# BAB 2
# TINJAUAN PUSTAKA

## 2.1 Tinjauan Perusahaan

Tahap ini merupakan peninjuan terhadap tempat penelitian studi kasus yang dilakukan di CV. Nusantara Technology (Nusantech). Tinjauan ini meliputi profil CV. Nusantara Technology (Nusantech), visi dan misi, logo perusahaan, struktur organisasi dan deskripsi jabatan yang ada di CV.Nusantara Technology.

### 2.1.1 Profil CV.Nusantara Technology

CV. Nusantara Technology (Nusantech) adalah perusahaan yang bergerak di bidang layanan IT serta penelitian inovasi produk-produk digital yang berpusat di Bandung, Jawa Barat. Nusantech berusaha untuk memberikan kontribusi terbaik untuk membangun ekosistem IT yang lebih baik di Indonesia dengan mengimplementasikan konsep smart city dengan memberdayakan seluruh potensi yang dimiliki oleh penduduk di setiap kota. Nusantech terdiri dari para profesional yang bekerjasama untuk membuat solusi digital untuk mengatasi permasalahan di Indonesia dan dunia. Nusantech juga membantu organisasi dalam menyelesaikan permasalahan dengan memanfaatkan teknologi secara efektif.

Nusantech berperan sebagai inkubator, IT consultant sekaligus tempat belajar berbasis komunitas. Nusantech berusaha menjamin kualitas SDM IT nasional untuk mampu lebih baik dari SDM luar. Sebagai inkubator, Nusantech mendorong pengembangan produk-produk yang dibuat oleh anggotanya sehingga mampu memberikan dampak yang optimal. Sebagai IT consultant, Nusantech secara profesional mengerjakan project-project pembuatan sistem berbasis IT untuk membantu mengatasi permasalahan personal, bisnis dan pemerintah.

Berawal dari rasa kekhawatiran CEO CV. Nusantara Technology Solution (Nusantech) Nicolas Novian Ruslim terhadap perkembangan IT di Indonesia yang dinilai belum siap dalam menghadapi Masyarakat Ekonomi Asean (MEA). maka pada awal Februari 2015 beliau mulai mengumpulkan mahasiswa-mahasiswa dalam sebuah lingkungan perkuliahan, sehingga terbentuklah tim dengan beranggotakan 9 orang dengan kemampuan dibidang IT yang terdiri dari berbagai

minat (website, mobile, robotika, jaringan, sistem kontrol, desain). Untuk menunjang perkembangan CV. Nusantara Technology Solution (Nusantech) beliau memakai gajinya selama 1 tahun sebagai peneliti di lab smart city ITB untuk membiayai penyewaan rumah serta alat-alat penunjang lainnya seperti meja, kursi, internet dan fasilitas lainnya Kemudian barulah pada tanggal 19 mei 2015 secara resmi CV. Nusantara Technology Solution (Nusantech) berdiri, dengan Akta Notaris Irma Rahmawati, SH. No 33, Dengan S. K. Menteri Kehakiman dan Hak Asasi Manusia Republik Indonesia No C – 409 – HT.03.01. Th 2001 Tanggal 30 Oktober 2001 dan S. K. Kepala Badan Pertanahan Nasional Republik Indonesia No 4 – X – 2002 Tanggal 15 April 2002. Berikut Profil CV.Nusantara Technology (Nusantech):

Nama Perusahaan : CV.Nusantara Technology
Alamat : Jl. Sekeloa Timur No.62 Bandung
Telepon : 0878-2329-7974
Email : hello@nusantech.co
Website : www.nusantech.co

### 2.1.2 Visi dan Misi CV. Nusantara Technology

Berikut ini adalah Visi dan Misi CV. Nusantara Technology akan dijelaskan sebagai berikut:

1. Visi

   Penggerak utama dalam pengembangan SDM profesionl bidang IT di Indonesia.

2. Misi
   a. Menciptakan ekosistem pertukaran informasi dan kolaborasi di bidang IT
   b. Memberikan bimbingan untuk menghasilkan profesional Indonesia dibidang IT
   c. Mendorong pengembangan produk IT nasional secara berkelanjutan
   d. Melakukan standarisasi dan kontrol kualitas produk IT yang dihasilkan

### 2.1.3 Logo Perusahaan

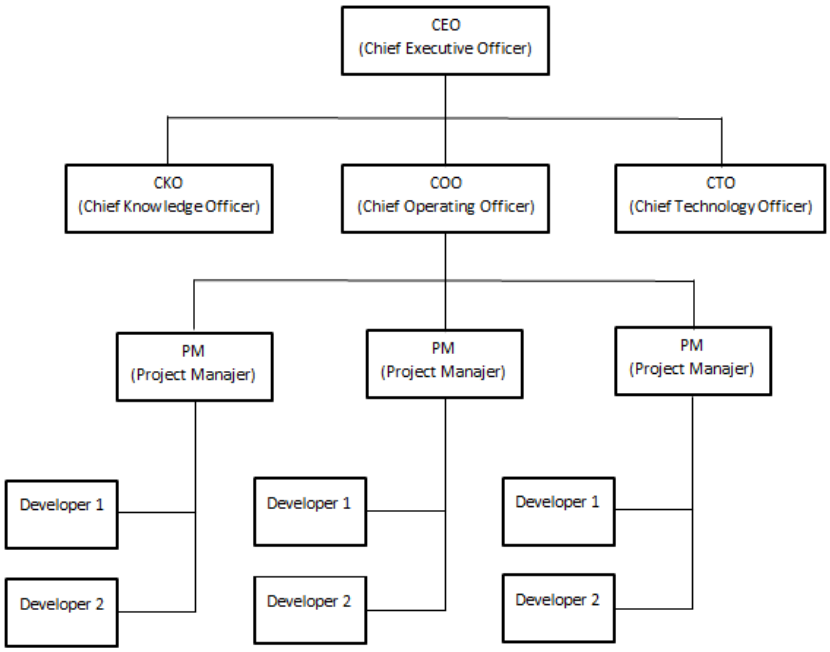
Berikut ini merupakan Logo dari CV. Nusantara Technology dapat dilihat pada gambar 2.1.

**Gambar 2. 1 Logo CV. Nusantara Technology**

### 2.1.4 Struktur Organisasi Perusahaan dan Deskripsi Jabatan

Struktur organisasi merupakan salah satu kerangka yang menggambarkan hubungan antar bidang-bidang kerja maupun orang-orang yang mempunyai kedudukan, wewnang, dan tanggung jawab dari bidang kerja yang dipimpinnya dalam suatu sistem kerjasama.

Struktur organisasi CV. Nusantara Technology dapat dilihat pada gambar 2.2.




**Gambar 2. 2 Struktur Organisasi CV. Nusantara Technology**

Berdasarkan struktur organisasi CV. Nusantara Technology pada gambar 2.2, dijelaskan deskripsi jabatan sebagai berikut:

1. CEO (Chief Executive Officer)

   CEO sebagai penanggung jawab perusahaan, yang mempunyai wewenang atas perusahaan, serta yang mengeluarkan kebijakan-kebijakan perusahaan.

2. COO (Chief Operating Officer)

   COO bertanggung jawab atas lancarnya kegiatan produksi dan produktifitas karyawan-karyawa sehingga perusahaan dapat berjalan dengan baik dan lancar. Serta bertugas menggantikan tugas CEO jika CEO tidak ada di perusahaan, semua tanggung jawab dan wewenang atas perusahaan diambil alih oleh COO.

3. CTO (Chief Technology Officer)

   CTO bertanggung jawab untuk penelitian dan pengembangan serta untuk perencanaan produk baru. Serta Mengembangkan strategi untuk meningkatkan penghasilan/pemasukan.

4. CKO (Chief Knowledge Officer)

   CKO bertugas mengumplkan, serta menggunakan pengetahuan yang ada untuk selalu memperbaiki proses bisnis dalam perusahaan. Serta memanage semua pengetahuan dari setiap pegawai untuk digunakan lagi oleh pegawai lain.

5. Mobile developer

   Tugas dan tanggung jawab *Mobile Developer*:

   a. Membangun aplikasi mobile (android, iOS) sesuai rancangan
   b. Mencari dan menerapkan algoritma terbaik dalam membangun aplikasi
   c. Melakukan perbaikan atas aplikasi atau pemeliharaan aplikasi
   d. Membuat spesifikasi program secara detail melalui diskusi dengan client
   e. Menjelaskan secara tepat apa tindakan (aksi) program yang diinginkan
   f. Menguraikan spesifikasi program ke dalam elemen-elemen sederhana dan menerjemahkan logikanya ke dalam bahasa pemrograman
   g. Memikirkan solusi yang mungkin untuk memprediksi masalah, mengevaluasi pilihan lain

h. Bekerja sebagai bagian dari tim, dimana mengadakan proyek khusus, untuk membuat bagian tertentu dari program
i. Mengkombinasikan semua elemen dari rancangan program dan mengujinya
j. Menguji *sample* data-set untuk memeriksa keluaran dari program sesuai dengan yang diinginkan.

6. Website Developer

Tugas dan tanggung jawab *Website Devloper*:

a. Membuat gambar-gambar yang akan disisipkan kedalam website dengan menggunakan Photoshop dan sebagainya.
b. Mengatur Konten Website sesuai dengan kategori dan diintegrasikan dengan *link navigasi*.
c. Membangun Website mulai dari halaman depan sampai akhir sesuai dengan struktur web yang dipilih.
d. Membagi tugas pemrograman dengan memecah-mecah setiap unit pemrograman.
e. Melakukan pengkodean terhadap setiap unit pemrograman tersebut.
f. Memasukkan Data kedalam Database berikut dengan table-teblenya.
g. Membuat *script* untuk mengakses Database & menghubungkan tabel-tabel dalam Database
h. Melakukan uji coba kinerja program pada setiap unit pemrograman
i. Menata Layout untuk pemasangan Banner, Link, Animasi dan sebagainya.
j. Melakukan uji coba setiap halaman & memeriksa kesalahan penulisan HTML, CSS, dan sebagainya

## 2.2 Landasan Teori

landasan teori pada penulisan skripsi ini akan menerangkan mengenai teori-teori yang berhubungan dengan Manajemen Proyek di CV. Nusantara Technology.

### 2.2.1 Manajemen

Manajemen merupakan suatu pengetahuan tentang seni memimpin organisasi yang terdiri atas kegiatan perencanaan, pengorganisasian, pelaksanaan dan pengendalian terhadap sumber-sumber daya yang terbatas dalam usaha mencapai tujuan dan sasaran yang efektif dan efisien .[1]

Tujuan manajemen yaitu untuk mendapatkan metode atau cara teknis yang paling baik agar dengan sumber-sumber daya yang terbatas diperoleh hasil maksimal dalam hal ketepatan, kecepatan, penghematan dan keselamatan kerja secara komprehensif. Adapun unsur-unsur manajemen yaitu: [1]

a. Tujuan : sasaran yang hendak dicapai dalam optimasi biaya, mutu, waktu dan keselamatan.
b. Pemimpin : mengarahkan organisasi dalam mencapai sasaran dan tujuan.
c. Sumber-sumber daya yang terbatas : manusia, modal/biaya,peralatan dan material.
d. Kegiatan : perencanaan, pengorganisasian, pelaksanaan dan pengendalian.
    1. Perencanaan (planning)

       Pada kegiatan ini dilakukan antisipasi tugas dan kondisi yang ada dengan menetapkan sasaran dan tujuan yang harus dicapai serta menentukan kebijakan pelaksanaan, program yang akan dilakukan, jadwal dan waktu pelaksanaan, prosedur pelaksanaan secara administratif dan operasional serta alokasi anggaran biaya dan sumber daya.
    2. Pengorganisasian (organizing)

       Pada kegiatan ini dilakukan identifikasi dan pengelompokan jenis-jenis pekerjaan, menentukan pendelegasian wewenang dan tanggung jawab personel serta meletakkan dasar bagi hubungan masing-masing unsur organisasi. Untuk menggerakkan organisasi, pemimpin harus mampu mengarahkan organisasi dan menjalin komunikasi antar pribadi dalam hirarki organisasi. semua itu dibangkitkan melalui tanggung jawab dan partisipasi semua pihak.
    3. Pelaksanaan (actuating)

       Kegiatan ini adalah mengimplementasi dari perencanaan yang telah ditetapkan, dengan melakukan tahapan pekerjaan yang sesungguhnya

secara fisik atau nonfisik sehingga produk akhir sesuai dengan sasaran dan tujuan yang telah ditetapkan. Karena kondisi perencanaan sifatnya masih ramalan dan subyektif serta masih perlu penyempurnaan, dalam tahapan ini sering terjadi perubahan-perubahan dari rencana yang telah ditetapkan.

4. Pengendalian (controlling)

   Kegiatan yang dilakukan pada tahap ini dimaksudkan untuk memastikan bahwa program dan aturan kerja yang telah ditetapkan dapat dicapai dengan penyimpangan paling minimal dan hasil paling memuaskan. Untuk itu dilakukan bentuk-bentuk kegiatan seperti berikut.

   a. Supervsi

      Melakukan serangkaian tindakan koordinasi pengawasan dalam batas wewenang dan tanggung jawab menurut prosedur dan organisasi yang telah ditetapkan, agar dalam operasional dapat dilakukan secara bersama-sama oleh semua personel dengan kendali pengawas.

   b. Inspeksi

      Melakukan pemeriksaan terhadap hasil pekerjaan dengan tujuan menjamin spesifikasi mutu dan produk sesuai dengan yang direncanakan.

   c. Tindakan Koreksi

      Melakukan perubahan dan perbaikan terhadap rencana yang telah ditetapkan untuk menyesuaikan dengan kondisi pelaksanaan. Proses dalam manajemen sifatnya umum dan dapat digunakan dalam berbagai kegiatan/bidang yang membutuhkan pengelolaan yang sistematis, terarah serta mempunya sasaran dan tujuan yang jelas. Macam dan bidang yang menggunakan ilmu manajemen adalah manajemen pemerintahan, manajemen industri, manajemen perusahaan, manajemen sumberdaya, manajemen proyek, dan lain sebagainya. Untuk manajemen proyek biasanya waktu dibatasi oleh program-program yang sifatnya sementara dan berakhir bila sasaran dan tujuan organisasi proyek sudah tercapai. Bila membuat proyek sejenis pada waktu sesudahnya, biasanya sasaran dan tujuannya lebih inovatif dengan memodifikasi program-progaram sebelumnya.

### 2.2.2 Proyek

Berikut adalah bahasan tentang proyek yang terdiri dari defenisi proyek, karakteristik proyek, stakeholder proyek dan organisasi proyek.

#### 2.2.2.1 Definisi Proyek

Proyek adalah gabungan dari sumber-sumber daya seperti manusia , material, peralatan dan modal/biaya yang dihimpun dalam suatu wadah organisasi sementara untuk mencapai sasaran dan tujuan. Proses manajemen proyek dimulai dari kegiatan perencanaan hingga pengendalian yang didasarkan pada input-input seperti tujuan dan sasaran proyek, informasi dan data yang digunakan, serta penggunaan sumber daya yang benar dan sesuai dengan kebutuhan yang diperlukan[1].

Dalam proses yang sesungguhnya, pemimpin dengan wewenang yang ada dalam organisasi proyek mengelola dan mengarahkan segala perangkat dan sumber daya yang ada dengan kondisi terbatas, tetapi berusaha memperoleh pencapaian paling maksimal sesuai dengan standar kinerja proyek dalam hal biaya, mutu, waktu dan keselamatan kerja yang telah ditetapkan sebelumnya. Untuk mendapatkan produk akhir yang maksimal, segala macam kegiatan pada proses manajemen proyek direncanakan dengan detail dan akurat untuk mengurangi penyimpangan-penyimpangan. Dan bila ada tindakan koreksi dalam proses selanjutnya, diusahakan koreksi tersebut tidak terlalu banyak[1].

#### 2.2.2.2 Karakteristik Proyek

Masing-masing proyek biasanya mempunyai karakteristik tersendiri dalam hal kegiatan yang dilakukan, tujuan dan saran, serta produk akhir. Untuk lebih jelas, berikut ini diuraikan jenis proyek berdasarkan komponen kegiatan utama dan produk akhir [1].

a. Proyek Konstruksi
b. Proyek Industri Manufaktur
c. Proyek Penelitian dan Pengembangan
d. Proyek Padat Modal
e. Proyek Pengembangan Produk Baru
f. Proyek Pelaksanaan Manajemen
g. Proyek Infrastruktur.

### 2.2.2.3 Stakeholder Proyek

Agar keinginan dan kebutuhan masing-masing pihak dalam suatu proyek dapat direalisasikan dalam suatu usaha bersama untuk pencapaian sasaran dan tujuan, perlu dilakukan identifikasi terhadap organisasi atau individual (Stakeholder), baik dari internal maupun eksternal, yang akan berperan mempengaruhi proyek dan harus diantisipasi selama proyek berlangsung. Stakeholder proyek secara umum diuraikan dibawah ini [1].

1. Manajer Proyek : orang yang bertanggung jawab mengelola proyek
2. Pelanggan *(customer) :* seseorang/oganisasi yang menggunakan produk proyek.
3. Organisasi Proyek : hierarki/susunan tugas dan wewenang individual.
4. Sponsor : Penyedia sumber dana untuk proyek.
5. Masyarakat : sebagai konsumen.

### 2.2.2.4 Organisasi Proyek

Oraganisasi Proyek adalah sebagai sarana dalam pencapaian tujuan dengan mengatur dan mengorganisasi sumber daya, tenaga kerja, material, peralatan dan modal secara efektif dan efisien dengan menerapkan sistem manajemen sesuai kebutuhan proyek. agar tujuan organisasi dapat dicapai, dilakukan proses sebagai berikut:

1. Identifikasi dan pembagian kegiatan

   Identifikasi dan pembagian kegiatan proyek perlu diketahui untuk menentukan volume pekerjaan, macam dan jenisnya, kebutuhan sumber daya, jadwal pelaksanaan serta anggarannya sehingga dapat dilaksanakan oleh penanggung jawab kegiatan sesuai dengan sasaran dan tujuan proyek.
2. Pengelompokan penanggung jawab kegiatan

   Agar hasilnya maksimal pemilihan penanggung jawab organisasi disesuaikan dengan keahlian, keterampilan dan kemampuan personel di bidangnya sehingga sasaran dan tujuan proyek dapat tercapai.

3. Penentuan wewenang dan tanggung jawab

   Setiap ersonel penggung jawab kegiatan harus mengetahui wewenang dan tanggung jawab pekerjaannya, dengan membuat penjabaran kerja serta standar prosedur operasional pekerjaan yang dikelolanya.

4. Menyusun mekanisme pengendalian

   Karena organisasi proyek melibatkan banyak pihak, maka agar tidak terjadi penyimpangan, mekanisme pengendalian dan koordinasi dibuat dalam format yang dapat menggerakkan organisasi dalam mengidentifikasi, memecahkan masalah, serta melakukan tindakan koreksi untuk mengatasi penyimpangan [1].

### 2.2.3 Manajemen Proyek

Berikut ini akan membahas tentang manajemen proyek yang terdiri dari definisi manajemen proyek dan tujuan manajemen proyek.

#### 2.2.3.1 Definisi Manajemen proyek

Manajemen proyek merupakan fungsi, disiplin dan tugas yang harus dikerjakan/dilakukan manajer/profesional dalam disiplin praktis dalam menjalankan fungsinya dan/atau melaksanakan tugasnya [1].

Manajemen proyek merupakan proses yang terdiri dari 8 aktifitas yang meliputi [1]:

1. Perencanaan : memutuskan apa yang harus dilakukan
2. Mengorganisasi : membuat penyusunan
3. Pengaturan orang : memilih orang yang tepat untuk sebuah pekerjaan
4. Mengarahkan : memberikan instruksi
5. Monitor : melakukan pengecekan progres
6. Kontrol : mengambil *action* untuk perbaikan
7. Inovasi : memberikan solusi baru
8. Mewakili : berhubungan dengan *user*
9. Dan unsur (5M) : Man, Material, Machine (termasuk pelayanan, fasilitas dan energi), Money dan Method.

### 2.2.3.2 Tujuan Manajemen Proyek

Adapun tujuan dari manajemen proyek diantaranya:

1 Efisiensi (biaya, sumber daya dan waktu)
2 Kontrol terhadap proyek lebih baik sehingga proyek dapat sesuai dengan scope, biaya, sumber daya dan waktu yang telah ditentukan
3 Meningkatkan kualitas
4 Meningkatkan kualitas
5 Meningkatkan produktifitas
6 Dapat menekan resiko yang timbul
7 Koordinasi internal menjadi lebih baik
8 Meningkatkan semangat,tanggung jawab serta loyalitas tim terhadap proyek, yaitu dengan penugasan yang jelas kepada masing-masing anggota tim.

### 2.2.4 Metode Earned Value Management (EVM)

Dalam penentuan kinerja proyek dengna cara earned value manajemen atau Nilai Hasil, informasi yang ditampilkan berupa indikator dalam bentuk kuantitatif, yang menampilkan informasi *progress* biaya dan jadwal proyek. Indikator ini menginformasikan posisi kemajuan proyek dalam jangka waktu tertentu serta dapat memperkirakan proyeksi kemajuan proyek pada periode selanjutnya. Indikator-indikator tersebut adalah sebagai berikut [2]:

1. BCWS *(Budgeted Cost of Work Shedule)*, menggambarkan anggaran rencana sampai pada periode tertentu terhadap volume rencana proyek yang akan dikerjakan.
2. BCWP *(Budgeted Cost of Work Performed)*, menggambarkan anggaran rencana proyek pada periode tertentu terhadap apa yang telah dikerjakan pada volume pekerjaan aktual.
3. ACWP *(Actual Cost of Work Performed)*, menggambarkan anggaran actual yang dihabiskan untuk pelaksanaan pekerjaan pada keadaan volume pekerjaan aktual.

Berbekal ketiga indikator tersebut pengukuran kinerja biaya dan waktu untuk metode earned Value menggunakan 3 jenis kurva S sebagai nilai kumulatif

biaya dengan fungsi waktu, yang terintegrasi dalam satu tampilan yang terdiri atas nilai kumulatif biaya: BCWS, BCWP, ACWP.

Istilah-istilah yang digunakan dalam metode EVM adalah sebagai berikut [2]:

1. Penyimpangan jadwal/waktu
   a. SV *Schedule Varience* = BCWP – BCWS

      SV > 0, progres aktual > rencana : terjadi percepatan proyek terhadap rencana *(chedule underrun)*

      SV < 0, progres aktual < rencana : terjadi keterlambatan proyek terhadap rencana *(chedule overrun)*

   b. SPI (Schedule Performance Index) = BCWP/BCWS

      SPI > 1, progres aktual > rencana : terjadi percepatan proyek terhadap rencana *(Schedule underrun)*

      SPI < 1, progres aktual < rencana : terjadi keterlambatan proyek terhadap rencana *(Schedule overrun)*
2. Penyimpangan Biaya
   a. CV (Cost Variace) = BCWP-ACWP

      CV > 0, biaya volume aktual > biaya aktual *(Cost Underrun)*

      CV < 0, biaya volume aktual < biaya aktual *(Cost Overrun)*

   b. CPI *(Cost Performance Index)* = BCWP/ACWP

      CPI > 1, biaya volume aktual > biaya aktual *(Cost Underrun)*

      CPI < 1, biaya volume aktual < biaya aktual *(Cost Overrun)*

Kemajuan proyek untuk waktu yang akan datang dapat diramalkan dengan cara seperti berikut :

*1.* Perkiraan penyelesaian proyek *(Estimated Completion Date)*

ECD = (Sisa Waktu/SPI) + Waktu Terpakai

Persentase keterlambatan/percepatan = 100% - ECD/Jadwal Rencana * 100%

*2.* Perkiraan biaya penyelesaian proyek *(Estimated at Completion)*

EAC = Sisa anggaran/CPI + ACWP

= (Total Biaya – BCWP) / CPI + ACWP

Persentase biaya penambahan/penurunan biaya aktual tehadap anggaran biaya

= 100% - EAC/Total Biaya * 100%

3. Earned Value ( Nilai hasil) = BCWP (Biaya penyelesaian volume pekerjaan pada periode tertentu)

Ketiga hal tersebut adalah indikator yang dapat dihitung pada *baseline/milestone* yang telah ditentukan, sehingga nilai-nilai yang didapat menunjukkan progres proyek yang pada periode tersebut dan progres proyek dari segi biaya dan waktu untuk penyelesaian pada masa yang akan datang.

### 2.2.5 Metode Function Point

*Function Point* adalah salah satu pendekatan pengukuran perangkat lunak untuk mengukur ukuran sistem berdasar kebutuhan sistem. *Function Point analysis* (FPA) adalah takaran tidak langsung untuk ukuran fungsional suatu sistem [3].

*Function Point Analysis* dikembangkan pertama kali oleh Allan J. Albrecht di pertengahan 1970 dengan tujuan mencoba menyelesaikan kesulitan terkait dengan *Line of Code* (LOC), dan membantu dalam pengembangan sebuah mekanisme untuk meramalkan beban (*effort*) terkait dengan pengembangan perangkat lunak. *Function Point* pertama dipublikasikan pada tahun 1979, kemudian tahun 1983. Tahun 1984, Albrecht memperbaiki metode ini dan sejak 1986, ketika *International Function Point User Group* (IFPUG) dibentuk, beberapa versi *Function Point Counting Practices Manual* diterbitkan oleh IFPUG. (Albrecht, 1983). Dalam metode Function Points, ukuran sebuah sistem dapat dihitung dengan 3 komponen, yaitu *information processing size* (*Unadjusted Function Points-UFP*), *Technical complexity adjustment factors* dan *Function Points*.

*Function Point* di dalam sebuah perangkat lunak diidentifikasi dan dikategorikan ke dalam salah satu dari lima tipe fungsi pengguna, yaitu: *External Input* (EI), *External Outputs* (EO), *Internal Logical File* (ILF), *External Interface Files* (EIF) dan *External Inquiry* (EQ), kemudian dinilai kompleksitasnya dan diberi sejumlah nilai *Function Points*.

*External Input* (EI) adalah proses dasar yang memproses data dan informasi kontrol yang datang dari luar batasan aplikasi. Data mungkin digunakan untuk memelihara satu atau lebih berkas logika internal. External Input dapat untuk mengubah perilaku sistem.

*External Outputs* (EO) merupakan sebuah proses dasar dimana hasil data dilewatkan dari dalam ke keluar dari batasan aplikasi, contohnya aplikasi menghasilkan berkas XML atau CSV. *External Outputs* menampilkan informasi melalui logika pemrosesan, tidak hanya mengambil data. *External Outputs* dapat berupa data laporan atau output file yang dikirim ke aplikasi lain. Laporan dan file tersebut dibuat dari satu atau lebih *Internal Logical File* (ILF) dan *External Interface Files* (EIF).

Fungsi utama *External Inquiry* (EQ) adalah menyediakan informasi ke user melalui pengambilan atau pemrosesan data atau informasi kontrol dari *Internal Logical File* (ILF) atau *External Interface Files* (EIF). Tampilan beberapa tipe laporan dan pencarian merupakan komponen yang tepat untuk *External Inquiry*. *External Inquiry* tidak mengubah ILF atau EIF, hanya mengambil data untuk ditampilkan.

*Internal Logical File* (ILF) adalah kelompok data atau kelompok informasi kontrol yang digunakan dalam aplikasi. Peran utama ILF yaitu menyimpan data yang dipelihara oleh satu atau lebih proses dalam aplikasi, contohnya tabel, file dan informasi kontrol seperti *user preferences*. *External Interface File* (EIF) adalah kelompok data berelasi atau informasi kontrol yang dirujuk oleh aplikasi, tapi dipelihara oleh aplikasi lain. Sebuah *External Interface File* yang dihitung untuk sebuah aplikasi harus merupakan *Internal Logical File* di aplikasi lain.

Setiap bagian dari tipe fungsi kemudian diklasifikasikan berdasarkan 3 (tiga) tingkatan kompleksitas (*simple, average, complex*). Tingkat kompleksitas menentukan bobot yang akan diaplikasikan pada jumlah fungsi untuk menentukan kuantitas *Unadjusted Function Points* (UFP). Adapun kriteria yang menentukan ketiga tingkatan kompleksitas, dapat dilihat pada tabel 2.1.

**Tabel 2. 1 Kriteria Tingkat Kompleksitas**

| Tingkat Kompleksitas | Kriteria |
|---|---|
| Simple | 1. Waktu yang diperlukan untuk pengerjaan sebentar<br>2. Jumlah sub menu sedikit<br>3. Dapat dikerjakan oleh 1 orang |
| Average | 1 . Waktu yang diperlukan untuk pengerjaan rata-rata<br>2 . Jumlah sub menu rata-rata<br>3 . Dapat dikerjakan oleh 1 orang walau memakan waktu agak lama dari seharusnya |
| Complex | 1. Waktu yang diperlukan untuk pengerjaan lama<br>2. Jumlah sub menu banyak<br>3. Harus dikerjakan bersama-sama<br>4. Menggunakan algoritma yang cukup rumit |

### 2.2.6 Manajemen Resiko

Konsep manajemen risiko mulai diperkenalkan di bidang keselamatan dan kesehatan kerja pada era tahun 1980-an setelah berkembangnya teori *accident model* dari ILCI dan juga semakin maraknya isu lingkungan dan kesehatan. Tujuan dari manajemen risiko adalah minimisasi kerugian dan meningkatkan kesempatan ataupun peluang. Bila dilihat terjadinya kerugian dengan teori *accident model* dari ILCI, maka manajemen risiko dapat memotong mata rantai kejadian kerugian tersebut, sehingga efek dominonya tidak akan terjadi. Pada dasarnya manajemen risiko bersifat pencegahan terhadap terjadinya kerugian maupun '*accident'* [3].

Ada beberapa ruang lingkup manajemen resiko Berikut ini adalah langkah-langkah yang dilakukan dalam menajemen resiko ,yaitu [3]:

1 Perencanaan Manajemen Resiko

Perencanaan Manajemen Resiko merupakan proses yang memutuskan tentang pendekatan yang akan dilakukan, dan bagaimana melaksanakan kegiatan manajemen resiko untuk suatu proyek. Pada tahap ini menentukan konteks kegiatan yang akan dikelola resikonya.

Masukan pada ruang lingkup ini adalah sebagai berikut :

a. Faktor-faktor lingkungan organisasi
b. Aset proses organisasi
c. Pernyataan cakupan proyek
d. Rencana manajemen proyek

*2* Identifikasi Resiko

Identifikasi resiko bertujuan mengidentifikasi serta membuat daftar resiko yang mungkin terjadi. Proses identifikasi kejadian ini dilakukan dengan pendekatan diskusi dan wawancara dengan pihak perusahaan yang menghasilkan daftar lengkap resiko. Identifikasi dikelompokan berdasarkan jenis resikonya.

3 Analisis Kemungkinan dan Konsekuensi Resiko

Pada tahap ini, berdasarkan identifikasi risiko yang telah dilakukan sebelumnya, dilakukan pengidentifikasian mengenai probabilitas terjadinyarisiko beserta dampak yang mungkin ditimbulkanjika risiko tersebut terjadi, sehingga akan dihasilkan tingkat kepentingan dari masing-masing risiko.

Penilaian risiko pada dasarnya mengacu pada dua faktor, yaitu kuantitas risiko dan kualitas risiko. Kuantitas risiko terkait dengan berapa banyak nilai, atau dampak, yang rentan terhadap risiko sedangkan kualitas risiko terkait dengan kemungkinan suatu risiko muncul. Tujuan penilaian risiko adalah untuk mendapatkan daftar risiko yang telah dinilai berdasarkan tingkat dampak dan kemungkinan terjadinya.

Hasil penilaian risiko tersebut kemudian dipetakan untuk mengetahui risiko-risiko utama yang harus menjadi menjadi prioritas untuk ditangani. Penilaian probabilitas dari setiap risikodan dampak yang ditimbulkan

dibuat dalam suatu skala yaitu 0 sampai 1, dimana skala tersebut menyatakan tingkatan dari rendah, sedang dan tinggi, seperti dijelaskan pada tabel 2.2.

**Tabel 2. 2 Skala dan Nilai Resiko**

| **Skala** | **Nilai Resiko** |
|---|---|
| 0-0,3 | Rendah |
| 0.4-0,7 | Sedang |
| 0.8-1 | Tinggi |

Pemberian nilai probabilitas dan dampak dari setiap risiko dilakukan dengan berdiskusi dengan pihak perusahaan. dimana nilai terdiri dari suatu skala yaitu 0 sampai 1 yang menyatakan tingkatan dari rendah, sedang dan tingginya probabilitas dan dampak dari masing-masing risiko. Tabel 2.3 berikut ini akan memperlihatkan hasil perhitungan probabilitas dan dampak risiko yang telah didiskusikan berdiskusi pihak perusahaan, beserta hasil perhitungan tingkat kepentingan dari masing-masing risiko (risk exposure). Tingkat kepentingan dari masing-masing risiko dihitung dengan menggunajan rumus (2.7) sebagai berikut:

***Risk Exposure = Probability (Outcome) * Loss(Outcome)  (2.7)***

Dimana :

*Risk Exposure* : Tingkat kepentingan resiko
*Probability(outcome)* : Nilai probabilitas resiko
*Loss(outcome)* : Nilai dampak yang ditimbulkan resiko

**Tabel 2. 3 Contoh Hasil Perhitungan Tingkat Kepentingan Resiko**

| Kode Resiko | Probabilitas | Dampak | Tingkat Kepentingan Resiko |
|---|---|---|---|
| R1 | 0.7 | 0.8 | 0.56 |
| R2 | 0.2 | 0.7 | 0.14 |
| R3 | 0.1 | 0.7 | 0.7 |
| R4 | 0.3 | 0.5 | 0.15 |
| R5 | 0.5 | 0.5 | 0.25 |

4 Mitigasi Resiko

Dalam melakukan penanganan terhadap risiko terdapat empat Alternatif tindakan yang dapat dilakukan, yaitu:

a. Menerima Resiko(*Acceptance*)

Acceptance adalah penerimaan risiko beserta konsekuensinya, yaitu tindakan perusahaan untuk menerima suatu risiko dengan tidak melakukan tindakan berarti yang memerlukan sumber daya yang besar. Tindakan ini biasanya diterapkan pada risiko-risiko yang tingkat risikonya rendah bagi perusahaan, sehingga apabila dilakukan penanganan residual risk menimbulkan biaya yang tidak sebanding dengan keuntungannya.

b. Menghidari Resiko*(Avoidance)*

*Avoidance* adalah tindakan perusahaan untuk tidak melakukan usaha tertentu yang mengandung risiko yang tidak diinginkan. Tindakan ini biasanya diterapkan pada risiko-risiko yang tingkat risikonyatidak dapat diterima oleh perusahaan atau berdampak sangat tinggi bagi perusahaan, Dimana penanganannya akan menimbulkan biaya yang sangat tinggi serta tidak efisien.

c. Mengurangi Resiko*(Mitigation)*

*Mitigation* adalah tindakan perusahaan dengan menggunakan semua sumber daya yang dimilikinya berusaha untuk dapat meminimalkan risiko tanpa menghilangkan peluang perusahaan untuk meraih keuntungan. Tindakanini dapat dilakukan terhadap salah satu dari kedua faktor, yaitu:

a. Mengurangi kemungkinan terjadinya risiko, biasanya dengan melakukan proses perubahan desain dan *engineering*, prosedur *quality assurance* atau audit secara periodik.
b. Mengurangi dampak akibat terjadinya suatu risiko, biasanya diterapkan pada risiko yang berdampak tinggi dan kemungkinannya rendah, antaralain dengan membuat rencana kontinjensi atau rencana evakuasi.

c. Membagi Resiko*(Transfer)*

*Transfer* adalah tindakan perusahaan untuk memindahkan risiko kepada pihak ketiga yang dapat mengelola risiko antara lain melalui kesepakatan kontrak dengan asuransi.

## 2.3 Internet

*Interconnected Network* adalah jaringan global yang terdiri dari ratusan bahkan ribuan komputer termasuk jaringan-jaringan lokal tersebut. Komputer-komputer ini terhubung menjadi satu melalui jaringan telepon. Sistem jaringan internet ini terbentuk tidak kurang dari 40 bahkan 50 juta komputer yang letaknya tersebar diseluruh benua termasuk indonesia.

Dari sisi teknis internet bisa didefenisikan sebagai rajanya jaringan *(notworks oj networks)*. Dan dari sisi yang lain orang juga menyatakan bahwa jaringan internet adalah jaringan tertua. Sedangkan dari segi pengetahuan, internet merupakan sebuah perpustakaan besar dengan segundang informasi-informasi

lengkap, bahkan internet bisa didefenisikan sebagai *shopping center* terbesar didunia bagi orang yang suka berbelanja. Terlepas dari semua istilah/defenisi yang kita berikan terhadap internet itu, hingga saat ini be;um ada kesepakatan resmi yang dapat menjawab pertanyaan, apakah internet itu? Hal ini dikarenakan internet berkembang sangat cepat dan mempunyai banyak defenisi, tergantung dari sudut pandangnya [4].

### *2.3.1 Aplikasi Web*

Aplikasi web merupakan salah satu layanan yang didapat oleh pemakai komputer yang terhubung dengan internet. Web pada awalnya adalah ruang informasi dalam internet, dengan menggunakan teknologi hyperteks, pemakai dituntun untuk menemukan informasi dengan mengikuti link yang disediakan dalam dokumen web yang ditampilkan dalam browser web. Web memudahkan pengguna komputer untuk berinteraksi dengan pelaku internet lainnya dan menelusuri (informasi) di internet. Selain itu web telah diadobsi oleh perusahaan sebagai sebgian dari strategi teknology informasi, karena beberapa alasan seperti akses informasi mudah, setup server lebih mudah, informasi mudah didistribusikan dan bebas platform; informasi dapat disajikan oleh browser web pada standar dokumen berbagai tipe data disajikan [5].

### 2.3.2 Server Web

Server web adalah komputer yang digunakan untuk menyimpan dokumen-dokumen web, komputer ini akan melayani permintaan dokumen web dari kliennya. Browser web seperti explorer ata navigator berkomunikasi melalui jaringan (termasuk jaringan internet) dengan server web, menggunakan HTTP.Browser akan mengirimkan request kepada server untuk meminta dokumen tertentu atau layanan lain yang disediakan oleh server. Server memberikan dokumen atau layanannya jika tersedia juga dengan menggunakan protokol HTTP[5].

### 2.3.3 Browser Web

Browser web adalah software yang digunakan untuk menampilkan informasi dari server web. Software ini telah dikembangkan dengan menggunakan user interface grafis, sehingga pemakai dapat dengan melakukan *‘point dan click’*

untuk pindah antar dokumen. Lynx adalah browser yang masih menggunakan mode teks, yng akibatnya adalah tidak ada gambar yang dapat ditampilkan. Lynx ini ada pada lingkungan DOS dan *nix (keluarga sistem operasi unix). Akan tetapi perkembangan dari browser mode teks ini tidaklah secepat browser web dengan GUI.

Dapat dikatakan saat ini hanya ada 2 browser web GUI yang populer: Internet Eksplorer dan Netscape Navigator. Kedua browser ini bersaing untuk merebut pemakainya, dengan berusaha untuk mendekati standar spesifikasi dokumen HTML yang direkomendasikan oleh W3C [5].

## 2.4 Perancangan *Database*

Merancang *database* merupakan suatu hal yang sangat penting. Kesulitan utama dalam merancang *database* adalah bagaimana merancang dan mengakomodir penyimpanan data yang baik sehingga data tersebut dapat memuaskan keperluan saat ini dan masa mendatang. Konsep *database* merupakan langkah untuk menentukan *database* yang diharapkan dapat mewakili seluruh kebutuhan pengguna [6].

### 2.4.1 Pengertian *Database*

Database adalah kumpulan file - file yang mempunyai kaitan antara satu file dengan file lain sehingga membentuk satu bangunan data untuk menginformasikan satu perusahaan, instansi dalam batasan tertentu.

Basis data sendiri dapat didefinisikan dalam sejumlah sudut pandang seperti [6]:

1. Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.
2. Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redudansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan.
3. Kumpulan *file*/tabel/arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronik.

Prinsip utama basis data adalah pengaturan data/arsip. Tujuan utamanya adalah kemudahan dan kecepatan dalam pengambilan kembali data/arsip. Yang

sangat ditonjolkan dalam basis data adalah pengaturan / pemilahan / pengelompokkan /pengorganisasian data yang akan disimpan sesuai fungsi/jenisnya. Pemilahan /pengelompokkan / pengorganisasian Ini dapat berbentuk sejumlah *file* / tabel terpisah atau dalam bentuk pendefinisian kolom-kolom / *field-field* data dalam setiap *file* / tabel. *Database* dapat diciptakan dan dapat pula ditiadakan. Setiap *Database* umumnya dibuat untuk mewakili sebuah semesta data yang spesifik. Misalnya, *Database* akademik, dan dalam basis data akademik kita dapat menempatkan *file* mahasiswa, file matakuliah, *file* dosen, *file* jadwal, *file* kehadiran, *file* nilai, dan sebagainya.

## *2.5 UML*

Untuk melakukan pemodelan sistem/perangkat lunak, dalam buku ini kita akan menggunakan notasi-notasi UML (*Unified Modeling Language).* Yang akan kita gambarkan secara elektronik. (dengan bantuan komputer) lewat sarana perangkat lunak rational rose. Dengan pemodelan menggunakan pemodelan ini, pengembang (juga perangkat lunak rational rose) dapat melakukan [7]:

1. Tinjauan umum bagaimana arsitektur sistem secara keseluruhan.
2. Penelaahan bagaimana objek-objek dalam sistem saling mingimkan pesan (message) dan saling bekerja sama satu sama lain.
3. Menguji apakah sistem/perangkat lunak sudah berfungsi seperti yang seharusnya.
4. Dokumentasi sistem/perangkat lunak untuk keperluan-keperluan tertentu dimasa yang akan datang.

### *2.5.1 Class Diagram*

*Class Diagram* adalah sebuah spesifikasi yang jika diinstansikan akan menghasilkan sebuah obyek dan merupakan inti dari pengembangan dan desain berorientasi objek.

*Class* menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metode/fungsi). Class diagram menggambarkan struktur dan deskripsi *class*, *package* dan *object* beserta hubungan satu sama lain seperti *containment*, *pewarisan*, *asosiasi*, dan lain-lain. Sebuah class memiliki tiga area pokok [7]:

1. Nama , merupakan nama dari sebuah kelas.
2. Atribut, merupakan properti dari sebuah kelas. Atribut melambangkan batas nilai yang mungkin ada pada obyek dari class.
3. Operasi, adalah sesuatu yang bisa dilakukan oleh sebuah atau yang dapat dilakukan oleh class lain terhadap sebuah class. Atribut dan metoda dapat memiliki salah sifat berikut:
   a. *Private*, tidak dapat dipanggil dari luar class yang bersangkutan.
   b. *Protected*, hanya dapat dipanggil oleh class yang bersangkutan dan anak-anak yang mewarisinya.
   c. *Public*, dapat dipanggil oleh siapa saja.
   d. *Package*, hanya dapat dipanggil oleh instance sebuah class pada paket yang sama.

*Class* juga dapat merupakan implementasi dari sebuah *interface*, yaitu *class abstrak* yang hanya memiliki metode. *Interface* tidak dapat langsung diintansikan, tetapi harus diimplementasikan dahulu menjadi sebuah class. Dengan demikian *interface* mendukung resolusi metoda pada saat *runtime*.

Relasi atau *relationship* merupakan keterhubungan antar kelas yang muncul pada saat sebuah kelas berinteraksi dengan kelas-kelas lainnya.

1. Asosiasi (*Association*)

   Para pemodel menggunakan pemahaman asosiasi adalah pada saat beberapa kelas saling terhubung satu sama lain secara konseptual. Asosiasi juga dapat menjadi lebih kompleks pada saat beberapa kelas terhubung ke satu kelas.
2. *Constraint* pada Asosiasi

   Kadangkala sebuah asosiasi diantara dua *class* harus mengikuti sebuah aturan dan aturan ini bias diletakkan dalam sebuah constraint pada garis asosiasi dan diletakkan dalam kurung kurawal. Bentuk lain dari tipe constraint adalah relasi OR yang ditulis dengan {*or*} dalam garis putus-putus yang menghubungkan dua garis asosiasi.
3. Kelas Asosiasi (*Association Class*)

   Sebuah asosiasi dapat memiliki atribut dan operasi sepertinya halnya sebuah *class*. Sebuah *association class* sebenarnya diperlukan apabila salah satu dari

kelas yang terhubung mempunya sebuah atau beberapa atribut yang tidak layak dimiliki kelas tersebut, karena secara logis atribut tersebut lebih layak dimiliki oleh asosiasi yang menghubungkan kedua kelas tersebut.

4. Multiplisitas (*Multiplicity*)

   Multiplisitas atau kardinalitas adalah jumlah banyaknya objek sebuah class yang berelasi dengan sebuah objek lain pada class lain yang berasosiasi pada class tersebut.

### *2.5.2 Use Case Diagram*

*Use case* adalah deskripsi fungsi dari sebuah sistem dari perspektif / sudut pandang para pengguna sistem. *Use case* mendefenisikan "apa" yang dilakukan oleh sistem dan elemen-elemennya saling berinteraksi. *Use case* bekerja dengan menggunakan "*scenario*", yaitu deskripsi urutan-urutan langkah yang menerangkan apa yang dilakukan pengguna terhadap sistem maupun sebaliknya.

Fungsi dari diagram us*e case* adalah menggambarkan fungsionalitas yang diharapkan dari sebuah sistem, dengan menekankan aspek-aspek apa yang dilakukan sistem bukan bagaimana sistem melakukannya. Sebuah *use case* merepresentasikan sebuah interaksi antara *actor* dengan sistem [7].

### *2.5.3 Activity Diagram*

*Activity diagram* adalah salah satu cara untuk memodelkan event-event yang terjadi dalam suatu *use case*. Diagram ini juga dapat diartikan dengan sejumlah teks. Namun, penggunaan teks kadang terlalu sulit dipahami; terutama jika aliran-aliran event berbelit-belit dan memiliki banyak alternatif. Dalam hal yang terakhir ini, seringkali *activity diagram* yang bersifat grafis, lebih mudah dimengerti.

*Activity diagram* digunakan untuk memodelkan aspek-aspek dinamis dari sistem. Activity diagram secara esensial mirip diagram alir *(flowchart)*, memperlihatkan aliran kendali dari suatu aktifitas ke aktifitas lainnya [7].

### *2.5.4 Sequence Diagram*

*Sequence diagram* adalah *interaction diagram* yang memperlihatkan event-event yang berurutan sepanjang berjalannya waktu; kita membaca diagram ini dari atas kebawah. Masing-masing *use case* akan memiliki beberapa aliran

alternatif. Masing-masing *sequence diagram* akan menggambarkan aliran-aliran pada suatu *use case*. Kita dapat membada diagram ini dengan melihat pada objek-objek dan pesan-pesan. Objek-objek yang berperan dalam aliran diperlihatkan pada kotak persegi panjang yang melintas pada bagian atas diagram [7].

### *2.5.5 Component Diagram*

Komponen perangkat lunak adalah bagian fisik dari sebuah sistem yang menetap di komputer. komponen merupakan implementasi *software* dari sebuah *class*. Komponen bisa berupa tabel, file data, file exe, file DLL, dokumen dan lain-lain.

*Component diagram* mengandung komponen, *interface* dan *relationship*. Komponen diagram ini digunakan pada saat anda ingin memecah sistem menjadi komponen-komponen dan ingin menampilkan hubungan-hubungan mereka dengan antarmuka atau pemecahan komponen menjadi struktur yang lebih rendah [7].

## *2.6 Tools*

Pada sub bab ini akan menjelaskan peralatan-peralatan yang dibutuhkan dalam membangun perangkat lunak.

### *2.6.1 Notepad++*

Notepad++ merupakan sebuah *text editor* yang sangat berguna bagi setiap orang dan khususnya bagi para developer dalam membuat program. Notepad++ menggunakan komponen Scintilla untuk dapat menampilkan dan menyuntingan teks dan berkas kode sumber berbagai bahasa pemrograman yang berjalan diatas sistem operasi Microsoft Windows.

Selain manfaat dan kemampuannya menangani banyak bahasa pemrograman, Notepad ++ juga dilisensikan sebagai perangkat *free*. Jadi, setiap orang yang menggunakannya tidak perlu mengeluarkan biaya untuk membeli aplikasi ini karena sourceforge.net sebagai layanan yang memfasilitasi Notepad ++ membebaskannya untuk digunakan.

Bebebapa daftar bahasa program yang didukung oleh Notepad++ adalah C, C++, Java, C#, XML, HTML, PHP, Javascript. Berikut dibawah ini adalah

penjelasan dari bahasa pemrograman dan library yang digunakan dalam pembangunan aplikasi ini [8].

1. PHP

PHP adalah sebuah bahasa pemograman berbasis web yang mempunyai banyak keunggulan dibandingkan dengan bahasa pemograman berbasis web yang lain. PHP merupakan bahasa pemograman yang bersumber Perl. Sedangkan Perl merupakan pengembangan dari bahasa C. Fungsi yang dimiliki oleh PHP sangat lengkap sehingga tidak perlu membuat fungsi sendiri karena daftar fungsi PHP yang lengkap menjadikan baris perintah semakin efisien [9].

2. HTML

HTML (*Hyper Text Markup Language*) adalah sebuah bahasa markup yang digunakan untuk membuat sebuah halaman web dan menampilkan berbagai informasi di dalam sebuah browser internet. HTML juga berupa kode – kode tag yang menginstruksikan browser untuk menghasilkan sesuai yang diinginkan. Sebuah file yang merupakan file HTML dapat dibuka dengan menggunakan browser web seperti Mozilla Firefox atau Microsoft Internet Explorer [8].

3. CSS

*Cassading Style Sheet* atau yang sering disebut CSS. Berisi rangkain intruksi yang menentukan bagaimana suatu text akan tertampil di halaman web. Perancangan desain text dapat dilakukan dengan mendifinisikan fonts (huruf), color (warna), margins (ukuran), latar belakang (background), ukuran font (font size) [9].

4. Jquery

JQuery adalah salah satu library javascript. Dengan JQuery, kita dapat melakukan banyak hal yang tidak bisa dilakukan oleh HTML maupun CSS. Misalya, menampilkan artikel tanpa me-reupload-halaman, memunculkan pop-up di tengah-tengah halaman, menyembunyikan artikel jika diklik, dan sebagainya [9].

### 2.6.2 Xampp

XAMPP adalah sebuah software *web server* apache yang didalamnya dusah tersedia database *server* MySQL dan dapat mendukung pemograman PHP. XAMPP merupakan *software* yang mudah digunakan gratis dan mendukung

instalasi di *linux* dan *windows*. Keuntungan lainnya adalah Cuma menginstal satu kali sudah tersedia *Apache* Web *server*, MySQL *Database Server*,PHP *Support* (PHP 4 dan PHP 5) dan beberapa *module* lainnya [10].