BAB 2
TINJAUAN PUSTAKA

2.1. Tinjauan Tempat Penelitian

Tinjauan umum tempat penelitian merupakan pembahasan mengenai profil, visi dan misi, struktur organisasi serta deskripsi jabatan pada PT Hegar Daya yang akan menjadi tempat dibangunnya sistem ini.

2.1.1. Profil Tempat Penelitian

PT Hegar Daya merupakan salah satu perusahaan konsultan swasta di Bandung yang khususnya bergerak dibidang jasa konsultansi konstruksi dan non konstruksi. Sejak berdirinya PT Hegar Daya secara resmi pada tanggal 25 September 1978, cita dan citra kerja yang ditampilkan adalah mengutamakan keberhasilan pelayanan kerja profesi yang bersifat tepat guna serta tanggung jawab. Dengan pengalaman yang dimiliki perusahaan, berbagai tender dan proyek telah diselesaikan baik konstruksi dan non kontruksi dari beberapa instansi pemerintah. Dari berbagai proyek tersebut, antara lain perencanan rekayasa, perencanaan penataan ruang, pengawasan rekayasa, konsultansi spesialis dan konsultansi lainnya. Selain proyek jasa konsultansi konstruksi PT Hegar Daya menyediaakan jasa konsultansi non konstruksi, antara lain jasa konsultansi perencanaan kepariwisataan, jasa survey, jasa studi, penelitian dan bantuan teknik, jasa khusus, jasa konsultansi manajemen, pengembangan pertanian dan perdesaan, transportasi, telematika, pertambangan dan energi, keuangan, dan pendidikan.

Klien dapat menunjuk PT. Hegar Daya untuk layanan yang bersifat menyeluruh ataupun sebagian, mulai dari studi pendahuluan, perencanaan, perancangan sampai pada pengawasan. Berikut profil PT. Hegar Daya :

Nama Perusahaan : PT Hegar Daya
Alamat : Jalan Taman Saturnus I No. 9 Kota Bandung 40286
Telepon : 022 - 7560416
Email : hegardaya@gmail.com

2.1.2. Visi dan Misi Tempat Penelitian

PT Hegar Daya memiliki Visi dan Misi sebagai berikut :

a. Visi

Menjadi perusahaan konsultan jasa konstruksi (engineering dan manajemen konstruksi) dan non konstruksi (studi kelayakan, lingkungan hidup dan manajemen non konstruksi)

b. Misi

“Memberikan pelayanan yang terbaik” dilandasi oleh kesadaran perlunya memadukan perencanaan proyek pembangunan dengan pertimbangan kelestarian lingkungan sehingga tercipta kegiatan pembangunan yang berkelanjutan dan berwawasan lingkungan.

2.1.3. Logo Tempat Penelitian

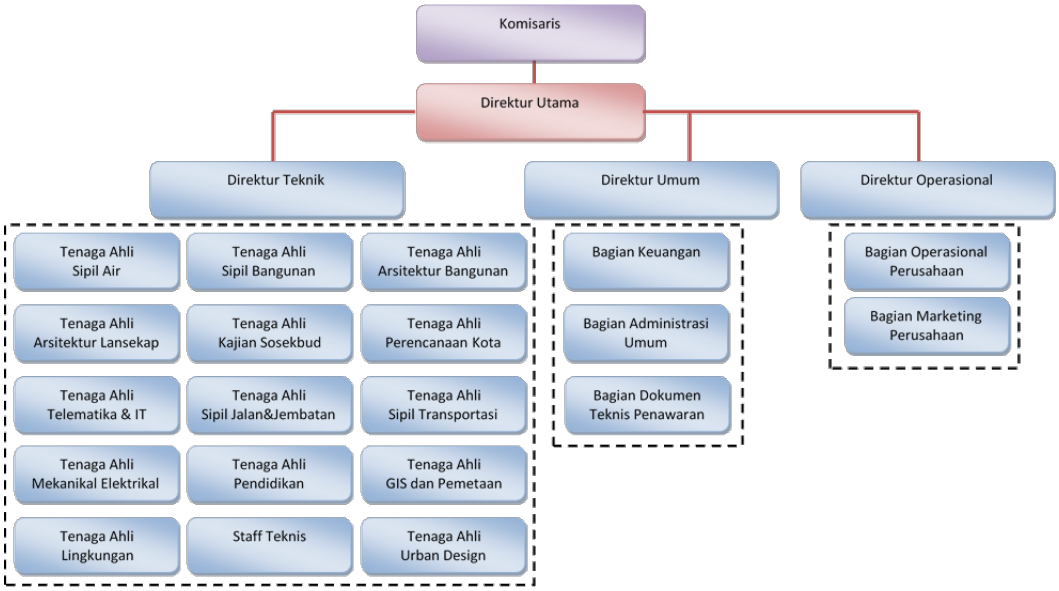
Logo PT. Hegar Daya dapat dilihat pada gambar 2.1.



Gambar 2. 1 Logo PT. Hegar Daya

2.1.4. Struktur Organisasi Perusahaan

Setiap organisasi mempunyai struktur yang berbeda disesuaikan dengan karakteristik masing-masing organisasi tersebut. Dengan adanya struktur organisasi setiap anggota akan mengetahui dimana posisinya sehingga mereka dapat melakukan aktifitas sesuai dengan tugas, wewenang dan tanggungjawab dari bidang kerja yang dipimpinnya dalam suatu sistem kerjasama. Struktur organisasi perusahaan pada PT Hegar Daya sebagai konsultan dapat dilihat pada gambar 2.2.



Gambar 2. 2 Struktur Organisasi Perusahaan

2.1.5. Deskripsi Pekerjaan

Deskripsi pekerjaan digunakan untuk mengetahui tugas, wewenang, tanggung jawab dari masing-masing bagian. Deskripsi pekerjaan dari struktur organisasi perusahaan pada PT. Hegar Daya adalah sebagai berikut :

1. Komisaris
   a. Melakukan pengawasan dan memberikan nasihat kepada direksi.
   b. Memastikan keberadaan dan kecukupan sumber keuangan.
   c. Mengesahkan anggaran tahunan.
2. Direktur Utama
   a. Memutuskan dan menentukan peraturan dan kebijakan tertingi perusahaan.
   b. Menandatangin surat perjanjian kontrak pekerjaan dan surat-surat penawaran keikutsetaan dalam proses dokumen lelang.
   c. Mengontrol kinerja setiap bagian
3. Direktur Teknik
   a. Menguasai detail dan spesifikasi teknis dokumen penawaran sebagai acuan dalam menyusun personil tenaga ahli yang terlibat dalam pekerjaan sesuai dengan dokumen pemilihan.

b. Menyiapkan detail materi penyusunan Rencana Anggaran Biaya (RAB) proyek.

4. Direktur Umum
   a. Membuat laporan keuangan dari setiap pekerjaan yang telah selesai dan melaporkannya kepada direktur utama.
   b. Mengatur, membuat dan melakuan dokumentasi mengenai urusan administrasi perusahaan.
   c. Memonitoring jadwal proses lelang yang telah diikutin.
5. Direktur Operasional
   a. Memastikan operasional perusahaan berjalan dengan baik.
   b. Melakukan komunikasi dengan dinas terkait proyek.
   c. Menghadiri undangan pembuktian kualifikasi perusahaan dan melakukan klarifikasi legalitas perusahaan ke pokja unit layanan pengadaan sesuai dengan pekerjaan yang telah diikuti.

2.2. Landasan Teori

Subbab ini berisikan teori-teori pendukung yang digunakan dalam proses analisis dan implementasi pada permasalahan yang ada di PT. Hegar Daya.

2.2.1. Pengenalan Sistem Informasi

Ada beberapa hal yang harus diketahui jika berbicara tentang sistem informasi diantaranya adalah pengertian tentang sistem dan pengertian informasi. Di bawah ini adalah penjelasan mengenai masing-masing teori yang berhubungan dengan sistem dan informasi dari beberapa tokoh.

2.2.1.1. Pengertian Sistem

Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan kegiatan atau untuk melakukan sasaran yang tertentu[2].

Pendekatan sistem yang merupakan jaringan kerja dari prosedur lebih menekankan urutan-urutan operasi di dalam sistem. Menurut Richard F. Neuschel suatu prosedur adalah suatu urutan operasi klerikal (tulis-menulis), yang melibatkan beberapa orang di dalam satu atau lebih departemen, yang diterapakan untuk menjamin penanganan yang seragam dari transaksi-transaksi bisnis yang terjadi.

2.2.1.2. Pengertian Informasi

Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi penerimanya[2].

Menurut Gordon B. Davis Informasi adalah data yang telah diolah menjadi suatu bentuk yang penting bagi si penerima dan mempunyai nilai nyata atau yang dapat dirasakan dalam keputusan-keputusan yang sekarang atau keputusan-keputusan yang akan datang[2].

2.2.1.3. Pengertian Sistem Informasi

Sistem Informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengelolaan transaksi harian, mendukung operasi, bersifat manajerial, dan kegiatan strategis dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang dibutuhkan[2].

2.2.1.4. Komponen Sistem Informasi

Sistem informasi terdiri dari komponen - komponen yang disebut blok bangunan *(building block*), yang terdiri dari komponen input, komponen model, komponen *output*, komponen teknologi, komponen *hardware*, komponen *software*, komponen basis data, dan komponen kontrol. Semua komponen tersebut saling berinteraksi satu dengan yang lain membentuk suatu kesatuan untuk mencapai sasaran[3].

1. Komponen *input*

   Input mewakili data yang masuk ke dalam sistem informasi. Input disini termasuk metode-metode dan media yang digunakan untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen dasar.

2. Komponen *model*

   Blok ini terdiri dari kombiasi prosedur, logika dan metode matematik yang akan memanipulasi data input dan data yang tersimpan di basis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang sudah diingginkan.

3. Komponen *output*

   Hasil dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua pemakai sistem.

4. Kompenen Teknologi

   Teknologi adalah *toolbox* dalam sistem informasi, teknologi digunakan untuk menerima *input*, menjalankan *model*, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran, dan membantu pengendalian dari sistem secara keseluruhan.

   a. Komponen *hardware*

      Hardware berperan penting sebagai suatu media penyimpanan vital bagi sistem informasi, yang berfungsi sebagai tempat untuk menampung database atau lebih mudah dikatakan sebagai sumber data dan informasi untuk memperlancar dan mempermudah kerja dari sistem informasi.

   b. Komponen *software*

      Software berfungsi sebagai tempat untuk mengolah, menghitung dan memanipulasi data yang diambil dari hardwere untuk menciptakan suatu informasi.

5. Komponen basis data

   Basis data merupakan kumpulan data yang saling berkaitan dan berhubungan satu dengan yang lain, tersimpan di perangkat keras komputer dan menggunakan perangkat lunak untuk memanipulasinya. Data perlu disimpan dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data di dalam basis data perlu diorganisasikan sedemikian rupa supaya informasi yang dihasilkan berkualitas. Organisasi basis data yang baik juga berguna untuk efisiensi kapasitas penyimpanannya. Basis data diakses atau dimanipulasi menggunakan perangkat lunak yang disebut DBMS (*Database Management System)*.

6. Komponen control

   Banyak hal yang dapat merusak sistem informasi, seperti bencana alam, api, temperatur, air, debu, kecurangan – kecurangan, kegagalan – kegagalan sistem itu sendiri, ketidak efisienan, sabotase dan lain sebagainya. Beberapa pengendalian perlu dirancang dan diterapkan untuk menyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi.

2.2.1.5. Karakteristik Sistem Informasi

Supaya sistem itu dikatakan sistem yang baik memiliki karakteristik yaitu mempunyai komponen, batas sistem (*boundary*), lingkungan luar sistem (*environment*), penghubung sistem (*interface*), masukan sistem (*input*), keluaran sistem (*output*), pengolah sistem, dan sasaran sistem[2]. Di bawah ini merupakan penjelasan dan masing-masing karakteristik tersebut:

1. Komponen

   Suatu sistem terdiri dari sejumlah komponen-komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan. Komponen sistem terdiri dari komponen yang berupa subsistem atau bagian-bagian dari sistem.

2. Batasan sistem (boundary)

   Batasan sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lain atau dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan. Batasan suatu sistem menunjukan ruang lingkup (scope) dari sistem tersebut.

3. Lingkungan luar sistem (environment)

   Lingkungan luar sistem (environment) adalah diluar batas dari sistem yang mempengarui operasi sistem. Lingkungan dapat bersifat menguntungkan yang harus tetap dijaga dan yang merugikan yang harus dijaga dan dikendalikan, kalau tidak akan menggangu kelangsungan hidup dari sistem.

4. Penghubung sistem (interface)

   Penghubung sistem merupakan media penghubung antara satu subsistem dengan subsistem lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari subsistem ke subsistem lain. Keluaran (output) dari subsistem akan menjadi masukan (input) untuk subsistem lain melalui penghubung.

5. Masukan sistem (input)

   Masukan adalah energi yang dimasukkan kedalam sistem, yang dapat berupa perawatan (maintenace input), dan masukkan sinyal (signal input). Maintenace input adalah energi yang dimasukkan agar sistem dapat beroperasi. Signal input

adalah energi yang diproses untuk didapatkan keluaran. Contoh dalam sistem computer program adalah maintenance input sedangkan data adalah signal input untuk diolah menajadi informasi.

6. Keluaran sistem (output)

   Keluaran sistem adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Contoh komputer menghasilkan panas yang merupakan sisa pembuangan, sedangkan informasi adalah keluaran yang dibutuhkan.

7. Pengolah sistem

   Suatu sistem menjadi bagian pengolah yang akan merubah masukkan menjadi keluaran. Sistem produksi akan mengolah bahan baku menjadi bahan jadi, system akutansi akan mengolah data menjadi laporan-laporan keuangan.

8. Sasaran sistem

   Suatu sistem pasti mempunyai tujuan (goal) atau sasaran (objective). Sasaran dari sistem sangat menentukan input yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem.

2.2.2. Pengertian Manajemen

Manajemen adalah suatu ilmu 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[4]. Proses dalam manajemen sifatnya umum dan dapat digunakan dalam berbagai kegiatan yang membutuhkan pengelolaan yang sistematis, terarah serta mempunyai tujuan yang jelas.

Tujuan dari manajemen adalah 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[4].

2.2.2.1. Pengertian 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[4]. Adapun sifat-sifat dari proyek

yaitu memiliki tujuan yang jelas dan unik, berlangsung sementara, membutuhkan sumber daya yang beragam, memiliki sponsor atau pelanggan sebagai penyedia sumber dana agar proyek sesuai dengan sasaran dan tujuan yang telah ditetapkan.

2.2.2.2. Pengertian Manajemen Proyek

Manajemen Proyek adalah penerapan ilmu pengetahuan, keahlian dan keterampilan, cara teknis yang terbaik dan sumber daya yang terbatas, untuk mencapai sasaran dan tujuan yang telah ditentukan agar mendapatkan hasil yang optimal dalam hal kinerja biaya, mutu dan waktu, serta keselamatan kerja[4]. Manajemen proyek adalah ilmu dan seni memimpin dan mengkoordinir sumber daya yang terbatas untuk mencapai tujuan yang efektif dan efisien dengan menggunakan teknik pengelolaan modern untuk mencapai sasaran yang telah ditentukan yaitu lingkup, mutu, jadwal dan biaya, serta memenuhi keingginan para stake holder.

Dalam manajemen proyek, penentuan waktu penyelesaian kegiatan ini merupakan salah satu kegiatan awal yang sangat penting dalam proses perencanaan karena penentuan waktu tersebut akan menjadi dasar bagi perencanaan yang lain yaitu:

1. Penyusunan jadwal (*scheduling*), anggaran (*budgeting*), kebutuhan sumber daya manusia (*manpower planning*), dan sumber organisasi yang lain.
2. Proses pengendalian (*controlling*).

   Manajemen Proyek meliputi tiga fase, yaitu:

   a. Perencanaan

      Fase ini mencakup penetapan sasaran dan tujuan, serta menentukan kebijakan pelaksanaan operasional anggaran biaya dan sumber daya.

   b. Penjadwalan

      Fasa ini menghubunkan orang, uang dan bahan untuk kegiatan khusus dan menghubungkan masing-masing kegiatan satu dengan yang lainnya.

   c. Pengendalian

      Perusahaan mengawasi sumber daya, biaya, kualitas, dan anggaran. Perusahaan juga merevisi atau mengubah rencana dan menggeser atau

mengelola kembali sumber daya agar dapat memenuhi kebutuhan waktu dan biaya

Adapun tujuan dari manajemen proyek diantaranya adalah sebagai berikut:

1. Tepat waktu (*on time*)
   yaitu waktu atau jadwal yang merupakan salah satu sasaran utama proyek, keterlambatan akan mengakibatkan kerugian, seperti penambahan biaya, kehilangan kesempatan produk memasuki pasar.
2. Tepat anggaran (*on budget*)
   yaitu biaya yang harus dikeluarkan sesuai dengan anggaran yang telah ditetapkan.
3. Tepat spesifikasi (*on specification*)
   dimana proyek harus sesuai dengan spesifikasi yang telah ditetapkan.

2.2.3. 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[4].

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*), *Relative complexity adjustment factors* dan *Function Points*.

1. *Unadjusted Function Points*(UFP)

Unadjusted 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[5].

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)[5].

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[5]. 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 dan pada konstanta bobot tiap tingkatan dapat di lihat pada tabel 2.1.

Tabel 2. 1 Kriteria Tingkatan Kompleksitas

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

Tabel 2. 2 Konstanta Bobot Level Kompleksitas

Komponen Sistem	**Level Kompleksitas**									**Total CFP**
	Simple			***Average***			***Complex***			
	A	B	AxB=C	D	E	DxE=F	G	H	DxH=I	CFP = C+F+I
Exsternal inputs(EI)	?	3	?	?	4	?	?	6	?	?
Exsternal Output(EO)	?	4	?	?	5	?	?	7	?	?
Internal Logical Files (ILF)	?	7	?	?	10	?	?	15	?	?
External interface Files (EIF)	?	5	?	?	7	?	?	10	?	?
External Inquiry (EI)	?	3	?	?	4	?	?	6	?	?
									Total	?

Keterangan :

- Kolom A , D dan G adalah jumlah inputan pada setiap proses.
- Kolom B, E dan H adalah nilai bobot dari setiap komponen level kompleksitas, nilai tersebut adalah konstanta yang dibuat oleh *Function Point International User Group*(IFPUG).

- Kolom C, F dan H adalah point yang di dapat dari hasil perkalian jumlah inputan dan bobot setiap komponen.

2. *Relative complexity adjustment factors (RCAF)*

Setelah mengakumulasi UFP kemudian menghitung akumulasi RCAF. RCAF dihitung berdasarkan pada keseluruhan kompleksitas sistem dengan menggunakan 14 *General System Characteristics*(GCS), dimana nilai masing-masing GCS berskala 0(nol) sampai 5(lima). Skala 0(nol) menunjukan tidak adanya pengaruh dan skala 5(lima) penunjukan adanya pengaruh yang luas tehadap keseluhan proyek. Pemberian nilai dari masing-masing GSC dilakukan secara subyektif, maka dari itu untuk menentukan pemberian pada bobot pada proses ini dibutuhkan orang yang sudah berpengalaman atau sudah pernah menangani aplikasi dengan spesifikasi sistem hampir mirip. Tabel 2.3 memperlihatkan 14 *General System Characteristics* yang akan diberikan penilaian. Penilaian Kompleksitas memiliki skala 0 s/d 5 yaitu : 0 = Tidak berpengaruh, 1 = Insidental, 2 = Moderat, 3 = Rata-rata, 4 = Signifikan, 5 = Essential.

Tabel 2. 3 14 General system characteristic

General System Characteristic	**Bobot**
Tingkat kompleksitas komunikasi data	[0/1/2/3/4/5]
Tingkat kompleksitas pemrosesan data	[0/1/2/3/4/5]
Tingkat kompleksitas performasnce	[0/1/2/3/4/5]
Tingkat kompleksitas konfigurasi	[0/1/2/3/4/5]
Tingkat frekuensi penggunaa software	[0/1/2/3/4/5]
Tingkat frekuensi input data	[0/1/2/3/4/5]
Tingkat kemudahaan penggunaan bagi *user*	[0/1/2/3/4/5]
Tingkat frekuensi update data	[0/1/2/3/4/5]
Tingkat kompleksitas prosesing data	[0/1/2/3/4/5]
Tingkat kemungkinan penggunaan kembali/*reusable* kode program	[0/1/2/3/4/5]
Tingkat kemudahan dalam instalasi	[0/1/2/3/4/5]
Tingkat kemudahan operasional *software*(*backup*,*recovery*,dsb)	[0/1/2/3/4/5]
Tingkat *software* dibuat untuk multi organisasi/perusahaan/klien	[0/1/2/3/4/5]
Tingkat kompleksitas dalam mengikuti perubahan/fleksibel	[0/1/2/3/4/5]
Total RCAF	**?**

Karakteristik diatas merupakan ketetapan atau konstanta yang dibuat oleh *Function Point Internasional User Group* (IFPUG).

3. *Function Point (FP)*

Function Point adalah proses melakukan perhitungan untuk mendapat nilai *Function point* dari *software* yang akan dibangun. FP didapat dari perkalian CFP dan RCAF dengan rumus :

$$FP = CFP*(0.65+(0.01*RCAF) \quad \textbf{. . . (2.1)}$$

Function *Point* (FP)

Keterangan :

Angka 0.65 dan 0.01 adalah ketetepan atau konstanta yang dibuat oleh *Function Point Internasional User Group* (IFPUG).

2.2.4. Metode *Earned Value Management*(EVM)

Metode EVM atau metode Nilai Hasil merupakan suatu metode pengendalian kinerja proyek yang lebih progresif. Metode ini bisa memberikan informasi mengenai posisi kemajuan proyek dalam jangka waktu tertentu serta dapat memperkirakan progres proyek pada periode selanjutnya baik dalam hal biaya maupun waktu penyelesaian proyek[4].

Metode ini menggunakan kurva S sebagai tampilan informasi dengan sumbu X menunjukan durasi proyek dan sumbu Y untuk menyatakan kumulatif biaya. Dimana kelebihan dari metode ini adalah sebagai berikut[4]:

1. Tampilan informasi metode *earned value* lebih progresif dibandingkan kurva S konvensional.
2. Metode ini bisa memprediksi kerugian biaya dan waktu berdasarkan progres kerja yang cenderung lambat, sehingga tambahan durasi proyek dan biaya akhir dapat dihitung dengan pendekatan sistematis.
3. Informasi dari predikisi Biaya penyelesaian akhir dan prediksi waktu penyelesaian akhir dapat digunakan untuk melakukan tindakan koreksi berupa mempercepat progress proyek dengan pertukaran biaya dan waktu atau dengan penambahan tenaga kerja atau lembur serta penjadwalan kembali sumber daya yang misalnya tenaga kerja, peralatan, serta material.

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

1. *Planned Value*(PV)

   PV atau yang dahulu di kenal dengan *budgeted Cost for Work Scheduled*(BCWS) merupakan biaya yang dianggarkan untuk pekerjaan yang dijadwalkan dalam suatu periode tertentu dan ditetapkan dalam anggaran. Diperoleh dengan mengalikan persentase progres rencana dengan anggaran total proyek (BAC)[6].

   PV = (%progres rencana) x BAC **. . . (2.2)**

   ***Planned Value*(PV)**

2. *Earned Value*(EV)

   EV atau *Budgeted Cost of Word Performed* (BCWP) merupakan biaya yang dianggarkan untuk pekerjaan yang telah selesai dilaksanakan. Diperoleh dengan mengalikan antara persentase progress yang telah dilaksanakan dengan anggaran total proyek(BAC)[6].

   EV = (%progres aktual) x BAC **. . . (2.3)**

   ***Earned Value*(EV)**

3. *Cost Variance* (CV)

   CV merupakan selisih antara nilai yang diperoleh setelah menyelesaikan paket-paket pekerjaan dengan biaya aktual yang terjadi selama pelaksanaan proyek.

   CV = EV – AC **. . . (2.4)**

   ***Cost Variance*(CV)**

4. *Schedule Variance* (SV)

   SV digunakan untuk menghitung penyimpangan antara EV dengan PV

   SV = EV – PV **. . . (2.5)**

   ***Schedule Variance*(SV)**

5. *Cost Performance Index* (CPI)

Merupakan faktor efisiensi biaya yang telah dikeluarkan dapat diperlihatkan dengan membandingkan nilai pekerjaan yang secara fisik telah iselesaikan(EV) dengan biaya yang telah dikeluarkan dalam periode yang sama (AC)[6].

$$CPI = EV / AC \quad (2.5) \qquad \textbf{. . . (2.6)}$$

***Cost Permormance Index*(CPI)**

6. *Scedule Performance Index* (SPI)

   Merupakan faktor efisiensi kinerja dalam menyelesaikan pekerjaan yang diperlihatkan dengan perbandingan antara nilai pekerjaan yang secara fisik telah diselesaikan (EV) dengan rencana pengeluaran biaya yang dikeluarkan berdasarkan rencana pekerjaan(PV)[6].

$$CPI = EV / PV \qquad \textbf{. . . (2.7)}$$

***Schedule Performance Index*(SPI)**

7. Prediksi biaya penyelesaian akhir (EAC)

   Predikisi biaya penyelesaian akhir proyek dapat dihitung dengan perbandingan antara Total anggaran(BAC) dengan CPI.

$$EAC = BAC/CP \qquad \textbf{. . . (2.8)}$$

Prediksi biaya penyelesaian akhir(EAC)

8. Prediksi waktu penyelesaian akhir (ETC)

   Prediksi waktu penyelesaian akhir dapat dihitung dengan perbandingan antara durasi proyek(OD) dengan SPI

$$ETC = OD/SPI \qquad \textbf{. . . (2.9)}$$

Prediksi waktu penyelesaian akhir (ETC)

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 segudang 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 belum 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[7].

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[7].

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[7].

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[7].

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[8].

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[8]:

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. Model Analisis

Model analisis yang digunakan untuk membangun sistem informasi manajemen proyek adalah sebagai berikut :

2.5.1 Flowmap

Flowmap merupakan diagram alir yang menunjukan aliran suatu dokumen, aliran data fisik entitas sistem inforamsi dan kegiatan operasi yang berhubungan dengan sistem informasi. Penggambaran biasanya diawali dengan mengamati dikumen apa yang menjadi media data atau informasi dan selanjutnya ditelusuri bagaimana dokumen termasuk ke bagian entitas nama dokumen tersebut, proses apa yang terjadi terhadap dokumen tersebut dan seterusnya[9].

2.5.2 Entity Relationship Diagram (ERD)

Entity Relationship Diagram atau biasa dikenal dengan diagram E-R secara grafis menggambarkan isi sebuah *database*. ERD terbagi atas tiga komponen, yaitu entitas (*entity*), atribut (*atribute*), dan relasi atau hubungan (*relation*). Secara garis besar entitas merupakan dasar yang terlibat dalam sistem. Atribut atau field

berperan sebagai penjelas dari entitas, dan relasi atau hubungan menunjukkan hubungan yang terjadi antara dua entitas [9].

1. Entitas (*Entity*)

   Entitas (*Entity*) menunjukan objek-objek dasar yang terkait di dalam sistem. Objek dasar dapat berupa orang, benda atau hal lain yang keterangannya perlu disimpan dalam basis data.

2. Atribut (*Atribute*)

   Atribut sering juga disebut sebagai properti (*property*) merupakan keterangan-keterangan yang terkait pada sebuah entitas yang perlu disimpan sebagai basis data. Atribut berfungsi sebagai penjelas sebuah entitas.

3. Relasi (*Relation*)

   Relasi atau hubungan adalah kejadian atau transaksi yang terjadi di antara dua entitas yang keterangannya perlu disimpan dalam basis data.

2.5.3 Diagram Konteks

Diagram konteks adalah diagram yang terdiri dari suatu proses dan menggambarkan ruang lingkup suatu sistem. Diagram konteks merupakan level tertinggi dari DFD yang menggambarkan seluruh input ke sistem atau output dari sistem. Diagram konteks akan memberi gambaran tentang keseluruhan sistem. Sistem dibatasi oleh *Boundary* (dapat digambarkan dengan daris putus). Dalam diagram konteks hanya ada satu proses. Tidak boleh ada *store* di dalam diagram konteks[9].

2.5.4 Data Flow Diafram (DFD)

DFD merupakan alat perancangan sistem yang berorientasi pada alur data dengan konsep dekomposisi yang dapat digunakan untuk menggambaran analisis maupun rancangan sistem yang mudah dikomunikasikan kepada pemakai maupun pembuat program. Bagian dari DFD adalah sebagai berikut [9]:

1. Arus Data (*Data Flow*)

   Arus data ini menunjukan arus data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem.

2. Proses

Proses adalah kegiatan yang dilakukan oleh manusia, mesin atau komputer dari hasil arus data yang masuk ke dalam proses untuk dihasilkan arus data yang akan keluar dari proses.

3. Kesatuan Luar (*External Entity*)

   Kesatuan luar merupakan kesatuan di lingkungan luar sistem yang dapat berupa manuasia, organisasi atau sistem lain yang akan memberikan masukan (*input*) atau menerima kaluaran (*output*) dari sistem.

4. *File*

   Kumpulan data yang disimpan dengan cara tertentu. Data yang mengalir disimpan dalam *file*. Aliran data di-*update* atau ditambahkan kedalam *file*.

2.5.5 Kamus Data

Kamus data atau data *dictionary* adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. Dengan menggunakan kamus data, analisis sistem dapat mendefinisikan data yang mengalir di sistem dengan lengkap, pada tahap perancangan sistem, kamus data dapat digunakan untuk merancang *input, output,* dan merancang *database program.* Kamus data dibuat berdasarkan arus data yang ada [9].

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[10].

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[11].

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[10].

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) [11].

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 [11].

2.6.2 Xampp

XAMPP adalah program aplikasi pengembang yang berguna untuk pengembangan website berbasis PHP dan MySQL. Software XAMPP dibuat dan dikembangkan oleh *Apache Friends*[12]. Perangkat lunak komputer ini memiliki kelebihan untuk bisa berperan sebagai *server* web *Apache* untuk simulasi pengembangan *website*. *Tool* pengembangan web ini mendukung teknologi web populer seperti PHP, MySQL, dan Perl. Melalui program ini, programmer web dapat menguji aplikasi web yang dikembangkan dan mempresentasikannya ke pihak lain secara langsung dari komputer, tanpa perlu terkoneksi ke internet. XAMPP juga dilengkapi fitur manajemen *database* PHPMyAdmin seperti pada *server hosting* sungguhan, sehingga pengembang web dapat mengembangkan aplikasi web berbasis *database* secara mudah. Program XAMPP banyak diaplikasikan dan digunakan oleh kalangan pengguna komputer di bidang pemrograman web. XAMPP merupakan *software* gratis. XAMPP dapat dijalankan di sistem operasi *Windows* 2000/XP/Vista/7/8 dan sistem operasi lain.

Fitur yang ada pada XAMPP adalah sebagai berikut :

1. Apache
2. Cgi-Bin
3. PHP
4. MySQL
5. FTP
6. PHP MyAdmin

2.7 Konsep Pengujian Perangkat Lunak

Ada beberapa jenis dan strategi dalam pengujian perangkat lunak, yang semuanya itu memiliki satu tujuan yang sama, yaitu meningkatkan kepercayaan diri pengembang perangkat lunak terhadap fungsi-fungsi perangkat lunaknya. Beranjak dari tujuan ini, suatu perangkat lunak dapat diuji untuk menerima berbagai perlakuan dalam rangka pengujian. Di antaranya adalah dengan melakukan tindakan-tindakan yang mungkin dilakukan pengguna perangkat lunak. Juga, perhitungan terhadap keandalan perangkat lunak selama operasi dilakukan.

Berdasarkan standar IEEE, pengujian perangkat lunak memiliki pengertian aktivitas yang dilakukan untuk mengevaluasi kualitas produk dan untuk mengembangkannya dengan mengidentifikasi kelemahan dan permasalahan yang tejadi. Definisi secara umum adalah "*Software testing consist of the dynamic verification of the behavior of program on a finite set of test cases, suitably selected from the usually infinite executions domain, againts the expected behavior"(*Pengujian perangkat lunak terdiri dari verifikasi dinamis perilaku program pada sekumpulan kasus-kasus pengujian yang terbatas,pada umumnya dipilih dengan tepat dari domain eksekusi yang tak terbatas, dan berlawanan dengan perilaku yang diharapkan).

Pengujian perangkat lunak adalah proses untuk menjalankan sebuah program atau sistem untuk mencari kesalahan. Atau, pengujian perangkat lunak melibatkan semua kegiatan yang bertujuan untuk mengevaluasi atribut atau kemampuan dari sebuah program atau sistem dan menentukan bahwa pengujian perangkat lunak bertemu dengan hasil yang diperlukan. Perangkat lunak ini tidak berbeda dengan proses fisik lainnya yang diterima, dengan masukan yang diterima dan keluaran yang dihasilkan [13].

2.7.1 Pengujian *White box*

Pengujian *White Box*, yang kadang-kadang disebut pengujian *Glass box*, adalah metode desain *test case* yang menggunakan struktur kontrol desain prosedural untuk memperoleh *text case* . Dengan menggunakan metode pengujian *white box*, perekayasa sistem dapat melakukan *test case* yang [14] :

1. Memberikan jaminan bahwa semua jalur independen pada suatu modul telah digunakan paling tidak satu kali.
2. Menggunakan semua keputusan login pada sisi *true* dan *false.*
3. Mengeksekusi semua *loop* pada batasan mereka dan pada batas operasional mereka.
4. Menggunakan struktur data internal untuk menjamin validitasnya

2.7.2 Pengujian *Black Box*

Pengujian *black box* berfokus pada persyaratan *fungsional* perangkat lunak. Dengan demikian, pengujian *black box* memungkinkan perekayasa perangkat lunak

mendapatkan serangkaian kondisi input yang sepenuhnya menggunakan semua persyaratan *funsgional* untuk suatu program. Pengujian *black box* bukan merupakan *alternatif* dari teknik *white box*, tetapi merupakan pendekatan komplementer daripada metode *white box*. Pengujian *black box* berusaha menemukan kesalahan dalam kategori sebagai berikut :

1. Fungsi-fungsi yang tidak benar atau hilang.
2. Kesalahan *interface*.
3. Kesalahan dalam struktur data atau akses database eksternal.
4. Kesalahan kierja.
5. Inisialisasi dan keselahan terminasi.

Tidak seperti pengujian *white box*, yang dilakukan pada saat awal proses pengujian, pengujian *black box* cenderung diaplikasikan selama tahap akhir pengujian[14]. Yaitu dengan pengujian alfa dan pengujian beta, dimana pengertiannya sebagai berikut :

a. Pengujian Alfa

   Pada jenis pengujian ini, pengguna akan diundang ke pusat pengembang. Pengguna akan menggunakan aplikasi dan pengembang akan mencatat setiap masukan atau tindakan yang dilakukan oleh pengguna. Semua jenis perilaku yang tidak normal dari sistem dicatat dan dikoreksi oleh para pengembang.

b. Pengujian Beta

   Pada jenis pengujian ini, perangkat lunak didistribusikan sebagai sebuah versi beta dengan pengguna yang menguji aplikasi di situs mereka. Pengecualian/cacat yang terjadi akan dilaporkan kepada pengembang. Pengujian beta dilakukan setelah pengujian alfa. Versi perangkat lunak yang dikenal dengan sebutan versi beta dirilis untuk pengguna yang terbatas di luar perusahaan. Perangkat lunak dilepaskan ke kelompok masyarakat agar dapat memastikan bahwa perangkat lunak tersbut memiliki bebrapa kesalahan atau *bug*.

2.8 Acuan Penelitian

Dalam skripsi ini terhadap artikel penelitian hasil pemikiran yang menjadi referensi skripsi ini. Adapun hal tersebut dapat dilihat pada tebel 2.4.

Tabel 2. 4 Acuan Penelitian

Judul Artikel	Sistem Informasi Manajemen Proyek di PT. Quicksteps Sarana Solusindo
Penulis	Husnul Hamidi
Judul Jurnal/Proceeding	Jurnal Ilmiah Komputer dan Informatika (KOMPUTA) / ISSN 2089-9033
Tahun Terbit	2016
Rangkuman	PT. Quickstepps Sarana Solusindo(QSS) merupakan sebuah perusahaan yang bergerak dibidang Teknologi Informasi (TI) dan Support System. Saat ini PT.QSS mendapat kesempatan untuk mengerjakan proyek Sistem Informasi Akuntansi. Metode Function Point digunakan untuk melakukan perhitungan estimasi tingkat kompleksitas. Metode Earned Value Management(EVM) digunakan untuk melakukan pengendalian waktu dan biaya proyek, EVM digunakan karena metode ini mampu mengintegrasikan waktu dan biaya sehingga bisa mengungkapkan kinerja kegiatan. Sedangkan untuk melakukan penilaian dengan menggabungkan peluang munculnya risiko dan dampaknya digunakan metode metode Probality Impact Matrix(PIM).
Judul Artikel	Pengukuran Volume Perangkat Lunak dengan Metode Function Point
Penulis	Wawa Wikusna
Judul Jurnal/Proceeding	InSearch, Universitas Informatika dan Bisnis Indonesia – 1 of 9 / ISS 2085-7993 – 1 of 9
Tahun Terbit	2012
Rangkuman	Mengukur volume perangkat lunak bertujuan untuk membantu perhitungan, kontrol kualitas, perkiraan produktivitas, & kontrol proyek, serta untuk membantu mengambil keputusan taktis pada saat proyek sudah berjalan.Dengan demikian manajer proyek dapat memanaj pengembangan Aplikasi Konseling untuk Siswa Sekolah Menengah Tingkat Atas dengan baik. Aplikasi Konseling untuk Siswa Sekolah Menengah Tingkat Atas dibangun melibatkan tim yang terdiri dari analis sistem, programmer, dan pengguna perangkat lunak, dan sumber daya yang dibutuhkan oleh tim.

Tabel 2. 5 Acuan Penelitian (Lanjutan)

	Betapa pentingnya aplikasi ini bagi kegiatan bimbingan dan konseling siswa SMA serta kualitas aplikasi yang hasilkan harus terjamin, maka proyek pengembangannya harus terkendali dengan baik. Untuk itulah perlu dilakukan pengukuruan volume Aplikasi Konseling untuk Siswa. Sekolah Menengah Tingkat Atas ini, sebagai dasar bagi pengelola dalam mengendalikan.
Judul Artikel	Perencanaan Sistem Informasi Pengendalian Proyek dengan Metode Earned Value Management (EVM)
Penulis	Zul Fadli
Judul Jurnal/Proceeding	ITS-paper-27702-3110106054-Paper, pp. 1-7
Tahun Terbit	2013
Rangkuman	Pembangunan harus mempunyai arah dan tujuan yang jelas sehingga hasil yang direncanakan dapat dicapai lebih optimal. Pengawasan, evaluasi, dan pengendalian perlu dilakukan terutama yang menyangkut masalah biaya dan waktu pelaksanaan proyek sehingga dapat diketahui kinerja proyek serta memprediksi besarnya biaya dan waktu berakhirnya pelaksanaan proyek jika proyek mengalami keterlambatan penyelesainnya, menggunakan metode Earned Value Management (EVM) dalam mengendalikan waktu dan biaya proyek. Akan tetapi dalam prakteknya pengelolaan proyek konstruksi yang dilakukan kontraktor di Indonesia belum menerapkan EVM sepenuhnya. Terutama penggunaan sistem informasi dalam pengelolaan proyek konstruksi belum optimal termasuk dalam menerapkan metode EVM dalam pengendalian proyek. Oleh karena itu dirancang sistem informasi pengendalian proyek dengan metode EVM ini menggunakan aplikasi Microsoft Visual Studio Ultimate 2010. Melalui perancangan sistem informasi ini, hasil dari output program EVM akan dapat dilakukan validasi dengan hasil perhitungan manual.