BAB II

LANDASAN TEORI

2.1. Tinjauan Pustaka

Tinjauan pustaka dilakukan dari referensi-referensi yang berkaitan dengan masalah yang dibahas penulis melakukan studi pustaka melalui sumber buku, jurnal-jurnal terkait, maupun pada sumber lainnya sebagai dasar pembahasan lebih lanjut untuk memperoleh bahan landasan-landasan teori dari sistem yang akan dikembangkan oleh penulis yang membahas tentang Sistem Informasi Pengolahan Data Kepegawaian Berbasis Web Pada PT. Argo Pantes Tbk.

A. Konsep Dasar Sistem Informasi

1. Definisi Sistem Informasi

Menurut Hutahean (2014 : 13) sistem informasi adalah suatu sistem didalam 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 dibutuhkan.

Menurut Sutarman dalam penelitian Esa Wijayanti (2014 : 14) Sistem informasi dapat didefinisikan dengan mengumpulkan, memproses, menyimpan, menganalisa, menyebarkan informasi untuk tujuan tertentu.

Dari pendapat diatas maka dapat disimpulkan bahwa sistem informasi adalah suatu kegiatan mengumpulkan, memproses, menyimpan, menganalisa dan menyebarkan suatu informasi untuk tujuan tertentu .

2. Konsep Sistem Informasi

Menurut Hutahean (2014 : 13) sistem informasi terdiri dari beberapa komponen-komponen yang disebut dengan istilah blok bangunan (*building block*) yaitu:

a. Blok Masukan (*input block*)

Input mewakili data yang masuk kedalam sistem informasi, input disini termasuk metode-metode dan media yang digunakan untuk menangkap data yang akan dimasukan, yang dapat berupa dokumen dasar.

b. Blok Model (*model block*)

Blok ini terdiri dari kombinasi prosedur, logika, dan model matematik yang akan memanipulasi data input dan data yang tersimpan di basis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang sudah diinginkan.

c. Blok Keluaran (*output block*)

Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

d. Blok Teknologi (*technology block*)

Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan

keluaran dan membantu pengendalian diri secara keseluruhan.

Teknologi terdiri dari unsur utama :

1. Teknisi (*human ware* atau *brain ware*)
 2. Perangkat lunak (*software*)
 3. Perangkat keras (*hardware*)
- e. Blok Basis Data (*data base block*)

Merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan diperangkat keras computer dan digunakan perangkat lunak untuk memanipulasinya.

- f. Blok Kendali (*control block*)

Banyak faktor yang dapat merusak sistem informasi, misalnya bencana alam, api, tempratur tinggi, air, debu, kecurangan-kecurangan, kejanggalan sistem itu sendiri, kesalahan-kesalahan ketidakefisienan, sabotase dan sebagainya. Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah atau bila terlanjur terjadi kesalahan dapat langsung diatasi.

B. Konsep Dasar Pemrograman

Menurut Shalahuddin dan Rosa (2011 : 62) Pemrograman terstruktur adalah konsep atau sudut pandang pemrograman yang membagi-bagi program berdasarkan fungsi-fungsi atau prosedur-prosedur yang dibutuhkan program

komputer”. Modul-modul biasanya dibuat dengan mengelompokkan fungsi-fungsi dan prosedur-prosedur yang diperlukan sebuah proses tertentu.

Desain-desain yang terdapat dalam pemrograman terstruktur adalah:

a. Desain *Modular*

Penyusunan program dengan desain *modular* dilakukan dengan memecah program kedalam beberapa modul, dimana setiap modul menunjukkan fungsi dan tugas tunggal. Dengan membagi masalah kedalam beberapa modul, maka masalah akan menjadi sederhana dan mudah dipahami. Desain dengan metode *modular* merupakan implementasi dan pengembangan dari desain *top down*. Setiap program memiliki sebuah modul program utama yang mengontrol semua proses yang terjadi, termasuk mengirimkan kontrol program kepada sub modul untuk melakukan suatu fungsi tertentu. Pemrograman *modular* diterapkan dengan menggunakan *sub routine*, yaitu sebuah kumpulan perintah yang melakukan tugas pemrosesan yang terbatas.

b. Desain *Top Down*

Penyusunan program desain *top down* mengacu kepada tujuan program secara menyeluruh, bukan pada bagaimana cara mencapainya. Setelah tujuan terdefinisi dengan jelas, barulah dibuat garis besar proses yang akan dilaksanakan program. Kemudian secara bertahap garis besar proses diuraikan menjadi tahapan-tahapan yang lebih terperinci, sehingga akhirnya semua program selesai. Keuntungan dari desain *top down* adalah bahwa setiap tahap pemrograman yang ada menjadi sederhana, karena setiap tingkat mengabaikan detail dari tingkat yang lebih rendah. Dengan cara ini, sebuah proses yang kompleks dipecah kedalam sebuah urutan langkah yang relatif mudah, hasil akhir adalah berupa sebuah modul yang lebih sederhana.

c. Desain *Bottom Up*

Penyusunan program dengan desain *bottom up* dilakukan dengan menyelesaikan semua pengkodean untuk modul-modul tingkat paling rendah, kemudian diuji. Jika benar, maka diintegrasikan ke tingkat berikutnya dan diuji lagi. Pengujian *bottom up* membutuhkan beberapa program kendali, yang memanggil modul dan memasukkan data penambahan signifikan didalam program. Pada dasarnya desain *bottom up* merupakan kebalikan dari desain *top down*, hanya saja pendefinisian tujuan dilakukan tetap pada awal penyusunan program.

C. PHP

Menurut Bahtiar (2008 : 17) PHP adalah sebuah bahasa pemrograman yang didesain agar dapat disisipkan dengan mudah ke halaman HTML. PHP memberikan solusi sangat mudah dan dapat berjalan di di berbagai jenis *platform* dengan luasnya cakupan sistem operasi yang mampu menjalankan PHP dan lengkapnya *function* yang dimilikinya tersedia lebih dari 400 *function* di PHP. Penemu bahasa pemrograman ini adalah Rasmus Lerdorf yang bermula dari keinginan sederhana Lerdorf untuk mempunyai alat bantu dalam memonitor pengunjung yang melihat situs web pribadinya, Inilah sebabnya pada awal pengembangannya. PHP merupakan singkatan dari *Personal Home Page Tools*, sebelum akhirnya menjadi *Hypertext Preprocessor*. Untuk dapat menjalankan script – script PHP sebuah sistem harus mempunyai *Apache Web Server*, PHP 4/5, dan database MySQL ketiganya adalah program *open source* yang tersedia secara gratis di internet dan dapat berjalan di berbagai *platform*.

D. Sistem Informasi Kepegawaian

Menurut Gecko (2008) manfaat pencatatan data kepegawaian adalah sebagai sarana untuk menyimpan data kepegawaian secara sistematis, sehingga memudahkan penemuan kembali jika diperlukan untuk bahan pembinaan pegawai.

Pengelolaan kepegawaian yang bersifat manajerial maupun teknis administratif selalu berhubungan dengan data, dalam bentuk yang tercetak maupun data elektronik. Kegiatan administrasi kepegawaian akan berpengaruh pada keadaan data perorangan pegawai maupun keseluruhan. Seringkali Perubahan – perubahan yang terjadi tidak segera diketahui para pelaksana administrasi yang lain. Keberadaan Perangkat Komputer tidak banyak membantu karena data disimpan dan dikelola oleh masing - masing pelaksana dan tidak ada kesatuan *platform* dalam penyimpanan. Akibatnya dalam hal data pokok sekalipun, bisa perlu waktu lama untuk menemukannya bahkan terjadi kesalahan *break*.

E. WSDM (Web Site Design Method)

Menurut Simarmarta (2010 : 217) WSDM model adalah pendekatan yang terpusat pada pengguna untuk pengembangan situs web, yaitu model-model aplikasi yang berdasarkan pada kebutuhan informasi dari kelompok pengguna. Proses pengembangan ini dibagi ke dalam empat fase, yaitu :

1. Pemodelan Pengguna

Dalam fase ini, pengguna diklasifikasikan dan dikelompokkan dalam pemelajaran kebutuhan sistem dengan masing – masing kelompok pengguna.

2. Desain Konseptual

Dalam fase ini, diagram kelas didesai untuk mewakili model statis sistem dan model navigasional untuk mewakili kemungkinan dari navigasi.

3. Desain Implementasi

Dalam fase ini, model – model dari desain konseptual diterjemahkan ke dalam suatu bahasa abstrak yang lebih mudah untuk dimengerti oleh komputer.

4. Implementasi

Dalam fase ini, hasil dari desain implementasi ditulis dalam bahasa komputer khusus.

F. UML (Unified Modelling Language)

Menurut Sukamto dan Shalahuddin (2013 : 133) UML (*Unified Modeling Language*) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek”. UML (*Unified Modeling Language*) untuk pemrograman terstruktur mendefinisikan diagram-diagram sebagai berikut:

1. Use Case 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”.

2. Activity Diagram

activity diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak.

3. Component Diagram

component diagram dibuat untuk menunjukkan organisasi dan ketergantungan diantara kumpulan komponen dalam sebuah sistem. *Component diagram* fokus pada komponen sistem yang dibutuhkan dan ada di dalam sistem.

4. *Deployment Diagram*

deployment diagram menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi.

G. ERD (*Entity Relationship Diagram*)

ERD menurut sutanta (2011 : 91) merupakan suatu model data yang dikembangkan berdasarkan objek, *Entity Relationship Diagram* digunakan untuk menjelaskan hubungan antar data dalam basis data kepada pengguna secara logis didasarkan pada suatu persepsi bahwa *real world* terdiri atas obyek – obyek dasar tersebut penggunaan *Entity Relationship Diagram* berguna untuk memodelkan sistem yang nantinya basis data akan dikembangkan. 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.

Komponen *Entity Relationship Diagram* 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 dalam basis data.

Untuk menggambarkan suatu 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 meyakinkan 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 sebagai berikut :

1. Atribut digambarkan dengan simbol *ellips*.
2. Nama atribut dituliskan dalam simbol *ellips*.
3. Nama atribut merupakan kata benda tunggal.
4. Nama atribut sedapat mungkin menggunakan nama yang mudah dipahami, dan dapat menyatakan makna 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 disimbolkan dalam 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.

H. Logical Record Structure

LRS Menurut Frieyadie (2007 : 13) adalah merupakan hasil dari pemodelan entity relational ship (ER) beserta atributnya sehingga bisa terlihat hubungan – hubungan antar entitas. Dalam pembuatan LRS terdapat 3 hal yang dapat mempegaruhi yaitu :

- a. Jika tingkat hubungan (*cardinality*) satu pada satu (*one-to-one*), maka di gabungkan dengan entitas yang lebih kuat atau (*strong entity*), atau digabungkan dengan entitas yang memiliki atribut lebih sedikit.
- b. Jika tingkat hubungan (*cardinality*) satu pada banyak (*one-to-many*), maka hubungan relasi atau digabungkan dengan entitas yang tingkat hubungannya banyak.
- c. Jika tingkat hubungan (*cardinality*) banyak pada (*many-to-many*), maka hubungan relasi tidak akan digabungkan dengan entitas manapun, melainkan menjadi sebuah LRS.

I. E-Commerce

Menurut Afrina dan Marpaung dalam Sunarto (2009 : 27) *e-commerce* atau lebih dikenal dengan *e-com* dapat diartikan sebagai cara berbelanja atau berdagang secara *online* atau *direct selling* yang memanfaatkan fasilitas internet di mana terdapat *website* yang dapat menyediakan layanan *get and deliver*.

“*E-commerce* merupakan satu set dinamis teknologi, aplikasi dan proses bisnis yang menghubungkan perusahaan, konsumen, dan komunitas tertentu melalui transaksi elektronik dan perdagangan barang, pelayanan, dan informasi yang dilakukan secara elektronik. Seperti yang dijelaskan oleh David Baum dalam Sunarto (2009 : 28).

Menurut Sunarto (2009 : 68) secara umum kita dapat mengklasifikasikan *e-commerce* menjadi beberapa jenis dan memiliki karakteristik yang berbeda, yaitu:

1. *Business to Business* (B2B)

- a. *Trading Partners* yang sudah diketahui dan umumnya memiliki hubungan yang cukup lama. Informasi hanya dipertukarkan dengan *partner* tersebut sehingga jenis informasi yang dikirimkan dapat disusun sesuai kebutuhan dan kepercayaan.
- b. Pertukaran data (*data exchange*) berlangsung berulang-ulang dan secara berkala, dengan format data yang sudah disepakati bersama sehingga memudahkan pertukaran data untuk dua entitas yang menggunakan *standart* yang sama.
- c. Salah satu pelaku dapat melakukan inisiatif untuk mengirimkan data, tidak harus menunggu *partner*.
- d. Model yang umum digunakan adalah *per-to-per* di mana *processing intelligence* dapat didistribusikan di kedua belah pihak.

2. *Business to Consumer* (B2C)

Bussines to Customer (B2C) merupakan transaksi ritel dengan pembeli individual. Selain itu *Bussines to Customer* (B2C) juga dapat berarti mekanisme toko *online* (*electronic shopping mall*) yaitu transaksi antara *e-mechant* dengan *e-customer*.

Karakteristik *Bussines to Consumer* (B2C) adalah sebagai berikut:

- a. Terbuka untuk umum di mana semua informasi disebarakan ke umum.

- b. Layanan yang diberikan bersifat umum (*generic*) dengan mekanisme yang dapat digunakan oleh khalayak ramai. Sebagai contoh, karena sistem *web* sudah umum digunakan maka layanan diberikan dengan menggunakan *web*.
- c. Layanan diberikan berdasarkan permohonan (*on demand*). Konsumen melakukan inisiatif dan produsen harus siap memberikan respon sesuai dengan permohonan.
- d. Pendekatan *client-server* sering digunakan di mana diambil asumsi klien (*consumer*) menggunakan sistem yang minimal (berbasis *web*) dan *processing (business procedure)* diletakkan di sisi *server*.

3. *Consumer to Consumer (C2C)*

Merupakan transaksi di mana konsumen menjual produk secara langsung kepada konsumen lainnya. Selain itu, seorang individu juga bisa mengiklankan produknya berupa barang atau jasa, pengetahuan, atau keahliannya di salah satu situs lelang.

4. *Consumer to Business (C2B)*

Merupakan individu yang menjual produk atau jasa kepada organisasi dan individu yang mencari penjual dan melakukan transaksi.

5. *Non-Business Electronic Commerce*

Meliputi kegiatan non bisnis seperti kegiatan lembaga-lembaga pendidikan, organisasi nirlaba, keagamaan dan lain-lain.

6. *Intrabusiness (Organizational) Electronic Commerce*

Kegiatan ini meliputi semua aktivitas internal organisasi melalui internet untuk melakukan pertukaran barang, jasa, dan informasi serta menjual produk perusahaan kepada karyawan, dan lain-lain.

J. Pengujian Black Box

Menurut Simanjutak dkk (2010) *black box testing* adalah metode pengujian perangkat lunak yang tes fungsionalitas dari aplikasi yang bertentangan dengan struktur internal atau kerja (lihat pengujian *white-box*). pengetahuan khusus dari kode aplikasi / struktur internal dan pengetahuan pemrograman pada umumnya tidak diperlukan. Uji kasus dibangun di sekitar spesifikasi dan persyaratan, yakni, aplikasi apa yang seharusnya dilakukan.

Menggunakan deskripsi eksternal perangkat lunak, termasuk spesifikasi, persyaratan, dan desain untuk menurunkan uji kasus. Tes ini dapat menjadi fungsional atau non-fungsional, meskipun biasanya fungsional. Perancang uji memilih input yang valid dan tidak valid dan menentukan output yang benar. Tidak ada pengetahuan tentang struktur internal benda uji itu.

Metode uji dapat diterapkan pada semua tingkat pengujian perangkat lunak: unit, integrasi, fungsional, sistem dan penerimaan. Ini biasanya terdiri dari kebanyakan jika tidak semua pengujian pada tingkat yang lebih tinggi, tetapi juga bisa mendominasi unit testing juga.

Metode uji coba *black box* memfokuskan pada keperluan fungsional dari software. Karena itu uji coba *black box* memungkinkan pengembang software untuk membuat himpunan kondisi input yang akan melatih seluruh syarat-syarat fungsional suatu program. Uji coba *black box* bukan merupakan alternatif dari uji coba *white box*, tetapi merupakan pendekatan yang melengkapi untuk menemukan kesalahan lainnya, selain menggunakan metode *white box*.

Uji coba *black box* berusaha untuk menemukan kesalahan dalam beberapa kategori, diantaranya :

- a. Fungsi-fungsi yang salah atau hilang.
- b. Kesalahan interface.
- c. Kesalahan dalam struktur data atau akses database eksternal.
- d. Kesalahan performa.

e. Kesalahan inisialisasi dan terminasi.

2.2. Penelitian Terkait

Menurut penelitian yang dilakukan oleh Zaenuri (2013:1) Pengolahan data perpustakaan Madrasah Aliyah Negeri Lasem masih dilakukan secara manual dan penyampaian masih dilakukan secara lisan menyebabkan proses memakan waktu lama karena kurang efektif dan efisien mengingat keterbatasan dari sumber daya manusia itu sendiri selain itu pengolahan data perpustakaan yang dilakukan secara manual sering terjadi kesalahan karena kesalahan petugas perpustakaan oleh karena itu dibuatkannya sistem pengolahan data perpustakaan dengan sistem komputerisasi yang diharapkan mengatasi kesalahan yang telah ada sehingga memberikan kemudahan dalam pengolahan data perpustakaan di Madrasah Aliyah Negeri Lasem dan dapat berjalan lebih efektif dan efisien.

Menurut penelitian yang dilakukan oleh Purnamasari (2013:2) pada unit pelaksanaan teknis taman kanak-kanak pada sekolah pringku pegolahan data pegawai dan penggajian masih menggunakan metode konvensional metode ini membutuhkan waktu yang lama dalam pegolahan data pegawai dan data penggajianya sehingga dapat terjadi kesalahan karena kurang efektif dan efisien dalam pengolahan data pegawai dan data penggajian. Dengan adanya hal tersebut unit pelaksana teknis taman kanak-kanak pringku perlu merubah metode konvensional menjadi metode komputerisasi dan otomatis sehingga menghasilkan sebuah sistem informasi yang dapat mengolah data pegawai dan data penggajian yang efektif dan efisien.