

BAB II

LANDASAN TEORI

2.1. Konsep Dasar Sistem

Konsep dasar sistem merupakan sekelompok komponen berbasis komputer yang dibuat oleh manusia dalam mengelola data, menyimpan, menghimpun kerangka kerja serta mengkoordinasikan sumber daya manusia dan komputer untuk mengubah sistem masukan menjadi sistem keluaran untuk mencapai tujuan dan sasaran yang ditetapkan sebelumnya.

Sistem mempunyai suatu konsep yang mendasari sebuah pengertian-pengertian yang di kemukakan oleh berbagai pakar untuk mendefinisikan dari suatu sistem itu sendiri. Sebelum mendefinisikan suatu sistem pakar harus mempunyai konsep dasar untuk memperkuat pendefinisiannya.

2.1.1. Pengertian Sistem

Menurut (Jogiyanto, 2017) menyimpulkan bahwa : Sistem merupakan kumpulan dari beberapa bagian yang memiliki keterkaitan dan saling bekerja sama serta membentuk suatu kesatuan untuk mencapai suatu tujuan dari sistem tersebut. Maksud dari suatu sistem adalah untuk mencapai suatu tujuan dan sasaran dalam ruang lingkup yang sempit. Dari pengertian di atas dapat diartikan bahwa sistem adalah sekumpulan sumber daya yang saling terkait untuk mencapai suatu tujuan.

Menurut (Muhammad Muslihudin & Oktafianto, 2016) “Sistem adalah sekumpulan komponen atau jaringan kerja dari prosedur-prosedur yang saling berkaitan dan saling bekerja sama membentuk suatu jaringan kerja untuk mencapai sasaran atau tujuan tertentu”. Sedangkan menurut (Mulyadi, 2016) “Sistem adalah “suatu jaringan

prosedur yang dibuat menurut pola yang terpadu untuk melaksanakan kegiatan pokok perusahaan”.

Dengan demikian disimpulkan bahwa sistem merupakan sekumpulan komponen/entitas yang berada dalam suatu jaringan kerja, saling berinteraksi, berhubungan dan ketergantungan antar komponen untuk mencapai suatu tujuan yang menghasilkan keluaran (output).

1. Karakteristik Sistem

Suatu sistem mempunyai ciri-ciri karakteristik yang terdapat pada sekumpulan elemen yang harus dipahami dalam mengidentifikasi pembuatan sistem. Adapun karakteristik sistem menurut (Hutahaean, 2015) adalah sebagai berikut:

a. Komponen Sistem (*components system*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerjasama membentuk satu kesatuan komponen sistem tersebut dapat berupa suatu sub-sistem atau bagian dari sistem.

b. Batasan Sistem (*boundary*)

Batasan sistem merupakan batasan suatu sistem dengan suatu sistem lainnya atau dengan lingkungan luarnya, batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan yang tidak dapat dipisahkan batas waktu sistem menunjukkan ruang lingkup dari sistem tersebut.

c. Lingkungan Luar Sistem (*Environment*)

Lingkungan luar sistem merupakan bentuk apapun yang ada diluar sistem yang mempengaruhi operasi sistem ini bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut.

d. Penghubung Sistem (*interface system*)

Penghubung sistem adalah sebagai media yang menghubungkan sistem dengan subsistem.

e. Masukan Sistem (*input system*)

Masukan sistem (*input system*) adalah energi yang dimasukkan *system* yang dapat berupa pemeliharaan (*maintenance input*) dan masukan sinyal (*signal input*). *Maintenance input* adalah energi yang dimasukkan agar sistem tersebut dapat beroperasi. *Signal input* adalah energi yang diproses untuk mendapatkan keluaran.

f. Keluaran Sistem (*output system*)

Hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan untuk subsistem yang lain.

g. Pengolahan Sistem (*process system*)

Pengolahan sistem (*process system*) merupakan suatu sistem yang dapat mempunyai suatu proses yang akan merubah masukan menjadi keluaran. Sebagai contoh sistem akuntansi, sistem ini akan mengolah data transaksi menjadi laporan-laporan yang dibutuhkan oleh pihak manajemen.

h. Sasaran Sistem (*objective system*)

Sasaran sistem merupakan suatu sistem yang mempunyai tujuan dan sasaran yang pasti dan bersifat deterministik (dapat diprediksi). Sasaran sangat menentukan kebutuhan akan masukan dan keluaran yang diharapkan. Jika suatu sistem tidak memiliki sasaran maka operasi sistem ;[tidak akan ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran dan tujuan yang telah direncanakan.

1. Klasifikasi Sistem

Sistem merupakan suatu bentuk integrasi antara satu komponen dengan komponen lainnya. Karena sistem memiliki sasaran yang berbeda untuk setiap kasus yang terjadi yang ada dalam sistem tersebut. Oleh karena itu sistem diklasifikasikan kedalam berdasarkan kriteria tertentu sebagai berikut:

a. Sistem Terbuka dan Tertutup

Sebuah sistem dikatakan terbuka apabila aktivitas didalam sistem tersebut dipengaruhi oleh lingkungannya. Sedangkan suatu sistem dikatakan tertutup apabila aktivitas didalam sistem tersebut tidak dipengaruhi oleh perubahan yang terjadi di lingkungannya.

b. Sistem Buatan Manusia dan Tuhan (Allah)

Suatu sistem apabila diklasifikasikan berdasarkan pembuat sistem bisa tuhan (sistem alamiah) bisa juga manusia.

c. Sistem Berjalan dan Konseptual

Suatu sistem belum diterapkan disebut sebagai sistem konseptual. Apabila kita merancang suatu sistem dan sistem tersebut belum diterapkan maka sistem tersebut hanyalah merupakan angan-angan atau masih berbentuk harapan yang mungkin secara akal sehat (konsep) penyusunnya sistem sudah benar, dibuat berdasarkan kebutuhan dan situasi kondisi yang ada. Sistem berjalan adalah sistem yang digunakan saat ini. Sistem yang benar adalah sistem yang tepat guna dan dapat digunakan oleh pemakai sistem untuk meningkatkan pengendalian, efisiensi, dan kecepatan.

d. Sistem Sederhana dan Komplek

Dilihat dari tingkat kesulitannya, sebuah sistem dapat merupakan sebuah sistem yang sederhana atau sistem yang komplek. Sistem sederhana adalah sistem yang

memiliki sedikit tingkatan dan subsistem. Sedangkan sistem kompleks adalah sistem yang memiliki banyak tingkatan dan subsistem.

e. Kinerja Dapat dan Tidak Dapat Dipastikan

Suatu sistem dapat pula diklasifikasikan berdasarkan kepada kinerja yang dihasilkannya. Sebuah sistem yang dapat dipastikan artinya dapat ditentukan pada saat sistem akan dan sedang dibuat. Dilain pihak, sebuah sistem mungkin tidak dapat dipastikan yang artinya tidak dapat ditentukan dari awal tergantung kepada situasi yang dihadapi.

f. Sementara dan Selamanya

Suatu sistem mungkin digunakan untuk selamanya atau untuk periode waktu tertentu saja. Sementara artinya sistem hanya digunakan periode waktu tertentu. Sebaliknya jika selamanya yang artinya sistem digunakan selama-lamanya untuk waktu yang tidak ditentukan.

g. Ada Secara Fisik dan Abstrak/Non Fisik

Akhirnya sistem dapat dilihat dari wujudnya. Kendaraan bermotor bukan hanya merupakan sistem buatan manusia akan tetapi juga merupakan sistem yang ada secara fisik. Ada secara fisik artinya disini dapat diraba. Perusahaan dan perguruan tinggi bukanlah organisasi yang dapat disentuh secara fisik. Kita dapat menyentuk foto, menunjuk apa yang ada difoto seperti mesin atau buku-buku, akan tetapi wujudnya adalah abstrak/non fisik. Abstrak artinya disini tidak dapat diraba.

h. Sistem, Subsistem dan Supersistem

Berdasarkan tingkatannya/hierarki sebuah sistem bisa merupakan komponen dari sistem yang lebih besar. Sistem yang lebih kecil yang ada dalam sebuah sistem

disebut sebagai subsistem. Sedangkan sistem yang sangat besar dan kompleks adalah supersistem.

i. Bisa Beradaptasi dan Tidak Bisa Beradaptasi

Berdasarkan fleksibelnya kita dapat membedakan karakteristik suatu sistem tersebut dapat beradaptasi terhadap perubahan yang terjadi di lingkungannya atau tidak. Suatu sistem bisa beradaptasi artinya bisa menyesuaikan diri terhadap perubahan lingkungan, sebaliknya jika suatu sistem tidak bisa menyesuaikan diri terhadap perubahan lingkungan disebut tidak bisa beradaptasi.

2.1.2. Siklus Hidup Pengembangan Sistem

Menurut (Munthe, 2019) “*System Development Life Cycle (SDLC)* merupakan suatu pendekatan yang memiliki tahap atau bertahap untuk melakukan analisa dan membangun suatu rancangan sistem dengan menggunakan siklus yang lebih spesifik terhadap kegiatan pengguna.”

Menurut (R. Susanto & Andriana, 2016) menyimpulkan bahwa : *System Development Life Cycle* atau yang lebih dikenal dengan istilah SDLC adalah metodologi umum yang digunakan untuk mengembangkan sistem informasi. SDLC terdiri dari beberapa fase yang dimulai dari fase perencanaan, analisis, perancangan, implementasi hingga pemeliharaan sistem. Konsep SDLC ini mendasari berbagai jenis model pengembangan perangkat lunak untuk membentuk suatu kerangka kerja untuk perencanaan dan pengendalian pembuatan sistem informasi. Model-model SDLC yang sering digunakan antara lain Waterfall dan Prototyping.

Menurut (Rosa & Shalahuddin, 2015) “*System Development Life Cycle* atau SDLC adalah aplikasi penerapan dari penemuan permasalahan (problem solving) yang didapat dari pendekatan sistem (system approach) menjadi pengembangan dari solusi sistem informasi terhadap masalah bisnis.”

Menurut (Widaryanto, Widiyanto, & Rifa'i, 2017) menyimpulkan bahwa : *System Development Life Cycle (SDLC)* adalah proses yang digunakan dalam pembuatan sistem. *Object Oriented Programming* bertujuan untuk mengombinasikan data dan mengasosiasikan method ke dalam satu unit yang bernama objek untuk membangun sistem yang kompleks.. Belum ada satu model yang efektif untuk semua penerapan, karena itu perlu memilih model yang sesuai untuk setiap project. UML saat ini menjadi tool paling penting untuk sistem dengan model berorientasi objek.

(*System Development Life Cycle* atau SDLC berfungsi untuk menggambarkan tahapan utama dan langkah-langkah di dalam setiap tahapan yang secara garis besar terbagi menjadi 3 kegiatan utama, yaitu” :

1. Analisis

Tahapan ini digunakan oleh analisis sistem untuk membuat keputusan. Apabila sistem saat ini sudah tidak bisa berfungsi secara baik, dan hasil analisisnya digunakan sebagai besar untuk memperbaiki sistem.

Kegiatan yang dilakukan pada tahap analisis adalah:

- a. Deteksi masalah (*Problem Detection*)
- b. Penelitian atau investigasi awal (*Initial Investigation*)
- c. Analisa kebutuhan sistem (*Requirment Analysis*)
- d. Mensortir kebutuhan sistem (*Generation of System Alternatives*)
- e. Memilih sistem yang baik (*Selection of Proper System*)

2. Perancangan atau Disain

Tahapan perancangan (*Design*) memiliki tujuan untuk mendisain sistem baru yang dapat menyelesaikan masalah-masalah yang dihadapi perusahaan yang diperoleh dari pemilihan alternatif sistem yang terbaik. Kegiatan yang dilakukan dalam tahap perancangan ini meliputi:

a. Perancangan Keluaran

Bertujuan menentukan keluaran-keluaran yang akan digunakan oleh sistem. keluaran tersebut berupa tampilan-tampilan layar, dan juga format dan frekuensi laporan yang di perlukan.

b. Perancangan Masukan

Bertujuan menentukan data-data masukan, yang akan digunakan untuk mengoperasikan sistem.

c. Perancangan *File*

Perancangan *file* diawali dengan merancang diagram hubungan antara entitas, setelah itu melakukan uji normalisasi dari 1-NF sampai ke 5-NF, minimal sampai 3-NF. Setelah lulus uji normalisasi maka dibuatkan spesifikasi data nya.

3. *Implementation*

Kegiatan yang dilakukan dalam tahap implementasi ini adalah pembuatan program dan tes data, pelatihan, dan pergantian sistem.

2.1.3. Model Pengembangan Perangkat Lunak

Menurut (Irmayani, 2019) menjelaskan bahwa :

Buku Rekayasa Perangkat Lunak ini membahas kegiatan-kegiatan yang dilakukan untuk membangun perangkat lunak menggunakan model-model proses yang ada dan yang sering digunakan oleh para pengembang perangkat lunak (*software developers*). Dalam rekayasa perangkat lunak umumnya ada beberapa kegiatan seperti yang dijelaskan pada model proses apa pun, yaitu identifikasi kebutuhan, desain, pengodean, implementasi, dan pemeliharaan. Selain itu, juga dibahas mengenai kualitas, pengujian, pemeliharaan.

Menurut (Shalahuddin, 2016) “Model SDLC air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequential linear*) 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 adalah gambaran model air terjun. beberapa tahapan dalam pengembangan perangkat lunak model *waterfall* yaitu:

1. Analisa Kebutuhan Perangkat Lunak Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh user.
2. Desain Proses multi langkah yang fokus pada desain pembuatan program perangkat lunak, termasuk struktur data, arsitektur perangkat lunak, representasi antar muka, dan prosedur pengodean.
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 dari segi logik dan fungsional untuk memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan dan memastikan keluaran yang dihasilkan sesuai dengan apa yang diinginkan.
5. Pendukung (*support*) dan Pemeliharaan (*maintenance*) 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. Dari kenyataan yang terjadi sangat jarang model air terjun dapat dilakukan sesuai alurnya karena sebab sebagai berikut :

1. perubahan spesifikasi perangkat lunak terjadi di tengah alur pengembangan.
2. Sangat sulit bagi pelanggan untuk mendefinisikan semua spesifikasi di awal alur pengembangan. Pelanggan sering kali butuh contoh (*prototype*) untuk menjabarkan spesifikasi kebutuhan sistem lebih lanjut.

3. Pelanggan tidak mungkin bersabar mengakomodasi perubahan yang diperlukan di akhir alur pengembangan. Dengan berbagai kelemahan yang dimiliki model air terjun tapi model ini telah menjadi dasar dari model-model yang lain dalam melakukan perbaikan model pengembangan perangkat lunak. Hal positif dari model air terjun adalah struktur tahap pengembangan sistem jelas, dokumentasi dihasilkan disetiap tahap pengembangan, dan sebuah tahap dijalankan setelah tahap sebelumnya selesai dijalankan (tidak ada tumpang tindih pelaksanaan tahap).

2.1.4. Pengertian Sistem Informasi

Pada perusahaan atau instansi yang mengikuti perkembangan teknologi dan informasi, sistem informasi dipakai sebagai alat terapan dan kombinasi antara teknologi dengan sistem yang dipakai dalam perusahaan atau instansi tersebut.

Menurut (Larasati & Masripah, 2017) mendefinisikan bahwa “sistem informasi merupakan kombinasi dari teknologi informasi dan aktivitas manusia untuk mendukung operasi dan manajemen”.

Menurut (A.S & Shalahuddin, 2015) “sistem informasi merupakan kombinasi dari teknologi informasi dan aktivitas manusia untuk mendukung operasi dan manajemen”.

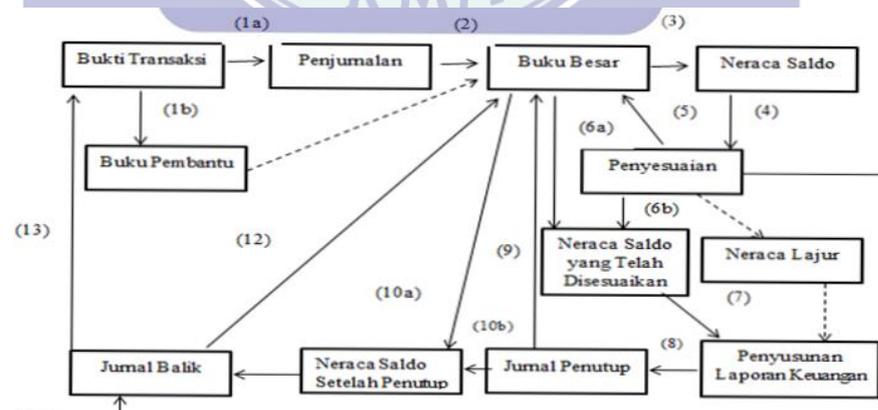
Menurut (Triandini, Jayanatha, Indrawan, Werla Putra, & Iswara, 2019) “Sistem informasi adalah cara-cara yang diorganisasi untuk mengumpulkan, memasukkan, mengolah, dan menyimpan data, serta untuk menyimpan, mengelola, mengendalikan, dan melaporkan informasi sedemikian rupa sehingga sebuah organisasi dapat mencapai tujuan yang telah ditetapkan”.

Menurut (Munthe, 2019) “Sistem Informasi merupakan suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan.”

Menurut (Abdullah, 2015) menjelaskan bahwa :
Sistem informasi adalah cara-cara yang diorganisasi untuk mengumpulkan, memasukkan, dan mengolah serta menyimpan data, dan cara-cara yang diorganisasi untuk menyimpan, mengelola, mengendalikan, dan melaporkan informasi sedemikian rupa sehingga sebuah organisasi dapat mencapai tujuan yang telah ditetapkan.” Sedangkan menurut (Pamungkas, 2017) ” Sistem Informasi berperan sebagai sistem karena mempunyai ruang lingkup yang relatif lebih luas dan lebih kompleks.

Dengan demikian dapat disimpulkan bahwa sistem informasi adalah suatu serangkaian prosedur pengumpulan data yang terdiri dari komponen-komponen yang terkait dengan *hardware*, *software* dan sumber daya manusia sehingga dapat menghasilkan informasi untuk mencapai suatu tujuan.

2.1.5. Pengertian Siklus Akuntansi



Sumber : (Astria Setia Mulyani, 2019)

Gambar II.1

Contoh Siklus Akuntansi

Siklus akuntansi adalah suatu proses penyediaan laporan keuangan perusahaan atau instansi untuk suatu periode waktu tertentu. Informasi berupa laporan keuangan dihasilkan melalui proses akuntansi yang panjang. Pada proses tersebut terdapat tahapan yang harus dipenuhi untuk mendapatkan hasil laporan yang baik, valid dan akuntabel.

Menurut (Suyenti, 2017) “Siklus akuntansi adalah tahapan – tahapan mulai dari terjadinya sampai dengan penyusunan laporan keuangan sehingga siap untuk pencatatan berikutnya”.

Dengan penjelasan diatas dapat disimpulkan bahwa siklus akuntansi adalah suatu kegiatan yang dilakukan terus menerus yang didalamnya terdapat berbagai proses, dimulai dari proses transaksi hingga penyusunan laporan keuangan untuk persiapan aktivitas periode akuntansi selanjutnya dengan pembuatan jurnal balik.

2.1.6. Pengertian Sistem Informasi Akuntansi

Sistem informasi akuntansi merupakan sistem pengolahan data akuntansi yang terdiri dari koordinasi manusia, alat dan metode yang berinteraksi dalam suatu wadah organisasi yang berstruktur untuk menghasilkan informasi akuntansi keuangan dan informasi akuntansi manajemen berstruktur.

Menurut (A. Susanto, 2017) “Sistem informasi akuntansi dapat didefinisikan sebagai kumpulan (integrasi) dari sub-sub sistem/ komponen baik fisik maupun nonfisik yang saling berhubungan dan bekerja sama satu sama lain secara harmonis untuk mengolah data transaksi yang berkaitan dengan masalah keuangan menjadi informasi keuangan”.

Menurut (Jermias, 2016) “Sistem Informasi Akuntansi membantu manajer dalam mengambil keputusan yang efektif. Keuntungan perusahaan tidak lepas dari

dukungan sumber daya manusia karena memberikan sumbangan tenaga, pikiran, dan keahliannya dalam operasi perusahaan.”

Menurut (Tresnawati, Sabijono, & Manossoh, 2017) “Sistem Informasi Akuntansi yang baik dapat membantu sebuah perusahaan untuk menyediakan kebutuhan perusahaan akan informasi-informasi akuntansi dan informasi-informasi lainnya yang berkaitan dengan proses bisnis yang ada dalam perusahaan tersebut.”

Sistem akuntansi dibuat bertujuan untuk menghasilkan informasi. Informasi tersebut dapat berupa laporan keuangan informasi yang dapat digunakan untuk menentukan tindakan yang harus diambil dalam sebuah keputusan. Sistem informasi menerima *input* yang biasa disebut dengan nama transaksi suatu sistem, yang kemudian diajukan atau dikirim melalui berbagai proses menjadi output informasi yang akan didistribusikan kepada para pemakai informasi yang akan mencatat dan mengolah data transaksi dan menyajikan informasi kepada pihak-pihak yang berhak dan berkepentingan untuk dibidang sistem informasi suatu perusahaan. Sistem informasi akuntansi terdiri dari lima komponen antara lain:

1. Orang-orang yang mengoperasikan sistem tersebut dan melaksanakan berbagai fungsi.
2. Prosedur-prosedur, baik manual maupun yang terotomatis, yang dilibatkan dalam mengumpulkan, memproses dan menyimpan data tentang aktivitas-aktivitas organisasi.
3. Data tentang proses-proses bisnis organisasi.
4. *Software* yang dipakai untuk memproses data organisasi.
5. Infrastruktur teknologi informasi, termasuk komputer, peralatan pendukung (*peripheral device*), dan peralatan untuk komunikasi jaringan.

Berdasarkan pengertian sistem informasi akuntansi yang telah dijelaskan, maka dapat disimpulkan bahwa sistem informasi akuntansi adalah sekumpulan komponen-komponen yang terintegrasi dan saling bekerja sama untuk mengolah data-data keuangan menjadi sebuah informasi yang dapat digunakan untuk mengambil keputusan oleh para pemegang kepentingan.

2.1.7. Pengertian Penjurnalan

Jurnal adalah rincian semua transaksi keuangan dan akun-akun yang mempengaruhi transaksi. Pencatatan semua transaksi keuangan suatu perusahaan atau instansi yang dicatat secara sistematis dan bertujuan untuk pendataan, termasuk di dalamnya jumlah transaksi, nama-nama transaksi baik mempengaruhi atau dipengaruhi dan waktu transaksi berjalan.

Menurut (Oliveira, 2016) “Jurnal Khusus adalah jurnal yang berfungsi untuk mencatat transaksi sejenis. Jurnal Khusus merupakan salah satu tahapan awal dalam siklus Akuntansi Perusahaan Dagang”.

Adapun jurnal penggajian yang merupakan catatan beban gaji yang telah dikeluarkan perusahaan. Jurnal untuk pencatatan penggajian tersebut adalah sebagai berikut:

Pencatatan jurnal gaji:

Biaya Gaji	xxx	
		Kas
		xxx

2.1.8. Pengertian Penggajian

Sumber daya manusia yang profesional diperlukan untuk mencapai tujuan perusahaan atau instansi dalam mengendalikan perusahaan atau instansi tersebut.

Sebagai imbalan kepada sumber daya manusia tersebut, perusahaan atau instansi harus menjanjikan suatu kontra prestasi dalam bentuk gaji sebagai penghargaan usaha tenaga kerja dengan jumlah yang telah ditentukan pada periode waktu tertentu.

Menurut (Hidayatun, 2016) “Gaji adalah salah satu aspek yang penting dalam perusahaan karena merupakan suatu bentuk kompensasi sebagai balas jasa yang diberikan perusahaan atas kinerja karyawannya”.

Menurut (Muhammad & Mulyani, 2017) menyimpulkan bahwa :

Gaji adalah suatu bentuk pembayaran periodik dari seorang majikan pada karyawannya yang dinyatakan dalam suatu kontrak kerja. Dari sudut pandang pelaksanaan bisnis, gaji dapat dianggap sebagai biaya yang dibutuhkan untuk mendapatkan sumber daya manusia untuk menjalankan operasi, dan karenanya disebut dengan biaya personel atau biaya gaji. Dalam akuntansi, gaji dicatat dalam akun gaji. Istilah lain dari gaji adalah honor dan upah. Gaji, honor ataupun upah dapat diterima pegawai di lingkungan kantor atau tempat kerja milik negara atau tempat swasta.

Menurut (Setiawan, Muhidin, & Ikhwana, 2017) “Penggajian atau pengelolaan gaji merupakan suatu proses tindakan balas jasa suatu organisasi kepada pegawai sebagai seorang yang memberikan sumbangan dalam tercapainya tujuan organisasi. Lamanya proses pembuatan slip gaji dan laporan dapat menghambat proses penggajian pegawai.”

Menurut (Moenir & Yuliyanto, 2017) “Gaji adalah suatu bentuk jasa yang diberikan secara teratur kepada seorang pegawai atas jasa dan hasil kerjanya.”

Menurut (Roring & Baday, 2017) “Sistem penggajian merupakan fungsi yang sangat penting untuk memberikan kompensasi kepada para pegawai berupa gaji sebagai kontribusi mereka kepada organisasi/ instansi. Penggajian merupakan salah satu proses dalam sebuah organisasi / instansi yang rentan terhadap masalah.”

Sedangkan menurut Mulyadi, sistem penggajian terdiri dari jaringan prosedur berikut ini:

1. Prosedur pencatatan waktu

Prosedur ini bertujuan untuk mencatat waktu hadir karyawan yang diselenggarakan oleh fungsi pencatatan waktu dengan menggunakan daftar hadir pada pintu masuk. Pencatatan waktu hadir dapat menggunakan daftar hadir biasa atau menggunakan kartu hadir yang diisi secara otomatis dengan menggunakan mesin pencatat waktu.

2. Prosedur pembuatan daftar gaji

Fungsi pembuat daftar gaji membuat daftar gaji karyawan. Data yang dipakai sebagai dasar pembuatan daftar gaji adalah surat-surat keputusan mengenai pengangkatan karyawan baru, kenaikan pangkat, pemberhentian karyawan, pembuatan pangkat daftar gaji bulan sebelumnya, dan daftar hadir.

3. Prosedur pembuatan bukti kas keluar

Biaya tenaga kerja didistribusikan kepada departemen-departemen yang menikmati manfaat tenaga kerja. Ini dimaksudkan untuk pengendalian biaya dan perhitungan harga pokok produksi.

4. Prosedur distribusi biaya gaji

Dalam distribusi biaya gaji, biaya tenaga kerja didistribusikan kepada departemen-departemen yang menikmati manfaat tenaga kerja. Ini dimaksudkan untuk pengendalian intern dan perhitungan harga pokok produk.

5. Prosedur pembayaran gaji

Prosedur pembayaran gaji melibatkan fungsi akuntansi dan fungsi keuangan. Fungsi akuntansi melibatkan perintah pengeluaran kas ke fungsi keuangan agar ditulis cek dan fungsi keuangan mencairkan cek ke bank untuk pembayaran gaji.

Dengan demikian disimpulkan bahwa penggajian adalah balas jasa yang diberikan oleh perusahaan atau instansi dalam bentuk pembayaran sebagai balasan atas seseorang karena telah melaksanakan tugas dan jasanya terhadap perusahaan atau instansi yang bersangkutan.

2.1.9. Metode Pengembangan *Waterfall*

Nama model *Waterfall* sebenarnya adalah *Linear Sequential Mode*. Model ini sering disebut dengan *classis life cycle* atau model *waterfall*. *Waterfall* pertama kali diperkenalkan oleh Winston Royce sekitar tahun 1970 sehingga sering dianggap kuno, tetapi merupakan model yang paling banyak dipakai didalam *software Engineering* (SE). *Waterfall* melakukan pendekatan secara sistematis dan berurutan. Disebut dengan *Waterfall* karena tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya dan berjalan berurutan.

Menurut (Pressman (2016) “*Waterfall* adalah aplikasi penerapan dari penemuan permasalahan (problem solving) yang didapat dari pendekatan sistem (system approach) menjadi pengembangan dari solusi sistem informasi terhadap masalah bisnis”.

Menurut (Staiger, 2018) “*Waterfall* yang terbagi menjadi empat tahapan, yaitu:

- a) Analisa kebutuhan perangkat lunak Proses pengumpulan kebutuhan yang dilakukan secara instensif untuk mespesifikasikan kebutuhan perangkat lunak agar dapat dipahami, seperti: Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu didokumentasikan.
- b) Desain Desain perangkat lunak adalah proses multi/langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur

perangkat lunak, representasi antarmuka, dan prosedur pengkodean. Penulis menggunakan Adobe Dreamweaver CS6, ERD, LRS, dan Struktur Navigasi.

- c) Pembuatan kode program Desain harus ditranslasikan kedalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain. Penulis menggunakan kode program dengan script PHP, java script, CSS.
- d) Pengujian Pengujian fokus pada perangkat lunak secara dari segi logik fungsional untuk memastikan bahwa semua bagian sudah diuji. Penulis menggunakan black box testing Hal ini dilakukan untuk meminimalisir kesalahan (error), dan memastikan keluaran yang dihasilkan sesuai denganyang diinginkan.

2.1.10. Database (Basis Data)

Menurut (Andaru, 2018) menjelaskan bahwa :

Database atau basis data adalah kumpulan informasi yang disimpan di dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut. Kegunaan utama sistem basis data adalah agar pemakai mampu menyusun suatu pandangan (*view*) abstraksi data. Hal ini bertujuan untuk menyederhanakan interaksi antara pengguna dengan sistemnya dan basis data dapat mempresentasikan pandangan yang berbeda kepada para pengguna, programmer, dan administratornya.

Relational Database Management System (RDBMS) adalah program yang melayani sistem basis data yang entitas utamanya terdiri dari tabel-tabel yang mempunyai relasi dari satu tabel ke tabel lain. Salah satu jenis *Relational Database Management System (RDBMS)* yang didistribusikan secara gratis dibawah lisensi GPL (*General Public Licence*) adalah MySQL. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam *Database* sejak lama, yaitu SQL (*Structured*

Query Language). SQL merupakan sebuah konsep pengoperasian database, terutama untuk pemilihan atau seleksi dan pemasukan data yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis. Contoh MySQL dan APACHE, XAMPP, Wamp Server.

Pada dasarnya, penggunaan data pada MySQL dilakukan atas dukungan *PHPMysqlAdmin* yang berperan sebagai administrator MySQL melalui *browser (web)*.

1. MYSQL

Menurut (Achmad Yusron Arif, 2019) “MySQL adalah sebuah software atau perangkat lunak system manajemen berbasis data SQL atau juga multi user dan DBMD Multithread. Pada dasarnya, MySQL ini sebenarnya adalah turunan yang berasal dari salah satu konsep utama dalam database yang memang telah ada sebelumnya yaitu SQL atau Structured Query Language”.

Beberapa kelebihan MySQL, antara lain-lain:

- a. MySQL dapat berjalan dengan stabil pada berbagai sistem operasi, seperti Windows, Linux, FreeBSD, Mac Os X Server, Solaris, dan masih banyak lagi.
- b. Bersifat *Open Source*, MySQL didistribusikan secara *Open Source* (gratis), dibawah lisensi *General Public Licence* (GPL).
- c. Bersifat *Multiuser*, MySQL dapat digunakan oleh beberapa *user* dalam waktu yang bersamaan tanpa mengalami masalah.
- d. MySQL memiliki kecepatan yang baik dalam menangani *query* (perintah SQL). Dengan kata lain dapat memproses lebih banyak SQL persatuan waktu.
- e. Dari segi *security* atau keamanan data, MySQL memiliki beberapa lapisan sekuriti, seperti *level subnet mask*, nama *host*, dan izin akses *user* dengan

sistem perizinan yang mendetail serta *password* yang dikriptasikan secara sistematis.

- f. Selain MySQL bersifat fleksibel dengan berbagai pemrograman, MySQL juga memiliki *interface* (antar muka) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan aplikasi API (*Application Programming Interface*).
- g. Dukungan banyak komunitas, biasanya tergabung dalam sebuah forum untuk saling berdiskusi membagi informasi tentang MySQL.

2. PHPMyAdmin

“*PhpMyAdmin* adalah sebuah aplikasi open source yang berfungsi untuk memudahkan manajemen MySQL. Dengan menggunakan *PhpMyAdmin*, dapat membuat database, membuat table, menginsert, menghapus dan mengupdate data dengan GUI dan terasa lebih mudah, tanpa perlu mengetikkan perintah SQL secara manual”. Madcoms (2016).

3. XAMPP

Menurut (Ratnasari, 2018) menyimpulkan bahwa :

XAMPP ialah perangkat lunak bebas yang mendukung banyak sistem operasi, merupakan campuran dari beberapa program. Yang mempunyai fungsi sebagai server yang berdiri sendiri (localhost), yang terdiri dari program MySQL database, Apache HTTP Server, dan penerjemah ditulis dalam bahasa pemrograman PHP dan Perl. Nama XAMPP merupakan singkatan dari X (empat sistem operasi), Apache, MySQL, PHP dan Perl. Program ini tersedia di bawah GNU General Public License dan bebas, adalah mudah untuk menggunakan web server yang dapat melayani tampilan halaman web yang dinamis.

4. JAVA

Menurut (Yuana, 2015) “Java adalah bahasa pemrograman dan platform komputasi pertama kali dirilis oleh Sun Microsystems pada tahun 1995. Java

merupakan teknologi yang mendasari kekuatan program untuk utilitas, permainan, dan aplikasi bisnis. Java berjalan pada lebih dari 850 juta komputer pribadi di seluruh dunia, dan pada miliaran perangkat di seluruh dunia, termasuk ponsel dan perangkat TV.”

2.2. Peralatan Pendukung (*Tools System*)

Peralatan pendukung merupakan suatu penjelasan peralatan-peralatan pendukung yang dipakai dalam membuat suatu sistem usulan. Media pendukung sistem merupakan alat yang dapat digunakan untuk menggambarkan bentuk logika model dari suatu sistem, dimana simbol-simbol, lambang-lambang dan diagram-diagram menunjukkan secara arti fisiknya.

2.2.1. Pengertian UML (*Unified Modeling Language*)

Unified Modeling Language (UML) adalah tujuan umum, perkembangan, bahasa pemodelan di bidang rekayasa perangkat lunak , yang dimaksudkan untuk menyediakan cara standar untuk memvisualisasikan desain sistem.

UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak.

Menurut (Kurniawan, 2018) “UML sudah menjadi bahasa pemodelan baku dalam pengembangan sistem perangkat lunak. Pemodelan dalam UML yang penting guna menjelaskan aspek fungsionalitas sistem adalah pemodelan *use case*, yang dideskripsikan secara tekstual dalam bentuk *use case scenario* untuk menggambarkan interaksi yang terjadi antara aktor dengan sistem”.

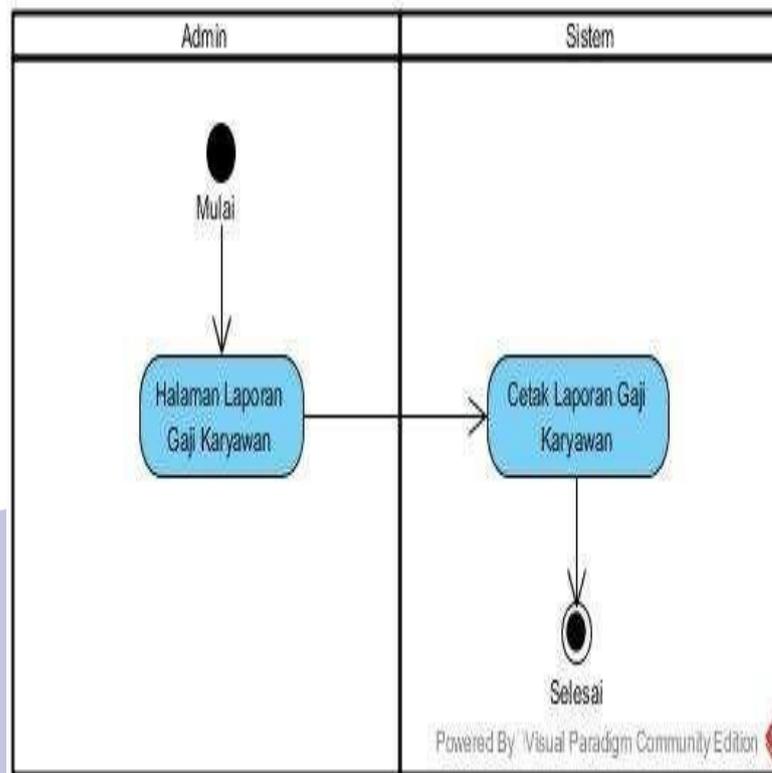
1. *Activity Diagram*

Activity Diagram menggambarkan tentang aktifitas yang terjadi pada sistem dari pertama sampai akhir.

Menurut (Aghniya, 2019) mendefinisikan bahwa, “activity diagram memodelkan workflow proses bisnis dan urutan aktifitas dalam sebuah proses. Diagram ini sangat mirip dengan flowchart karena memodelkan workflow dari suatu aktifitas lainnya atau dari aktifitas ke status”.

Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut:

- a. Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
- b. Urutan atau pengelompokan tampilan dari sistem/ *user interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan.
- c. Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya,
- d. Rancangan menu yang ditampilkan pada perangkat lunak.



Sumber : (Moenir, Ardianto dan Yuliyanto Fajar: 2017)

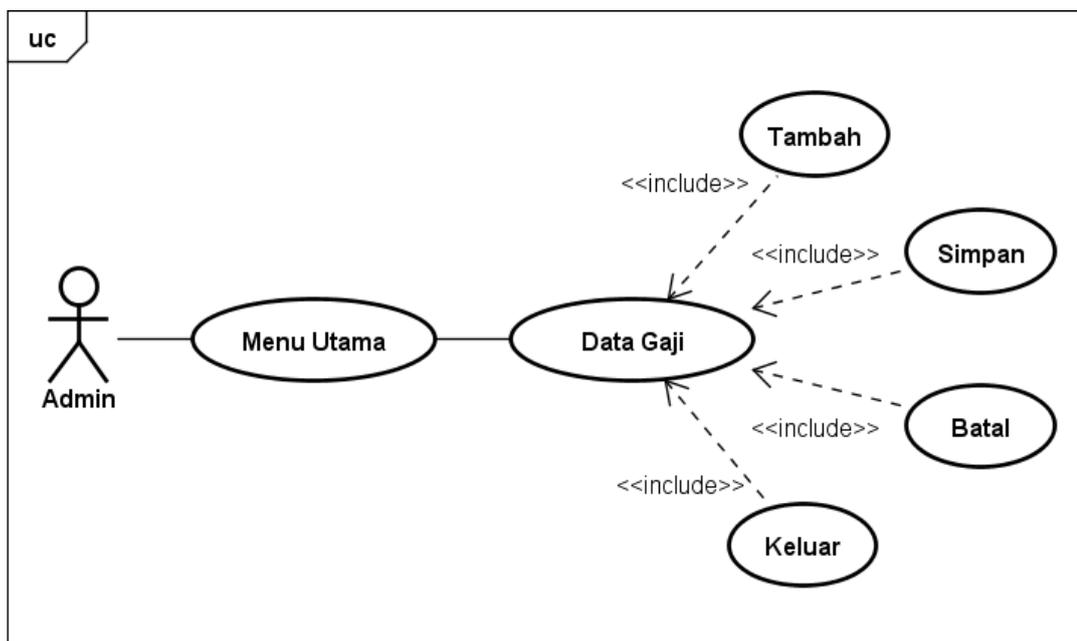
Gambar II.2

Contoh Activity Diagram Laporan Gaji Karyawan

2. *UseCase Diagram*

UseCase Diagram digunakan untuk menggambarkan secara ringkas siapa yang menggunakan sistem dan apa saja yang bisa dilakukannya.

Menurut (Aghniya, 2019) “Menyimpulkan bahwa, “use case adalah rangkaian atau uraian sekelompok yang saling terkait dan membentuk sistem secara teratur yang dilakukan atau diawasi oleh sebuah aktor”.



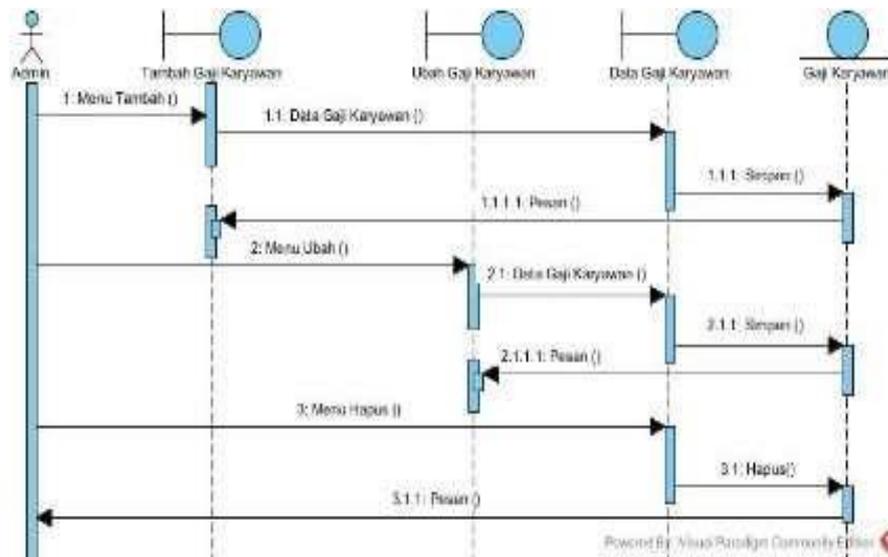
Sumber : (Astria Setia Mulyani, 2019)

Gambar II.3
Contoh Usecase Diagram Data Gaji

3. *Sequence Diagram*

Sequence Diagram adalah sebuah diagram yang menggambarkan interaksi antar objek di dalam sebuah sistem. Interaksi tersebut berupa *message* yang digambarkan terhadap waktu.

Menurut (Hendini, 2018) “*Sequence Diagram* menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek”. Untuk membangun database system informasinya system ini perlu melibatkan Konsep data model ERD (entity relational diagram) dan LRS (logical record structure).



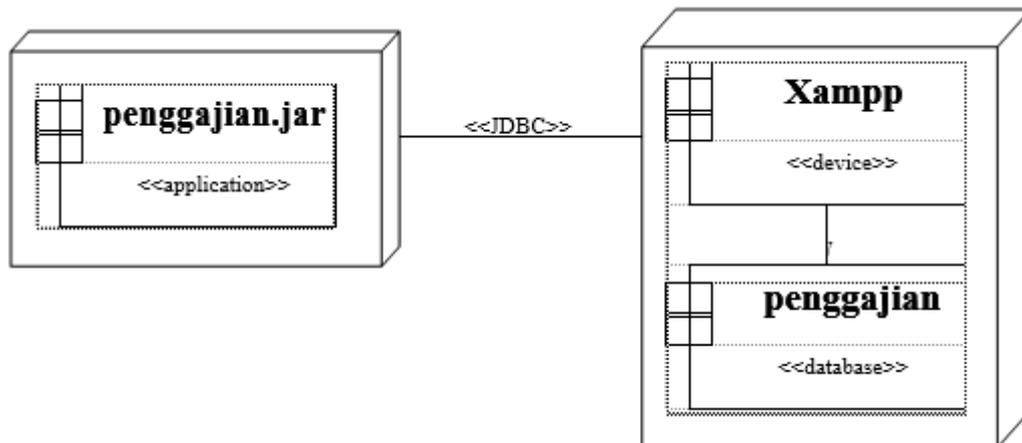
Sumber : (Moenir, Ardianto dan Yuliyanto Fajar: 2017)

Gambar II.4
Contoh Sequence Diagram Gaji Karyawan

4. Deployment Diagram

Diagram *deployment* atau *deployment diagram* yaitu salah satu diagram pada UML yang menunjukkan tata letak suatu sistem secara fisik, dapat juga dikatakan untuk menampilkan bagian-bagian software yang terdapat pada hardware dan digunakan untuk menerapkan suatu sistem dan hubungan antara komponen hardware. Jadi deployment diagram intinya untuk menunjukkan letak software pada hardware yang digunakan sistem.

Menurut (Shalahuddin & Rosa, 2015) “*Deployment Diagram* menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi”.



Sumber : (Astria Setia Mulyani, 2019)

Gambar II.5

Contoh Deployment Diagram Sistem Pembayaran Gaji

Berdasarkan pengertian, dapat disimpulkan bahwa UML merupakan bahasa yang mempermudah pembaca dalam memahami cara berjalannya sebuah sistem perangkat lunak dengan menggunakan sebuah gambar.

2.2.2. Pengertian Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) adalah model teknik pendekatan yang menyatakan atau menggambarkan hubungan suatu model.

Menurut (Ibeng, 2018) “ERD adalah suatu model untuk menjelaskan mengenai hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang memiliki hubungan antar relasi. Entity Relationship Diagram oyi untuk memodelkan struktur data serta hubungan antar data, untuk dapat menggambarkannya digunakan beberapa notasi serta simbol”.

Komponen-komponen dalam *Entity Relationship Diagram* (ERD) adalah sebagai berikut :

1. Entitas (*Entity*)

Entitas merupakan data inti yang akan disimpan; bakal *table* pada baris data; benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel.

2. Atribut

Field atau kolom data yang disimpan dalam suatu entitas.

3. Atribut Kunci Primer

Field atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses record yang diinginkan; biasanya berupa id; kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama).

4. Atribut Multinilai (*Multivalue*)

Field atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu.

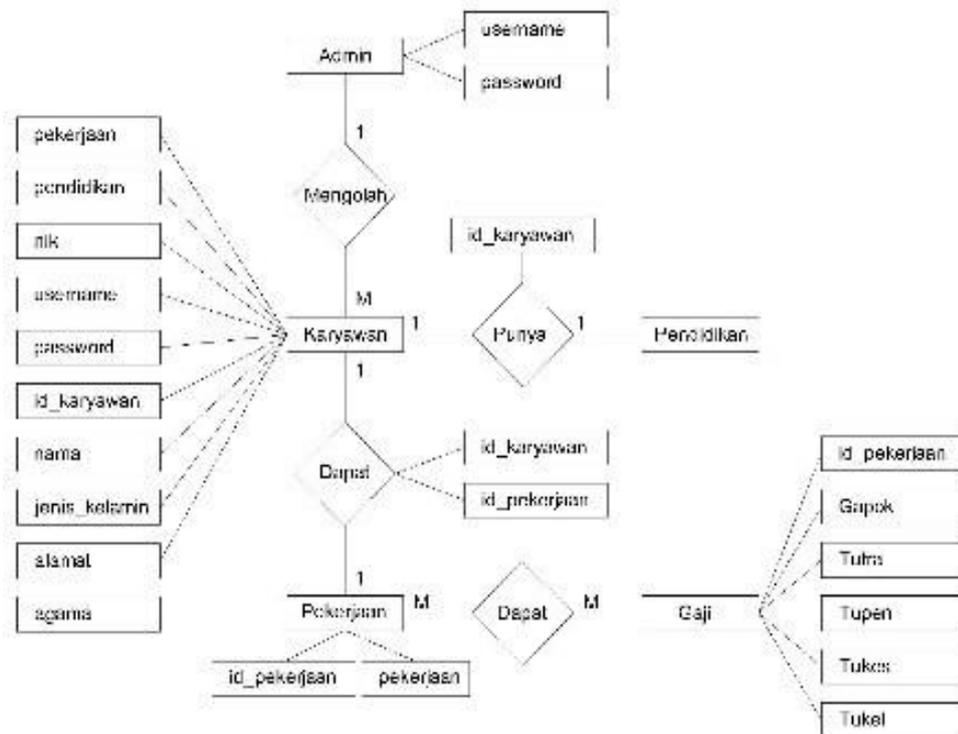
5. Relasi (*Relation*)

Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja.

6. Asosiasi (*Association*)

Penghubung antara relasi dan entitas di mana di kedua ujungnya memiliki *multiplicity* kemungkinan jumlah pemakaian.

Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas 1 ke N atau sering disebut dengan *one to many* menghubungkan entitas A dan entitas B.



Sumber : (Moenir, Ardianto dan Yuliyanto Fajar: 2017)

Gambar II.6

Contoh Gambar ERD Sistem Penggajian Karyawan

Dari penjelasan di atas dapat disimpulkan bahwa *Entity Relationship Diagram* (ERD) pemodelan basis data dengan menggunakan diagram relasi antar entitas, untuk memodelkan struktur data dan hubungan antar data dan digambarkan dengan menggunakan beberapa notasi dan simbol.

2.2.3. Pengertian Logical Record Structure (LRS)

Logical Record Structure (LRS) merupakan representasi dari struktur *record-record* pada tabel-tabel yang terbentuk dari hasil kelas antar himpunan entitas pada diagram E-R.

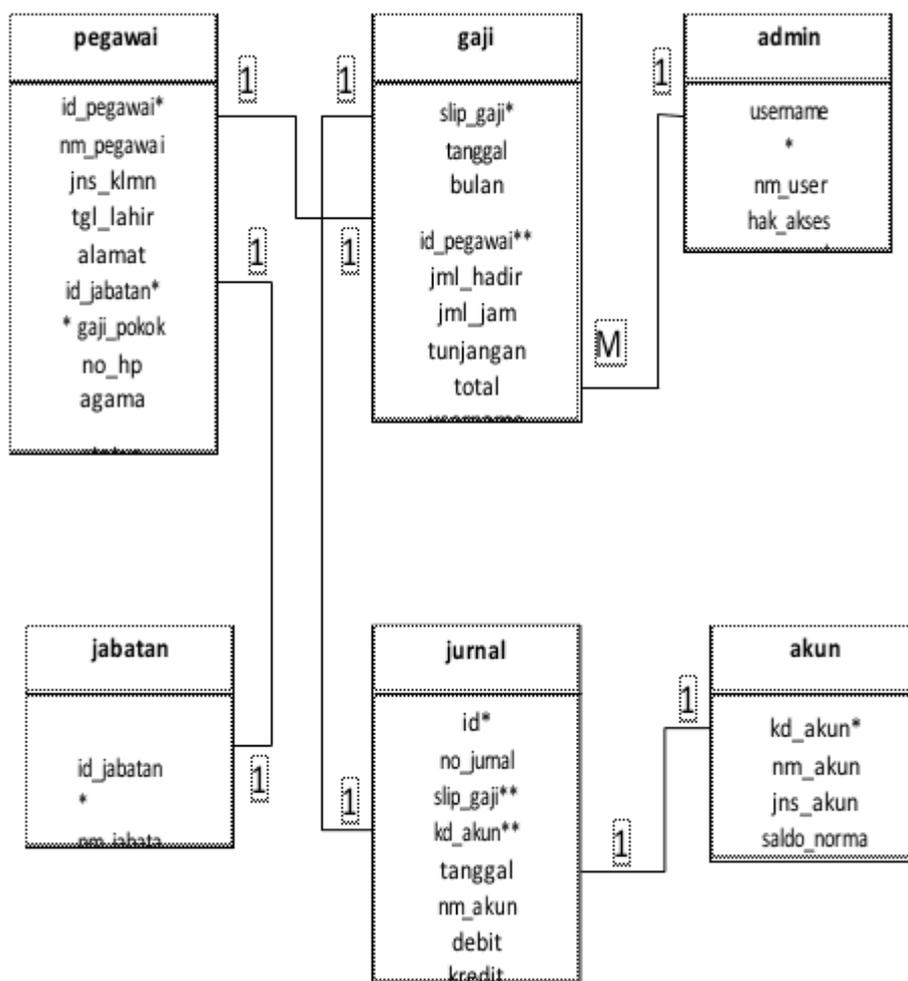
Menurut (Rahmawati, 2016) menyimpulkan bahwa :
Logical Record Structure (LRS) terdiri dari link-link tipe record. Link ini menunjukkan arah dari satu tipe record lainnya. Banyak link dari LRS yang diberi tanda field-field yang kelihatan pada kedua link tipe record. Penggambaran LRS mulai dengan menggunakan model yang dimengerti". Menurut pendapat di atas dapat disimpulkan Logical Record Structure (LRS) merupakan transformasi ERD ke LRS dimulai proses kardinalitas dan menghilangkan atribut-atribut yang saling berelasi dan memberikan batasan LRS adalah model system yang digambarkan dengan sebuah ER yang mengikuti pola permodelan tertentu dalam kaitannya dengan konvensi ke LRS.

Berikut adalah cara membentuk skema database atau *Logical Record Structured*

(LRS) berdasarkan *Entity Relationship Diagram* :

1. Jika relasinya satu-ke-satu maka *foreign key* diletakan pada salah satu dari dua entitas yang ada atau menyatukan kedua entitas tersebut.
2. Jika relasinya satu-ke-banyak, maka *foreign key* dilekatkan pada entitas *Many*.
3. Jika relasinya banyak-ke-banyak, maka dibuat file konektor yang berisi dua *foreign key* yang berasal dari kedua entitas.





Sumber : (Astria Setia Mulyani, 2019)

Gambar II.7
Contoh LRS Sistem Pembayaran Gaji

Dari penjelasan diatas dapat disimpulkan bahwa *Logical Record Structured* (LRS) merupakan suatu *record* dari tabel yang ada pada *database* dan direlasikan untuk mempermudah logika dari suatu program yang dibuat.

2.2.4. User Interface

Menurut (Ramadhan, 2017) menjelaskan bahwa :

Antarmuka Pemakai (*User Interface*) merupakan mekanisme komunikasi antara pengguna (*user*) dengan sistem. Antarmuka pemakai (*User Interface*) dapat menerima informasi dari pengguna (*user*) dan memberikan informasi kepada pengguna (*user*) untuk membantu mengarahkan alur penelusuran masalah sampai ditemukan suatu solusi. Tujuan dari user interface adalah sebagai penerjemah yang menghubungkan manusia dengan sistem operasi. sebagai media komunikasi antara pengguna dengan sistem operasi, maka sebaiknya ui di tampilkan dalam bentuk yang sederhana dan menarik, agar pengguna mengerti dan dapat menggunakan sistem tersebut dengan baik. Selain itu tampilan ui yang menarik membuat pengguna menjadi nyaman.

The screenshot shows a web application window titled "DATA GAJI". At the top right, there is a field for "Admin" with the value "Astria SM". Below this is a form with several input fields: "ID Slip Gaji", "Tanggal Cetak" (2019-09-08), "Bulan" (September), "ID Pegawai", "Nama Pegawai", "Jabatan", "Gaji Pokok", "Jumlah Hadir", "Jumlah Jam", "Tunjangan", and "Total Gaji". A "Pilih" button is next to the "ID Pegawai" field. Below the form are four buttons: "Tambah", "Simpan", "Batal", and "Keluar". At the bottom, there are two tables. The first table shows transactions with columns: Tanggal, No Jurnal, ID Slip Gaji, Kode Akun, Nama Akun, Debet, and Kredit. The second table shows employee data with columns: ID Slip Gaji, Tanggal, Bulan, ID Pegawai, Hadir, Jam, Tunjangan, Total, and Admin.

Tanggal	No Jurnal	ID Slip Gaji	Kode Akun	Nama Akun	Debet	Kredit
2019-09-08	J0809190...	G080919...	511	Biaya Gaji	1540000	0
2019-09-08	J0809190...	G080919...	101	Kas	0	1540000

ID Slip Gaji	Tanggal	Bulan	ID Pegawai	Hadir	Jam	Tunjangan	Total	Admin
G080919001	2019-09-08	September	G01	0	48	100000	1540000	Astria SM

Sumber : (Astria Setia Mulyani, 2019)

Gambar II.8

Contoh User Interface Pembayaran Gaji

2.2.5. Code Generation

Perancangan Program (*Code Generation*) Pembuatan program harus sesuai dengan perancangan dan desain yang telah dibuat sebelumnya. Penulisan program dilakukan menggunakan bahasa pemrograman PHP dengan *Adobe Dreamweaver CS6* dan basis data *Mysql*.

2.2.6. Blackbox Testing

Metode *Blackbox testing* memfokuskan pada keperluan fungsional dari *software*. Karena itu *Blackbox testing* memungkinkan pengembangan *software* untuk membuat himpunan kondisi input yang akan melatih seluruh syarat-syarat fungsional suatu program.

Menurut (Suri Karuniawati, Sri Widowati, Ir., MT.2, Iman Lukmanul Hakim SMB, 2015) menjelaskan bahwa :

Pengembangan sebuah perangkat lunak harus melalui proses penjaminan mutu perangkat lunak. Untuk mendapatkan perangkat lunak dengan kualitas yang bagus maka pengembangan perangkat lunak harus melalui proses penjaminan mutu. Salah satu proses penjaminan mutu perangkat lunak adalah pengujian perangkat lunak. Salah satu aspek penting dalam pengujian perangkat lunak adalah pembangkitan kasus uji. Dalam pembangkitan kasus uji terdapat beberapa teknik yang digunakan diantaranya *Whitebox testing* dan *Blackbox testing*. *Blackbox testing* merupakan pengujian yang dapat membangkitkan kasus uji dengan menggunakan spesifikasi kebutuhan perangkat lunak. Pada Tugas Akhir ini dibangun sebuah aplikasi yang mengimplem ntasikan metode *Cause Effect Graphing* pada teknik *Blackbox testing* yang dapat menghasilkan kasus uji dengan menggunakan spesifikasi kebutuhan perangkat lunak dengan tingkat kebenaran sebesar 100 % dan dapat mereduksi kasus uji sekitar 90%.

No	Skenario Pengujian	Test Case	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
1	<i>Input Data Gaji</i> (Data yang di input tidak lengkap) lalu klik Simpan	Data tidak diisi lengkap	Sistem tidak akan menyimpan ketika kolom tidak terisi semua, maka akan menampilkan pesan "Terdapat inputan yang kosong !"	Sesuai	Valid
2	<i>Input Numerik</i> (Jumlah Jam / Jumlah Hadir) dengan Huruf	Data Numerik diisi dengan Huruf	Sistem tidak mendukung untuk menampilkan <i>field</i> Total	Sesuai	Valid
3	<i>Input Data Gaji</i> (data yang diisi lengkap) lalu klik Simpan	Data diisi lengkap	Sistem menyimpan dan menampilkan pesan "Data Tersimpan."	Sesuai	Valid

Sumber : (Astria Setia Mulyani, 2019)

Gambar II.9

Contoh Blackbox Testing Pembayaran Gaji

