

BAB II

LANDASAN TEORI

2.1. Konsep Dasar Sistem

Pada dasarnya sistem adalah kerangka dari prosedur-prosedur yang saling berhubungan, yang disusun sesuai dengan skema yang menyeluruh untuk melaksanakan suatu kegiatan atau fungsi utama dari perusahaan yang dihasilkan oleh suatu proses tertentu yang bertujuan mengambil keputusan manajemen operasi perusahaan dari hari ke hari serta menyediakan informasi yang layak untuk pihak diluar perusahaan.

2.1.1. Pengertian Sistem

Sistem adalah suatu jaringan prosedur yang di buat menurut pola yang terpadu untuk melaksanakan kegiatan pokok perusahaan. Semua sistem memiliki input, proses, output, dan umpan balik. Menurut (Susanto, 2017) mengemukakan bahwa “Sistem adalah kumpulan atau *group*/dari sub sistem/bagian/komponen apapun baik fisik atau pun non fisik yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai satu tujuan tertentu”.

Hal ini sejalan dengan penelitian (Friadi et al., 2017) “Sistem adalah suatu hal yang terpenting dalam membuat perancangan sistem informasi”. Pada umumnya setiap organisasi selalu mempunyai sistem informasi untuk mengumpulkan, menyimpan, melihat, dan menyalurkan informasi.

Berdasarkan pengertian sistem diatas penulis mengambil kesimpulan bahwa sistem adalah sekumpulan komponen yang bekerja sama untuk mencapai suatu tujuan tertentu.

2.1.2. Konsep Dasar Informasi

Informasi merupakan hal yang sangat penting bagi suatu perusahaan. Suatu organisasi yang kurang mendapatkan informasi akan menjadi lemah dan akhirnya berakhir. Oleh karena itu, penting untuk mengetahui lebih jauh mengenai informasi.

2.1.3. Pengertian Informasi

Informasi merupakan hasil pemrosesan data (fakta) menjadi sesuatu yang bermakna dan bernilai untuk mengambil keputusan. Dalam kehidupan sehari-hari, segala aktivitas pengambilan keputusan menjadi mudah dengan adanya informasi. Menurut (Susanto, 2017) adalah sebagai berikut: “Informasi adalah hasil pengolahan data yang memberikan arti dan manfaat”.

Dari uraian tentang informasi ini ada tiga hal penting yang harus di perhatikan disini yaitu:

1. Informasi merupakan hasil pengolahan data
2. Memberikan makna atau arti
3. Berguna atau bermanfaat

Berdasarkan pengertian informasi diatas penulis mengambil kesimpulan bahwa informasi adalah data yang telah diolah dan menghasilkan keputusan yang bermanfaat bagi seseorang yang menerima.

2.1.4. Kualitas Informasi

Kualitas dari suatu informasi tergantung dari 3 hal yaitu informasi harus akurat, tepat pada waktunya dan relevan. Hal ini sejalan dengan penelitian (Pawirosumarto, 2016) Kualitas informasi berkaitan dengan system use, user satisfaction, dan net benefits (DeLone dan McLean 1992, 2003). Kualitas informasi mempunyai atribut-

atribut seperti informasi yang diperoleh dari sebuah sistem, keakuratan informasi, relevansi informasi, ketepatan waktu, dan kelengkapan informasi.

Jadi, semakin tinggi kualitas informasi yang digunakan, diprediksi akan berpengaruh terhadap semakin tinggi tingkat kepuasan pengguna akhir sistem informasi tersebut.

2.1.5. Pengertian Sistem Informasi

Informasi merupakan hal yang sangat penting bagi manajemen dalam mengambil keputusan. Sistem informasi merupakan serangkaian komponen berupa manusia, prosedur, data, dan teknologi yang digunakan untuk menghasilkan informasi yang bernilai untuk mengambil keputusan. Menurut (Susanto, 2017) Sistem informasi adalah “Kumpulan dari sub sistem baik fisik maupun non fisik yang saling berhubungan satu sama dan bekerja sama secara harmonis untuk mencapai satu tujuan yaitu mengolah data menjadi informasi yang berguna”.

Menurut (M. E. Putri & Wulandari, 2016) “Sistem Informasi merupakan sistem yang berada pada organisasi yang didalamnya terdapat sekelompok orang-orang, teknologi, media, fasilitas, prosedur-prosedur dan pengendalian yang digunakan untuk tujuan mendapat jalur komunikasi, memproses transaksi secara rutin, memberi sinyal kepada manajemen mengenai kejadian-kejadian internal dan eksternal dan menyediakan informasi yang dapat digunakan untuk sebagai pengambilan keputusan”.

Jadi, berdasarkan pengertian diatas dapat disimpulkan bahwa sistem informasi adalah kumpulan data yang terintegritasi dan saling melengkapi dengan menghasilkan *output* yang baik guna untuk memecahkan masalah dan pengambilan keputusan.

2.1.6. Pengertian Sistem Informasi Akuntansi

Seorang pemimpin dalam melaksanakan dan mengelola perusahaan harus mengambil keputusan yang tepat di antara alternatif yang akan diproses. Menurut (Susanto, 2017) Sistem informasi akuntansi merupakan “Kumpulan (integrasi) dari

sub sistem/bagian/ komponen apapun baik fisik atau non fisik yang saling berhubungan dan bekerja satu sama lain sama secara harmonis untuk mengolah data transaksi yang berkaitan dengan masalah keuangan menjadi informasi keuangan”.

Dengan demikian, dilihat dari definisi di atas sistem akuntansi dan sistem informasi akuntansi mempunyai pengertian yang sama yaitu serangkaian kegiatan administrative perusahaan dalam melaksanakan berbagai aktivitas sehari-hari.

2.1.7. Pengertian Akuntansi

Semua kejadian atau peristiwa ekonomi yang diakibatkan oleh aktivitas suatu organisasi perusahaan yang berpengaruh terhadap posisi keuangan tersebut disebut transaksi akuntansi. Menurut Warren dan Fess (Susanto, 2017), mendefinisikan “Akuntansi sebagai sistem informasi yang menghasilkan informasi atau laporan untuk berbagai kepentingan baik individu atau kelompok tentang aktivitas/operasi/peristiwa ekonomi atau keuangan suatu organisasi”. Menurut “(Mamintade, Sam, & Manado, 2019) Akuntansi adalah sistem informasi yang menyediakan laporan bagi para pihak mengambil keputusan (*stakeholder*) mengenai aktivitas ekonomi dan kondisi perusahaan”. Akuntansi merupakan sebuah sistem informasi yang memberikan laporan kepada pengguna informasi.

Jadi, transaksi bisnis atau akuntansi pada dasarnya merupakan penyeleksian semua aktivitas ekonomi kedalam aktivitas yang hanya berkaitan dengan suatu organisasi perusahaan, aktivitas ekonomi lainnya yang tidak berkaitan dikesampingkan.

2.1.8. Pengertian Jurnal

Transaksi keuangan yang terjadi akan dianalisa dan selanjutnya akan dicatat dalam sebuah buku harian yang biasa disebut dengan istilah jurnal umum. Menurut (R. A. Putri, Syafina, Kunci, & Informasi, 2018) Jurnal umum adalah buku pertama

proses akuntansi yang merupakan catatan pertama yang bersifat kronologis tentang transaksi maupun kejadian yang menunjukkan perkiraan apa yang di debet atau perkiraan apa yang harus di kredit beserta keterangan yang berkaitan dengan transaksi maupun kejadian tersebut.

2.1.9. Pengertian Basis Data

Pada umumnya basis data merupakan salah satu aspek yang sangat penting dalam sistem informasi, dimana basis data dijadikan sebagai gudang penyimpanan data yang akan di olah lebih lanjut. Basis data yaitu kumpulan data yang disusun secara sistematis di dalam komputer menggunakan perangkat lunak untuk menghasilkan informasi. Basis data juga merupakan kumpulan data yang umumnya menggambarkan aktivitas-aktivitas dan pelakunya dalam suatu organisasi.

Menurut (Sukamto, 2015) “Basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan”. Pada dasarnya basis data adalah media untuk menyimpan data agar dapat diakses dengan mudah dan cepat.

Kebutuhan basis data dalam sistem informasi meliputi;

1. Memasukkan, menyimpan, dan mengambil data.
2. Membuat laporan berdasarkan data yang telah di simpan.

Jadi, untuk merancang tabel-tabel yang akan dibuat maka dibutuhkan pola pikir penyimpanan data nantinya jika dalam bentuk baris-baris data (*record*) dimana setiap baris terdiri dari beberapa kolom.

2.1.10. Pengertian Java

Java merupakan bahasa berorientasi objek untuk pengembangan aplikasi mandiri, aplikasi berbasis internet, aplikasi untuk perangkat cerdas yang dapat berkomunikasi lewat internet atau jaringan komunikasi.

Menurut (Fajarianto & Dosen, 2016) “Java adalah bahasa pemrograman yang berorientasi obyek, bukan seperti Pascal, Basic, atau C yang berbasis prosedural”.

Java adalah bahasa pemrograman yang multi *platform* dan multi *device*. Sekali menuliskan program dengan menggunakan *java*, maka dapat menjalankannya hampir disemua komputer dan perangkat lain yang support *java*, dengan sedikit perubahan atau tanpa perubahan sama sekali dalam kodenya.

Banyak aplikasi dan situs yang tidak akan bekerja jika tidak ada perangkat lunak *java* pada komputer.

2.1.11. Pengertian MySQL

MySQL adalah perangkat lunak sistem manajemen basis data *SQL* atau *DBMS* yang *multithread*, *multiuser*. Menurut (Sukamto, 2015) mengungkapkan bahwa “*Structured Query Language (SQL)* adalah bahasa yang digunakan untuk mengelola data pada *RDBMS*”. *SQL* awalnya dikembangkan berdasarkan teori aljabar relasional dan kalkulus. Hampir semua *Database Management System* (*DBMS*) mengadopsi *SQL* sebagai bahasa untuk mengelola pada *DBMS*.

2.1.12. Pengertian Netbeans

Netbeans IDE merupakan alat pengembangan aplikasi yang terintegrasi, *netbeans* IDE mendukung pengembangan program yang menggunakan Bahasa *Java* dari semua versi. Menurut (Dhika, Isnain, & Tofan, 2019) “*Netbeans* yaitu lingkungan pengembangan yang bebas, open source, terintegrasi (*IDE*) yang memungkinkan untuk mengembangkan aplikasi desktop, seluler dan web. *IDE* mendukung pengembangan aplikasi dalam berbagai bahasa, seperti *Java*, *HTML5*, *PHP* dan *C ++*”.

Netbeans juga dapat digunakan programmer untuk menulis, mencari kesalahan dan menyebarkan program *netbeans* yang ditulis ke dalam bahasa pemrograman

lainnya dan program ini pun bebas digunakan dan untuk membuat *profesional dekstop, enterprise, web and mobile applications* dengan *java language, C++* dan bahkan *dynamic languages* seperti *PHP, JavaScript*.

2.1.13. Pengertian Waterfall

Dalam perancangan sistem pada tugas akhir ini penulis menggunakan metode *waterfall*. Menurut (Sukamto dan Shalahuddin, 2016) menjelaskan bahwa “Model *SDLC* air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequential linier*) atau alur hidup klasik (*classic life cycle*)”. Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengkodean, pengujian dan tahap pendukung (*support*). Berikut ini penjelasannya:

1. Analisis Kebutuhan Perangkat Lunak

Proses pengumpulan kebutuhan dilakukan secara intensif untuk memspezifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti yang dibutuhkan user. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu didokumentasikan.

2. Desain

Desain perangkat lunak adalah proses mutli langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitekstur perangkat lunak, representasi antarmuka, dan prosedur pengkodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.

3. Pembuatan Kode Program

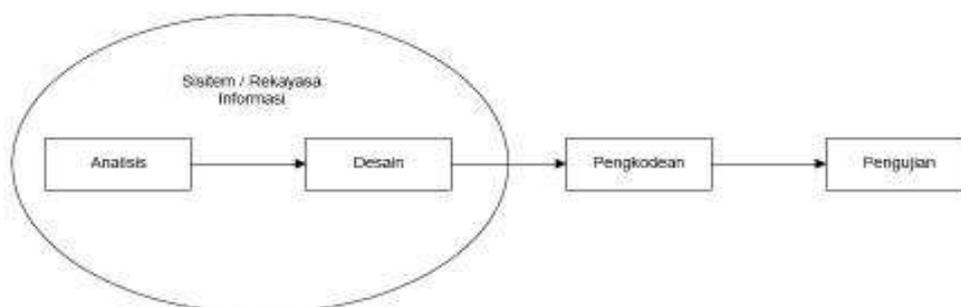
Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

4. Pengujian

Pengujian fokus pada perangkat lunak secara dari segi logik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluran yang dihasilkan sesuai dengan yang diinginkan.

5. Pendukung (*support*) dan Pemeliharaan (*maintanance*)

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke user. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.



Sumber : Sukamto dan Shalahudin (2016:29)

Gambar II.1.

Model air terjun (*waterfall*)

2.1.14. Pengertian XAMPP

Menurut “(AP, 2019) XAMPP adalah sebuah *software web server apache* yang didalamnya sudah tersedia *database server mysql* dan *support php programming*. XAMPP merupakan *software* yang mudah di- gunakan, gratis dan mendukung instalasi di *Linux* dan *Windows*”. Hal ini sejalan dengan penelitian menurut “(Dhika et al., 2019) XAMPP di artikan sebagai *software web apache* yang terdapat *database mysql dan php programming*. XAMPP merupakan *software* yang mudah digunakan, gratis dan mendukung instalasi di *linux* dan *windows*”.

2.2. Peralatan Pendukung (*Tools System*)

Peralatan pendukung (*Tools System*) merupakan alat yang dapat digunakan untuk menggambarkan bentuk logika model dari suatu sistem. Adapun *Tools System* yang digunakan penulis untuk merancang model sistem adalah:

2.2.1. Pengertian *Unified Modeling Language (UML)*

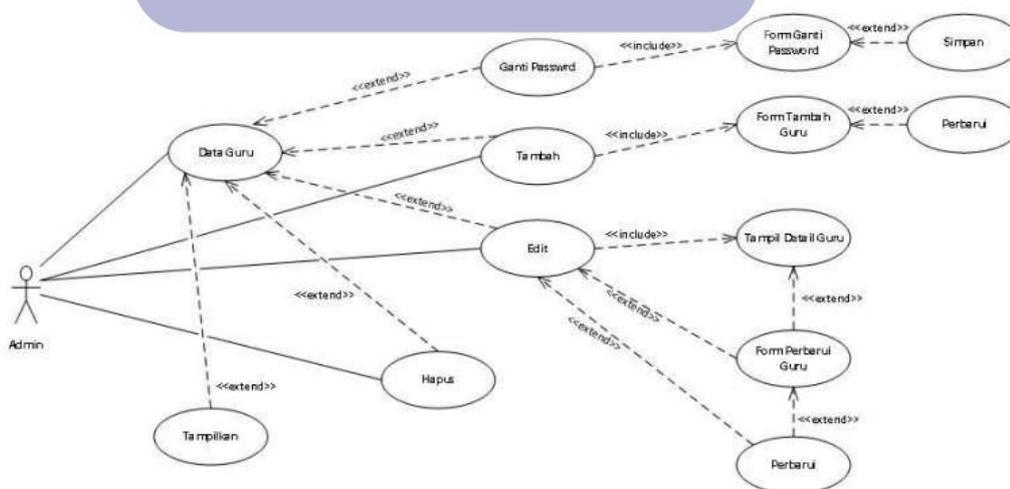
Pada perkembangan teknik pemograman berorientasi objek, munculah sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemograman berorientasi objek, yaitu *Unified Modeling Language (UML)*. Menurut (Sukanto dan Shalahudin, 2015), “*Unified Modeling Language (UML)* adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analis dan desain, serta menggambarkan arsitektur dalam pemograman berorientasi objek”.

Unified Modeling Language merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun dan dokumentasi dari perangkat

lunak. UML hanya berfungsi untuk melakukan pemodelan. Jadi penggunaan *Unified Modeling Language* tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya *Unified Modeling Language* paling banyak digunakan pada metodologi berorientasi objek.

1. Use Case Diagram

Use case adalah diagram yang bersifat statis dan menunjukkan himpunan *use case* dan aktor (suatu jenis khusus dari kelas). *Use case* menggambarkan kebutuhan sistem dari sudut pandang *user*. Menurut Sukanto dan Shalahudin (2015:155) “*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 di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Diagram *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut.

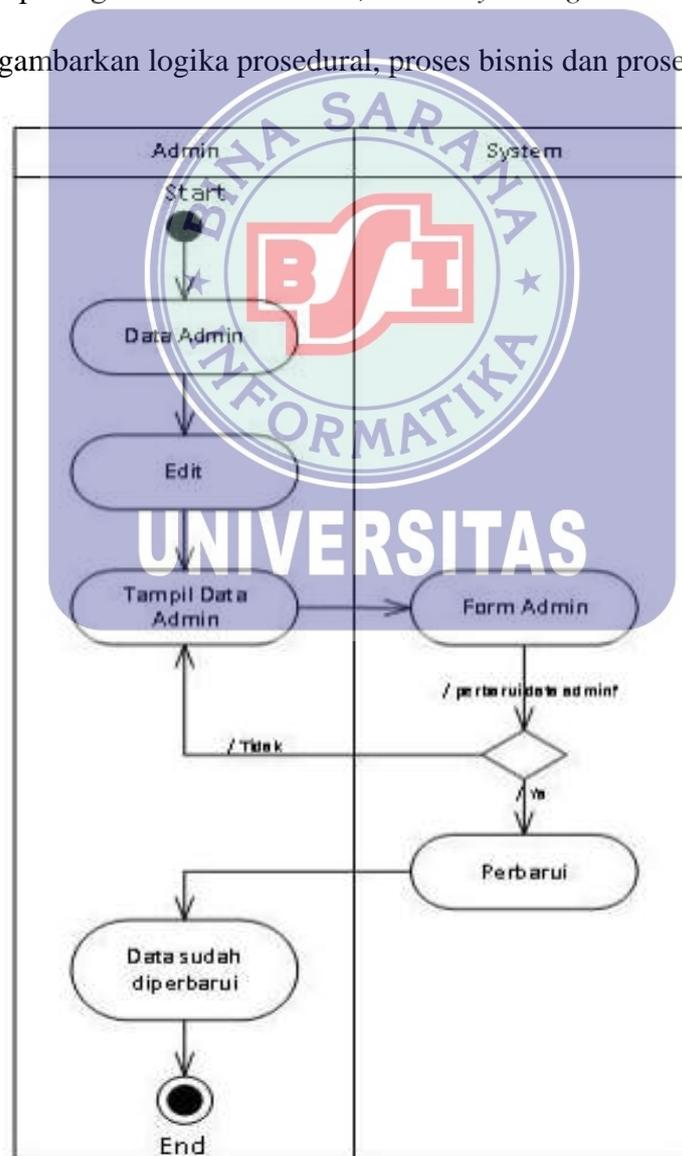


Sumber : Darmawati dan Hartini (2017)

Gambar II.2.
Contoh Use Case Diagram

2. Activity Diagram

Activity Diagram merupakan sebuah aliran kerja atau aktivitas dari sebuah sistem yang ada pada perangkat lunak. Menurut Sukamto dan Shalahudin (2015:161) “Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak”. Jadi, *activity diagram* adalah teknik untuk menggambarkan logika prosedural, proses bisnis dan proses kerja.



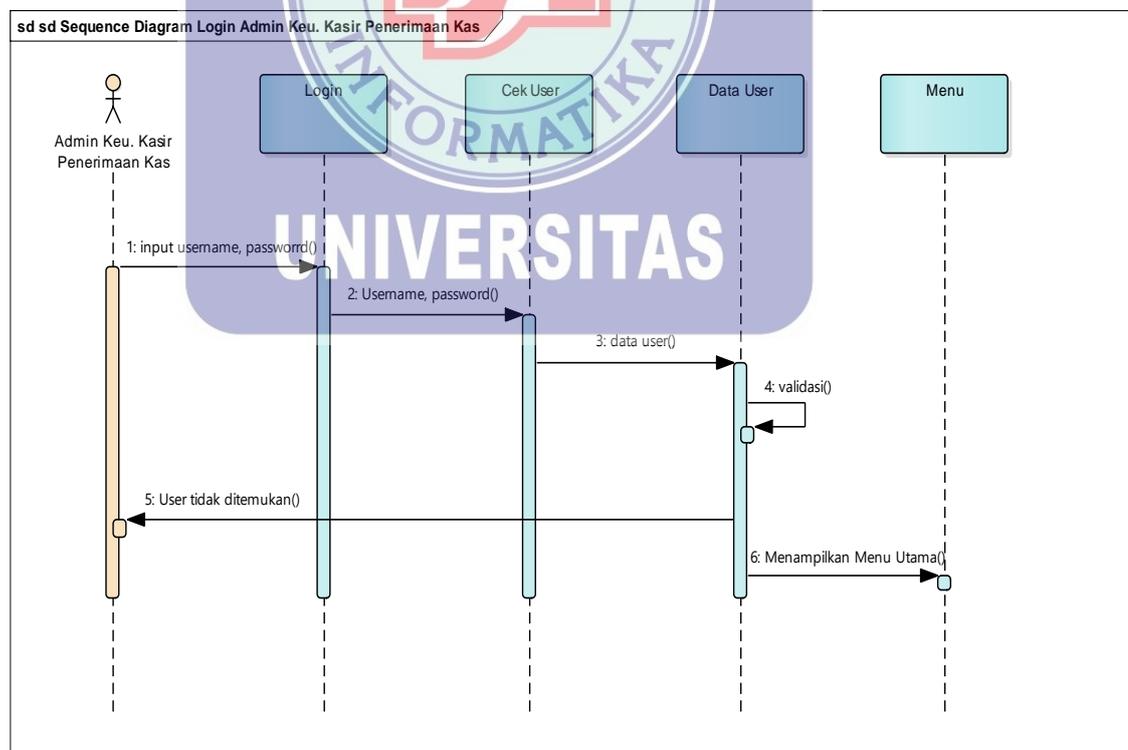
Sumber : Darmawati dan Hartini (2017)

Gambar II.3.

Contoh Activity Diagram

3. Sequence Diagram

Sequence Diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Menurut Sukanto dan Shalahudin (2015:165) “Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dengan *message* yang dikirimkan dan diterima antar objek”. Oleh karena itu untuk menggambarkan diagram skuen, maka harus diketahui objek-objek yang terlibat dalam sebuah use case beserta metode-metode yang memiliki kelas yang diinstansiasi menjasi objek itu.



Sumber : Syara, Chintya (2018)

Gambar II.4.

Contoh Sequence Diagram

4. Deployment Diagram

Deployment Diagram menunjukkan perangkat keras sistem dan perangkat lunak tersebut. *Diagram deployment* menggambarkan *detail* bagaimana komponen disebar di *deployment* ke dalam infrastruktur sistem. Menurut Sukamto dan Shalahudin (2015:154) “*Diagram Deployment* atau *Deployment diagram* menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi”. *Diagram deployment* berhubungan erat dengan diagram komponen dimana diagram ini memuat satu atau lebih komponen-komponen. Diagram ini sangat berguna saat aplikasi kita berlaku sebagai aplikasi yang dijalankan pada banyak mesin (*distributed computing*). *Diagram deployment* juga dapat digunakan untuk memodelkan hal-hal berikut:

- a. Sistem tambahan (*embedded system*) yang menggambarkan rancangan *device*, *node*, dan *hardware*.
- b. Sistem *client/server* misalnya seperti gambar berikut:



Sumber : Sukamto dan Shalahuddin (2015:154)

Gambar II.5.

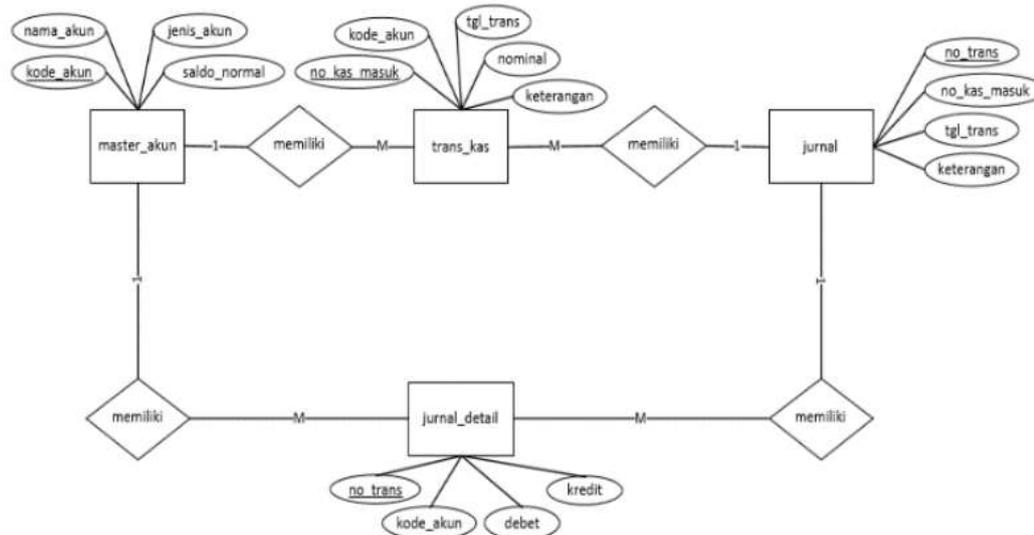
Contoh Deployment Diagram

2.2.2. Pengertian *Entity Relationship Diagram* (ERD)

Entity Relationship Diagram (ERD) dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data relasional. Sehingga jika penyimpanan basis data menggunakan OODBMS maka perancangan basis data tidak perlu menggunakan ERD.

Menurut (Sukamto dan Shalahuddin, 2015) mengatakan bahwa “ERD adalah bentuk paling awal dalam melakukan perancangan basis data relasional”. Menurut “(Dermawan, Juniadi; Hartini, 2017), ERD adalah alat pemodelan data utama dan akan membantu mengorganisasi data dalam suatu proyek ke dalam entitas- entitas dan menentukan hubungan antar entitas”. Proses memungkinkan analisis menghasilkan struktur basisdata yang baik, sehingga data dapat disimpan dan diambil secara efisien. Proses memungkinkan analisis menghasilkan struktur basis data yang baik, sehingga data dapat di simpan dan diambil secara efisien.

Dari pemahaman para ahli dapat disimpulkan ERD merupakan diagram yang menggambarkan penyusunan basis dalam bentuk entitas-entitas yang akan disusun dalam bentuk entitas-entitas yang selanjutnya dilakukan proses melengkapi atribut-atribut yang sesuai pada entitas.



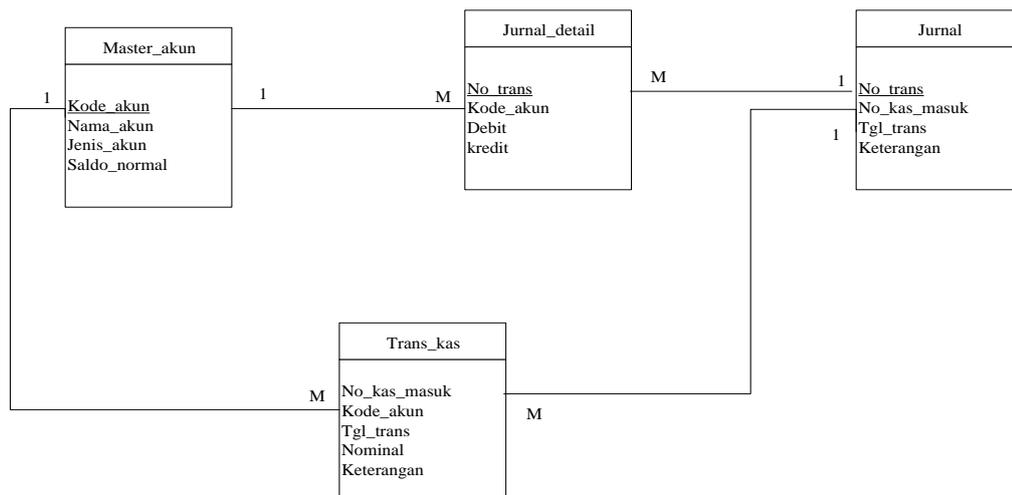
Sumber: Syara, Chintya (2018)

Gambar II.6.
Contoh Entity Relationship Diagram

2.2.3. Logical Record Structure (LRS)

Logical Record Structure dibentuk dengan nomor *type record*. Beberapa *type record* digambarkan oleh kotak empat persegi panjang dan dengan nama yang 23 unik-unik. *Logical Record Structure* terdiri dari link-link diantara *type record*. Link ini menunjukkan arah dari satu *type record* lainnya. Menurut “(Kuryanti & Sandra, 2016) *Logical Record Structure* (LRS) dibentuk dengan nomor dari *type record*”. Beberapa *type record* digambarkan oleh kotak persegi panjang dan dengan nama yang unik. Perbedaan LRS dengan E-R diagram adalah nama *type record* berada diluar kotak field *type record* ditempatkan. LRS terdiri dari link-link diantara *type record*.

Dari pemahaman para ahli dapat disimpulkan *Logical Record Structure* (LRS) adalah “Model system berbentuk gambar dengan nomer *type record* dan beberap *type record* digambarkan oleh kotak empat persegi panjang. LRS yang dibuat adalah sebagai berikut:



Sumber: Syara, Chintya (2018)

Gambar II.7.

Contoh Logical Record Structure

2.2.4. Black-box Testing

Pengujian yang dilakukan adalah *blackbox testing*. *Blackbox testing* memfokuskan pada keperluan fungsional dari software. Pengujian yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari software. *Black-box testing* adalah metode pengujian perangkat lunak yang tes fungsionalitas dari aplikasi yang bertentangan dengan struktur internal atau kerja.

Menurut "(Rismadewi, Wikusna, & Nugroho, 2018) *Black-box testing* adalah metode pengujian perangkat lunak yang tes fungsionalitas dari aplikasi yang bertentangan dengan struktur internal atau kerja. Pengetahuan khusus dari kode aplikasi / struktur internal dan pengetahuan pemrograman pada umumnya tidak diperlukan".

Maka *Black Box Testing* adalah pengujian *software* untuk mengetahui setiap fungsi berjalan dengan baik atau tidak sesuai kebutuhan.