**BAB II**

**LANDASAN TEORI**

**2.1 Tinjauan Pustaka**

**A. Konsep Dasar Sistem Informasi**

Beberapa definisi Sistem Informasi adalah sebagai berikut:

Menurut O’brian dalam Yakub (2012:16) menjelaskan bahwa: “sistem informasi (*information system)* merupakan kombinasi teratur dari orang-orang, perangkat keras (*hardware),* perangkat lunak (*software),* dan sumber daya mengumpulkan dan menyebarkan informasi dalam sebuah organisasi”.

 Menurut Sutabri (2012:38) mengemukakan bahwa:

 Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan laporan-laporan yang diperlukan oleh pihak luar tertentu.

 Dari pengertian diatas maka dapat disimpulkan bahwa sistem informasi yaitu kesatuan sistem yang bertujuan untuk menghasilkan informasi dan sebuah sistem yang terdiri dari pengumpulan, pemasukan, pemrosesan data, penyimpanan, pengolahan, pengendalian, dan pelaporan sehingga tercapai sebuh informasi yang mendukung pengambilan keputusan didalam suatu organisasi untuk mencapai sasaran dan tujuannya.

**B. Pengertian Penggajian**

Gaji merupakan salah satu hal yang mendorong atau memotivasi karyawan untuk tetap bekerja atau mengabdi secara menyeluruh terhadap perusahaan. Berikut beberapa definisi gaji menurut sumber literatur:

 Menurut Soemarso (2009:307) menjelaskan bahwa ”Gaji adalah imbalan kepada pegawai yang diberikan atas tugas-tugas administrasi dan pimpinan yang jumlahnya biasanya tetap secara bulanan”.

 Menurut Mardi (2011:107) menjelaskan bahwa “Gaji adalah sebuah bentuk pembayaran atau sebuah hak yang diberikan oleh sebuah perusahaan atau instansi kepada pegawai”.

 Menurut Mulyadi (2008:370) menjelaskan bahwa “Gaji umumnya merupakan pembayaran atas penyerahan jasa yang dilakukan oleh karyawan yang mempunyai jenjang jabatan manajer”.

 Dari definisi diatas dapat disimpulkan bahwa gaji merupakan suatu kompensasi yang dibayarkan oleh perusahaan kepada karyawan sebagai balas jasa atas kinerja yang telah diberikan terhadap perusahaan. Kompensasi tersebut diberikan secara periodik (bulanan) kepada karyawan sebagai bentuk imbalan yang berdasarkan *level* jabatan, kinerja, dan atas dasar pertimbangan lainnya. Gaji merupakan salah satu faktor pendorong motivasi karyawan dalam melaksanakan kegiatan dalam pekerjaannya, karyawan akan lebih giat dalam melaksanakan pekerjaan dan kewajiban-kewajiban lainnya.

**C. Bahasa Pemrograman**

**1. *Hypertext Preproccesor (PHP)***

 Menurut Anhar dalam Wijayanti (2014:32) mendefinisikan ,”*PHP* singkatan dari *Hypertext Preprocessor* yaitu bahasa pemrograman *web server-side* yang bersifat *open source”.*

 *PHP* merupakan *script* yang berintegrasi dengan *HTML ,* dan berada pada *server (server side HTML embeded scripting). PHP* adalah *script* yang digunakan untuk membuat halaman *website* yang dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat saat halaman itu dimana oleh *client.* Mekanisme ini menyebabkan informasi yang diterima *client* selalu yang terbaru (*up to date).* Semua *script* dieksekusi pada *server* dimana *script* tersebut dijalankan .

**2. *HypertText Markup Language (HTML)***

 Menurut Oktavian (2010:13),menjelaskan bahwa*“HTML(Hypertext Markup Language)* adalah suatu bahasa yang dikenali oleh *web browser* untuk menampilkan informasi dengan lebih menarik dibandingkan dengan tulisan teks biasa *(plain text)”*.

 Menurut Simarmata (2010:52),menjelaskan bahwa: ”*HTML* adalah bahasa *markup* untuk menyebaran informasi pada *web*. Ketika merancang *HTML*, ide ini diambil dari S*tandart Generalized Markup Language (SGML)*. *SGML* adalah cara yang terstandarisasi dari pengorganisasian dan informasi yang terstruktur di dalam dokumen atau sekumpulan dokumen. Walaupun *HTML* tidak dengan mudah dapat dipahami kebanyakan orang, ketika diterbitkan penggunaanya menjadi jelas”.

 Menurut Arief (2011:23),”*HTML* atau *HyperTextMarkup Language* merupakan salah satu format yang digunakan dalam pembuatan dokumen dan aplikasi yang berjalan di halaman *web*. Dokumen ini dikenal sebagai *web page*. Dokumen *HTML* merupakan dokumen yang disajikan pada *web browser*”.

 Berdasarkan pendapat yang dikemukakan diatas dapat ditarik kesimpulan bahwa, *HTML* adalah bahasa yang sangat tepat dipakai untuk menampilkan informasi pada halaman *web,* karena *HTML* menampilkan informasi dalam bentuk *hypertext* dan juga mendukung sekumpulan perintah yang dapat digunakan untuk mengatur informasi tersebut, sesuai dengan namanya bahasa ini menggunakan tanda *markup* untuk menandai perintah-perintahnya.

**3.** ***Dreamweaver CS6***

Menurut Madcoms (2010:11), “untuk memahami elemen ruang kerja *Dreamweaver CS6*, bukanlah sebuah file *HTML* baru dalam *Dreamweaver CS6”*. Setelah *Dreamweaver CS6* berhasil dijalankan, pilih file baru *(create new)* tipe *HTML* di dalam kotak dialog *New Document* yang tampil. Ruang kerja dasar *Dreamweaver CS6*, yaitu:

1. *Application Bar*, berada di bagian paling atas jendela aplikasi *Dreamweaver* *CS6*. Baris ini berisi tombol *workspace (workspace switcher)*, CS Live, menu, dan aplikasi lainnya.
2. *Toolbar Document*, berisi tombol-tombol yang digunakan untuk mengubah tampilan jendela dokumen, sebagai contoh tampilan *design* atau tampilan *code*. Juga dapat digunakan untuk operasi-operasi umum, misalnya untuk melihat hasil sementara halaman *web* pada jendela *browser*.
3. Jendela Dokumen adalah lembar kerja tempat membuat dan mengedit *design* halaman *web*.
4. *Workspace Switcher*, digunakan untuk mengubah tampilan ruang kerja *Dreamweaver CS5*. Sebagai contoh mengubah tampilan menjadi tampilan classic, yaitu tampilan ruang kerja *dreamweaver* versi sebelumnya.
5. *Panel Groups*, adalah kumpulan panel yang saling berkaitan, panel-panel ini dikelompokan pada judul-judul tertentu berdasarkan fungsinya. Panel ini digunakan untuk memonitor dan memodifikasi pekerjaan. Secara default, panel group berisi panel *Adobe BrowserLab*, *Adobe Bussiness Catalyst*, *Insert, CSS* *Styles, Asset, AP Element*, dan *Files*.
6. *Tag Selector*, diletakkan di bagian bawah jendela dokumen, satu baris dengan status bar. Bagian ini menampilkan hirarki pekerjaan yang sedang terpilih pada jendela dokumen, dapat juga digunakan untuk memilih objek pada jendela design berdasarkan jenis atau kategori objek tersebut. *Tag selector* juga menampilkan informasi format dari bagian yang sedang aktif pada lebar kerja *desig*n.
7. *Property Inspector*, digunakan untuk melihat dan mengubah berbagai properti objek atau teks pada jendela *design*. Properti untuk satu objek dengan objek lainnya selalu berbeda-beda. Jendela ini tidak dapat diuraikan pada tampilan jendela *code*.
8. *Toolbar Standart*, baris toolbar ini berisi tombol-tombol yang mewakili perintah pada menu *file* dan *edit*, diantaranya perintah *New, Open, Save, Save* *All, Cut, Copy, Paste, Undo*, dan *Redo*.
9. *Toolbar Style Rendering*, secara default disembunyikan. *Toolbar* ini berisi tombol-tombol untuk menampilkan *design* dalam media berbeda. Selain itu juga digunakan untuk mengaktifkan dan menonaktifkan style *CSS*.
10. *Toolbar Coding*, berisi tombol-tombol yang digunakan untuk melakukan operasi kode-kode standar. *Toolbar* ini hanya tampil pada jendela *code*.
11. *Toolbar Browser Navigation*, *toolbar* ini adalah toolbar baru yang ada di dalam *Dreamweaver CS6*, dan letaknya tepat berada di atas jendela dokumen. *Toolbar* ini berisi tombol-tombol yang digunakan sebagai navigasi di dalam *browser*.
	* 1. **CSS (*Cascanding Style Sheet*)**

Menurut Alexander (2013 : 112) “*Cascading Style Sheet* (CSS) dikembangkan untuk menata gaya pengaturan halaman *website.* Pada awalnya CSS dikembangkan pada SGML pada tahun 1970, dan terus dikembangkan hingga saat ini CSS telah mendukung banyak bahasa *Markup* seperti: HTML, XHTML, XML, SVG (*Scalable Vector Graphics*) dan *Mozilla XUL* (*XML User Interface* *Language*). Mengacu dari arti bahasa, *Cascanding Style Sheet* (CSS) memiliki arti gaya menata halaman bertingkat, yang berarti setiap satu elemen yang telah diformat dan memiliki anak dan telah diformat, maka anak dari elemen tersebut secara otomatis mengikuti format elemen induknya.

Pada bulan Desember tahun 1996 W3C mengenalkan spesifikasi CSS level 1 atau juga dikenal CSS1 yang mendukung format huruf, warna pada teks. Kemudian bulan Mei tahun 1998 W3C mengeluarkan CSS2yang didalamnya terdapat fungsi pengaturan tata letak elemen, dan saat ini W3C telah memperbaiki serta meningkatkan kemampuan CSS2 menjadi CSS3. *Cascanding Style Sheet* (CSS) terdiri dari *Selector,* Properti dan Nilai. Seperti halnya HTML PHP dan bahasa pemograman lainnya, CSS juga memiliki aturan penulisan.

**D. *Database***

1. Definisi *Database*

 Menurut Prasetio (2012:181),”*Database* adalah sebuah struktur yang umumnya dikategorikan dalam 2 (dua) hal, sebuah *database flat* dan sebuah *database relasional. Database relasional* lebih disukai karena lebih masuk akal dibandingkan *database flat*”.

 Menurut Raharjo (2013:3),”*Database* adalah kumpulan data yang terintegrasi dan diatur sedemikian rupa sehingga data tersebut dapat dimanipulasi, diambil, dan dicari secara cepat”.

 Menurut Kustiyaningsih (2011:146),”*Database* adalah struktur penyimpanan data untuk menmbah, mengakses dan mmproses data yang disimpan dalam sebuah *database* komputer, diperlukan sebuah sistem manajemen *database* seperti *MySQL* *Server”.*

 Berdasarkan beberapa pendapat para ahli yang dikemukakan di atas dapat ditarik kesimpulan bahwa *database* adalah sekelompok data yang mempunyai ciri-ciri khusus dan dapat dikelola sedemikian rupa sehingga bisa menghasilkan sebuah format data yang baru.

1. Jenis *Database* yang digunakan
2. *Web Server*

Menurut Anhar (2010:4), “Definisi *Web Server* adalah sebagai berikut: *Web* *server* adalah aplikasi yang berfungsi untuk melayani permintaan pemanggilan alamat dari pengguna melalui *web browser*, dimana *web server* mengirimkan kembali informasi yang diminta tersebut melalui *HTTP (Hypertext Transfer* *Protocol)* untuk ditampilkan ke layar *monitor* komputer kita. Agar kita dapat mengubah isi dari *website* yang dibuat, kita membutuhkan program *PHP”. Script-script PHP* tersebut yang berfungsi membuat halaman *website* menjadi dinamis. Dinamis artinya pengunjung *web* dapat memberikan komentar saran atau masukan pada *website* kita. *Website* yang kita buat menjadi lebih hidup karena ada komunikasi antara pengunjung dan kita sebagai *web master*nya.

 Menurut Oktavian (2010:11), “*Web Server* adalah aplikasi yang berguna untuk menerima permintaan informasi dari pengguna melalui *web browser*, dan mengirimkan permintaan kembali informasi yang diminta melalui *HTTP* (*HyperText Transfer Protocol*). Biasanya *web server* diletakkan di komputer tertentu pada *web hosting*”.

 Berdasarkan beberapa pendapat yang dikemukakan diatas, maka dapat disimpulkan *web server* merupakan sebuah perangkat lunak yang bertugas menerima permintaan client melalui port *HTTP* maupun *HTTPS* dan merubah isi yang ada ke dalam format *HTML*.

1. XAMPP

 XAMPP merupakan sebuah *tools* yang menyediakan beberapa paket perangkat lunak ke dalam satu buah paket. Dengan meng*instal* XAMPP tidak perlu lagi melakukan instalas dan konfigurasi *web server Apache, PHP,* dan *MySQl* secara manual. *XAMPP*  akan menginstalasi dan mengkonfigurasi secara otomatis. Fungsinya adalah sebagai *server* yang berdiri sendiri (*localhost)*, yang terdiri atas program *Apache HTTP Server, MySQL database,* dan penerjemah bahasa yang ditulis dengan bahasa pemrograman *PHP* dan *Perl.* Nama *XAMPP* merupakan singkatan dari X (empat sistem operasi apapun), *Apache, MySQL ,PHP* dan *Perl.* Program ini tersedia dalam GNU (*General Public License)* dan bebas, merupakan *web server* yang mudah digunakan yang dapat melayani tampilan halaman *web* yang dinamis.

1. *PhpMyAdmin*

Menurut Nugroho (2010:88), “*PhpMyAdmin* adalah suatu aplikasi *Open* *Source* yang berbasis *web*, aplikasi ini dibuat menggunakan program *PHP,* fungsi aplikasi ini adalah untuk mengakses database *MySQL*”.

 Dengan adanya aplikasi ini akan sangat mempermudah dan mempersingkat kerja kita dalam mengelola *database MySQL*. Dengan adanya kelebihan yang dimilikinya mengakibatkan para pengguna awam tidak harus mampu untuk mengetahui perintah-perintah *MySQL* dalam pembuatan *database* dan *tabel*.

 Menurut Arief (2011:429), “*PhpMyAdmin* adalah salah satu aplikasi berbasis *GUI (Graphical User Interface)* yang digunakan untuk mengelola *database MySQL*”.

 Menurut Prasetio (2012:53), “*PhpMyadmin* merupakan *tools berbasis web* yang berguna untuk mengelola *database MySQL*”.

 Berdasarkan beberapa pendapat yang dikemukakan di atas, maka dapat disimpulkan bahwa dengan menggunakan *PhpMyAdmin*, maka *aplikasi* ini dapat membantu dalam menavigasi beberapa *database, table, log,* dan beberapa hal lainnya.

1. *MySQL*

 Menurut Nugroho (2009:10),”*MySQL* adalah suatu sistem manajemen basis data *relasional (RDBMS-Relational Dabatase Management System)* yang mampu bekerja dengan cepat, kokoh, dan mudah digunakan”.

 Sedangkan menurut Wahana Komputer (2010:111)*,”MySQL* merupakan suatu aplikasi yang digunakan untuk memanajemen suatu data dan banyak digunakan khalayak diseluruh dunia”.

 Menurut Wahana Komputer (2010:7), *MySQL* memiliki beberapa kelebihan dan keuntungan dibandingkan *database* lain, diantaranya adalah:

a. Banyak ahli berpendapat *Mysql* merupakan server tercepat.

b. *Mysql* merupakan sistem manajemen database yang Open Source (kode sumbernya terbuka), yaitu software ini bersifat free atau bebas digunakan oleh perseorangan atau instansi tanpa harus membeli atau membayar kepada pembuatnya.

c. *Mysql* mempunyai performa yang tinggi tapi simple.

d. *Mysql* dapat diakses melalui protocol [*ODBC*](https://widuri.raharja.info/index.php?title=ODBC&action=edit&redlink=1) (Open Database Connectivity) buatan *Microsoft*. Ini menyebabkan *Mysql* dapat diakses oleh banyak software.

e. Semua klien dapat mengakses server dalam satu waktu, tanpa harus menunggu yang lain mengakses database.

f. Database *Mysql* dapat diakses dari semua tempat di internet dengan hal akses tertentu.

g. *Mysql* merupakan database yang mampu menyimpan data berkapasitas besar, sampai berukuran Gigabyte.

h. *Mysql* dapat berjalan di berbagai operating system seperti *L*[*inux*](https://widuri.raharja.info/index.php/Linux)*,* [*Windows*](https://widuri.raharja.info/index.php/Windows)*,* [*Solaris*](https://widuri.raharja.info/index.php?title=Solaris&action=edit&redlink=1), dan lain-lain.

**E. Metode Pengembangan Perangkat Lunak**

 Menurut Pressman (2010:39) menjelaskan bahwa “model *waterfall* adalah model klasik yang bersifat sistematis, berurutan dalam membangun *software”.*

Fase-fase dalam model *waterfall* menurut Pressman(2010:39), sebagai berikut:



Sumber : Pressman (2010:39)

**Gambar II.1.**

***Waterfall* Model**

* 1. *Communication*

 Langkah ini merupakan analisis terhadap kebutuhan *software*, dan tahap untuk

 mengadakan pengumpulan data dengan melakukan pertemuan dengan customer, maupun mengumpulkan data-data tambahan baik yang ada di jurnal, artikel, maupun dari internet.

* 1. *Planning*

 Proses *planning* merupakan lanjutan dari proses *communication (analysis requirement).* Tahapan ini akan menghasilkan dokumen *user requirement* atau bisa dikatakan sebagai data yang berhubungan dengan keinginan *user* dalam pembuatan *software*, termasuk rencana yang akan dilakukan.

1. *Modeling*

Proses *modeling* ini akan menerjemahkan syarat kebutuhan ke sebuah perancangan *software* yang dapat diperkirakan sebelum dibuat *coding.* Proses ini berfokus pada rancangan struktur data, arsitektur *software,* representasi *interface,* dan detail (algoritma) prosedural. Tahapan ini akan menghasilkan dokumen yang disebut *software requirement.*

1. *Construction*

*Construction* merupakan proses membuat kode. *Coding* atau pengkodean merupakan penerjemahan desain dalam bahasa yang bisa dikenali oleh komputer. *Programmer* akan menerjemahkan transaksi yang diminta oleh *user.* Tahapan inilah merupakan tahapan secara nyata dalam mengerjakan suatu *software,* artinya penggunaan komputer akan dimaksimalkan dalam tahapan ini. Setelah pengkodean selesai maka akan dilakukan *testing* terhadap sistem yang telah dibuat. Tujuan *testing* adalah menemukan kesalahan-kesalahan terhadap siste tersebut untuk kemudian bisa diperbaiki.

1. *Deployment*

Tahapan ini bisa dikatakan final dalam pembuatan sebuah *software* atau sistem. Setelah melakukan analisis, desain dan pengkodean maka sistem yang sudah jadi akan digunakan oleh *user.* Kemudian *software* yang telah dibuat harus dilakukan pemeliharaan secara berkala.

**F. Unified Modelling Language (UML)**

1. Definisi *Unified Modelling Language (UML)*

Menurut Nugroho (2009:6) menyatakan bahwa: “*Unified Modeling Language*  adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berpradigma berorientasi objek, pemodelan (*modeling)* sesungguhnya digunakan untuk penyederhanaan permasalahan- permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami”.

 Berdasarkan pendapat yang dikemukakan diatas dapat disimpulkan bahwa *Unified Modeling Languge (UML)* adalah sebuah bahasa pemodelan yang berorientasi objek dan menjadi standar dalm visualisasi, merancang, dan mendokumentasikan sistem perangkat lunak untuk penyederhanaan permasalahan-permasalahan yang kompleks.

2. Jenis Diagram *UML*

 Menurut Widodo dkk (2011:10) menjelaskan bahwa:”Beberapa literature menyebutkan bahwa UML menyediakan sembilan jenis diagram, yang lain menyebutkan delapan karena ada beberapa diagram yang digabung, misanya diagram komunikasi, diagram urutan dan diagram pewaktuan digabung menjadi diagram interaksi”.

 Namun demikian model-model itu dapat dikelompokkan berdasarkan sifatnya yaitu statis atau dinamis. Jenis diagram itu antara lain:

a. Diagram use-case *(Usecase Diagram)*

 Bersifat statis. Diagram ini memperlihatkan himpunan *usecase* dan aktor-aktor (suatu jenis khusus dari kelas). Diagram ini terutama sangat penting untuk mengorganisasi dan memodelkan perilaku suatu sistem yang dibutuhkan serta diharapkan pengguna.

b. Diagram aktivitas *(Activity Diagram)*

 Bersifat dinamis. Diagram aktivitas adalah tipe khusus dari diagram status yang memperlihatkan aliran dari suatu suatu aktivitas ke aktivitas lainnya dalam suatu sistem. Diagram ini terutama penting dalam pemodelan fungsi-fungsi suatu sistem dan memberi tekanan pada aliran kendali antar objek.

c. Diagram komponen *(Component Diagram)*

Bersifat statis. Diagram komponen ini memperlihatkan organisasi serta kebergantungan sistem/perangkat lunak pada komponen-komponen yang telah ada sebelumnya.

1. Diagram deployment *(deployment diagram)*

 Bersifat statis. Diagram ini memperlihatkan konfigurasi saat aplikasi dijalankan *(run-time)*. Memuat simpul-simpul beserta komponen-komponen yang di dalamnya. Kesembilan diagram ini tidak mutlak harus digunakan dalam pengembangan perangkat lunak, semuanya dibuat sesuai kebutuhan. Pada *UML* dimungkinkan kita menggunakan diagram-diagram lainnya misalnya data *flow* *diagram, entity relationship diagram,* dan sebagainya.

**G. *Entity Relationship Diagram* (*ERD)***

 Menurut Rosa dan Shalahuddin (2013:53) “ERD adalah bentuk paling awal dalam melakukan perancangan basis data relasional. Jika menggunakan *OODBMS* maka peracangan ERDtidak perlu dilakukan”.

 ERD bersama-sama dengan detail pendukung merupakan model data yang pada gilirannya digunakan sebagai spesifikasi untuk *database.*

Komponen-komponen yang terdapat pada *entity relationship diagram* sebagai berikut :

1. Entitas (*Entity)*

 Digambarkan dengan kotak persegi panjang dan digunakan untuk menunjukan sekumpulan orang, tempat, objek, atau konsep yang menunjukan dimana data dicatat atau disimpan.

1. *Relationship (*Relasi)

Digambarkan dengan kotak *diamond* atau belah ketupat dengan garis yang menghubungkan ke entitas yang terkait. Sehingga *relationship* diberi nam dengan kata kerja. Hubungan atau relasi menunjukan abstraksi dari sekumpulan hubungan yang mengaitkan antara entitas yang berbeda.

1. Atribut

Menunjukan karakteristik dari tiap entitas atau sesuatu yang menjelaskan entitas atau hubungan sehingga atribut dikatakan elemen data dari entitas dan *relationship.* Dari setiap atribut-atribut entitas terdapat satu atribut yang dijadikan kunci (*key).*

1. Tingkat hubungan (*Cardinality)*

 Di dalam *entity relationship* diagram terdapat kardinalitas atau derajat relasi yang menunjukkan jumlah maksimum entitas yang dapat berelasi dengan entitas pada himpunan entitas yang lain,

 Jenis kardinalitas atau derajat relasi pada *entity relationship diagram* adalah sebagai berikut:

a. *One to One*

Tingkat hubungan satu ke satu, dinyatakan dengan satu kejadian pada entitas pertama, hanya mempunyai satu hubungan dengan satu kejadian pada entitas yang kedua dan sebaliknya.

b. *One to Many*

Tingkat hubungan satu ke banyak adalah sama dengan banyak kesatu, tergantung dari mana hubungan tersebut dilihat. Untuk satu kejadian pada entitas pertama dapat mempunyai banyak hubungan dengan kejadian pada entitas kedua, dan sebaliknya.

c. *Many to Many*

Tingkat hubungan kebanyakan terjadi jika setiap kejadian pada setiap entitas akan mempunyai banyak hubungan dengan kejadian pada entitas lainnya.

**H. LRS (*Logical Record Structure)***

Model E-R adalah rincian yang merupakan representasi logika dari data pada suatu organisasi atau area bisnis tertentu. *Logical Record Strukture* (LRS) terdiri dari hubungan-hubungan diantara tipe record. Link ini menunjukan arah dari suatu tipe record lainya. Banyak hubungan dari LRS yang diberi tanda filed-filed yang akan kelihatan pada kedua link tipe record. Pengembangan LRS dimulai dengan menggunakan model yang dimengerti, yaitu:

1. Transformasi diagram ERD ke LRS

2. Transformasi LRS relasi table.

 Penggambaran LRS mulai dengan menggunakan model yang mudah dimengerti. Dua metode yang dapat digunakan dimulai dengan hubungan kedua

model yang dapat dikonversikan ke LRS, metode yang lain dimulai dengan ER diagram dan langsung dikonversikan ke LRS.

**2.2 Penelitian Terkait**

Menurut IrwanYulistiawan dkk (2011:7) menjelaskan bahwa:

 Sistem informasi penggajian yang dirancang menghasilkan sistem penggajian yang lebih efisien dan efektif dalam pencatatan informasi terkait dengan aktivitas penggajian, pembuatan, laporan serta dapat mengatasi masalah human error. Sistem informasi penggajian yang dirancang juga memiliki pembatasan hak akses bagi setiap user dengan tujuan untuk meningkatkan internal control dan mencegah pihak-pihak yang tidak berwenang mengakses data. Sistem informasi penggajian yang dirancang menghasilkan data yang mengintegrasikan data dari berbagai aktivitas yang berhubungan dengan penggajian seperti data absensi, lembur, pinjaman, uang makan dan transport, pajak, dan zakat.

 Menurut Devi Lestari (2014:25) menjelaskan bahwa:

Dengan diterapkannya sistem informasi penggajian karyawan pada Pr. Tunas Mandiri Kabupaten Pacitan dapat membantu permasalahan yang sering terjadi di Pr. Tunas Mandiri Kabupaten Pacitan karena dapat meminimalkan kesalahan dalam menghitung gaji karyawan. Pembangunan sistem informasi penggajian karyawan pada Pr.Tunas Mandiri Kabupaten Pacitan sudah sesuai dengan kebutuhan yang diperlukan Pr. Tunas Mandiri Kabupaten Pacitan. Dengan adanya analisis dan perancangan sistem informasi penggajian Pr. Tunas Mandiri Kabupaten Pacitan dapat memberikan alternatif pemecahan masalah dengan diciptakannya sebuah aplikasi sistem informasi penggajian Pr. Tunas Mandiri Kabupaten Pacitan agar lebih efektif dan efisien sehingga data yang dihasilkan lebih akurat.