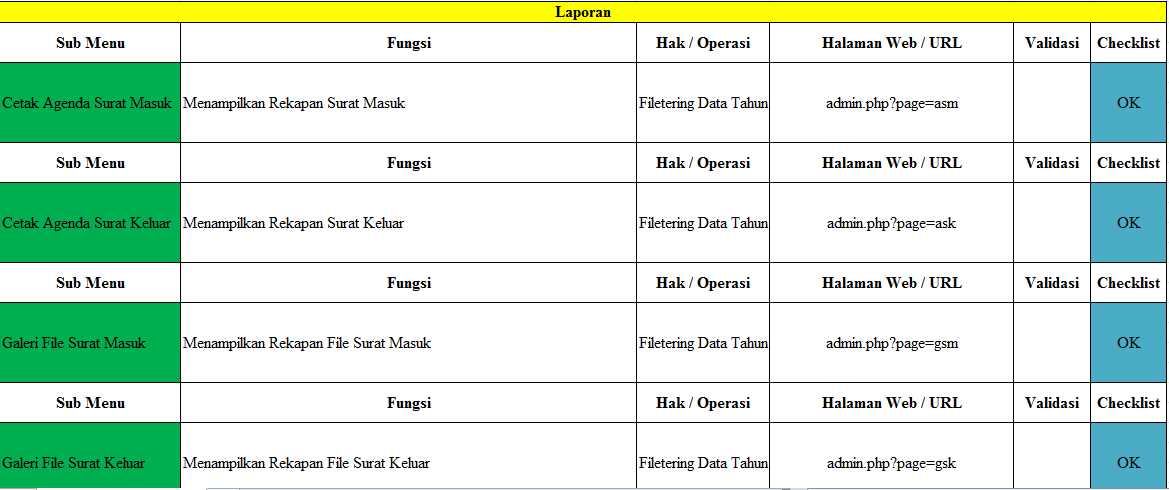
### Hasil pengujian dengan Desk Checking

1. Transaksi Data



Gambar IV.1 Desk Checking Transaksi Data

1. Laporan



Gambar IV.2 Desk Checking Laporan

# 

SIMPULAN DAN SARAN

## Kesimpulan

Sistem surat menyurat yang cepat, tepat, dan akurat sangat penting bagi perkembangan suatu instansi/perusahaan. Dengan munculnya sistem surat menyurat dengan menggunakan komputer sangat membantu dalam mendapatkan informasi yang cepat, tepat dan akurat.

Berdasarkan uraian yang telah dibahas dalam tugas ini, maka dapat diambil kesimpulan yaitu :

1. Perancangan program disposisi surat pada sekolah digunakan sebagai alat bantu untuk mempermudah dalam surat menyurat.
2. Penyimpanan data dengan menggunakan media penyimpanan komputer dinilai lebih aman.
3. Kesalahan-kesalahan yang terjadi dalam proses surat menyurat dapat diperkecil dibanding secara manual.
4. Sebuah program yang baik adalah program aplikasi yang mampu memenuhi segala kebutuhan dari pengguna serta dalam menampung data dalam jumlah besar.

## Saran

Adapun saran-saran yang disampaikan adalah sebagai berikut :

1. Diharapkan untuk melakukan pelatihan terlebih dahulu kepada pengguna sehingga dapat menjalankan program aplikasi ini dengan baik dan benar serta tidak menemukan kesulitan pada saat menggunakannya.
2. Diharapkan pengguna teliti dalam memasukan data, karena tanpa adanya ketelitian hasil yang didapat tidak akan optimal.
3. Meningkatkan pengetahuan Admin selaku pengguna sistem komputer, baik masalah *operasional* maupun pengetahuan lain yang berhubungan dengan dunia komputer.
4. Lakukan pemeriksaan secara rutin terhadap pengoperasian sistem juga perangkat-perangkat pendukung lainnya, sehingga apabila terjadi kesalahan bisa segera diatasi.

DAFTAR PUSTAKA

Alatas, Husein. Responsive Web *Design dengan PHP dan Bootstrap*,Jakarta: Lokomedia, 2013.

A.S., Rosa dan Shalahudin, M. 2013. *Rekayasa Perangat Lunak Terstruktur Dan Berorientasi Objek*. Informatika, Bandung.

Budiman, Agustiar. 2012. *Pengujian Perangkat Lunak dengan Metode Black Box Pada Proses Pra Regristasi User Via Website*. Makalah, halaman: 4.

Hasugian, H., S Shidiq, A.N. (2012). *Rancangan Bangun Sistem Kreatif bidang penyewaan sarana olahraga*, 2012 (semantik), 606-612.

Madcoms. (2013). *Kupas Tuntas Adobe Dreamweaver CSS Dengan Pemograman PHP&MYSQL*. Yogyakarta: Andi Publisher.

Mahendra, Alit 2011. *Macam-macam Struktur Navigasi*. Jakarta

Saputra, Agus, Ridho Taufiq Subagio, dan Saluky. 2012. *Membangun Aplikasi E-Library untuk Panduan Skripsi*. Jakarta: PT, Elex Media Komputindo.

Sibero, Alexander F.K. 2011*Kitab Suci Web Programing*. Jakarta: Mediakom..

Puspitasari. 2011. *Pemrograman Web Database dengan PHP&MYSQL*. Jakarta: Skripta

Prabowo, PudjonWidodo. 2011. *Menggunakan UML*. Informatika Bandung.