

BAB II

LANDASAN TEORI

2.1. Konsep Dasar

Didalam konsep dasar ini, penulis menambahkan beberapa teori dari para ahli yang berkaitan dengan penyusunan tugas akhir ini. Termasuk hal-hal yang berkaitan dengan kepenulisan tugas akhir, diantaranya pengertian dari konsep dasar sistem dan program.

2.1.1. Sistem

Menurut (Tohari, 2014) mengemukakan bahwa “Sistem adalah kumpulan atau himpunan dari unsur atau variabel-variabel yang saling terkait, saling berinteraksi, dan saling tergantung satu sama lain”.

Karakteristik suatu sistem menurut (Tohari, 2014) adalah:

1. Komponen atau elemen (*components*)

suatu sistem terdiri dari komponen-komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan.

2. Batas Sistem (Boundary)

Batas sistem merupakan daerah yang membatasi antara sistem yang satu dengan sistem yang lainnya atau dengan lingkungan luarnya. Adanya batas sistem, maka sistem dapat membentuk suatu kesatuan, karena dengan batas sistem ini, fungsi dan tugas dari subsistem satu dengan yang lainnya berbeda tetapi tetap saling berinteraksi. Dengan kata lain, batas sistem merupakan ruang lingkup atau *scope* dari sistem atau subsistem itu sendiri.

3. Lingkungan luar subsistem (*Environment*)

Lingkungan luar sistem adalah segala sesuatu diluar batas sistem yang mempengaruhi operasi suatu sistem.

4. Penghubung sistem (*interface*)

Penghubung sistem merupakan suatu media (penghubung) antara subsistem dengan subsistem lainnya yang membentuk satu kesatuan, sehingga sumber-sumber daya mengalir dari subsistem yang satu ke subsistem lainnya.

5. Masukan (*input*)

Input adalah energi atau sesuatu yang dimasukkan ke dalam suatu sistem yang dapat berupa masukan yaitu energi yang dimasukkan supaya sistem dapat beroperasi atau masukan sinyal yang merupakan energi yang diproses untuk menghasilkan suatu luaran.

6. Luaran (*output*)

Merupakan hasil dari energi yang diolah dan diklasifikasikan menjadi luaran yang berguna, juga merupakan luaran atau tujuan akhir dari sistem.

7. Pengolah (*process*)

Suatu sistem mempunyai bagian pengolah yang akan mengubah *input* menjadi *output*.

8. Sasaran (*objekive*)

dari sistem sangat menentukan sekali masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem.

Sedangkan menurut (Abdussomad, 2016) mengemukakan bahwa “suatu sistem adalah jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu.”

2.1.2. Informasi

Menurut (Tohari, 2014) mengemukakan bahwa “Informasi merupakan data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan”.

Sedangkan kualitas dari satu informasi tergantung dari tiga hal menurut (Tohari, 2014) adalah:

1. Akurat artinya informasi harus bebas dari kesalahan-kesalahan.
2. Tepat artinya informasi yang datang pada penerima tidak boleh terlambat.
3. *Relevan* artinya informasi harus mempunyai manfaat bagi pemakainnya.

Sedangkan menurut (Mardi, 2014) menyimpulkan bahwa, “informasi adalah data yang diolah kemudian menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya, menggambarkan suatu kejadian (*event*), dan kesatuan nyata (*fact and entity*) serta digunakan untuk pengambilan keputusan”.

2.1.3. Sistem Informasi

Menurut (Puspitasari, 2015) menyimpulkan bahwa “Sistem Informasi adalah sebuah yang bertujuan untuk menyimpan, memproses dan mengkomunikasikan informasi. Sistem informasi digunakan oleh seluruh pengguna yang ada dalam organisasi”.

Menurut Kenneth dan Jane dalam (Sunyoto, 2014) menjelaskan bahwa, “sistem informasi secara teknis dapat didefinisikan sebagai kumpulan komponen yang saling berhubungan, mengumpulkan atau mendapatkan, memproses, menyimpan dan mendistribusikan informasi untuk menunjang pengambilan keputusan dan pengawasan dalam suatu organisasi”.

Bagian dimensi sistem informasi menurut Kenneth dan Jane dalam (Sunyoto, 2014), adalah sebagai berikut:

1. Organisasi

Sistem organisasi merupakan bagian yang tak terpisahkan dari organisasi. Bagi sebuah perusahaan seperti perusahaan pembuat laporan kredit, bisnisnya tidak akan berjalan tanpa sebuah sistem informasi.

2. Teknologi

Teknologi informasi adalah satu dari banyak alat yang digunakan manajer untuk menghadapi perubahan.

3. Manajemen

Tugas manajemen adalah untuk berusaha memahami banyak keadaan yang dihadapi oleh organisasi, mengambil keputusan dan memutuskan rencana kegiatan untuk memecahkan permasalahan organisasi.

2.1.4. Akuntansi

Menurut (Hery, 2015) mengemukakan bahwa “Akuntansi merupakan sebuah sistem informasi yang memberikan laporan kepada para pengguna informasi akuntansi atau kepada pihak-pihak yang memiliki kepentingan terhadap hasil kinerja dan kondisi keuangan perusahaan”.

Pembuatan *record* dan pemrosesan data transaksi akuntansi menurut George M.Scott dalam (Sunyoto, 2014) terdiri atas beberapa tahapan berikut:

1. Mencatat semua rincian transaksi dalam sebuah jurnal buku pencatatan pertama, berbagai fakta tentang transaksi yang dihilangkan dibagian ini biasanya tidak dapat direkam pada tahap selanjutnya.

2. Posting transaksi dalam suatu akun pada buku besar tambahan. Melakukan posting berarti memindahkan atau memasukkan sejumlah uang transaksi.
3. Posting semua buku besar tambahan ke dalam buku besar umum, yang mana dalam sistem manual bagian ini merupakan satu buku besar yang mencatat buku besar tambahan sebagai satu akun dalam sebuah buku.

2.1.5. Sistem Informasi Akuntansi

Menurut (Tresnawati, dkk 2017) mengemukakan “Sistem Informasi Akuntansi adalah komponen organisasi yang dirancang untuk mengolah data keuangan menjadi informasi atau laporan keuangan, yang di tujukan kepada pihak internal maupun eksternal perusahaan”.

Menurut (Mardi, 2014) mengemukakan bahwa, “Sistem Informasi Akuntansi dapat diartikan sebagai kegiatan yang berintegrasi yang menghasilkan laporan dibentuk data transaksi bisnis yang diolah dan disajikan sehingga menjadi sebuah laporan keuangan yang memiliki arti bagi pihak yang membutuhkannya”.

Tiga tujuan sistem informasi akuntansi menurut (Mardi, 2014) yaitu:

1. Guna memenuhi setiap kewajiban sesuai dengan otoritas yang diberikan kepada seseorang (*to fulfill obligations relating to stewardship*).
2. Setiap informasi yang dihasilkan merupakan bahan yang berharga bagi pengambilan keputusan manajemen (*to support decision making by internal decision makers*).
3. Sistem informasi diperlukan untuk mendukung kelancaran operasional perusahaan sehari-hari (*to support the-day-to-day operations*).

2.1.6. Pembelian

Menurut (Heripracoyo, 2014) mengemukakan bahwa “Pembelian merupakan kegiatan pemilihan sumber, pemesanan dan perolehan barang dan jasa sebagai salahsatu aktivitas utama operasi bisnis perusahaan”.

Menurut (Sudrajat, 2016) “Jurnal umum adalah sebuah daftar khusus untuk mencatat setiap aktivitas transaksi yang dilakukan suatu usaha secara kronologis sesuai urutan tanggal ke dalam jumlah yang harus didebet dan dikredit. Jurnal umum merupakan tempat pertama kali untuk mencatat transaksi”.

Adapun jurnal pembelian yang terdiri dari jurnal pembelian secara tunai dan kredit adalah sebagai berikut:

Jurnal Pembelian Secara Tunai

Pembelian

Kas

Pembelian Secara Kredit

Pembelian

Hutang Dagang

2.1.7. Program

Menurut (Kadir, 2014) megemukakan bahwa “Program adalah kumpulan intruksi yang ditujukan untuk komputer agar komputer dapat melakukan tujuan tertentu yang diharapkan oleh pemakai”.

Sedangkan menurut (Sukamto dan Shalahuddin, 2018) menjelaskan bahwa, “Pemrograman terstruktur adalah konsep atau paradigma atau sudut pandang pemrograman yang membagi-bagi program berdasarkan fungsi-fungsi atau prosedur-prosedur yang dibutuhkan program komputer”.

2.1.8. Bahasa Pemrograman Java

Menurut (Rusmayanti, 2014) mengemukakan bahwa “Java merupakan bahasa pemrograman berorientasi objek dan bebas platform dikembangkan oleh SUN Micro system dengan jumlah keunggulan yang memungkinkan java dijadikan sebagai bahasa pengembang enterprise”.

Menurut (Kadir, 2014) mengemukakan bahwa “Java merupakan bahasa yang bersifat multiplatform, mengingat bahasa ini dapat digunakan diberbagai platform, dari windows hingga linux”.

2.1.9. Netbeans IDE 8.2

Menurut (Rusmayanti, 2014) mengemukakan bahwa “*Netbeans* merupakan salah satu IDE yang dikembangkan dengan bahasa pemrograman java. *Netbeans* mempunyai lingkup pemrograman java terintegrasi dalam suatu perangkat lunak yang didalamnya menyediakan pembangunan GUI, *text editor*, *complier*, dan *interpreter*”.

Pada tahun 1997, Rohman Stanek mengkomersialkan IDE NetBeans dan kemudian dibeli oleh Sun Microsystem pada tahun 1999. Pada tahun 2010, Sun (dan NetBeans) diakuisisi oleh Oracle. NetBeans merupakan IDE *open-source* yang dapat digunakan secara gratis. NetBeans mempunyai sekumpulan *software* modul yang dipakai untuk membuat suatu aplikasi .modul merupakan aris Java (Java Archive) yang memuat kelas-kelas Java yang berinteraksidengan NetBeans Open API. Pembuatan program Java dapat menggunakan *tect editor* seperti notepad ++ dan lain sebagainya. Untuk *compile* dan *running* dapat dilakukan melalui *command promt*. Penggunaan aplikasi NetBeans akan memudahkan *developer software* untuk membuat kode program, meng-*compile*, danmenjalankan program.

2.1.10. Basis Data (*Database*)

Menurut (Sukamto dan Shalahuddin, 2018) mengemukakan bahwa “Basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan”.

Menurut (Nurrahman, dkk 2018) menyimpulkan bahwa Basis data dapat dipahami sebagai suatu kumpulan data terhubung (*interrelated data*) yang disimpan bersama-sama dalam suatu media, tanpa mengatap satu sama lain atau tidak perlu satu kerangkaan data”

2.2. Peralatan Pendukung (*Tools System*)

Peralatan pendukung merupakan alat yang digunakan untuk menggambarkan bentuk logika model dari sistem yang akan dibuat. Adapun peralatan yang dipakai penulis sebagai berikut:

2.2.1. Konsep Dasar *Unitified Modelling Language*(UML)

UML muncul karena adanya suatu kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi *software*.

Menurut (Puspitasari, 2017) mengemukakan bahwa “ UML adalah sebuah bahasa yang telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem perangkat lunak”.

Menurut (Abdussomad, 2016) “*UML (Unified Modeling Language)* adalah bahasa pemodelan dalam sistem atau perangkat lunak yang berparadigma berorientasi objek”

Menurut (Sukamto dan Shalahuddin, 2018) menyimpulkan bahwa “UML (*Unified Modeling Language*) adalah salah standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement* (kebutuhan), membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek”.

2.2.2. *Use Case Diagram*

Menurut (Sukamto dan Shalahuddin, 2018) mengatakan bahwa “*Use Case* atau diagram *use case* merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat. *Use Case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu”.

Syarat penamaan *use case* adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada *use case* yaitu pendefinisian apa yang disebut aktor dan *use case* menurut (Sukamto dan Shalahuddin, 2018), adalah:

1. Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.
2. *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.

2.2.3. *Activity Diagram*

Menurut (Tohari, 2014) mengemukakan bahwa “*Activity Diagram* memodelkan *workflow* proses bisnis dan urutan aktivitas dalam sebuah proses”.

Elemen-elemen dari *activity diagram* menurut (Tohari, 2014) meliputi :

1. Status *start* (mulai) dan *end* (akhir)
2. Aktivitas yang merepresentasikan sebuah langkah dalam *workflow*
3. *Transition* menunjukan terjadinya perubahan status aktivitas
4. Keputusan yang menunjukan *alternative* dalam *workflow*.
5. *Synchronization bars* yang menunjukan *subflow parallel*.

6. *Swimlanes* yang merepresentasikan *role* bisnis yang bertanggung jawab pada aktivitas yang berjalan.

Menurut (Sukamto dan Shalahuddin, 2018) mengatakan bahwa “Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak”.

Berikut menurut (Sukamto dan Shalahuddin, 2018) adalah:

1. Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
2. Urutan atau pengelompokan tampilan dari sistem/*user interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan.
3. Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.
4. Rancangan menu yang ditampilkan pada perangkat lunak.

2.2.4. *Sequence Diagram*

Menurut (Tohari, 2014) mengemukakan bahwa “*Sequence diagram* menggambarkan interaksi antara sejumlah objek dalam urutan waktu”.

Banyaknya diagram sekuen yang harus digambar adalah minimal sebanyak pendefinisian *use case* yang memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada diagram sekuen sehingga semakin banyak *use case* yang didefinisikan maka diagram sekuen yang harus dibuat juga semakin banyak. Oleh karena itu untuk menggambar diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu.

Menurut (Sukamto dan Shalahuddin, 2018) mengemukakan bahwa “Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek”.

2.2.5. *Deployment Diagram*

Menurut (Tohari, 2014) menjelaskan bahwa, “Diagram *deployment* menunjukkan tata letak sebuah sistem secara fisik. Diagram ini akan menampilkan bagian-bagian *software* yang berjalan pada bagian-bagian *hardware* yang digunakan untuk mengimplementasikan sebuah sistem dan kethubungan antara komponen-komponen *hardware* tersebut”.

Untuk menggambar diagram *deployment*, ada beberapa hal yang harus diidentifikasi terlebih dahulu menurut (Tohari, 2014) yaitu:

1. Menentukan *node*.
2. Hubungan antar *node*.

Penentuan *node*, diantaranya dapat dilakukan dengan beberapa pedoman menurut (Tohari, 2014) sebagai berikut:

1. *Node* merupakan elemen fisik yang sudah tersedia dalam sistem atau aplikasi. Amatilah dan jadikan elemen fisik sebagai *node* di diagram *deployment*.
2. *Node* mewakili sumber-sumber komputasi.
3. *Node* dilambangkan dengan kubus.

Menurut (Sukamto dan Shalahuddin, 2018) mengemukakan bahwa “Diagram *deployment* atau *deployment diagram* menunjukan konfigurasi komponen dalam proses eksekusi aplikasi”.

2.2.6. *Entity Relationship Diagram (ERD)*

Menurut (Abdussomad, 2016) pada dasarnya ada tiga simbol yang digunakan sebagai berikut:

1. Entitas digambarkan dengan kotak persegi panjang.
2. Hubungan atau Relasi digambarkan dengan kotak berbentuk diamond..
3. Atribut digambarkan dengan bentuk elips.
4. Tingkat Hubungan atau Cardinality,

Entity Relationship Diagram (ERD) juga menunjukkan tingkat hubungan yang terjadi

Menurut (Priyadi, 2014) menjelaskan bahwa, “Pemodelan basis data dengan menggunakan diagram relasi antar entitas, dapat dilakukan dengan menggunakan suatu pemodelan basis data yang bernama *Diagram Entity- Relationship*”.

Tahapan proses pemodelan yang digunakan pada Diagram E-R (ERD) menurut (Priyadi, 2014), adalah sebagai berikut:

1. Identifikasi entitas.
2. Menentukan atribut *key*.
3. Identifikasi relasi.
4. Membuat desain perkiraan.
5. Menentukan kardinalitas.
6. Melengkapi desain dengan atribut deskriptif.

Menurut (Nurahman, dkk 2018) mengemukakan bahwa “Sebuah diagram ER/ER_D tersusun atas tiga komponen, yaitu entitas, atribut, dan kerelasian antar entitas.

2.2.7. *Logical Record Structure (LRS)*

Menurut (Junaidi, 2016) mengemukakan bahwa “*Logical Record Structure* (LRS) merupakan hasil dari pemodelan *entity relationship* (ER) beserta atributnya sehingga bisa terlihat hubungan-hubungan antarentitas”.

Logical Record Structure dapat dikatakan sebagai perubahan bentuk yang disebut transformasi dari ERD yang sudah dibuat.

Menurut (Puspitasari, 2016) mengemukakan bahwa “LRS merupakan sebuah model sistem yang digambarkan dengan sebuah diagram-ER akan mengikuti pola atau aturan pemodelan tertentu dalam kaitannya dengan konversi LRS, maka perubahan yang terjadi adalah mengikuti aturan-aturan tertentu”.

2.2.8. *User Interface*

Menurut (Sukanto dan Shalahuddin, 2018) mengemukakan bahwa, “Antarmuka atau *interface* sangat mirip dengan kelas, tapi tanpa atribut kelas dan memiliki metode yang dideklarasikan tanpa isi”.

Sedangkan menurut (Tohari, 2014) menjelaskan bahwa, “*Interface* adalah satu *set operation* yang memberikan spesifikasi beberapa aspek dari perilaku dan operasi disuatu *class* ke *class* yang lain”.

Misalnya, *keyboard* pada komputer sebenarnya merupakan *interface* yang bisa dipakai ulang karena tombol-tombol *keyboard* sebenarnya berasal dari mesin ketik, hanya saja mungkin ada beberapa operasionalisasi tombol-tombol yang berbeda yang sudah ditransfer ke sistem yang lain. Misalnya tombol *control*, *page up*, *page down* dan lain-lain.

2.2.9. Konsep Dasar *Black Box Testing*

Menurut (Sukamto dan Shalahuddin, 2018) menjelaskan bahwa, “*Black-Box Testing* (pengujian kotak hitam) yaitu menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan”.

Pengujian kotak hitam dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai dengan spesifikasi yang dibutuhkan. Kasus uji yang dibuat untuk melakukan pengujian kotak hitam harus dibuat dengan kasus benar dan kasus salah, misalkan untuk kasus proses *login* maka kasus uji yang dibuat menurut (Sukamto dan Shalahuddin, 2018) adalah:

1. Jika *user* memasukkan nama pemakai (*username*) dan kata sandi (*password*) yang benar.
2. Jika *user* memasukkan nama pemakai (*username*) dan kata sandi (*password*) yang salah, misalnya nama pemakai benar tapi kata sandi salah, atau sebaliknya, atau keduanya salah.

Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan. Metode uji coba *Black Box* memfokuskan pada keperluan fungsional dari *software*.