

## **BAB II**

### **LANDASAN TEORI**

#### **2.1. Tinjauan Pustaka**

Di dalam tinjauan pustaka, penulis akan menjelaskan tentang semua teori teori yang berhubungan dengan judul yang akan di bahas.

##### **A. Sistem Informasi**

Terdapat beberapa pendapat mengenai pengertian sistem, antara lain yaitu pendapat dari Mulyadi (2008: 5) yang menyatakan bahwa suatu sistem pada dasarnya adalah sekelompok unsur yang erat terhubung satu dengan yang lainnya, yang berfungsi bersama-sama untuk mencapai tujuan tertentu. Secara lebih rinci dapat dijelaskan bahwa sistem terdiri dari unsur-unsur yang lebih kecil, unsur-unsur tersebut merupakan bagian terpadu dari sistem yang bersangkutan. Unsur tersebut bekerja sama untuk mencapai suatu tujuan sistem. Suatu sistem pada dasarnya merupakan bagian dari sistem lain yang lebih besar dan luas cakupannya.

Menurut Sutabri (2007:14) sistem informasi adalah suatu system didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manejerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan.

Menurut Al fatta (2007:14 ) sistem informasi merupakan system dengan komponen-komponen yang bekerja untuk mengolah data menjadi informasi.

## B. Penggajian

Definisi Gaji menurut Sikula (2007:119) Gaji adalah balas jasa dalam bentuk uang yang diterima karyawan sebagai konsekuensi dari statusnya sebagai seorang karyawan yang memberikan kontribusi dalam mencapai tujuan perusahaan.

## C. *Unified Modelling Language (UML)*

UML singkatan dari *Unified Modeling Language* yang berarti bahasa permodelan standar. Menurut Chonoles dalam Widodo dan Herlawati (2011:6) mengatakan sebagai bahasa, berarti UML memiliki *sintaks* dan *semantic*. Ketika kita membuat model menggunakan konsep UML ada aturan-aturan yang harus diikuti. Bagaimana elemen pada model-model yang kita buat berhubungan satu dengan lainnya harus mengikuti standar yang ada.

Menurut Widodo dan Herlawati (2011:10), Ada 5 diagram yang digunakan dalam penulisan skripsi ini yaitu *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, *Component Diagram* dan *Deployment Diagram*. Berikut penjelasan dari masing-masing diagram :

### 1. *Use Case Diagram*

Diagram *Use Case*. Bersifat statis. Diagram ini memperlihatkan himpunan *use case* dan aktor-aktor (suatu jenis khusus dari kelas). Diagram ini terutama sangat penting untuk mengorganisasi dan memodelkan perilaku suatu sistem yang di butuhkan serta diharapkan pengguna.

### 2. *Activity Diagram*

Diagram Aktivitas (*Activity Diagram*). Bersifat dinamis. Diagram Aktivitas adalah tipe khusus dari diagram status yang memperlihatkan aliran dari 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.

### 3. *Sequence Diagram*

Diagram interaksi dan *Sequence* (urutan). Bersifat dinamis. Diagram urutan adalah diagram interaksi yang menekankan pada pengiriman pesan dalam suatu waktu tertentu.

### 4. *Component Diagram*

Diagram Komponen (*Component Diagram*). Bersifat statis. Diagram komponen ini memperlihatkan organisasi serta kebergantungan sistem atau perangkat lunak pada komponen-komponen yang telah ada sebelumnya. Diagram ini berhubungan dengan diagram kelas dimana komponen secara tipikal dipetakan kedalam satu atau lebih kelas-kelas, antarmuka-antarmuka serta kolaborasi-kolaborasi.

### 5. *Deployment Diagram*

Diagram Deployment (*Deployment Diagram*). Bersifat statis. Diagram ini memperlihatkan konfigurasi saat aplikasi dijalankan (*run-time*). Memuat simpul-simpul beserta komponen-komponen yang ada didalamnya. 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*).

#### **D. *Entity Relationship Diagram (ERD)***

Menurut Sutanta (2011:91) “*Entity Relationship Diagram (ERD)* merupakan suatu model data yang dikembangkan berdasarkan objek.” *Entity Relationship Diagram (ERD)* digunakan untuk menjelaskan hubungan antar data dalam basis data kepada pengguna secara logis. *Entity Relationship Diagram (ERD)* didasarkan pada suatu persepsi bahwa *real world* terdiri atas objek-objek dasar tersebut. Penggunaan *Entity Relationship Diagram (ERD)* relatif mudah dipahami, bahkan oleh para pengguna yang awam. Bagi perancang atau analis sistem, *Entity Relationship Diagram (ERD)* berguna untuk memodelkan sistem yang nantinya, basis data akan di kembangkan. Model ini juga membantu perancang atau analis sistem pada saat melakukan analisis dan perancangan basis data karena model ini dapat menunjukkan macam data yang dibutuhkan dan kerelasiaan antardata didalamnya.

##### **1. Komponen *Entity Relationship Diagram (ERD)***

Komponen *Entity Relationship Diagram* menurut Sutanta (2011: 91) adalah sebagai berikut :

###### **a. Entitas**

Entitas merupakan suatu objek yang dapat dibedakan dari lainnya yang dapat diwujudkan dalam basis data. Objek dasar dapat berupa orang, benda, atau hal yang keterangannya perlu disimpan didalam basis data. Untuk menggambarkan sebuah entitas digunakan aturan sebagai berikut :

- 1) Entitas dinyatakan dengan simbol persegi panjang.
- 2) Nama entitas dituliskan didalam simbol persegi panjang.
- 3) Nama entitas berupa kata benda, tunggal.

- 4) Nama entitas sedapat mungkin menggunakan nama yang mudah dipahami dan dapat menyatakan maknanya dengan jelas.

b. Atribut

Atribut merupakan keterangan-keterangan yang terkait pada sebuah entitas yang perlu disimpan dalam basis data. Atribut berfungsi sebagai penjelas pada sebuah entitas. Untuk menggambarkan atribut digunakan aturan sebagai berikut:

- 1) Atribut digambarkan dengan simbol *ellips*.
- 2) Nama atribut dituliskan didalam simbol *ellips*.
- 3) Nama atribut merupakan kata benda, tunggal.
- 4) Nama atribut sedapat mungkin menggunakan nama yang mudah dipahami dan dapat menyatakan maknanya dengan jelas.

c. Relasi

Relasi merupakan hubungan antara sejumlah entitas yang berasal dari himpunan entitas yang berbeda. Aturan penggambaran relasi adalah sebagai berikut :

- 1) Relasi dinyatakan dengan simbol belah ketupat.
- 2) Nama relasi dituliskan didalam simbol belah ketupat.
- 3) Nama relasi berupa kata kerja aktif.
- 4) Nama relasi sedapat mungkin menggunakan nama yang mudah dipahami dan dapat menyatakan maknanya dengan jelas.

### E. *Logical Record Structure (LRS)*

Menurut Simarmata (2007:115) *Logical Record Structure* adalah representasi dari struktur *record-record* pada table-tabel yang terbentuk dan hasil relasi antara himpunan entitas”. Beberapa tipe *record* digambarkan oleh kotak empat persegi panjang dan dengan nama yang unik. Beda LRS dengan diagram *entity relationship* nama tipe *record* berbeda di luar kotak *field* tipe *record* ditempatkan. LRS terdiri dari link-link di antara tipe *record*. *Link* ini menunjukkan arah dan 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. Dua metode yang dapat digunakan, dimulai dengan hubungan kedua model yang dapat dikonversikan ke LRS. Metode yang lain dimulai dengan *entity relationship diagram* dan langsung dikonversikan ke LRS.

1. Konversi ERD ke LRS. Diagram *entity relationship* diagram harus ke bentuk LRS (struktur record secara logika). Dan bentuk LRS inilah yang nantinya dapat ditransformasikan ke bentuk relasi (tabel).
2. Konversi ERD ke LRS sebuah model sistem yang digambarkan dengan sebuah ERD akan mengikuti pemodelan tertentu. Dalam kaitannya 16 dengan konversi ke LRS, untuk perubahan yang terjadi adalah mengikuti aturan-aturan berikut:
  - a. Setiap Entitas diubah ke bentuk kotak dengan nama entitas, berbeda di luar kotak dan atribut berada di dalam kotak.

- b. Sebuah *Relationship* kadang disatukan, dalam sebuah kotak bernama entitas, kadang sebuah kotak bersama-sama dengan entitas, kadang disatukan dalam sebuah kotak tersendiri.
3. Konversi LRS ke relasi (tabel) relasi atau tabel adalah bentuk pernyataan data secara grafis 2(dua) dimensi, yang terdiri dari kolom dan baris. Relasi adalah bentuk *visual* dari sebuah file, dan tiap tuple dalam sebuah *field* atau dalam bentuk lingkaran *diagram entity relationship* dikenal dengan sebutan atribut. Konversi dan *logical record structure*, dikenal dengan cara:
- a. Nama *logical record structure* menjadi relasi.
  - b. Tiap atribut menjadi sebuah kolom di dalam relasi.

## **F. Bahasa Pemrograman Yang Digunakan**

### **1. PHP (*Hypertext Preprocessor*)**

Menurut Adelheid dan Khairil (2012:2) PHP atau *Hypertext Preprocessor* adalah bahasa yang berbentuk *script* yang ditempatkan dalam server dan dieksekusi didalam *server* untuk selanjutnya ditransfer dan dibaca oleh *client*. Ada pun keuntungan dari bahasa pemrograman PHP adalah sebagai berikut :

- a. Bahasa pemrograman PHP adalah sebuah bahasa *script* yang tidak melakukan sebuah kompilasi dalam penggunaannya.
- b. *Web server* yang mendukung PHP dapat ditemukan dimana-mana dari mulai IIS sampai dengan *apache*, dengan konfigurasi yang *relative* mudah.
- c. Dalam sisi pengembangan lebih mudah , karena banyaknya milis-milis dan developer yang siap membantu dalam pengembangan.

- d. Dalam sisi pemahaman, PHP adalah bahasa *scripting* yang paling mudah karena referensi yang banyak. PHP adalah bahasa *open source* yang dapat digunakan diberbagai mesin (*linux, unix, windows*) dan dapat dijalankan secara *runtime* melalui *console* serta juga dapat menjalankan perintah-perintah sistem.

## **G. Perangkat Lunak Pendukung**

### **1. Xampp**

Menurut Nugroho (2008:18) Filosofi dibalik XAMPP adalah untuk membangun sebuah mudah untuk menginstal distribusi bagi pengembang untuk masuk ke dunia Apache. Untuk membuatnya nyaman untuk pengembang XAMPP dikonfigurasi dengan semua fitur diaktifkan. Konfigurasi standar tidak baik dari sudut pandang security dan tidak cukup aman untuk lingkungan produksi – jangan gunakan XAMPP dilingkungan tersebut. Sejak LAMPP 0.9.5 Anda dapat membuat instalasi XAMPP anda aman dengan menghubungkan >>/ opt / lamp / lamp<< keamanan. Apache *web server* merupakan tulang punggung dari *World Wide Web* (WWW). Web server mengunggu permintaan klien yang menggunakan *browser* seperti *Netscape, Navigasi, Mozilla, Link* dan lain-lain. *Web server* dalam berkomunikasi dengan kliennya, menggunakan protocol HTTP (*Hyper Text Transfer Protocol*).

Xampp merupakan produk yang istimewa karena berisi hal-hal yang diperlukan oleh sebuah *web*, dapat diakses, dapat diandalkan dan gratis. Xampp merupakan *web server* yang *open source*. Ini berarti setiap yang menggunakan produk ini mempunyai akses ke sumber kode (*source code*). Jika ingin

menambahkan sesuatu tinggal dituliskan ke dalam sumber kode yang menjalankannya. Apabila ditemukan kesalahan (*bug*) dapat segera diperbaiki atau segera dilaporkan.

## **2. MySQL (*Structured Query Language*)**

Menurut Nugroho (2008:29) MySQL merupakan *database* yang berbasis *server*. Anda bisa menggunakan database MySQL apabila memiliki izin hak akses didalamnya. Hal ini seperti halnya pada saat anda hendak menggunakan klien MySQL untuk masuk pada server MySQL.

Keunggulan dari MySQL adalah :

- a. Bersifat *open source*.
- b. Sistem *software*-nya tidak memberatkan kerja *server* atau komputer karena dapat bekerja di *background*.

## **3. Framework Codeigniter**

Menurut Rahmadiansyah dan Irwan (2012:2-1) Code Igniter merupakan sebuah *framework* pemrograman web dengan menggunakan Bahasa php. *Framework* ini ditulis dengan menggunakan bahasa php versi 4 dan versi 5 oleh Rick Ellislab yang menjadi CEO Ellislab, Inc. dan dipublikasikan dengan lisensi di bawah Apache/BSD Open Source.

## **4. Adobe Dreamweaver CS5**

Menurut Madcoms (2011:2) memberikan batasan bahwa *dreamweaver* adalah sebuah HTML *editor profesional* untuk mendesain web secara *visual* dan mengelola situs atau halaman web. Saat ini terdapat *software* dari kelompok *adobe* yang belakangan banyak digunakan untuk mendesain suatu situs web. Versi terbaru dari *Adobe Dreamweaver* saat ini adalah *Dreamweaver CS5*,

terdapat beberapa kemampuan bukan hanya sebagai *software* untuk desain web saja tetapi juga untuk menyunting kode serta pembuatan aplikasi web dengan menggunakan berbagai bahasa pemrograman web, antara lain: JPS, PHP, ASP dan *ColdFusion*.

## **5. Pengertian Basis Data**

Menurut Kadir (2014:218) basis data (*database*) adalah suatu pengorganisasian sekumpulan data yang saling terkait sehingga memudahkan aktivitas untuk memperoleh informasi. Basis data dimaksudkan untuk mengatasi *problem* pada sistem yang memakai pendekatan berbasis berkas.

## **6. DBMS (*Database Management System*)**

Menurut Kadir (2014:218) DBMS adalah perangkat lunak sistem yang memungkinkan para pemakai membuat, memelihara, mengontrol dan mengakses basis data dengan cara praktis dan efisien. DBMS dapat digunakan untuk mengakomodasikan berbagai macam pemakai yang memiliki kebutuhan akses yang berbeda-beda.

## **7. DML (*Data Manipulation Language*)**

Menurut Kadir (2014:227) DML adalah perintah-perintah yang digunakan untuk melakukan hal-hal seperti berikut :

1. Mengambil data pada basis data.
2. Menambahkan data pada basis data.
3. Mengubah data pada basis data.
4. Menghapus data pada basis data.

DML dapat dibagi menjadi dua kelompok, yaitu prosedural dan non-prosedural.

- a. DML prosedural adalah perintah-perintah yang memungkinkan pemakai menentukan data apa saja yang diperlukan dan bagaimana cara mendapatkannya. Untuk mendapatkan hasil yang dikehendaki, pemakai memberikan perintah-perintah yang mengikuti suatu algoritma (langkah yang rinci untuk menyelesaikan masalah).
- b. DML non-prosedural adalah perintah-perintah yang memungkinkan pemakai menentukan data apa saja yang diperlukan, tanpa perlu menyebutkan cara mendapatkannya.

#### **H. *Black Box Testing***

Menurut Rizky (2011:264) *black box testing* adalah tipe *testing* yang memperlakukan perangkat lunak yang tidak diketahui kinerja internalnya. Sehingga para *tester* memandang perangkat lunak seperti layaknya sebuah “kotak hitam” yang tidak penting dilihat isinya, tapi cukup dikenal proses *testing* di bagian luar.

#### **2.2. Penelitian Terkait**

Menurut observasi dan wawancara yang penulis lakukan, Sekolah Menengah Pertama (SMP) PGRI Kebonagung dalam mengolah data gaji karyawan masih menggunakan cara yang konvensional, yakni mencatat pada pembukuan sehingga sering terjadi kesalahan dalam menghitung jumlah gaji guru yang didasarkan pada berbagai rincian yang berbeda-beda, hal ini dapat menghambat proses kinerja bendahara instansi tersebut. Kesalahan dalam menghitung gaji dapat berakibat fatal karena dapat berpengaruh dalam pembuatan laporan keuangan. Selain itu bendahara juga kesulitan dalam pembuatan laporan keuangan, karena petugas harus melihat data pada pembukuan, kemudian mengolahnya menggunakan Ms. Excel. Hasil penelitian perancangan sistem informasi penggajian pada sekolah Menengah pertama (smp) PGRI Kebonagung oleh (Saputra dan Bukhori 2014: 6).

Pada suatu perusahaan sistem penggajian sangatlah penting, untuk menunjang kelancaran pembayaran gaji karyawan, maka dari itu dibutuhkan sebuah program aplikasi penggajian untuk memberikan kemudahan dalam penggajian karyawan. Disimpulkan

bahwa prosedur penggajian karyawan yang dipakai oleh PT Arus Global Security Service Jakarta pada dasarnya belum terkomputerisasi dengan baik, sehingga pengolahan data semakin rumit. Dengan adanya program aplikasi penggajian, maka penginputan data dan penyusunan laporan diperlukan ketelitian, karena data yang masuk dapat bertambah dan berubah sewaktu-waktu. Berdasarkan hal diatas, maka diperlukan adanya suatu program aplikasi penggajian yang mampu mengolah data secara cepat dan tepat, sehingga mempermudah bagian staff keuangan dalam proses pembayaran gaji karyawan dan dapat meningkatkan kinerja karyawan. Hasil penelitian sitem informasi penggajian karyawan (studi kasus : pt. arus global security service jakarta) oleh (Zulhanis 2016: 206).