



BAB II

TINJAUAN PUSTAKA

2.1 Pengertian Monitoring

Monitoring dari sistem yang akan dibuat merupakan pengamatan jalannya disposisi surat pimpinan Politeknik Negeri Sriwijaya. Menurut Handoko dalam Herliana dan Rasyid (2016:43) Monitoring merupakan sebuah kegiatan untuk menjamin akan tercapainya semua tujuan organisasi dan manajemen. Dalam kesempatan lain menurut Subtari dalam Herliana dan Rasyid (2016:43) monitoring juga didefinisikan sebagai langkah untuk mengkaji apakah kegiatan yang dilaksanakan telah sesuai dengan rencana, mengidentifikasi masalah yang timbul agar langsung dapat diatasi, melakukan penilaian apakah pola kerja dan manajemen yang digunakan sudah tepat untuk mencapai tujuan, mengetahui kaitan antara kegiatan dengan tujuan untuk memperoleh ukuran kemajuan. Menurut Widiastuti dan Susanto (2014:196) Monitoring memiliki beberapa tujuan, yaitu :

1. Mengkaji apakah kegiatan- kegiatan yang dilaksanakan telah sesuai dengan rencana.
2. Mengidentifikasi masalah yang timbul agar langsung dapat diatasi.
3. Melakukan penilaian apakah pola kerja dan manajemen yang digunakan sudah tepat untuk mencapai tujuan kegiatan.
4. Mengetahu ikaitan antara kegiatan dengan tujuan untuk memperoleh ukuran kemajuan.
5. Menyesuaikan kegiatan dengan lingkungan yang berubah, tanpa menyimpang dari tujuan.

2.2 Pengertian Disposisi

Menurut Aji (2014:29) Disposisi adalah petunjuk tertulis mengenai tindak lanjut pengelolaan surat bersama lembar disposisi diantarkan oleh kurir ke dinas atau biro yang dituju, kemudian apabila diperlukan pejabat yang berwenang dapat melakukan disposisi lanjutan kepada bawahannya hingga surat sampai ke tangan



pelaksana untuk ditindaklanjuti. Perangkat singkat dari pimpinan terhadap surat masuk. Terdapat 4 jenis disposisi, yaitu :

1. Disposisi Langsung

Yaitu disposisi yang langsung ditulis pada surat tersebut di tempat yang kosong.

2. Disposisi Tidak Langsung

Yaitu disiapkan lembar disposisi secara tersendiri.

3. Disposisi Reproduksi (Rep) / Tindak Lanjut

Yaitu disposisi yang memerlukan tindak lanjut.

4. Disposisi Deponeren (Dep) / Simpan

Yaitu disposisi yang tidak memerlukan tindak lanjut.

(<http://apriiabosblog.blogspot.co.id/2014/11/pengurusan-surat-masuk-keluar-sistem.html>)

2.3 Pengertian Surat

Menurut Silmi dalam Aji dkk (2014 : 27) Surat adalah sehelai kertas atau lebih yang digunakan untuk mengadakan komunikasi secara tertulis. Dalam kesempatan lain menurut Aji dkk (2014:27) Surat merupakan jembatan pengertian dan alat komunikasi bagi seseorang dan orang lain, surat-surat harus disusun secara singkat dan padat tetapi jelas dan tegas. Menurut Ramelan dalam Dewi dan Oktavia (2017:8-9) ciri-ciri surat adalah sebagai berikut :

1. Surat merupakan pesan tertulis
2. Isi pesan dalam surat berupa informasi atau persuasi
3. Surat memiliki bagian-bagian yang standar
4. Surat memiliki bentuk yang standar
5. Surat memiliki satu pesan inti
6. Gaya bahasa surat bias formal ataupun informal

2.4 Pengertian Android

Android pada sistem yang akan dibuat digunakan sebagai platform untuk mengakses webview sistem disposisi surat yang dibuat. Menurut Athoillah dan



Irawa (2013:2) Android adalah sistem operasi berbasis kernel linux. Google mengibaratkan Android sebagai tumpukan software dimana setiap tumpukan berisi program yang mendukung fungsi spesifik dari sistem operasi, adapun susunan lapisan tersebut dari bawah ke atas adalah sebagai berikut :

1. Linux sebagai kernel
2. Android runtime dan libraries berisi Dalvik Virtual Machine dan kode-kode librari dalam bahasa C/C++
3. Application framework berisi program untuk mengatur fungsi-fungsi dasar smartphone
4. Application.

Android diciptakan oleh sebuah perusahaan kecil bernama Android Inc pada tahun 2000, yang kemudian perusahaan tersebut dibeli oleh Google Inc, untuk mengembangkan android lebih lanjut, dibentuklah Open Handset Alliance (OHA) yang terdiri dari 34 perusahaan software, hardware dan telekomunikasi diantaranya yaitu Google, HTC, Intel, Motorola, T-Mobile dll. Sampai saat ini Android telah banyak berkembang hingga beberapa versi. Android versi 1.1 adalah versi pertama yang dirilis pada tahun 2009, kemudian berturut – turut muncul versi yang lain yang merupakan perbaikan demi perbaikan dari versi yang sebelumnya diantaranya ialah versi 1.5 (Cupcake), versi 1.6 (Donut), versi 2.0/2.1 (Eclair), versi 2.2 (Froyo), versi 2.3 (Gingerbread), versi 3.0/3.1 (Honeycomb), versi 4.0 (ICS), hingga yang terbaru saat ini adalah versi 4.1 (Jellybean).

2.5 Pengertian PHP

PHP digunakan sebagai bahasa pemrograman inti untuk server dari sistem yang dibuat. Menurut Sunoto (2015:149) PreProcessor (PHP) adalah biasanya script yang menyatu dengan tag-tag HTML yang dieksekusi di server dan dikembangkan untuk pembuatan web dinamis. Web dinamis adalah suatu *web* dimana data yang ada didalamnya dapat diubah atau di *update* sewaktu-waktu tergantung kebutuhannya. Ciri dari suatu web dinamis adalah dapat berinteraksi dengan pengguna layanan atau pengunjung informasi ditampilkan dari *database* yang tersimpan di *web server* tersebut. PHP merupakan maintenance untuk

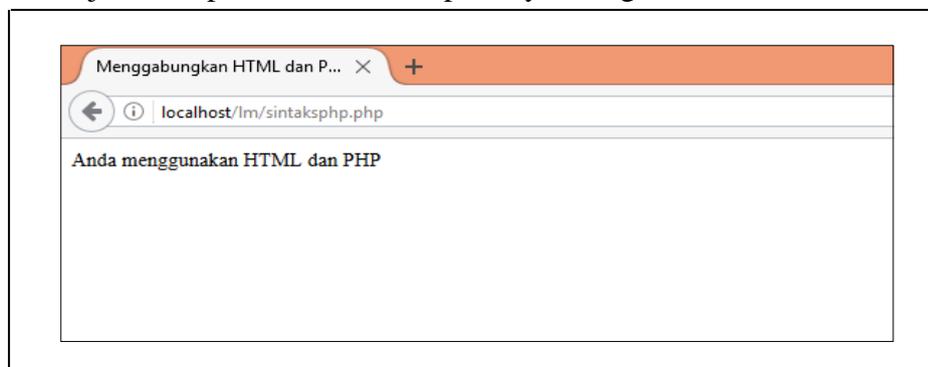


membuat suatu situs web dengan mudah. PHP juga merupakan bahasa script server side yang disisipkan pada HTML.

Menurut Athoillah dan Irawa (2013:2-3) Cara kerja PHP diawali dengan client yang memanggil berdasarkan URL melalui browser, kemudian browser mendapat alamat dari web server, yang nantinya akan memberikan segala informasi yang dibutuhkan web browser. Web browser yang sudah mendapat informasi segera melakukan proses penterjemahan kode PHP dan menampilkannya ke layar pemakai. Ada beberapa aturan dasar dalam penulisan sintaks PHP. Script PHP selalu dimulai dan diakhiri dengan `<?php` dan `?>`, dan untuk setiap statement harus diakhiri dengan tanda `;` atau disebut semicolon, sedangkan untuk pergantian spasi dalam PHP tidak akan mempengaruhi tampilan hasilnya. Contoh Script PHP :

```
<html>
<head>
<title>Menggabungkan HTML dan PHP</title>
</head>
<body>
    <?php
        echo "Anda menggunakan HTML dan PHP";
    ?>
</body>
</html>
```

Pada saat dijalankan pada browser, tampilannya sebagai berikut:



Gambar 2.1. Tampilan PHP sederhana



2.6 Pengertian MySQL

MYSQL sebagai database yang digunakan dari sistem yang dibuat. Menurut Hendry (2015:7), MySQL merupakan turunan salah satu konsep utama dalam basis data yang telah ada sebelumnya, yaitu SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoperasian basis data, terutama untuk pemilihan atau seleksi dan pemasukkan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis. Hendry (2015:79), 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 server 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 cukup besar, dengan jumlah rekaman (*records*) lebih dari 50 juta dan 60 ribu table, 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 socket (Unix), atau Named Pipes (NT).

10. Lokalisasi

MySQL dapat mendeteksi pesan kesalahan pada klien dengan menggunakan lebih dari dua puluh bahasa. Meskipun demikian, bahasa Indonesia belum termasuk di dalamnya.

11. Antarmuka

MySQL memiliki antarmuka (*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 lebih fleksibel dalam menangani ALTER TABLE, dibandingkan basis data lainnya seacam PostgreSQL ataupun Oracle.

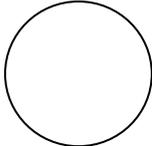
2.7 Pengertian DFD (*Data Flow Diagram*)

Sukamto dan Shalahudin (2013:70), *data flow diagram* (DFD) atau dalam bahasa Indonesia menjadi Diagram Alir Data (DAD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*).

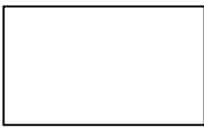
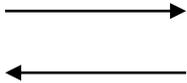


Sukanto, Shalahuddin (2013:71) menjelaskan notasi pada DFD adalah sebagai berikut:

Tabel 2.1 Simbol-simbol *Data Flow Diagram*

| No. | Simbol | Keterangan |
|-----|---|--|
| 1 |  | <p>Proses atau fungsi atau prosedur; pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya menjadi fungsi atau prosedur di dalam kode program</p> <p>catatan :</p> <p>nama yang diberikan pada sebuah proses biasanya berupa kata kerja</p> |
| 2 |  | <p><i>File</i> atau basis data atau penyimpanan (<i>storage</i>); pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya dibuat menjadi tabel-tabel basis data yang dibutuhkan, tabel-tabel ini juga harus sesuai dengan perancangan tabel-tabel pada basis data (<i>Entity Relationship Diagram (ERD)</i>, <i>Conceptual Data Model (CDM)</i>, <i>Physical Data Model (PDM)</i>)</p> <p>catatan :</p> <p>nama yang diberikan pada sebuah penyimpanan biasanya kata benda</p> |

Lanjutan **Table 2.1** Simbol-simbol *Data Flow Diagram*

| | | |
|---|---|---|
| 3 |  | <p>Entitas luar (<i>external entity</i>) atau masukan (<i>input</i>) atau keluaran (<i>output</i>) atau orang yang memakai/berinteraksi dengan perangkat lunak yang dimodelkan atau sistem lain yang terkait dengan aliran data dari sistem yang dimodelkan</p> <p>catatan :</p> <p>nama yang digunakan pada masukan (<i>input</i>) atau keluaran (<i>output</i>) berupa kata benda</p> |
| 4 |  | <p>Aliran data; merupakan data yang dikirim antar proses, dari penyimpanan ke proses, atau dari proses ke masukan (<i>input</i>) atau keluaran (<i>output</i>)</p> <p>catatan :</p> <p>nama yang digunakan pada aliran data biasanya berupa kata benda, dapat diawali dengan kata data misalnya “data siswa” atau tanpa kata data misalnya “siswa”</p> |

(Sumber : Sukamto, Shalahuddin (2013:71))

Sukamto dan Shalahudin (2013:72) menjelaskan tentang tahapan-tahapan perancangan dengan menggunakan *Data Flow Diagram* yaitu :

1. Membuat DFD Level 0 atau sering disebut juga *Context Diagram*

DFD Level 0 menggambarkan sistem yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun sistem lain. DFD Level 0 digunakan untuk menggambarkan interaksi antara sistem yang akan dikembangkan dengan entitas luar.

2. Membuat DFD Level 1

DFD Level 1 digunakan untuk menggambarkan modul-modul yang ada dalam sistem yang akan dikembangkan. DFD Level 1 merupakan hasil *breakdown* DFD Level 0 yang sebelumnya sudah dibuat.



3. Membuat DFD Level 2

Modul-modul pada DFD Level 1 dapat di-*breakdown* menjadi DFD Level 2. Modul mana saja yang harus di-*breakdown* lebih detail tergantung pada kedetailan modul tersebut. Apabila modul tersebut sudah cukup detail dan rinci maka modul tersebut sudah tidak perlu di-*breakdown* lagi. Untuk sebuah sistem, jumlah DFD Level 2 sama dengan jumlah modul pada DFD Level 1 yang di-*breakdown*

4. Membuat DFD Level 3 dan seterusnya

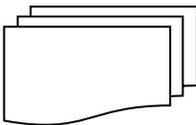
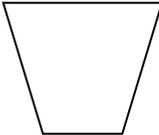
DFD Level 3,4,5, dan seterusnya merupakan *breakdown* dari modul pada DFD Level di-atasnya. *Breakdown* pada level 3,4,5, dan seterusnya aturannya sama persis dengan DFD Level 1 atau 2.

2.8 Pengertian *Blockchart*

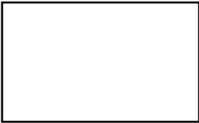
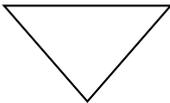
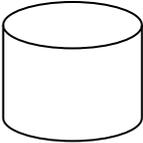
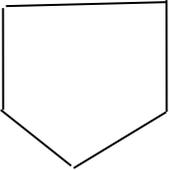
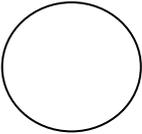
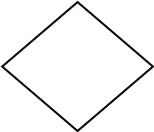
Kristanto (2011:68) menjelaskan, *blockchart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu. Pembuatan *Blockchart* harus memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi.

Kristanto (2011:68) juga menjelaskan tentang Simbol-simbol yang sering digunakan dalam *blockchart* dapat dilihat pada tabel berikut ini :

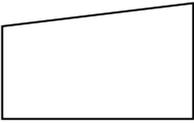
Tabel 2.2 Simbol-simbol *Blockchart*

| No | Simbol | Keterangan |
|----|---|--|
| 1. |  | Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/bendel/berkas atau cetakan |
| 2. |  | Multi dokumen |
| 3. |  | Proses Manual |

Lanjutan **Tabel 2.2** Simbol-simbol *Blockchart*

| | | |
|-----|---|---|
| 4. |  | Proses dilakukan oleh komputer |
| 5. |  | Menandakan dokumen yang diarsipkan (arsip manual) |
| 6. |  | Data penyimpanan (<i>Storage</i>) |
| 7. |  | Proses apa saja yang tidak terdefinisi termasuk aktivitas fisik. |
| 8. |  | Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang lain |
| 9. |  | Terminal yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama |
| 10. |  | Terminal yang menandakan awal dan akhir dari suatu aliran |
| 11. |  | Pengambilan keputusan (<i>Decision</i>) |

Lanjutan **Tabel 2.2** Simbol-simbol *Blockchart*

| | | |
|-----|---|---------------------------------|
| 12. |  | Layar peraga (<i>monitor</i>) |
| 13. |  | Pemasukkan data secara manual |

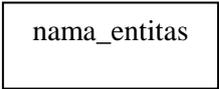
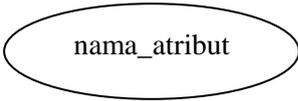
(Sumber : Kristanto (2011:68))

2.9 Pengertian ERD (*Entity Relationship Diagram*)

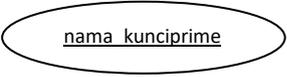
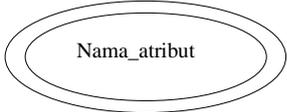
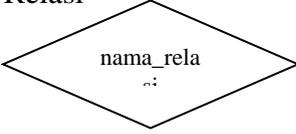
Sukanto dan Shalahuddin (2013:50), ERD adalah pemodelan awal basis data yang dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data relasional.

Sukanto dan Shalahuddin (2013:50) menjelaskan simbol-simbol yang digunakan dalam ERD, yaitu :

Tabel 2.3 Simbol-simbol *Entity Relationship Diagram* (ERD)

| No | Simbol | Keterangan |
|----|---|--|
| 1. | Entitas/ <i>entity</i>  | Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya adar dapat diakses oleh aplikasi komputer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel |
| 2. | Atribut  | <i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas |

Lanjutan **Tabel 2.3** Simbol-simbol *Entity Relationship Diagram* (ERD)

| | | |
|----|---|--|
| 3. | Atribut kunci primer  | <i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> 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) |
| 4. | Atribut multivali / <i>multivalve</i>  | <i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu |
| 5. | Relasi  | Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja |
| 6. | Asosiasi / <i>association</i>  | Penghubung antara relasi dan entitas di mana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian. Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas. Misalkan ada kardinalitas 1 ke N atau sering disebut dengan <i>one to many</i> menghubungkan entitas A dan entitas B |

(Sumber : Sukamto, Shalahuddin (2013:75))



2.10 Pengertian Kamus Data

Sukamto dan Shalahuddin (2013:73), kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum (memiliki standar cara penulisan).

Sukamto dan Shalahuddin (2013:74), menjelaskan simbol-simbol yang di gunakan dalam kamus data, yaitu :

Table 2.4 Simbol-simbol dalam Kamus Data

| No | Simbol | Arti |
|----|------------------|---------------------------------|
| 1 | = | disusun atau terdiri atas |
| 2 | + | Dan |
| 3 | [[]] | baik ...atau... |
| 4 | { } ⁿ | n kali diulang/ bernilai banyak |
| 5 | () | data operasional |
| 6 | *...* | batas komentar |

(Sumber : Sukamto, Shalahuddin (2013:74))

2.11 Framework for the Application of System Thinking (FAST)

Menurut Whitten dkk (2014:81) *Framework Application of System Thinking* atau FAST merupakan Kerangka untuk pemerapan pemikiran sistem yang cukup fleksible untuk menyediakan tipe-tipe berbeda proyek maupun strategi dan berisi gabungan dari praktik praktik penggunaan metode pengembangan sistem yang dapat ditemui dalam banyak metode refensi dan komersial. Terdapat empat fase yang harus diselesaikan untuk semua proyek pengembangan sistem yaitu :

1. Permulaan sistem

Perencanaan awal sebuah proyek untuk mendefinisikan lingkup, tujuan jadwal dan anggaran bisnis awal.

2. Analisis sistem

Studi domain proses bisnis untuk merekomendasikan perbaikan dan menentukan persyaratan dan prioritas bisnis untuk solusi



3. Desain sistem

Spesifikasi atau kontruksi solusi berbasis computer yang teknis untuk persyaratan bisnis yang diidentifikasi dalam analisis sistem.

4. Implementasi sistem

Kontruksi, instalasi, pengujian dan pengiriman sebuah sistem ke dalam operasi.

2.12 Referensi Jurnal

Menurut Budi dan Muhammad (2014) dengan menggunakan metode pembangunan sistem menggunakan model *waterfall*, penelitian ini bertujuan untuk membangun sistem yang dapat membantu didalam tata kelola surat dan memudahkan didalam pendelegasian tugas. Dari hasil penelitian ini diharapkan memudahkan petugas dalam pencarian surat, serta tidak terjadi kehilangan surat, sehingga pekerjaan dapat diselesaikan tepat waktu.

Menurut Sapto dan Miguani (2014) penelitian ini menggunakan metode pengembangan *Sekuel*, aplikasi ini untuk membantu kinerja staf administrasi dalam mengelola surat melalui pencatatan dan penomoran otomatis, penyimpanan pada database dan juga mempersiapkan surat balasan. Para pengguna sistem informasi ini bias mengakses melalui *device* yang terhubung dengan internet, juga mendapatkan pemberitahuan melalui *email* dengan memanfaatkan aplikasi *push e-mail client* di *smartphone* masing-masing.

Menurut Riedsa dan Pradana (2018) penelitian ini bertujuan untuk membantu menganalisa permasalahan dan mendokumentasikan perancangan sistem yang diinginkan PT Mitra Pinasthika Mulia. Pengembangan sistem ini menggunakan FAST (*Framework for the Application of System Thinking*) Pada tiga fase pertama dihasilkan hasil analisis kebutuhan dengan PIECES sebagai kerangka untuk klasifikasi masalah, pada fase desain logis dihasilkan usecase, activity diagram, sequence diagram, wireframe, class diagram, CDM, dan PDM. Selain itu, pada proses evaluasi perancangan menggunakan metode consistency analysis terbukti memiliki nilai presentase 100% konsisten dan termasuk kategori correctness pada uji correctness yang membuktikan bahwa kebutuhan sistem dengan perancangan sistem bersifat konsisten dan benar.



Menurut Syarif dan Setiawan (2013) sistem ini merupakan sarana pelayanan yang lebih baik. Jika saja hal tersebut dapat dimaksimalkan maka kegiatan perusahaan pun akan menjadi lebih efektif dan efisien. Perancangan sistem informasi berbasis website menggunakan metode pengembangan sistem yaitu FAST (Framework for the Application System Thinking) terdapat satu halaman inti dimana pada halaman inti tersebut terbagi menjadi 6 menu, yaitu Home, Golf, Sport Club, Family Club, Birdie Cafe dan Gallery. Kemudian factor internal kekuatan (strength) meliputi Sistem berbasis web, bahasa pemrograman, kemudahan dalam akses informasi, layanan pendaftaran online, security personil, pengelolaan content terjamin, display yang menarik, dan cara pembayaran via transfer. Sedangkan pada faktor internal kelemahan (weakness) meliputi Sistem open sources, bukan sistem java, dokumentasi pendaftaran online, dan sistem validasi.

Menurut Gantini dan Griffin (2011) Bahasa pemrograman yang digunakan adalah Visual C# dan *Database Management System (DBMS) SQL Server*, penelitian ini bertujuan untuk mempermudah mencari dan menemukan data surat beserta disposisi.