

BAB II

LANDASAN TEORI

2.1 Konsep Dasar Sistem

Menurut Gelinas dan Dull (2012 : 13), sistem adalah seperangkat elemen independen yang bersama-sama mencapai tujuan spesifik. Sistem juga dikatakan sebagai kumpulan dari bagian yang saling terintegrasi satu dengan yang lain.

Menurut Fery Wongso (2016) “Sistem adalah kumpulan atau rangkaian komponen-komponen yang saling berhubungan, bekerja sama dan saling berinteraksi untuk mencapai suatu tujuan dengan melalui tiga tahapan input (masuk), proses dan output (keluar)”.

2.1.1 Karakteristik Sistem

Menurut Jeperson Hutahaen (2015:3), agar sistem itu dikatakan sistem yang baik memiliki karakteristik sebagai berikut:

1. Komponen

Suatu sistem terdiri dari sejumlah komponen-komponen yang saling berinteraksi, yang artinya saling bekerja sama dalam bentuk 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 menunjukkan ruang lingkup (*scope*) dari sistem tersebut.

3. Lingkungan luar sistem (*environment*)

Lingkungan luar sistem adalah diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan dapat bersifat menguntungkan yang harusnya tetap dijaga dan yang merugikan yang harus dijaga dan dikendalikan, kalau tidak akan mengganggu kelangsungan hidup dari sistem.

4. Tampilan 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 di masukan kedalam sistem, yang dapat berupa perawatan (*maintenance input*), dan masukan sinyal (*signal input*). *Maintenance input* adalah energi yang dimasukkan agar sistem dapat beroperasi. *Signal input* adalah energi yang diproses untuk didapatkan keluaran. Contoh dalam sistem komputer program adalah *maintenance input* sedangkan data adalah *signal input* untuk diolah menjadi 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 masukan menjadi keluaran. Sistem produksi akan mengolah bahan baku menjadi bahan jadi, sistem akuntansi 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.

2.1.2 Pengertian Informasi

Informasi merupakan proses lebih lanjut dari data yang sudah memiliki nilai tambah. Menurut Sutabri (2012), informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan. Sistem pengolahan informasi akan mengolah data menjadi informasi atau mengolah data dari bentuk tak berguna menjadi berguna bagi yang menerimanya.

2.1.3 Pengertian Sistem Informasi

Terdapat beberapa macam pengertian sistem informasi menurut para ahli, diantaranya yaitu sebagai berikut:

Menurut Laudon dan Laudon (2013), sistem informasi secara teknis merupakan serangkaian komponen yang saling berhubungan yang mengumpulkan, menyimpan, memproses, dan mendistribusikan informasi untuk mendukung pengambilan keputusan dan pengawasan di sebuah organisasi. Sistem informasi juga membantu manajer dan karyawan dalam menganalisis masalah, menggambarkan hal-hal yang rumit, juga menciptakan produk atau inovasi baru.

Sistem informasi berisi informasi-informasi penting berupa, orang, tempat/lokasi, dan hal-hal penting lainnya yang berkaitan dengan organisasi dan lingkungan luar organisasi tersebut.

2.2 Konsep Dasar Proyek

Proyek adalah proses yang membutuhkan sumber daya untuk menghasilkan produk, memiliki siklus hidup, dan memiliki titik awal dan akhir yang jelas.

2.2.1 Pengendalian Proyek

Menurut Husen (2009) pengendalian dapat didefinisikan sebagai usaha yang sistematis untuk menentukan standar yang sesuai dengan sasaran dan tujuan perencanaan, merancang sistem informasi, membandingkan pelaksanaan dengan standar yang telah ditetapkan, menganalisa kemungkinan terjadinya penyimpangan, kemudian melakukan tindakan koreksi yang diperlukan agar sumber daya dapat digunakan secara efektif dan efisien dalam rangka mencapai sasaran dan tujuan.

2.2.2 Manajemen Biaya Proyek

Menurut Biemo W. Soemardi dkk Dalam Suhendar dkk., N.d. (2006) Manajemen biaya proyek (project cost management) adalah suatu pengendalian proyek untuk memastikan bahwa proyek selesai sesuai dengan anggaran biaya yang disetujui. Hal-hal yang perlu dipertimbangkan ketika mengelola biaya proyek adalah sebagai berikut :

1. Perencanaan Sumber Daya

Perencanaan sumber daya adalah proses menentukan sumber daya fisik (manusia, peralatan, bahan) dan kuantitas yang dibutuhkan untuk menyelesaikan

suatu kegiatan proyek. Proses ini erat kaitannya dengan proses penilaian.

2. Estimasi biaya

Estimasi biaya adalah suatu metode yang terlibat dengan menilai biaya aset yang dibutuhkan untuk menyelesaikan usaha. Dengan asumsi usaha selesai pada premis kesepakatan, penting untuk mengakui biaya dinilai dan biaya kesepakatan (kontrak). Perkiraan biaya termasuk menganalisa biaya yang dikeluarkan untuk menyelesaikan proyek. Sedangkan biaya kontak adalah keputusan dari perspektif bisnis, perkiraan biaya yang diperoleh dalam proses penilaian merupakan salah satu faktor yang diperhitungkan saat mengambil keputusan.

3. Penganggaran

Perencanaan anggaran adalah suatu proses yang terlibat dengan menetapkan biaya untuk setiap tindakan dari semua biaya yang dikeluarkan selama siklus penilaian, Hasil dari proses ini adalah cost baseline yang digunakan untuk mengukur kemajuan proyek.

4. Kontrol biaya

Pengontrol biaya dilakukan selama proyek berlangsung untuk mengetahui apakah biaya real proyek menyimpang dari rencana atau tidak. Semua penyebab terjadinya penyimpangan nilai harus didokumentasikan dengan baik sehingga tindakan korektif dapat diambil.

2.3 Cost And Schedule Control System Criteria (C/S-CSC)

2.3.1 Pengertian Metode Cost And Schedule Control System Criteria (C/S-CSC)

Kriteria sistem pengendalian biaya dan jadwal (cost and schedule control system criteria – C/S-CSC) adalah penerapan dari konsep nilai hasil dengan memasukkan dan mengkaitkan unsur-unsur anggaran, pengeluaran, jadwal, nilai hasil, lingkup kerja, dan pelaksana (Soeharto,1999). C/S-CSC dapat digunakan untuk memperkirakan besarnya biaya dan waktu berakhirnya sebuah proyek dengan cara melakukan peninjauan terhadap progress sebuah proyek baik itu harian, mingguan dan bulanan. Estimate All Schedule (EAS) adalah prakiraan total waktu proyek dimana prakiraan tersebut sangat mengacu pada hasil laporan produktivitas aktual selama proyek itu berlangsung.

2.3.2 Analisa Indikator Pada Metode Cost And Schedule Control System Criteria (C/S-CSC)

Analisa Indikator-Indikator Earned Value Ada tiga indikator-indikator dasar yang menjadi acuan dalam menganalisa kinerja dari proyek berdasarkan konsep earned value. Ketiga indikator tersebut adalah:

1. Planned Value (PV) Merupakan anggaran biaya yang dialokasikan berdasarkan rencana kerja yang telah disusun terhadap waktu tertentu. Disebut juga dengan BCWS (Budget Cost of Work Scheduled). PV dapat dihitung dari akumulasi anggaran biaya yang direncanakan untuk pekerjaan dalam periode waktu tertentu.
2. Earned Value (EV) Merupakan nilai yang diterima dari penyelesaian pekerjaan selama periode waktu tertentu. Disebut juga BCWP (Budget Cost of Work

Performed). EV ini dapat dihitung berdasarkan akumulasi dari pekerjaan-pekerjaan yang telah diselesaikan.

3. Actual Cost (AC) Merupakan representasi dari keseluruhan pengeluaran yang dikeluarkan untuk menyelesaikan pekerjaan dalam periode tertentu. Atau disebut juga dengan ACWP (Actual Cost of Work Performed). AC dapat berupa kumulatif hingga periode perhitungan kinerja atau jumlah biaya pengeluaran dalam waktu tertentu.

2.3.3 Analisa Varians

Analisa varians digunakan untuk mengetahui sampai sejauh mana hasil yang diramalkan dari apa yang diperkirakan. Analisa varians terdiri dari :

1. Schedule Variance (SV) Adalah hasil pengurangan dari Earned value(EV) dengan Planned Value(PV). Hasil dari Schedule Variance ini menunjukkan tentang pelaksanaan pekerjaan proyek. Harga SV sama dengan nol ($SV = 0$) ketika proyek sudah selesai karena semua Planned Value telah dihasilkan.

$$SV = EV - PV$$

2. Cost Variance (CV) Adalah hasil pengurangan antara Earned Value(EV) dengan Actual Cost(AC). Nilai Cost Variance pada akhir proyek akan berbeda antara BAC (Budgeted At Cost) dan AC(Actual Cost) yang dikeluarkan atau dipergunakan.

$$CV = EV - AC$$

2.3.4 Analisa Indeks Performansi

Indeks performansi digunakan untuk mengetahui efisiensi penggunaan sumber daya. Analisa Indeks performansi terdiri dari :

1. Schedule Performance Index (SPI) Adalah Faktor efisiensi kinerja dalam menyelesaikan pekerjaan dapat diperlihatkan oleh perbandingan antara nilai pekerjaan yang secara fisik telah diselesaikan (EV) dengan rencana pengeluaran biaya yang dikeluarkan berdasar rencana pekerjaan (PV). Rumus untuk Schedule Performance Index adalah :

$$SPI = EV / PV$$

Dimana,

SPI = 1 : proyek tepat waktu

SPI > 1 : proyek lebih cepat

SPI < 1 : proyek terlambat

2. Cost Performance Index (CPI) Adalah Faktor efisiensi biaya yang telah dikeluarkan dapat diperlihatkan dengan membandingkan nilai pekerjaan yang secara fisik telah diselesaikan (EV) dengan biaya yang telah dikeluarkan dalam periode yang sama (AC). Rumus untuk CPI adalah :

$$CPI = EV / AC$$

Dimana,

CPI = 1 : biaya sesuai rencana

CPI > 1 : biaya lebih kecil/hemat

CPI < 1 : biaya lebih besar/boros

Perkiraan dihitung berdasarkan kecenderungan kinerja proyek pada saat peninjauan dan mengasumsikan bahwa kecenderungan tersebut tidak mengalami perubahan kinerja proyek sampai akhir proyek atau kinerja proyek berjalan konstan. Perkiraan ini berguna untuk memberikan suatu gambaran ke depan kepada pihak kontraktor, sehingga dapat melakukan langkah-langkah perbaikan yang diperlukan.

1. Estimate to Complete (ETC) ETC merupakan prakiraan biaya untuk pekerjaan tersisa, dengan asumsi bahwa kecenderungan kinerja proyek akan tetap (konstan) sampai akhir proyek. ETC untuk progress fisik > 50 % Dimana, BAC (Budget at Completion) adalah biaya total proyek yang telah dianggarkan.

$$ETC = (BAC - EV) / CPI$$

2. Estimate at Completion (EAC) EAC Merupakan prakiraan biaya total pada akhir proyek yang diperoleh dari biaya aktual (AC) ditambahkan dengan ETC.

$$EAC = AC + ETC$$

3. Time Estimated (TE) TE Merupakan waktu perkiraan penyelesaian proyek. Dimana, TE (Time Estimated) : Perkiraan Waktu Penyelesaian ATE (Actual Time Expended) : Waktu yang telah ditempuh OD (Original Duration) : Waktu yang direncanakan

$$TE = ATE + ((OD - (ATE \times SPI)) / SPI)$$

2.3.5 Manfaat Metode Cost And Schedule Control System Criteria (C/S-CSC)

C/S-CSC dapat digunakan untuk memperkirakan besarnya biaya dan waktu berakhirnya sebuah proyek dengan cara melakukan peninjauan terhadap progres sebuah proyek baik itu harian, mingguan dan bulanan.

2.4 Peralatan Pendukung Sistem

Pada penelitian ini penulis menggunakan beberapa peralatan pendukung sebagai penunjang kegiatan, yaitu :

2.4.1 Basis Data

Menurut Kustiyaningsih (2011:146), “Database adalah Struktur penyimpanan data. Untuk menambah, mengakses dan memproses data yang disimpan dalam sebuah database komputer, diperlukan sistem manajemen database seperti MYSQL Server”

2.4.2 Entity Relationship Diagram

Menurut Yanto (2016:32) “ERD adalah suatu diagram untuk menggambarkan desain konseptual dari model konseptual suatu basis data relasional. ERD juga merupakan gambaran yang merelasikan antara objek yang satu dengan objek yang lain dari objek di dunia nyata yang sering dikenal dengan hubungan antar entitas”.

2.4.3 MySQL

Adapun Arief (2022) mengatakan pengertian MySQL adalah suatu jenis database server yang sangat terkenal dan banyak digunakan untuk

membangun aplikasi web yang menggunakan database sebagai sumber dan pengelolaan datanya. Mysql bersifat open source dan menggunakan SQL (Structured Query Language). MySQL biasa dijalankan diberbagai platform misalnya windows Linux, dan lain sebagainya.

2.4.4 PHP

Menurut Madcoms (2011:49), PHP merupakan pemrograman interpreter untuk proses penerjemahan barisan kode mesin yang dimengerti komputer secara langsung saat dijalankan. Dari pernyataan tersebut dapat disimpulkan bahwa PHP adalah bahasa pemrograman yang digunakan secara luas dalam menangani pembuatan dan/atau pengembangan suatu situs web dan dapat digunakan bersamaan dengan HTML.

2.4.5 HTML

Menurut Anhar (2010:40), HTML (*Hypertext Markup Language*) adalah sekumpulan simbol-simbol atau tag-tag yang dituliskan dalam sebuah file untuk menampilkan halaman pada *web browser*. Pada dokumen HTML, kita tidak memerlukan membaca dokumen tersebut berurutan dari atas ke bawah ataupun sebaliknya, tetapi kita dapat menuju topik tertentu secara langsung menggunakan teks penghubung yang akan membawa kita ke suatu topik lain atau dokumen lain. Dokumen ini umumnya berisi informasi atau tampilan (*interface*) aplikasi.

2.4.6 XAMPP

Menurut Madcoms (2011:31), XAMPP merupakan salah satu paket software untuk menginstall *web server* yang disediakan secara gratis. Dengan paket *software* instalasi ini, maka sudah dapat melakukan beberapa instalasi *software*

pendukung *web server*, yaitu Apache, PHP, phpMyAdmin, dan database MySQL. Berfungsi sebagai server yang berdiri sendiri (*localhost*), yang dimana terdiri dari program Apache HTTP Server, MySQL database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMPP sendiri ialah singkatan dari X (empat sistem operasi), Apache, MySQL, PHP, dan Perl. Program ini tersedia dan bebas digunakan untuk umum.

2.4.7 UML (Unified Model Language)

Menurut Nugroho (2010:6), UML (*Unified Modeling Language*) merupakan bahasa pemodelan untuk sistem atau perangkat lunak berorientasi objek. Pemodelan (*modeling*) berfungsi untuk menyederhanakan permasalahan-permasalahan yang kompleks sehingga dapat lebih mudah untuk dipelajari dan dipahami.

2.4.8 Model Diagram UML

Model-model pada diagram UML dapat diklasifikasikan berdasarkan sifatnya, yang antara lain:

1. *Use Case Diagram*

Menurut Setiawan & Khairuzzaman (2017:110), diagram *use case* menyajikan keterkaitan antara *use case* dan *actor*. Aktor disini dapat berupa orang, peralatan atau sistem lain yang berkorelasi dengan sistem yang sedang dibangun. *Use case* menggambarkan fungsionalitas sistem yang harus dipenuhi sistem dari pandangan aktor.

2. *Scenario*

Menurut Munawar (2010:6), skenario pada diagram UML ialah suatu dokumentasi akan kebutuhan fungsional dari sebuah sistem. Form skenario yang ada merupakan penjelasan atau penjabaran penulisan *use case* dari sudut pandang *actor*.

3. *Activity Diagram*

Menurut Sukamto dan Shalahuddin (2014:161), diagram aktivitas menggambarkan suatu aktifitas atau aliran kerja dari sebuah sistem atau proses atau menu yang terdapat pada perangkat lunak. Diagram aktivitas menggambarkan aktivitas oleh sistem dan bukan apa yang dilakukan oleh *actor*.

2.4.9 Balsamiq mockup

Balsamiq Mockups menurut (Faranello, 2012) adalah salah satu software yang digunakan dalam pembuatan desain atau prototyping dalam pembuatan tampilan user interface sebuah aplikasi.

2.5 Metode Pengembangan Perangkat Lunak

2.5.1 Metode Waterfall

Menurut Pressman (2015:42), Model waterfall adalah model Klasik yang bersifat sistematis, berurutan dalam membangun software. Nama model ini sebenarnya Adalah “Linear Sequential Model”. Model ini sering disebut juga dengan “classic life cycle” atau metode waterfall. Model ini termasuk ke dalam model generic pada rekayasa perangkat lunak dan pertama kali diperkenalkan oleh Winston Royce sekitar tahun 1970 sehingga sering dianggap kuno, tetapi merupakan model yang paling banyak dipakai dalam Software Engineering (SE). Model Ini melakukan pendekatan secara sistematis dan berurutan. Disebut

Dengan waterfall karena tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya dan berjalan berurutan.

2.5.2 Tahapan Metode Waterfall

Ian Sommerville (2011) menjelaskan bahwa ada lima tahapan pada Metode Waterfall, yakni Requirements Analysis and Definition, System and Software Design, Implementation and Unit Testing, Integration and System Testing, dan Operational and Maintenance.

1. Analysis

Sebelum melakukan pengembangan perangkat lunak, seorang pengembang harus mengetahui dan memahami bagaimana informasi kebutuhan pengguna terhadap sebuah perangkat lunak. Metode pengumpulan informasi ini dapat diperoleh dengan berbagai macam cara diantaranya, diskusi, observasi, survei, wawancara, dan sebagainya. Informasi yang diperoleh kemudian diolah dan dianalisa sehingga didapatkan data atau informasi yang lengkap mengenai spesifikasi kebutuhan pengguna akan perangkat lunak yang akan dikembangkan.

2. Design

Informasi mengenai spesifikasi kebutuhan dari tahap Requirement Analysis selanjutnya di analisa pada tahap ini untuk kemudian diimplementasikan pada desain pengembangan. Perancangan desain dilakukan dengan tujuan membantu memberikan gambaran lengkap mengenai apa yang harus dikerjakan. Tahap ini juga akan membantu pengembang untuk menyiapkan kebutuhan hardware dalam pembuatan arsitektur sistem perangkat lunak yang akan dibuat secara keseluruhan.

3. Implementation

Tahap implementation and unit testing merupakan tahap pemrograman. Pembuatan perangkat lunak dibagi menjadi modul-modul kecil yang nantinya akan digabungkan dalam tahap berikutnya. Disamping itu, pada fase ini juga dilakukan pengujian dan pemeriksaan terhadap fungsionalitas modul yang sudah dibuat, apakah sudah memenuhi kriteria yang diinginkan atau belum.

4. Testing

Setelah seluruh unit atau modul yang dikembangkan dan diuji di tahap implementasi selanjutnya diintegrasikan dalam sistem secara keseluruhan. Setelah proses integrasi selesai, selanjutnya dilakukan pemeriksaan dan pengujian sistem secara keseluruhan untuk mengidentifikasi kemungkinan adanya kegagalan dan kesalahan sistem.

5. Maintenance

Pada tahap terakhir dalam Metode Waterfall, perangkat lunak yang sudah jadi dioperasikan pengguna dan dilakukan pemeliharaan. Pemeliharaan memungkinkan pengembang untuk melakukan perbaikan atas kesalahan yang tidak terdeteksi pada tahap-tahap sebelumnya. Pemeliharaan meliputi perbaikan kesalahan, perbaikan implementasi unit sistem, dan peningkatan dan penyesuaian sistem sesuai dengan kebutuhan.