

APLIKASI MANAJEMEN ABSENSI UNTUK REALISASI PENGGAJIAN PEGAWAI

Ellen Dwi Pertiwi¹, Supriady², Hariandi Maulid³
Jurusan Teknik Informatika
Politeknik Pos Indonesia

funtimes_89@yahoo.com, goku_sakti2005@yahoo.com, hariandi.maulid@yahoo.co.id

ABSTRAK

Dalam setiap instansi maupun perusahaan terdapat bagian yang mengelola penggajian dimana bagian ini mengolah tunjangan, potongan, gaji kotor, data kehadiran dan akhirnya menghasilkan gaji bersih untuk setiap pegawai. akan tetapi masih ada beberapa instansi maupun perusahaan dalam pengolahan data gaji pegawai tersebut masih memanfaatkan *Microsoft Excel* yang menyebabkan terjadinya redudansi data, proses perhitungan gaji tidak efektif, dan belum adanya database untuk menyimpan arsip data penggajian sehingga keamanan data tidak terjamin.

Adapun Tujuan pembuatan aplikasi penggajian pegawai ini adalah untuk mempermudah dalam pengelolaan dan perincian gaji pegawai. bahasa pemrograman yang digunakan untuk membangun aplikasi ini yaitu menggunakan bahasa pemrograman PHP dengan database *mysql*, *WEB browser* yang digunakan adalah *Appserv* dan perancangan menggunakan metode *Data Flow Diagram (DFD)*, tools yang digunakan untuk melakukan perancangan sistem ini adalah menggunakan *Power Designer 6*.

Oleh karena itu, dengan dibuatnya Aplikasi Manajemen Absensi Untuk Realisasi Penggajian Pegawai dapat mengatasi permasalahan dalam perhitungan gaji pegawai, bagian staff keuangan dimudahkan dalam perhitungan gaji pegawai, sehingga staff keuangan tidak perlu lagi menghitung uang tunjangan, bonus, maupun lembur pegawai karena pada aplikasi ini perhitungan akan terhitung otomatis sehingga staff keuangan tidak perlu lagi menghitung ulang gaji bersih pegawai sehingga perhitungan gaji menjadi lebih cepat.

Kata Kunci: Aplikasi penggajian, *PHP*, *MySQL*, *Power Designer 6*.

I. PENDAHULUAN

1.1 Latar Belakang

Kondisi dunia pada saat ini memasuki era globalisasi, dimana Kebutuhan akan penyajian informasi yang cepat dan akurat semakin dibutuhkan oleh berbagai pihak, baik dikalangan pemerintah, swasta, maupun perorangan. Berkaitan dengan itu diperlukan suatu media yang menyajikan informasi tersebut yaitu komputer. Dengan adanya komputerisasi, baik dalam dunia kerja maupun bisnis, efisiensi waktu dan kerja sangat diutamakan agar dapat tercapai produktivitas kerja yang maksimal. Dalam aplikasi kerja saat ini komputerisasi sudah ada di setiap bidang pekerjaan apalagi yang berkaitan dengan data-data yang penting.

Gaji atau sering disebut upah adalah suatu bentuk kompensasi yang merupakan imbalan jasa yang diberikan secara teratur atas hasil kerja yang diberikan kepada seorang pegawai. Besar kecilnya suatu gaji yang diperoleh dilihat dari beberapa hal baik itu latar belakang pendidikan, jabatan, maupun

pengalaman kerja yang pernah didapat sebelumnya. Dalam suatu penggajian terdapat besarnya gaji pokok yang diberikan, tunjangan baik itu tunjangan jabatan, tunjangan lembur, dan tunjangan lainnya tergantung instansi maupun perusahaan yang membuat kebijakannya.[1]

Dalam setiap instansi maupun perusahaan terdapat bagian yang mengelola penggajian dimana bagian ini mengolah tunjangan, potongan, gaji kotor, data kehadiran dan akhirnya menghasilkan gaji bersih untuk setiap pegawai. akan tetapi masih ada beberapa instansi maupun perusahaan dalam pengolahan data gaji pegawai tersebut masih memanfaatkan *Microsoft Excel* yang menyebabkan terjadinya redudansi data, proses perhitungan gaji tidak efektif, dan belum adanya database untuk menyimpan arsip data penggajian sehingga keamanan data tidak terjamin. Selain itu, Sering terjadi keterlambatan dalam pengelolaan data mengakibatkan lambatnya penyajian informasi sehingga terjadi keterlambatan pembayaran gaji dan lambatnya pembuatan laporan penggajian karena

proses pembuatan laporan dilakukan dengan *entry* ulang arsip.

Berawal dari uraian di atas maka akan dibuat sebuah aplikasi pengelolaan gaji yang berjudul Aplikasi Manajemen Absensi Untuk Realisasi Penggajian Pegawai yang diharapkan dapat memudahkan dalam pengelolaan dan perincian gaji pegawai.

1.2 Identifikasi Masalah

Berdasarkan uraian dari latar belakang diatas, maka Masalah yang dapat diidentifikasi anantara lain adalah sebagai berikut.

1. Pengolahan data penggajian yang cukup sulit karena masih memanfaatkan *Microsoft Excel* yang menyebabkan terjadinya redudansi data, proses perhitungan gaji tidak efektif.
2. Belum digunakannya database dalam penyimpanan arsip data penggajian sehingga keamanan data tidak terjamin.
3. Tidak adanya keamanan hak akses sehingga data dengan mudahnya diambil maupun dirubah oleh orang yang tidak berhak mengakses.
4. Sering terjadi keterlambatan dalam pembuatan laporan penggajian karena proses pembuatan laporan dilakukan dengan *entry* ulang arsip.

1.3 Tujuan

Tujuan pembuatan aplikasi penggajian pegawai adalah sebagai berikut :

1. Merancang aplikasi yang dapat mengelola data penggajian.
2. Merancang aplikasi dengan perancangan penggunaan database untuk penyimpanan data penggajian.
3. Merancang aplikasi dengan pembagian hak akses.
4. Merancang aplikasi yang dapat memudahkan pembuatan laporan secara tepat waktu.

1.4 Ruang Lingkup

Pembuatan tugas akhir ini merupakan suatu langkah kerja untuk menyelesaikan permasalahan dalam pembuatan program aplikasi untuk mempermudah dalam kegiatan Aplikasi Manajemen Absensi Untuk Realisasi Penggajian Pegawai. Adapun aplikasi yang dibuat dalam pembuatan tugas akhir ini mencakup:

1. Pengelolaan Data Pegawai.
2. Pengelolaan Data Tunjangan.
3. Pengelolaan Data Bonus.
4. Pengelolaan Data Kasbon.
5. Pengelolaan Data Bagian.
6. Pengelolaan Data Jabatan.
7. Pengelolaan Data Penggajian Pegawai.
8. Pengelolaan Pembagian Hak akses.

9. Laporan.

II. LANDASAN TEORI

2.1 Penggajian

Gaji atau sering disebut upah adalah suatu bentuk kompensasi yang merupakan imbalan jasa yang diberikan secara teratur atas hasil kerja yang diberikan kepada seorang pegawai. Dalam suatu penggajian terdapat besarnya gaji pokok yang diberikan, tunjangan baik itu tunjangan jabatan, tunjangan lembur, dan tunjangan lainnya. Besarnya gaji pokok yang diberikan kepada seorang pegawai, biasanya sangat tergantung pada latar belakang pendidikan yang dimiliki, kemampuan maupun pengalaman kerjanya.[1]

Sistem penggajian dan pengupahan melibatkan fungsi kepegawaian, fungsi keuangan dan fungsi akuntansi. Fungsi kepegawaian bertanggung jawab dalam mencari karyawan baru, menyeleksi calon karyawan, pengangkatan karyawan, penetapan jabatan, penetapan tarif gaji dan upah, promosi dan penurunan pangkat, mutasi karyawan, penetapan berbagai tunjangan kesejahteraan karyawan serta perhitungan gaji dan upah. Dalam struktur organisasi, fungsi kepegawaian berada di bawah Departemen Personalia dan Umum. Fungsi keuangan bertanggung jawab atas pelaksanaan pembayaran gaji dan upah serta tunjangan kesejahteraan karyawan. Fungsi akuntansi bertanggung jawab atas pencatatan biaya tenaga kerja dan distribusi biaya tenaga kerja untuk kepentingan perhitungan harga pokok produk dan penyediaan informasi guna pengawasan biaya tenaga kerja.[2]

2.2 Prosedur Sistem Penggajian

Gaji pegawai yang ada disetiap perusahaan , pada umumnya dibagiakan setiap akhir bulan. Dengan demikian, kegiatan yang dilakukan oleh bagian keuangan adalah sebagai berikut :[1]

1. Mengumpulkan Data Lembur Pegawai.
Data lembur pada setiap pegawai , pada setiap bulan akan dikumpulkan guna dipehitungkan dengan pembayaran yang harus dibayarkan oleh pihak perusahaan.
2. Menghitung Rupiah Lembur Masing-masing Pegawai.
Berdasarkan data lembur, bagian keungan akan menghitung nilai rupiah yang harus dibayarkan kepada pegawai pada setiap akhir bulan. Dikarenakan standar lembur masing-masing pegawai tidaklah sama, maka bagian keuangan perlu mencari keterangan lain yang dalam hal ini diambil dari kartu induk pegawai. seperti diketahui, pada kartu induk pegawai terdapat keterangan mengenai berapa besar gaji pokok

- pegawai yang bersangkutan dan berapa rupaiah yang harus dibayarkan untuk 1 jam lembur.
3. Membuat Rekapitulasi Gaji.
Hasil perhitungan lembur dan potongan setiap pegawai yang dipadukan dengan gaji pokok, pada akhirnya akan menghasilkan gaji bersih yang akan diterima pada bulan yang bersangkutan.
 4. Membuat Slip Gaji Untuk Masing-masing Gaji.
Setelah rekapitulasi gaji, maka langkah berikutnya adalah membuat slip gaji untuk masing-masing pegawai. Slip gaji dibuat dua rangkap, dimana satu lembar di serahkan kepada pegawai beserta uang yang mereka terima dan satu lembar berikutnya di simpan sebagai arsip keuangan.
 5. Mengambil Uang Dari Bank
Berdasarkan data keuangan yang muncul dan tertera pada rekapitulasi gaji pegawai, pihak direksi kemudian membuka cek dari bank yang bersangkutan guna diuangkan secara tunai. Uang inilah yang akan digunakan untuk membayar gaji pegawai.
 6. Memasukkan Uang Kedalam Amplop.
Setelah uang tunai yang didapatkan dari bank, tugas berikutnya adalah memasukkan uang tersebut kedalam amplop gaji pegawai yang besarnya sesuai dengan apa yang tertulis pada slip gaji pegawai.
 7. Membagikan Uang Kepada Pegawai
Pegawai akan dipanggil satu per satu untuk menerima gaji mereka, dan diminta untuk menandatangani kedua slip gaji yang telah disediakan. Satu lembar slip bersama amplop gaji kemudian diserahkan kepada pegawai, dan satu lembar lagi disimpan bagian keuangan yang digunakan sebagai arsip.

2.3 Apache, PHP, dan MySQL

Server HTTP *Apache* atau Server Web/WWW Apache adalah server web yang dapat dijalankan di banyak sistem operasi (Unix, BSD, Linux, Microsoft Windows dan Novell Netware serta platform lainnya) yang berguna untuk melayani dan memfungsikan situs web. Protokol yang digunakan untuk melayani fasilitas web/www ini menggunakan HTTP. Apache merupakan perangkat lunak sumber terbuka dikembangkan oleh komunitas terbuka yang terdiri dari pengembang-pengembang dibawah naungan Apache Software Foundation. *PHP* yang bisa disebut juga *Hypertext Preprocessor* adalah sebuah bahasa pemrograman *open source* yang digunakan secara luas terutama untuk pengembangan web yang disisipkan pada HTML. *MySQL* adalah *Relational Database Management System* (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (*General Public License*).

Dimana setiap orang bebas untuk menggunakan MySQL, namun tidak boleh dijadikan produk turunan yang bersifat closed source atau komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam database sejak lama, yaitu SQL (*Structured Query Language*).[3]

Hubungan antara Apache, PHP, dan MySQL adalah PHP merupakan sebuah bahasa pemrograman untuk membuat suatu aplikasi situs web, sedang MySQL adalah sebuah sistem basisdatanya, PHP sebagai *client* akan melakukan *query* (permintaan/perintah) ke sebuah MySQL server. Apache merupakan server web yang dapat dijalankan di banyak sistem operasi yang berguna untuk melayani dan memfungsikan situs web yang telah dibuat dengan PHP dan MySQL.

2.4 Microsoft Visio

Microsoft Visio (atau sering disebut Visio) adalah sebuah program aplikasi komputer yang sering digunakan untuk membuat diagram, diagram alir (*flowchart*), *brainstorm*, dan skema jaringan yang dirilis oleh Microsoft Corporation. Aplikasi ini menggunakan grafik vektor untuk membuat diagram-diagramnya.[4]

2.5 Flow Chart

Alur jalannya informasi maupun sistem kerja dari suatu aplikasi perlu digambarkan, agar user bisa tahu bagaimana cara menggunakan suatu aplikasi. Flow chart adalah bagan yang menggambarkan urutan-urutan instruksi untuk dilakukan proses dengan komputer dan hubungan antara suatu proses dengan proses lainnya, dengan menggunakan simbol-simbol tertentu.[4]

Dalam membuat program flow chart, tidak ada suatu rumus atau ketentuan dan patokan yang mutlak, karena pada dasarnya program flow chart merupakan gambaran hasil pemikiran seseorang dalam melakukan analisa penyelesaian suatu masalah dengan komputer.

2.6 Diagram Konteks

Diagram konteks adalah suatu gambaran yang sangat umum mengenai sebuah sistem, dimana digambarkan sebagai proses tunggal.

2.6.1 Data Flow Diagram

Data Flow Diagram adalah suatu diagram yang menggambarkan arus data dan cara kerja sistem. *Data Flow Diagram* menjelaskan aliran data, informasi proses, basis data dan sumber atau tujuan anda.

Data Flow Diagram menggunakan beberapa simbol untuk mewakili, yaitu: [5]

1. Kesatuan Luar (*External Entity*)

Setiap sistem pasti mempunyai batas sistem (*boundary*) memisahkan satu sistem dengan lingkungan luarnya. Sistem akan menerima input dan menghasilkan *output* kepada lingkungan luarnya. Kesatuan luar merupakan kesatuan di lingkungan luar sistem yang berupa orang, organisasi, atau sistem lainnya yang berada di lingkungan luarnya yang akan memberi input atau menerima output dari sistem.

2. Arus Data (*Data Flow*)

Arus data di DFD diberi simbol suatu panah. Arus data ini mengalir diantara proses, simpanan data, dan kesatuan luar. Arus data ini menunjukkan arus dari data yang berupa masukan untuk sistem atau hasil dari proses sistem dan dapat berbentuk sebagai berikut:

- a. Formulir atau dokumen yang digunakan di perusahaan.
- b. Masukan untuk komputer.
- c. Surat atau memo.

3. Proses

Proses adalah kegiatan atau kerja yang dilakukan oleh orang, mesin, atau komputer dari hasil suatu arus data yang masuk ke dalam proses untuk dihasilkan arus data yang akan keluar dari proses. Suatu proses dapat ditunjukkan dengan simbol lingkaran atau empat persegi panjang tegak dengan sudut-sudut tumpul. Setiap proses harus diberi penjelasan yang lengkap meliputi:

- a. Identifikasi proses
Umumnya berupa suatu angka yang menunjukkan nomor acuan dari proses dan ditulis pada bagian atas di simbol proses.
- b. Nama proses
Menunjukkan apa yang dikerjakan oleh proses tersebut. Nama dari proses harus jelas dan lengkap menggambarkan kegiatan prosesnya.
- c. Pemroses
Untuk *Physical Data Flow Diagram* (PDFD) menunjukkan siapa atau dimana suatu proses dilakukan.
- d. Simpanan Data (*Data Store*)

Simpanan data merupakan data yang dapat berupa file atau database di sistem komputer, arsip atau catatan manual, tabel acuan manual, agenda atau buku.

2.7 Pengujian

Pengujian adalah proses menjalankan program dengan maksud untuk mencari kesalahan (*error*). Pengujian dikatakan berhasil bila dapat memunculkan kesalahan yang belum diketahui. Jadi, pengujian yang baik bukan untuk memastikan tidak ada kesalahan tetapi untuk mencari sebanyak mungkin kesalahan yang ada di program. Pengujian tidak dapat menunjukkan ketidakhadiran *defect*, pengujian hanya menunjukkan bahwa kesalahan perangkat lunak ada. [6]

Adapun prinsip dasar pengujian adalah sebagai berikut : [6]

1. Semua pengujian harus dapat dirunut ke *requirement*.
2. Pengujian harus direncanakan jauh sebelum dilakukan.
3. Prinsip Pareto berlaku pada pengujian perangkat lunak yaitu 80 % dari keseluruhan kesalahan yang didapat saat pengujian dapat dirunut ke 20 % dari keseluruhan modul tetapi persoalannya bagaimana menemukan 20% modul tersebut.
4. Pengujian harus dimulai dari lingkup kecil ke lingkup besar.
5. Tidak mungkin ada pengujian yang *exhaustive*.
6. Supaya efektif, pengujian harus dilakukan oleh pihak lain yang *independent*.

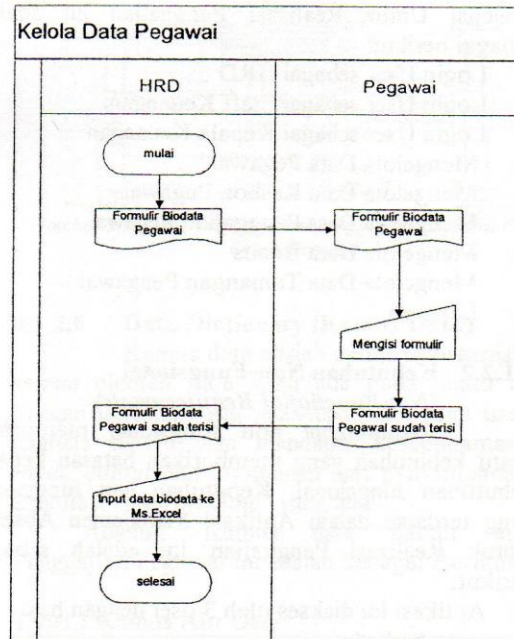
Pengujian yang baik harus memiliki peluang yang besar dalam menemukan kesalahan, tidak berganda (*redundant*), mewakili semua kemungkinan, tidak terlalu sederhana dan tidak terlalu rumit. Berikut adalah strategi pengujian :

1. Spesifikasikan kebutuhan (*requirement*) produk dalam bentuk yang dapat diukur (*quantifiable*) jauh sebelum pengujian dimulai.
2. Nyatakan tujuan pengujian secara eksplisit.
3. Pahami pengguna perangkat lunak dan buatlah *profil* dari tiap kategori pengguna.
4. Buat rencana pengujian yang menekankan pada *rapid cycle testing*.
5. Buat perangkat lunak *robust* yang dapat menguji dirinya sendiri.
6. Gunakan *formal technical review* sebagai penyaring sebelum pengujian dilakukan.
7. Gunakan *formal technical review* untuk menilai strategi pengujian dan kasus uji.
8. Kembangkan rancangan peningkatan berlanjut untuk proses pengujian.

Sedangkan hal-hal yang harus diuji dalam pengujian perangkat lunak adalah :

1. Antarmuka modul : memastikan bahwa informasi yang berasal dari dan keluar dari modul yang diuji mengalir dengan benar.

2. Struktur data local : memastikan bahwa data yang disimpan sementara menjaga integritasnya selama eksekusi perintah dalam modul dan mencari kesalahan-kesalahan dalam bentuk penggunaan tipe yang tidak tepat, inisialisasi yang salah atau nilai pasti (*default*), nama peubah yang salah, tipe data yang tidak tepat, *underflow*, *overflow*, dan *addressing exceptions*.
3. Kondisi batas : memastikan bahwa modul beroperasi dengan benar pada batas-batas pemrosesan yang ditentukan.
4. Jalur-jalur bebas : memastikan bahwa semua kemungkinan jalur kontrol yang mungkin dieksekusi dengan benar paling tidak sekali dan mencari kesalahan-kesalahan dalam bentuk penghitungan/pemrosesan yang salah, perbandingan yang salah dan alur kontrol yang tidak tepat.
5. Jalur-jalur penanganan kesalahan : perancangan perangkat lunak yang baik menuntut agar kondisi salah dapat diantisipasi dan memiliki penanganan kesalahan agar pemrosesan dapat berhenti dengan bersih (*antibugging*).



Gambar 1 Flowmap Kelola Data Pegawai

III. ANALISIS DAN PERANCANGAN

3.1 Analisis

Analisis merupakan langkah awal untuk pengembangan aplikasi, karena perancangan bahkan pengembangan implementasi aplikasi tidak akan terwujud dengan baik tanpa adanya analisa terhadap aplikasi yang sedang digunakan saat ini.

Dreamweaver adalah aplikasi yang digunakan untuk menuliskan kode-kode bahasa visual yang membentuk suatu aplikasi yang *user friendly*, seperti *PHP* yang bisa disebut juga *Hypertext Preprocessor* adalah sebuah bahasa pemrograman *open source* yang digunakan secara luas terutama untuk pengembangan web yang disisipkan pada HTML. [3]

Aplikasi yang kedua, yaitu *MYSQL* merupakan software DBMS (*Database Management System*) sebagai tempat menyimpan semua data mengenai objek tertentu. Dengan *MYSQL* sebuah database akan mudah dikelola walaupun jumlah datanya kompleks dan banyak. [3]

3.1.1 Analisis Sistem yang Sedang Berjalan

Tahap ini menjelaskan hasil pengamatan terhadap sistem yang sedang berjalan terdiri dari deskripsi prosedur kerja dan deskripsi dokumen yang digunakan.

3.1.1.1 Analisis Prosedur/Flowmap yang Sedang Berjalan

3.1.1.1.1 Flowmap Kelola Data Pegawai

Keterangan :

1. HRD memberikan formulir biodata pegawai kosong kepada pegawai.
2. Pegawai kemudian mengisi formulir kosong tersebut, setelah mengisi formulir kemudian diberikan kembali ke HRD.
3. Kemudian HRD menginputkan data formulir yang telah terisi ke Ms.Excel untuk menjadi data yang lebih rapi.

3.1.1.1.2 Analisis Dokumen yang Digunakan.

Analisis dokumen ini menguraikan dokumen-dokumen yang sedang berjalan, antara lain yaitu :

1. Dokumen formulir data pegawai

Fungsi : Keterangan untuk mengetahui biodata pegawai

Sumber : HRD

Rangkap : 1 (satu)

Distribusi : Pegawai

Frekuensi : Setiap Pegawai

Isi dokumen: nik, nama karyawan, tempat tanggal lahir, jenis kelamin, alamat, agama, status perkawinan, email, no tlp, pendidikan terakhir.

3.1.2 Analisis Sistem yang akan Dibangun

3.1.2.1 Kebutuhan Fungsional (*Functional Requirements*)

Requirement fungsional atau biasa juga disebut dengan kebutuhan fungsional, merupakan suatu kebutuhan yang berhubungan dengan kebutuhan software yang akan dibangun. Kebutuhan fungsional yang terdapat dalam Aplikasi Manajemen

Absensi Untuk Realisasi Penggajian ini adalah sebagai berikut:

- 1) Login User sebagai HRD
- 2) Login User sebagai Staff Keuangan
- 3) Login User sebagai Kepala Keuangan
- 4) Mengelola Data Pegawai.
- 5) Mengelola Data Kasbon Pegawai.
- 6) Mengelola Data Penggajian Pegawai.
- 7) Mengelola Data Bonus
- 8) Mengelola Data Tunjangan Pegawai
- 9) Laporan

3.1.2.2 Kebutuhan Non-Fungsional (NonFunctional Requirements)

Requirement non fungsional merupakan suatu kebutuhan yang memberikan batasan kepada kebutuhan fungsional. Kebutuhan non fungsional yang terdapat dalam Aplikasi Manajemen Absensi Untuk Realisasi Penggajian ini adalah sebagai berikut:

- 1) Aplikasi ini diakses oleh 3 user dengan hak yang berbeda
- 2) User sebagai HRD mempunyai hak akses penuh terhadap semua data yang terdapat didalam database kecuali perhitungan gaji.
- 3) User sebagai Dinas Kepala Keuangan mempunyai hak perhitungan gaji, kelola jabatan, dan report.
- 4) User sebagai Kepala Keuangan mempunyai hak mencetak dan melihat informasi report.
- 5) Setiap User yang menginginkan akses harus terdaftar terlebih dahulu

Aspek yang penting dari requirement non fungsional adalah merubah kebutuhan user menjadi jelas, ringkas, dan dapat diverifikasi. Sehingga, requirement yang baik menyatakan sesuatu yang dibutuhkan, dapat diverifikasi, memungkinkan, dan jelas.

Adapun asumsi yang buruk terjadi karena requirement tidak mempunyai akses yang cukup terhadap informasi yang dibutuhkan. Hal ini bisa dikurangi dengan cara mendokumentasikan informasi yang kritis seperti:

- 1) **Karakteristik Pengguna**
Menjelaskan karakteristik umum yang diinginkan terhadap pengguna produk
- 2) **Batasan**
Berisi deskripsi umum dari bagian lain yang akan membatasi pilihan pengembangan.
- 3) **Lingkungan Operasi**
Berisi tentang deskripsi operating sistem, DBMS, dan aplikasi client server yang akan berfungsi dengan spesifikasi server client OS, atau DBMS.

3.1.2.2.1 Karakteristik Pengguna

Minimal sebuah tabel dengan kolom mengenai karakteristik dan ciri dari setiap

pengguna. Terdiri dari 3 pengguna yang terdiri dari HRD, Staff Keuangan dan Kepala Keuangan.

3.1.2.2.2 Batasan-batasan

Batasan-batasan dalam pembuatan aplikasi ini adalah :

- 1) Aplikasi ini dapat diakses oleh HRD, Staff Keuangan, dan Kepala Keuangan
- 2) Aplikasi ini berfokus pada proses pengelolaan gaji pegawai. Sehingga dapat memudahkan dalam pengelolaan dan perincian gaji pegawai.

3.1.2.2.3 Asumsi dan Kebergantungan

Semua pengguna yang bisa login hanya HRD, Staff Keuangan, dan Kepala Keuangan yang telah terdaftar.

Lingkungan Operasi

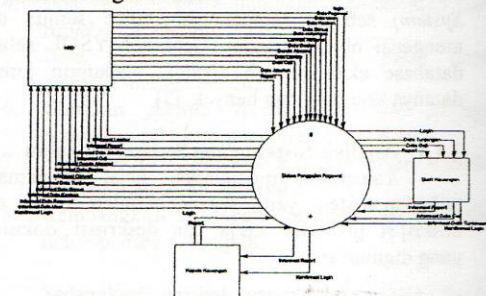
Kebutuhan perangkat lunak yang akan digunakan dalam pembangunan sistem ini antara lain :

- a. Sistem Operasi : Microsoft Windows XP
- b. Web Server : Appserv
- c. Data Base : MySQL
- d. Design Web : Dreamweaver 8

3.1.2.3 Context Diagram

Context diagram merupakan tingkatan tertinggi dalam diagram aliran data dan hanya memuat satu proses, menunjukkan sistem secara keseluruhan. Proses tersebut diberi nilai nol. Semua entitas eksternal yang ditunjukkan pada diagram konteks berikut aliran data – aliran data utama menuju dan dari sistem. Diagram tersebut tidak memuat penyimpanan data dan tampak sederhana untuk diciptakan, begitu entitas-entitas eksternal serta aliran data – aliran data menuju dan dari sistem diketahui penganalisis dari wawancara dengan user sebagai hasil analisis dokumen.

Gambar Context Diagram untuk Aplikasi Manajemen Absensi Untuk Realisasi Penggajian adalah sebagai berikut:



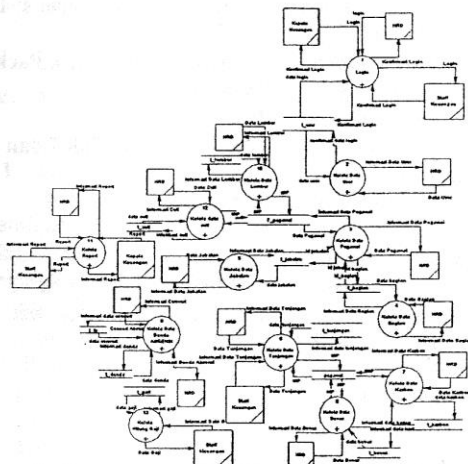
Gambar 2 Context Diagram Sistem Penggajian Pegawai

3.1.2.4 Data Flow Diagram (DFD)

Data Flow Diagram (DFD) adalah suatu diagram yang menggunakan notasi-notasi untuk arus

dari data sistem, yang penggunaannya sangat membantu untuk memahami sistem secara logika, terstruktur dan jelas. Atau DFD juga bisa dikatakan untuk menggambarkan darimana asal data dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut.

3.1.2.4.1 Data Flow Diagram Level 1 Sistem Penggajian Pegawai

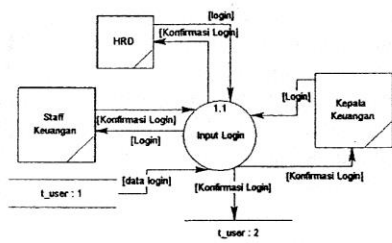


Gambar 3 DFD Level I Aplikasi Penggajian Pegawai

3.1.2.5 Proses Spesifikasi

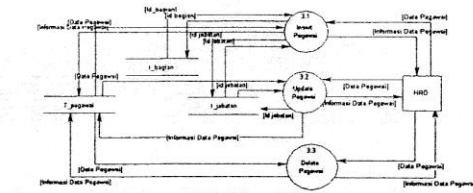
Proses Spesifikasi adalah menjelaskan tentang spesifikasi dari proses dalam perancangna aplikasi Penggajian Pegawai ini

3.1.2.5.1 DFD Proses Login Level 2



Gambar 4 DFD Level II Aplikasi Penggajian Pegawai Proses Login

3.1.2.5.2 DFD Proses Kelola Data Pegawai Level 2



Gambar 5 DFD Level II Aplikasi Penggajian Pegawai Proses Kelola Pegawai

3.1.2.6 Data Dictionary (Kamus Data)

Kamus data adalah daftar terorganisir dari semua elemen data yang ada pada suatu sistem dengan definisi yang jelas/tepat, sehingga user dan analisis sistem bisa mendapat kesepahaman dari input, output dan komponen dari penyimpanan dan kalkulasi "intermediate" yang ada.

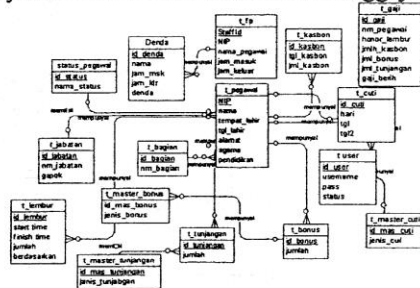
Adapun Kamus data dalam aplikasi Penggajian Pegawai ini adalah Sebagai Berikut:

Tabel 1 Kamus Alir Data

User	
Deskripsi	Menyatakan Data user
Struktur Data	t_user = {id user + status + password }
Pegawai	
Deskripsi	Menyatakan Data Pegawai
Struktur Data	t_Pegawai = {NIP+nama_pegawai+tempat lahir+tanggal lahir+alamat+agama+status+pendidikan+id_bagian+id_jabatan+foto}
Bagian	
Deskripsi	Menyatakan Data Bagian
Struktur Data	t_Bagian = {id_bagian,nama_bagian}
Jabatan	
Deskripsi	Menyatakan Data Jabatan
Struktur Data	t_pendaftaran = {id_jabatan+nama_jabatan}

3.1.2.7 CDM

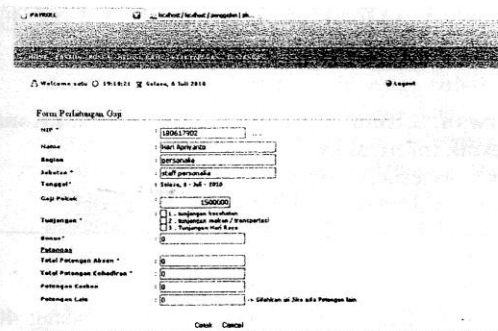
Berikut ini adalah CDM dari Aplikasi Manajemen Absensi Untuk Realisasi Penggajian



Gambar 6 CDM Aplikasi Penggajian Pegawai

3.1.2.8 PDM

Berikut ini adalah PDM dari Aplikasi Manajemen Absensi Untuk Realisasi Penggajian



Gambar 10 tampilan Hitung Gaji

IV. KESIMPULAN DAN SARAN

4.1 Kesimpulan

Kesimpulan yang dapat diambil dari Pembuatan aplikasi yang berjudul Aplikasi Manajemen Absensi Untuk Realisasi Penggajian Pegawai adalah sebagai berikut :

1. Dengan Dibuatnya aplikasi ini, masalah perhitungan gaji pegawai, bagian staff keuangan dimudahkan dalam perhitungan gaji pegawai, staff keuangan tidak perlu lagi menghitung uang tunjangan, bonus, maupun lembur pegawai karena pada aplikasi ini perhitungan akan terhitung otomatis sehingga staff keuangan tidak perlu lagi menghitung ulang gaji bersih pegawai sehingga perhitungan gaji menjadi lebih cepat.
2. Adanya database untuk penyimpanan arsip data penggajian sehingga keamanan data dapat terjamin.
3. Adanya keamanan hak akses untuk melihat dan merubah data.
4. Dengan dibuatnya aplikasi ini tidak akan terjadi keterlambatan dalam pembuatan laporan penggajian. Karena aplikasi ini dapat lebih cepat dan lebih mudah karena user tidak perlu lagi mengetik ulang untuk membuat laporan, pada aplikasi ini untuk mencetak laporan hanya tinggal memilih menu cetak laporan dan memilih laporan apa yang akan dicetak.

4.2 Saran

Adapun saran dalam pembuatan aplikasi ini yaitu:

1. Aplikasi ini diharapkan dapat dikembangkan agar dapat terintegrasi langsung dengan hardware seperti Fingerprint untuk proses absensinya.
2. Aplikasi selanjutnya diharapkan dapat dikombinasikan dengan layanan *sms-gateway* untuk memberikan informasi dari perusahaan kepada seluruh karyawan *via sms*.

Daftar Pustaka

1. [1] http://id.wikipedia.org/wiki/Sistem_informasi (diakses tanggal 14 juni 2010)
2. [2] Andi. 2003. *Aplikasi Database dengan VB 6.0*. Bandung: Informatika.
3. [3] Betha Sidik, Ir. 2006. *Pemrograman WEB dengan PHP*. Bandung: Informatika Bandung.
4. [4] Nugroho, Adi. 2004. *Konsep Pengembangan Sistem Basis Data*. Bandung: Informatika Bandung.
5. [5] Iskandar Pohan, Husni, Saiful Bahar, Kushasriyanto. 1997. *Pengantar Perancangan Sistem*. Jakarta: Erlangga.
6. [6] Nirwan, Saepudin. 2005. *Handout Rekayasa Perangkat Lunak*. Bandung: Politeknik Pos Indonesia