

BAB II

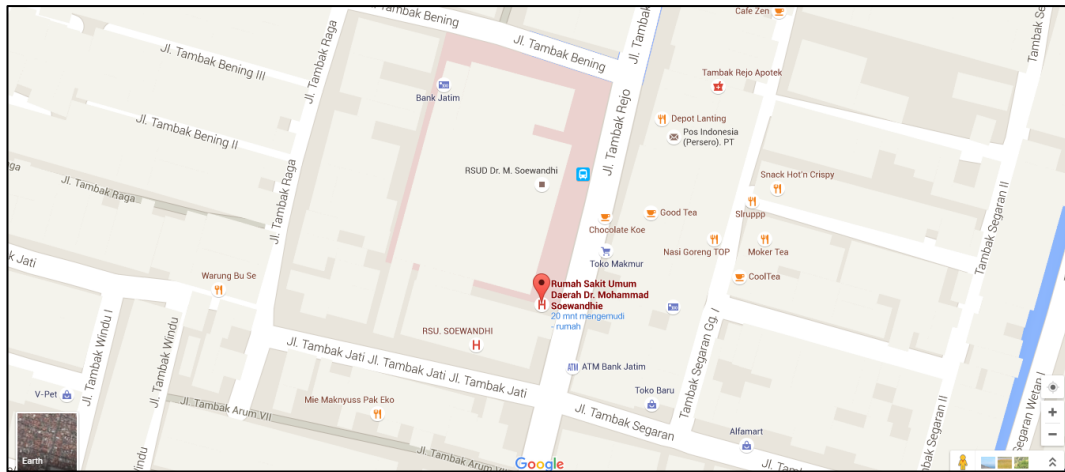
LANDASAN TEORI

2.1 Rumah Sakit Dr. M. Soewandhie Surabaya

RSUD Dr. Mohammad Soewandhie Surabaya atau lebih dikenal masyarakat Surabaya dengan nama RSUD Dr. Soewandhie yang terletak di Jl. Tambak Rejo 45-47, kecamatan Simokerto, kelurahan Tambakrejo, Kota Surabaya. Awal berdiri pada tahun 1964 sebagai pelayanan kesehatan di Kecamatan Tambaksari yang dikenal dengan nama Poli Penyakit Kelamin Tambakrejo, yang dipimpin oleh dr. Margono. RSUD dr. Mohamad Soewandhie sebagai rumah sakit milik Pemerintah Kota Surabaya telah melewati banyak perkembangan, sehingga saat ini mampu menjadi rumah sakit pilihan masyarakat Kota Surabaya, terutama untuk mendukung Sistem Jaminan Sosial Nasional (SJSN).

Berdasarkan SK Menteri Kesehatan RI Nomor : 371/MENKES/SK/V/2009 tanggal 13 Mei 2009 RSUD Dr. Mohamad Soewandhie Surabaya yang sebelumnya adalah Rumah Sakit Umum Daerah dengan klasifikasi kelas C berubah menjadi Rumah Sakit Umum kelas B. Selain itu juga memperoleh status Badan Layanan Umum pada tanggal 23 Juli 2009 melalui SK Walikota No. 188.45/251/436.1.2/2009 tentang Penerapan Pola Pengelolaan Keuangan Badan Layanan Umum Daerah Pada Rumah Sakit Umum Daerah dr. Mohamad Soewandhie Kota Surabaya.

Peta lokasi RSUD Dr. Soewandhie Surabaya tepatnya dapat dilihat pada Gambar 2.1 di bawah ini.



Gambar 2.1 Peta Lokasi RSUD Dr. Soewandhie Surabaya

Tenaga medis di RSUD Dr. Soewandhie Surabaya terdiri dari :

1. Dokter umum : 32 orang
2. Dokter spesialis : 43 orang
3. Dokter gigi : 3 orang
4. Dokter gigi spesialis : 3 orang
5. Ners : 47 orang
6. Perawat gigi : 3 orang
7. Bidan : 23 orang
8. Apoteker : 6 orang
9. Analis farmasi : 39 orang
10. Radiografer : 9 orang
11. Analis kesehatan : 28 orang
12. Rekam medik : 14 orang

- | | |
|-----------------|------------|
| 13. Sanitasi | : 4 orang |
| 14. Nutrisi | : 11 orang |
| 15. Fisioterapi | : 3 orang |

2.1.1 Visi dan Misi

1. Visi

Menjadi Rumah Sakit Pilihan dan Unggulan di Kota Surabaya

2. Misi

Memberikan layanan rumah sakit yang profesional dengan memperhatikan nilai sosial kemasyarakatan. Membentuk Sumber Daya Manusia rumah sakit yang profesional, penguasaan teknologi yang memadai dengan pengembangan diklat dan litbang, produktif dan pembelajar serta santun dan berintegritas. Membangun upaya kerjasama atau kemitraan dengan institusi Pemerintah dan Swasta yang harmonis dan saling menguntungkan.

2.1.2 Motto

"KAMI ADA UNTUK ANDA MELAYANI SEPENUH HATI"

2.1.3 Nilai Dasar Organisasi RSUD dr.Mohamad Soewandhie Surabaya:

"SIP"

- SANTUN
- INTEGRITAS
- PROFESIONAL

2.1.4 Pelayanan Instalasi Rawat Jalan RSUD Dr. Soewandhie

Pelayanan rawat jalan dengan kata lain dapat diartikan bahwa fasilitas yang digunakan sebagai tempat konsultasi, penyelidikan, pemeriksaan, dan pengobatan pasien oleh dokter ahli di bidang masing-masing yang disediakan untuk pasien yang membutuhkan waktu singkat untuk penyembuhannya atau tidak memerlukan pelayanan perawatan. Adapun PERDA No. 9 Tahun 2008 tentang Pelayanan Kesehatan pada Rumah Sakit Umum Daerah Dokter Mohamad Soewandhie pasal 26 dijelaskan bahwa:

1. Jenis pelayanan rawat jalan terdiri dari:

- a. Umum
- b. Spesialis
- c. Darurat
- d. Eksekutif.

2. Alur Pelayanan

Alur pelayanan yaitu kemudahan dan kepastian tahapan pelayanan yang diberikan kepada masyarakat. Alur pelayanan yang harus ada adalah alur pelayanan Rumah Sakit, alur pelayanan kamar bersalin, alur pelayanan laboratorium dan lain-lain. Alur pelayanan pasien gawat darurat disesuaikan dengan kasus agar segera mendapatkan penanganan.

3. Alur Rujukan

Sesuai dengan jenis upaya kesehatan yang diselenggarakan oleh Rumah Sakit ada dua macam rujukan yang dikenal, yakni:

- a. Rujukan Upaya Kesehatan Perorangan

Cakupan rujukan pelayanan kesehatan perorangan adalah rujukan kasus penyakit. Apabila Rumah Sakit tidak mampu menanggulangi satu kasus penyakit tertentu, maka Rumah Sakit tersebut wajib merujuknya ke sarana pelayanan kesehatan yang lebih mampu.

b. Rujukan Upaya Kesehatan Masyarakat

Cakupan rujukan pelayanan kesehatan masyarakat adalah rujukan masalah kesehatan masyarakat, misalnya kejadian luar biasa, pencemaran lingkungan dan bencana. Rujukan ini dilakukan bila Rumah Sakit tidak mampu menyelenggarakan dan tidak mampu menanggulangi upaya kesehatan masyarakat, maka Rumah Sakit wajib merujuknya Dinas Kesehatan Kabupaten/Kota.

2.2 Sistem Administrasi

Administrasi berdasarkan etimologis (asal kata) bersumber dari bahasa latin, yang terdiri ad + ministrate, yang secara operasional berarti melayani, membantu dan memenuhi. Dalam bahasa asalnya dari perkataan itu dapat terbentuk kata benda administratio dan kata sifat administrativus. Perkataan itu masuk ke dalam bahasa Inggris menjadi administration yang lebih banyak dikenal oleh para ilmuwan dan praktisi sekarang ini. Di bawah ini terdapat beberapa pendapat, arti atau definisi dari administrasi, yaitu:

1. Menurut Haryadi (2009:1) dalam bukunya Administrasi Perkantoran untuk Manajer & Staf mengatakan bahwa ada dua pengertian administrasi, yaitu dalam arti sempit dan administrasi dalam arti luas.

- a. Administrasi dalam arti sempit adalah kegiatan penyusunan dan pencatatan data dan informasi secara sistematis dengan tujuan untuk menyediakan keterangan serta memudahkan memperolehnya kembali secara keseluruhan dan dalam satu hubungan satu sama lain.
 - b. Administrasi dalam arti luas adalah kegiatan kerja sama yang dilakukan sekelompok orang berdasarkan pembagian kerja sebagaimana ditentukan dalam struktur dengan mendayagunakan sumber daya untuk mencapai tujuan secara efektif dan efisien.
2. Menurut Siagin (1947:2), Administrasi adalah "Keseluruhan proses pelaksanaan daripada keputusan yang telah diambil dan pelaksanaan itu pada umumnya dilakukan oleh dua orang manusia atau lebih untuk mencapai tujuan yang telah ditentukan sebelumnya".
3. Menurut Wayong (1961), Administrasi adalah "Kegiatan yang dilakukan untuk mengendalikan suatu usaha (pemerintah) agar tujuan tercapai". Administrasi seperti telah banyak disebutkan dalam uraian terdahulu adalah rangkaian kegiatan atau proses pengendalian suatu organisasi agar secara keseluruhan selalu terarah pada pencapaian tujuannya. Dengan demikian administrasi berarti rangkaian kegiatan atau proses pengendalian cara atau sistem kerja sama sejumlah orang agar berlangsung efektif dan efisien dalam mewujudkan tujuan bersama.

2.3 Rekayasa Perangkat Lunak

Menurut Yasin (2012:2), Perangkat Lunak adalah seluruh perintah yang digunakan untuk memproses informasi.

Perangkat lunak dapat berupa program atau prosedur. Program adalah kumpulan perintah yang dimengerti oleh komputer sedangkan prosedur adalah perintah yang dibutuhkan oleh pengguna dalam memproses informasi (O'Brien, 1999).

Pengertian Rekayasa Perangkat Lunak (RPL) adalah suatu disiplin ilmu yang membahas semua aspek produksi perangkat lunak, mulai tahap awal yaitu analisis kebutuhan pengguna, menentukan spesifikasi dari kebutuhan pengguna, desain, pengkodean, pengujian sampai pemeliharaan sistem setelah digunakan. Dengan pengertian ini jelaslah bahwa Rekayasa Perangkat Lunak tidak hanya berhubungan dengan cara pembuatan program komputer. Pernyataan "semua aspek produksi" pada pengertian di atas, mempunyai arti semua hal yang berhubungan dengan proses produksi seperti manajemen proyek, penentuan personil, anggaran biaya, metode, jadwal, kualitas sampai dengan pelatihan pengguna merupakan bagian dari Rekayasa Perangkat Lunak (RPL).

2.3.1 Tujuan Rekayasa Perangkat Lunak

Secara umum tujuan Rekayasa Perangkat Lunak tidak berbeda dengan bidang rekayasa yang lain. Hal ini dapat kita lihat pada Gambar 2.2 di bawah ini.



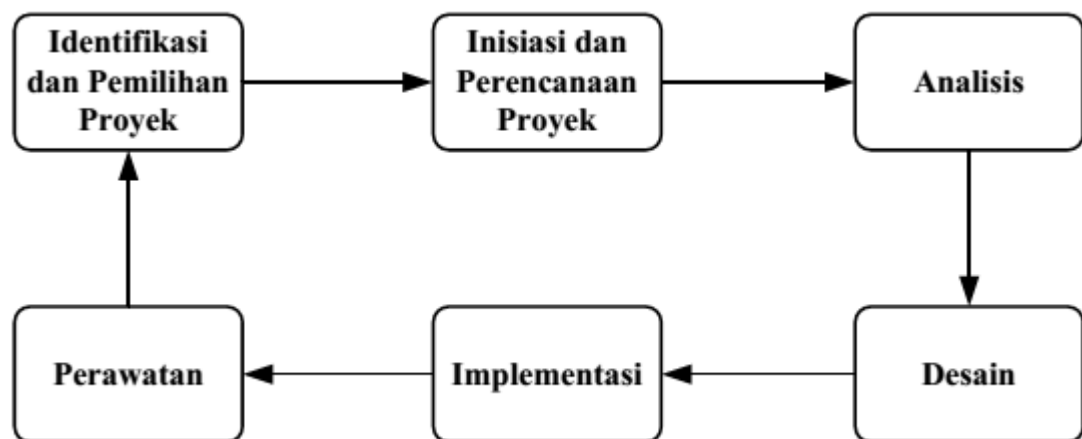
Gambar 2.2 Tujuan Rekayasa Perangkat Lunak

Dari Gambar 2.2 dapat diartikan bahwa bidang rekayasa akan selalu berusaha menghasilkan output yang kinerjanya tinggi, biaya rendah dan waktu penyelesaian yang cepat. Secara lebih khusus kita dapat menyatakan tujuan RPL adalah:

1. Memperoleh biaya produksi perangkat lunak yang rendah.
2. Menghasilkan perangkat lunak yang kinerjanya tinggi, andal dan tepat waktu.
3. Menghasilkan perangkat lunak yang dapat bekerja pada berbagai jenis platform.
4. Menghasilkan perangkat lunak yang biaya perawatannya rendah.

2.3.2 Metode Rekayasa Perangkat Lunak

Pada rekayasa perangkat lunak, banyak model yang telah dikembangkan untuk membantu proses pengembangan perangkat lunak. Model-model ini pada umumnya mengacu pada model proses pengembangan sistem yang disebut *System Development Life Cycle* (SDLC) seperti terlihat pada Gambar 2.3 berikut ini.



Gambar 2.3 *System Development Life Cycle (SDLC)*

Penjelasan dari Gambar 2.3 di atas adalah:

1. Kebutuhan terhadap definisi masalah yang jelas. Input utama dari setiap model pengembangan perangkat lunak adalah pendefinisian masalah yang jelas.
2. Tahapan - tahapan pengembangan yang teratur. Meskipun model-model pengembangan perangkat lunak memiliki pola yang berbeda-beda, biasanya model-model tersebut mengikuti pola umum *analysis – design–coding – testing – maintenance*.
3. *Stakeholder* berperan sangat penting, dapat berupa pengguna, pemilik, pengembang, pemrogram dan orang-orang yang terlibat dalam rekayasa perangkat lunak tersebut.
4. Dokumentasi merupakan bagian penting karena masing-masing tahapan dalam model biasanya menghasilkan sejumlah tulisan, diagram, gambar, atau bentuk-bentuk lain yang harus didokumentasi dan merupakan bagian tak terpisahkan dari perangkat lunak yang dihasilkan.
5. Keluaran dari proses pengembangan perangkat lunak harus bernilai ekonomis. Efek dari penggunaan perangkat lunak yang telah dikembangkan haruslah memberi nilai tambah bagi organisasi.

Menurut Kendall (2007), *Systems Development Life Cycle* (SDLC) atau siklus hidup pengembangan sistem adalah pendekatan melalui beberapa tahap untuk menganalisis dan merancang sistem yang dimana sistem tersebut telah dikembangkan dengan sangat baik melalui penggunaan siklus kegiatan penganalisis dan pemakai secara spesifik.

Siklus pengembangan sistem dibagi atas tujuh tahap, antara lain :

1. Mengidentifikasi masalah, peluang dan tujuan Dalam tahap ini penganalisis menentukan dengan tepat masalah-masalah dalam bisnis mereka, mengukur peluang guna mencapai sisi kompetitif atau menyusun standar-standar industri, dan tujuan-tujuan yang harus dicapai.
2. Menentukan syarat-syarat informasi Dalam tahap ini, penganalisis berusaha untuk memahami informasi apa yang dibutuhkan pemakai agar bisa ditampilkan dalam pekerjaan mereka. Orang-orang yang terlibat adalah penganalisis dan pemakai, manajer operasi dan pegawai operasional. Penganalisis sistem perlu tahu detail-detail fungsi-fungsi sistem yang ada yaitu: siapa, apa, dimana, kapan dan bagaimana dari bisnis yang sedang dipelajari.
3. Menganalisis kebutuhan sistem Dalam tahap ini, penganalisis menganalisis keputusan terstruktur yang dibuat. Penganalisis juga menyiapkan suatu proposal sistem yang berisikan ringkasan apa saja yang ditemukan, analisis biaya keuntungan alternatif yang tersedia serta rekomendasi atas apa saja yang harus dilakukan.
4. Merancang sistem yang direkomendasikan Dalam tahap ini, penganalisis merancang *data-entry* sedemikian rupa sehingga data yang dimasukkan ke dalam sistem informasi benar-benar akurat. Penganalisis juga merancang file-file basis data yang menyimpan data yang diperlukan oleh pembuat keputusan dan penganalisis bekerja sama dengan pemakai untuk merancang *output*. Terakhir penganalisis juga merancang prosedur-prosedur *back-up* dan kontrol untuk melindungi

sistem dan data serta membuat paket-paket spesifikasi program bagi pemrogram.

5. Mengembangkan dan mendokumentasikan perangkat lunak Dalam tahap ini, penganalisis bekerja sama dengan pemrogram mengembangkan suatu perangkat lunak awal yang diperlukan. Penganalisis juga bekerja sama dengan pemakai untuk mengembangkan dokumentasi perangkat lunak yang efektif, mencakup melakukan prosedur secara manual, bantuan *online* dan *website*.
6. Menguji dan mempertahankan sistem Dalam tahap ini, sistem yang telah dibuat harus dilakukan pengujian terlebih dahulu. Sebagian pengujian dilakukan oleh pemrogram sendiri dan lainnya dilakukan oleh penganalisis sistem.
7. Mengimplementasikan dan mengevaluasi sistem Tahap ini merupakan tahap terakhir yang melibatkan pelatihan bagi pemakai untuk pengendalian sistem. Pelatihan dilakukan oleh *vendor*, namun kesalahan pelatihan merupakan tanggung jawab penganalisis sistem. Proses ini mencakup pengubahan file-file dari format lama ke format baru atau membangun suatu basis data, menginstall peralatan, dan membawa sistem baru untuk diproduksi.

2.3.3 Tahapan Rekayasa Perangkat Lunak

Meskipun dalam pendekatan berbeda-beda, namun model-model pendekatan memiliki kesamaan, yaitu menggunakan pola tahapan *analysis – design – coding (construction) – testing – maintenance*.

1. Analisis Sistem adalah sebuah teknik pemecahan masalah yang menguraikan sebuah sistem menjadi komponen-komponennya dengan tujuan mempelajari seberapa bagus komponen-komponen tersebut bekerja dan berinteraksi untuk meraih tujuan mereka.
2. Model Proses adalah model yang menunjukkan aliran data yang masuk dan keluar pada suatu proses. Biasanya model ini digambarkan dalam *Data Flow Diagram / DFD*.
3. Desain Perangkat Lunak adalah tugas, tahapan atau aktivitas yang difokuskan pada spesifikasi detail dari solusi berbasis komputer (Whitten et al, 2004).
4. Konstruksi adalah tahapan menerjemahkan hasil desain logis dan fisik ke dalam kode-kode program komputer.
5. Pengujian sistem melibatkan semua kelompok pengguna yang telah direncanakan pada tahap sebelumnya. Pengujian tingkat penerimaan terhadap perangkat lunak akan berakhir ketika dirasa semua kelompok pengguna menyatakan bisa menerima perangkat lunak tersebut berdasarkan kriteria-kriteria yang telah ditetapkan. Perawatan dan Konfigurasi ketika sebuah perangkat lunak telah dianggap layak untuk dijalankan, maka tahapan baru menjadi muncul yaitu perawatan perangkat lunak.

2.4 Analisis Dan Perancangan Sistem

Menurut Kristanto (2003:5), Analisis sistem adalah seseorang yang mempunyai kemampuan untuk menganalisis sebuah sistem yang meliputi

mempelajari masalah yang timbul dan menentukan kebutuhan pemakai sistem. Untuk mencapai tujuan dari suatu sistem yang dibuat, dibutuhkan 3 perangkat atau alat yang dapat meningkatkan kinerja dari sebuah sistem sehingga tujuan dari sistem tersebut dapat dicapai. Tiga perangkat tersebut meliputi : perangkat keras, perangkat lunak dan perangkat manusia. Perangkat keras data berupa komputer, sedangkan perangkat lunak adalah program. Sedangkan perangkat manusia dapat berupa manajer, analisis sistem, programmer dan sebagainya. Dimana ketiga unsur tersebut bersama-sama membangun sistem yang efisien untuk mengatasi masalah yang dihadapi pemakai sistem.

Menurut Jogiyanto (2001:129) “Analisis Sistem adalah penguraian dari suatu sistem informasi yang utuh kedalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya”. Berdasarkan penjelasan diatas, analisis sistem adalah sebuah tahap yang paling penting dalam suatu pemrograman dimana tahap ini untuk mengevaluasi permasalahan yang ada dan kendala-kendala yang dihadapi. Tahap analisis sistem dilakukan setelah tahap perencanaan sistem dan sebelum tahap desain sistem atau perancangan sistem.

Menurut Kendall (2003:7), analisis sistem dilakukan dengan tujuan untuk dapat mengidentifikasi dan mengevaluasi permasalahan yang terjadi dan kebutuhan yang diharapkan, sehingga dapat diusulkan perbaikannya.

Perancangan sistem merupakan penguraian suatu sistem informasi yang utuh ke dalam bagian komputerisasi yang dimaksud, mengidentifikasi dan mengevaluasi permasalahan, menentukan kriteria, menghitung konsistensi terhadap kriteria yang ada, serta mendapatkan hasil atau tujuan dari masalah tersebut serta mengimplementasikan seluruh kebutuhan operasional dalam membangun aplikasi.

Analisis dan Perancangan Sistem dipergunakan untuk menganalisis, merancang, dan mengimplementasikan peningkatan-peningkatan fungsi bisnis yang dapat dicapai melalui penggunaan sistem informasi terkomputerisasi.

2.5 Konsep Dasar Basis Data

2.5.1 Database

Menurut Yuswanto (2005:2), database merupakan sekumpulan data yang berisi informasi yang saling berhubungan. Pengertian ini sangat berbeda antara database Relasional dan Non Relasional. Pada database Non Relasional, sebuah database hanya merupakan sebuah file.

Menurut Marlinda (2004:1), database adalah suatu susunan/kumpulan data operasional lengkap dari suatu organisasi/perusahaan yang diorganisir/dikelola dan disimpan secara terintegrasi dengan menggunakan metode tertentu menggunakan komputer sehingga mampu menyediakan informasi optimal yang diperlukan pemakainya.

Penyusunan satu database digunakan untuk mengatasi masalah-masalah pada penyusunan data yaitu redundansi dan inkonsistensi data, kesulitan pengaksesan data, isolasi data untuk standarisasi, multiple user (banyak pemakai),

masalah keamanan (*security*), masalah integrasi (*kesatuan*), dan masalah data independence (*kebebasan data*).

2.5.2 Sistem Basis Data

Menurut Marlinda (2004:1), sistem basis data adalah suatu sistem menyusun dan mengelola record menggunakan komputer untuk menyimpan atau merekam serta memelihara dan operasional lengkap sebuah organisasi/perusahaan sehingga mampu menyediakan informasi optimal yang diperlukan pemakai untuk proses mengambil keputusan.

Pada sebuah sistem basis data terdapat komponen-komponen utama yaitu Perangkat Keras (*Hardware*), Sistem Operasi (*Operating System*), Basis Data (*Database*), Sistem (Aplikasi atau Perangkat Lunak) Pengelola Basis Data (DBMS), Pemakai (*User*), dan Aplikasi (Perangkat Lunak) lain (bersifat opsional).

Keuntungan sistem basis data adalah:

1. Mengurangi kerangkapan data, yaitu data yang sama disimpan dalam berkas data yang berbeda-beda sehingga update dilakukan berulang-ulang.
2. Mencegah ketidakkonsistenan.
3. Keamanan data dapat terjaga, yaitu data dapat dilindungi dari pemakai yang tidak berwenang.
4. Integritas dapat dipertahankan.
5. Data dapat dipergunakan bersama-sama.
6. Menyediakan *recovery*.
7. Memudahkan penerapan standarisasi.

8. Data bersifat mandiri (*data independence*).
9. Keterpaduan data terjaga, memelihara keterpaduan data berarti data harus akurat. Hal ini sangat erat hubungannya dengan pengontrolan kerangkapan data dan pemeliharaan keselarasan data.

Kerugian sistem basis data adalah :

1. Diperlukan tempat penyimpanan yang besar.
2. Diperlukan tenaga yang terampil dalam mengolah data.
3. Perangkat lunaknya mahal.
4. Kerusakan sistem basis data dapat mempengaruhi departemen yang terkait.

2.5.3 Database Management System

Menurut Marlinda (2004:6), *Database Management System* (DBMS) merupakan kumpulan file yang saling berkaitan dan program untuk mengelolanya. Basis Data adalah kumpulan datanya, sedang program pengelolanya berdiri sendiri dalam suatu paket program yang komersial untuk membaca data, menghapus data, dan melaporkan data dalam basis data.

Bahasa-bahasa yang terdapat dalam DBMS adalah:

1. *Data Definition Language (DDL)*

Pola skema basis data dispesifikasikan dengan satu set definisi yang diekspresikan dengan satu bahasa khusus yang disebut DDL. Hasil kompilasi perintah DDL

2. *Data Manipulation Language (DML)*

Bahasa yang memperbolehkan pemakai mengakses atau memanipulasi data sebagai yang diorganisasikan sebelumnya model data yang tepat.

3. *Query*

Pernyataan yang diajukan untuk mengambil informasi. Merupakan bagian DML yang digunakan untuk pengambilan informasi.

DBMS memiliki fungsi sebagai berikut :

1. *Data Definition*, DBMS harus dapat mengolah pendefinisian data.
2. *Data Manipulation*, DBMS harus dapat menangani permintaan-permintaan dari pemakai untuk mengakses data.
3. *Data Security dan Integrity*, DBMS dapat memeriksa *security* dan *integrity* data yang didefinisikan oleh *Database Administrator* (DBA).
4. *Data Recovery dan Concurrency*, DBMS harus dapat menangani kegagalan-kegagalan pengaksesan basis data yang dapat disebabkan oleh kesalahan sistem, kerusakan disk, dan sebagainya. DBMS harus dapat mengontrol pengaksesan data yang konkuren yaitu bila satu data diakses secara bersamaan oleh lebih dari satu pemakai pada saat yang bersamaan.
5. *Data Dictionary*, DBMS harus menyediakan *data dictionary*.

2.6 MySQL

Menurut Rudianto (2011) MySql adalah salah satu jenis *database server* yang sangat terkenal dan banyak digunakan untuk membangun aplikasi web yang menggunakan database sebagai sumber dan pengolahan datanya”

MySQL dikembangkan oleh perusahaan swedia bernama MySQL AB yang pada saat ini bernama Tcx sekitar tahun 1994-1995, namun cikal bakal kodenya sudah ada sejak tahun 1979. Awalnya Tcx merupakan perusahaan

pengembang software dan konsultan database, dan saat ini MySQL sudah diambil alih oleh Oracle Corp.

Kepopuleran MySQL antara lain karena MySQL menggunakan SQL sebagai bahasa dasar untuk mengakses databasenya sehingga mudah untuk digunakan, kinerja *query* cepat, dan mencukupi untuk kebutuhan database perusahaan-perusahaan yang berskala kecil sampai menengah, MySQL juga bersifat *open source* (tidak berbayar).

2.7 Sejarah Microsoft Visual Studio 2010

Microsoft Visual Basic (sering disingkat sebagai VB saja) merupakan sebuah bahasa pemrograman yang bersifat event driven dan menawarkan Integrated Development Environment (IDE) visual untuk membuat program aplikasi berbasis sistem operasi Microsoft Windows dengan menggunakan model pemrograman Common Object Model (COM). Visual Basic merupakan turunan bahasa BASIC dan menawarkan pengembangan aplikasi komputer berbasis grafik dengan cepat, akses ke basis data menggunakan Data Access Objects (DAO), Remote Data Objects (RDO), atau ActiveX Data Object (ADO), serta menawarkan pembuatan kontrol ActiveX dan objek ActiveX. Beberapa bahasa skrip seperti Visual Basic for Applications (VBA) dan Visual Basic Scripting Edition (VBScript), mirip seperti halnya Visual Basic, tetapi cara kerjanya yang berbeda.

Pada tahun 2003 Visual Studio 2003 dirilis untuk memperbaiki kinerja dari visual Studio 2002 dengan meluncurkan .NET Framework versi 1.1. Hingga

pada tahun 2009 Visual Studio sudah pada versi 2010, dimana didalamnya sudah ditambahkan bahasa pemrogramana yang baru yaitu F#.

2.8 Testing dan implementasi

Menurut Standar ANSI/IEEE 1059, Testing adalah proses menganalisis suatu entitas software untuk mendeteksi perbedaan antara kondisi yang ada dengan kondisi yang diinginkan (defects/error/bugs) dan mengevaluasi fitur-fitur dari entitas software.

Menurut Romeo (2003:3), Testing software adalah proses mengoperasikan software dalam suatu kondisi yang dikendalikan untuk:

1. Verifikasi, menentukan verifikasi telah berlaku sebagaimana yang ditetapkan (menurut spesifikasi).
2. Mendeteksi error.
3. Validasi, menentukan validasi berdasarkan spesifikasi yang ditetapkan telah memenuhi keinginan atau kebutuhan pengguna yang sebenarnya.

Menurut Romeo (2003:6), Test Case merupakan tes yang dilakukan berdasarkan pada suatu inisialisasi, masukan, kondisi ataupun hasil yang telah ditentukan sebelumnya (2003:33). Metode testing yang digunakan adalah *Black Box Testing*.

Menurut Romeo (2003:52), Metode uji coba black box memfokuskan pada keperluan fungsional dari software. Karena itu uji coba black box memungkinkan pengembang software untuk membuat himpunan kondisi input yang melatih seluruh syarat-syarat fungsional suatu program. Uji coba black box bukan merupakan alternatif dari ujicoba white box, tetapi merupakan pendekatan yang

melengkapi untuk menemukan kesalahan lainnya, selain menggunakan metode white box.

Uji coba Black Box berusaha untuk menemukan kesalahan dalam beberapa kategori, diantaranya :

1. Fungsi-fungsi yang salah atau hilang.
2. Kesalahan interface.
3. Kesalahan dalam struktur data atau akses database eksternal.
4. Kesalahan performa.
5. Kesalahan inisialisasi dan terminasi.

Tidak seperti metode white box yang dilaksanakan diawal proses, uji coba black box diaplikasikan di beberapa tahapan berikutnya. Karena uji coba black box dengan sengaja mengabaikan struktur kontrol, sehingga perhatiannya difokuskan pada informasi domain. Uji coba didesain untuk dapat menjawab pertanyaan berikut : (Romeo,2003:52)

1. Bagaimana validitas fungsionalnya diuji?
2. Jenis input seperti apa yang menghasilkan kasus uji yang baik ?
3. Apakah sistem secara khusus sensitif terhadap nilai input tertentu ?
4. Bagaimana batasan-batasan kelas data diisolasi?
5. Berapa rasio data dan jumlah data yang dapat ditoleransi oleh sistem?
6. Apa akibat yang timbul dari kombinasi spesifik data pada operasi sistem?

Dengan mengaplikasikan uji coba black box, diharapkan dapat menghasilkan sekumpulan kasus uji yang memenuhi kriteria berikut :

1. Kasus uji yang berkurang, jika jumlahnya lebih dari 1 (satu), maka jumlah dari uji kasus tambahan harus didesain untuk mencapai uji coba yang cukup beralasan.
2. Kasus uji yang memberitahukan sesuatu tentang keberadaan atau tidaknya suatu jenis kesalahan, daripada kesalahan yang terhubung hanya dengan suatu uji coba yang spesifik.

Black Box Testing menurut Romeo (2003:62), dilakukan tanpa pengetahuan detail struktur internal dari sistem atau komponen yang dites. Black box testing juga disebut sebagai behavioral testing, specification-based testing, input/output testing atau functional testing. Black box testing berfokus pada kebutuhan fungsional pada software, berdasarkan pada spesifikasi kebutuhan dari software. Dengan adanya black box testing, perekayasa software dapat menggunakan sekumpulan kondisi masukan yang dapat secara penuh memeriksa keseluruhan kebutuhan fungsional pada suatu program. Black box testing bukan teknik alternatif daripada white box testing. Lebih daripada itu, ia merupakan pendekatan pelengkap dalam mencakup error dengan kelas yang berbeda dari metode white box testing.

Kategori error yang diketahui melalui black box testing adalah:






1. Fungsi yang hilang atau tak benar.
2. Error dari antar-muka.
3. Error dari struktur data atau akses eksternal database.
4. Error dari kinerja atau tingkah laku.
5. Error dari inisialisasi dan terminasi.


2.9 Bagan Alir Dokumen

Definisi bagan alir dokumen menurut Jogiyanto (2001:795) adalah bagan yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika. Bagan alir digunakan untuk mengkomunikasikan aliran data dan prosedur proses informasi yang diperlukan dalam sistem informasi. Penggunaannya dilakukan dengan menggunakan simbol-simbol yang dihubungkan dengan panah-panah untuk menunjukkan aktivitas proses informasi.







Terdapat dua jenis di dalam bagan alir dokumen, yaitu *System Flow* dan *Document Flow*. *Document Flow* yaitu bagan alir formulir yang menunjukkan arus dari laporan dan formulir termasuk tembusan-tembusannya. *System Flow* adalah bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem. Dapat dilihat pada tabel 2.1 serta tabel 2.2

Tabel 2.1 Simbol *Document Flow*

Simbol	Nama	Keterangan
	Terminator	Digunakan untuk memulai atau mengakhiri program
	Decision	Perbandingan pernyataan untuk melanjutkan langkah berikutnya
	Dokumen	Sumber data
	Manual proses	Digunakan untuk mendefinisikan proses yang dilakukan secara manual
	Aliran dokumen	Digunakan untuk mengetahui aliran dokumen di dalam suatu program

Simbol	Nama	Keterangan
	Penyimpanan file	Digunakan untuk mendefinisikan dokumen yang diarsipkan

Tabel 2.2 Simbol *System Flow*

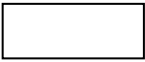
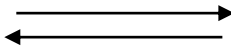
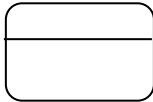

Simbol	Nama	Keterangan
	Input data	Digunakan untuk melakukan proses input data
	Decision	Perbandingan pernyataan untuk melanjutkan langkah berikutnya
	Dokumen	Sumber data
	Proses	Digunakan untuk mendefinisikan proses yang dilakukan oleh sistem
	Aliran data	Digunakan untuk mengetahui aliran data di dalam suatu program
	Penyimpanan data	Digunakan untuk mendefinisikan data-data yang disimpan ke dalam data base

2.10 Diagram Aliran Data

Definisi diagram aliran data menurut Jogiyanto (2001:795) adalah model yang dibuat agar penganalis sistem dapat merepresentasikan proses-proses data di dalam organisasi. Pendekatan aliran data menekankan logika yang mendasari sistem. Dengan menggunakan simbol-simbol diagram aliran data, penganalis

sistem dapat menciptakan suatu gambaran proses-proses yang bisa menampilkan dokumentasi yang solid. Dapat dilihat pada tabel 2.3

Tabel 2.3 Simbol *Data Flow Diagram*

Simbol	Nama	Keterangan
	Entitas eksternal	Unit yang berinteraksi dengan system
	Aliran data	
	Proses	Proses untuk melakukan transformasi data
	Data store	Tempat penyimpanan data