

BAB II

Landasan Teori

2.1 Tinjauan Objek Penelitian

Penelitian terdahulu adalah upaya peneliti untuk mencari perbandingan dan selanjutnya untuk menemukan inspirasi baru untuk penelitian selanjutnya di samping itu kajian terdahulu membantu penelitian dalam memposisikan penelitian serta menunjukkan orsinalitas dari penelitian. Pada bagian ini peneliti mencantumkan berbagai hasil penelitian terdahulu yang terkait dengan penelitian yang hendak dilakukan, kemudian membuat ringkasannya, baik penelitian yang sudah terpublikasikan atau belum terpublikasikan (skripsi, tesis, disertasi dan sebagainya). Dengan melakukan langkah ini, maka akan dapat dilihat sejauh mana orisinalitas dan posisi penelitian yang hendak dilakukan.

2.1.1 Penelitian Terdahulu

a. Penggunaan metode SDLC *Waterfall* pada Sistem Informasi Penggajian Karyawan Borongan pada PT Indokom Samudra Persada

Penelitian yang dilakukan oleh Intan Pertiwi pada tahun 2022 memiliki tujuan utama adalah untuk mempermudah pengelolaan data hasil kerja para karyawan dan pembuatan laporan yang lebih efisien serta aman, dengan harapan dapat mengurangi potensi kesalahan dalam penyampaian laporan. Aplikasi penggajian karyawan yang dikembangkan dalam penelitian ini diharapkan akan memberikan informasi yang berguna dalam pengeluaran perusahaan, sehingga dapat meningkatkan kelancaran operasional perusahaan.

Dalam pengembangan sistem ini, penulis menerapkan metode pendekatan berorientasi objek, yaitu metode *Waterfall*. Untuk mengumpulkan data yang diperlukan, penulis menggunakan berbagai teknik, seperti observasi dan wawancara. Namun, penting untuk dicatat bahwa dalam penerapan sistem ini, peneliti hanya menggunakan metode tersebut untuk mengelola penyimpanan data karyawan dan absensi. Padahal, salah satu aspek yang sangat penting dalam penggajian karyawan borongan adalah penghitungan gaji yang harus dilakukan setiap harinya. Sayangnya, sistem yang dikembangkan belum mencakup penghitungan hasil kerja harian ini, yang merupakan aspek kunci dalam pengelolaan penggajian karyawan borongan.

b. Rancang Bangun Sistem Informasi Penggajian Karyawan PT.

Boetomix Nusantara Lestari Berbasis *WEB*

Penelitian yang dilakukan oleh Taufiq dkk pada tahun 2021 memiliki tujuan untuk mengembangkan sistem informasi penggajian berbasis *web* yang lebih efisien dan efektif untuk PT. Boetomix Nusantara Lestari. Metode pengembangan sistem yang digunakan dalam penelitian ini adalah metode *Waterfall*, yang merupakan pendekatan sistematis yang dimulai dari komunikasi, pembuatan rencana, pembuatan model, pembuatan program aplikasi, hingga pengantaran sistem dan perawatan secara bertahap agar sistem bisa berjalan dengan baik. Sistem yang dibuat sudah diujicoba dan diterapkan pada PT. Boetomix Nusantara Lestari. Tetapi masih

memiliki kelemahan yaitu input data dilakukan dengan manual, tidak menggunakan alat *scan barcode*, sehingga masih rawan untuk terjadi kesalahan input data.

c. Sistem Informasi Penggajian Pegawai Harian Lepas (PHL) Berbasis Website Di CV. IBM

Penelitian yang dilakukan oleh Rizqy dkk pada tahun 2022 bertujuan untuk menyederhanakan proses perhitungan gaji pegawai harian lepas di CV. IBM. Dalam penelitian ini, mereka berhasil mengembangkan sebuah sistem berbasis *web* yang menggunakan bahasa pemrograman PHP dan *database* MySQL untuk mengelola data gaji pegawai harian lepas. Metode pengembangan sistem yang mereka pilih adalah metode *Waterfall*, yang melibatkan langkah-langkah yang terstruktur dalam pengembangan sistem.

Namun, perlu diperhatikan bahwa penelitian ini memiliki kelemahan utama. Kelemahan tersebut adalah bahwa perhitungan gaji pegawai harian lepas dilakukan berdasarkan kehadiran pegawai, bukan berdasarkan kinerja mereka. Ini berarti bahwa sistem yang dikembangkan hanya memperhatikan kehadiran pegawai, sementara faktor kinerja yang seharusnya menjadi pertimbangan dalam perhitungan gaji tidak dimasukkan. Oleh karena itu, dalam pengembangan sistem selanjutnya, penting untuk mempertimbangkan penilaian kinerja pegawai sebagai salah satu faktor dalam perhitungan

gaji agar sistem dapat memberikan gambaran yang lebih akurat tentang penggajian pegawai harian lepas di CV. IBM.

2.2 Konsep Dasar Web

Saat Ini fasilitas yang banyak digunakan orang untuk keperluan promosi bisnis seperti mengakses berbagai informasi dan keperluan komunikasi yaitu dengan menggunakan internet. Secara sederhana menurut Harjono (2009:1) "Internet adalah kumpulan dari beberapa komputer, bahkan jutaan komputer dari jutaan komputer diseluruh dunia yang saling berhubungan atau terkoneksi satu sama lainnya". Media yang digunakan bisa menggunakan kabel atau serat optic, satelit atau melalui sambungan telepon. Pendapat ini mengartikan bahwa internet merupakan media komunikasi dan informasi modern yang dapat dimanfaatkan secara global oleh pengguna diseluruh dunia dalam inter koneksi antar jaringan komputer yang terbentuk melalui sarana berupa penyedia akses (provider) internet, sehingga internet sebagai media informasi dapat menjadi yang efektif dan efisien untuk melakukan transaksi dan penyebaran informasi tanpa terhalang oleh jarak, perbedaan waktu dan juga faktor geografis bagi seseorang yang ingin mengakses informasi. Untuk itu pada bab ini akan membahas mengenai internet serta aplikasi yang bisa digunakan untuk membuat sebuah *website* dengan mudah yaitu *Macromedia Dreamweaver* dan bahasa pemrograman yang digunakan dalam membuat sebuah *website* yaitu PHP. Selain itu pembahasan ini mencakup juga peralatan pendukung untuk membuat *database* yang sangat diperlukan oleh sebuah *website* seperti struktur navigasi, ERD, LRS dan sebagainya.

2.1.1 Website

Menurut Rachdian (2008:1) mengemukakan bahwa “*website* adalah suatu pengenalan ruang informasi dimana sumber-sumber daya yang berguna diidentifikasi global yang disebut *Uniform Resource Identifier* (URI) atau lebih dikenal dengan istilah yang lebih populer yaitu *Uniform Resource Locator* (URL)”. Perkembangan *World Wide Web* (WWW) yang sangat pesat ditandai dengan munculnya berbagai macam *website* dengan halaman *web* yang interaktif. Penggunaan teknologi internet, khususnya *website* karena penggunaannya yang mudah, tampilan yang menarik dan cangkupan yang luas karena *website* dapat dilihat di seluruh dunia, selama mereka mempunyai koneksi internet (Gani, 2022).

a. Internet

Pengertian internet Menurut Priyanto (2009:9) mengemukakan bahwa “Internet adalah (*Interconnected Network*) kumpulan jaringan komputer diseluruh dunia yang saling berhubungan satu dengan yang lainnya”. Dimulai dari suatu proyek pada sekitar tahun 60’an Departemen pertahanan Amerika Serikat melalui *Advance Reasearc Projects Agency* (ARPA) merintis sebuah sistem jaringan yang dinamakan ARPANET atau *Advance projects network*. Tahun 1980’an, internet mulai digunakan secara terbatas untuk menghubungkan beberapa Universitas terkemuka di Amerika Serikat, kemudian beberapa universitas di Amerika Serikat mengerjakan proyek ini, dan awalnya berhasil menghubungkan 4 komputer di lokasi universitas yang berbeda. Perkembangan ARPANET ini cukup pesat jika dilihat dari perkembangan komputer pada saat ini.

Pada Tahun 1977, ARPANET telah menghubungkan lebih dari 100 main fraem komputer, dan saat ini terdapat sekitar 4 juta host jaringan yang terhubung dengan jaringan tersebut.

b. *Web server*

Pengertian *Web server* menurut Nugraha dkk, (2018) adalah “*Web server* merupakan *software* yang memberikan layanan data yang berfungsi menerima permintaan HTTP atau HTTPS dari klien yang dikenal dengan browser *web* dan mengirim kembali hasilnya dalam bentuk halaman-halaman *web* yang umumnya berbentuk dokumen HTML”.

Penggunaan *webservice* dalam mendukung penetrasi layanan teknologi informasi sangat besar, terutama dalam menyampaikan informasi kepada pengguna(Faudji dkk, 2023). Banyak sekali vendor-vendor penyedia *webservice* mulai dari yang berbayar maupun gratis (*opensource*), dengan masing-masing menawarkan kelebihan yang dimilikinya. Vendor-vendor tersebut seperti *Internet Information Services (IIS)* yang merupakan kepemilikan *Microsoft*, *Apache*, *Nginx* dan *Google Web server*. Namun yang populer digunakan adalah *Apache* dan *IIS*.

- *Apache Web server*

Menurut Firdaus (2010:5) “*Web server* terbaik berdasarkan riset dari Netcraft dengan pengguna terbanyak”. *Apache* memiliki tingkat keahandalan dan kestabilan yang tinggi serta fitur-fitur yang terbilang lengkap, hal ini yang membuat banyak pemakai mempercayakan *apache* sebagai *web server* mereka. Selain itu sifatnya sebagai *open*

source membuat *web server* ini menjadi *web server* yang gratis. Sifat yang *open source* ini membuat para pengguna *web server* memfavoritkannya dan banyak dukungan-dukungan yang datang dari komunitas komunitas dan sponsor untuk ikut mengembangkan *web server* ini sehingga menjadikan *web server* ini semakin handal. *Apache web server* dapat berjalan pada sistem operasi berbasis *Windows* atau *Unix*. 10

- *Internet Information Services (IIS)*

Menurut mohamad Iqbal (2007:3) “*Internet Information Services* Adalah *web server* buatan *Microsoft*”. Berbeda dengan *Apache* yang *open source*, untuk mendapatkan dukungan teknis penuh maka otomatis kita sebagai pengguna harus membeli lisensi *IIS* dari *Microsoft*. *IIS* menawarkan sistem jaringan antar muka yang mudah, aman dan stabil. Sesuai dengan perusahaan dimana *IIS* dikembangkan, maka *IIS* hanya bisa berjalan pada sistem operasi berbasis *Windows*.

- *Nginx Web server*

Menurut Purnomo (2013:9) “Alternatif lain *web server* yang bersifat *open source* adalah *Nginx* (dibaca *engine-x*)”. Kelebihan dari *Nginxweb server* hampir sama seperti *Apache*, bahkan beberapa *review* dari pengguna adalah konfigurasi *Nginx* yang lebih mudah dari pada *apache*. Jadi untuk pengguna yang ingin menjalankan *web server*

yang ringan dengan sistem pengoperasian yang mudah, maka cukup menggunakan *Nginx*.

- *Google Web server (GWS)*

Menurut Nugroho (2015:11) “*Google Web server* Adalah *web server* buatan *Google Inc* yang bersifat tertutup dan hanya di gunakan oleh pihak *Google* sendiri sebagai *web server* mereka”. Banyak info yang bisa didapat mengenai *web server* yang satu ini dikarenakan masih dalam riset *Google*.

2.3 Bahasa Pemrograman

2.3.1 *Personal Home page (PHP)*

Personal Home page adalah teknologi yang diperkenalkan tahun 1994 oleh Rasmus Lerdof. Menurut Purnomo (2009:3), PHP merupakan bahasa skrip yang 11 ditempatkan dalam server dan diproses diserver. PHP sendiri mempunyai fungsi untuk menterjemahkan *script/kode* PHP yang kita buat. Dengan PHP ini dapat dibuat beragam aplikasi *web*, mulai dari halaman *web* yang sederhana sampai dengan aplikasi kompleks yang membutuhkan koneksi ke *database*. Sampai saat ini telah banyak *database* yang didukung oleh PHP dan kemungkinan akan terus bertambah. *Database* tersebut adalah *Base*, *DBM*, *mSQL*, *SQL*, *ODBC*, *Oracle*, *Postgres*, *Sybase*, *Velocois*, *HTML* (E. S. Purnomo & Samopa, 2013).

Menurut Peranginangin (2011:2) mengemukakan bahwa “PHP adalah suatu program yang digunakan sebagai bahasa *Script server-side* dalam pengembangan *web* yang disisipkan pada dokumen *HTML*”. Sistem kerja dari program ini adalah sebagai *interpreter* dan bukan sebagai *compiler*. Dikatakan

sebagai bahasa *interpreter*, *script* mentahnya tidak harus diubah kedalam bentuk *source code*. Sehingga pada saat menjalankan program, kode dasar secara langsung akan dijalankan tanpa harus melalui perubahan kedalam bentuk *source code* (A. S. Purnomo, 2015). Diantara kelebihan-kelebihan dari PHP:

- a. Mudah dibuat dan kecepatan akses tinggi.
- b. Dapat berjalan dengan *web server* yang berbeda dan dalam sistem operasi yang berbeda pula seperti *Unix*, *Windows NT* dan *Macintosh*.
- c. Diterbitkan secara gratis.
- d. Dapat berjalan pada *web server* *Microsoft Personal Web server* dan *Apache*.
- e. Termasuk bahasa yang ditempel atau diletakan dalam tag HTML (*embedded*).
- f. Kemampuan untuk mengolah keluaran gambar (seperti file PDF, dan *Movie flash*).

2.3.2 *CodeIgniter*

Menurut Budi Raharjo (2015:3), “*CodeIgniter* adalah *framework web* untuk bahasa pemrograman PHP yang dibuat oleh Rick Ellis pada tahun 2006, penemu dan pendiri *EllisLab*. *EllisLab* adalah suatu tim kerja yang berdiri pada tahun 2002 dan bergerak di bidang pembuatan *software* dan *tool* untuk para pengembang *web*”.

CodeIgniter memiliki banyak fitur (fasilitas) yang membantu para pengembang (*developer*) PHP untuk dapat membuat aplikasi *web* secara mudah dan cepat. Dibandingkan dengan *framework web* PHP lainnya, harus diakui

bahwa *CodeIgniter* memiliki desain yang lebih sederhana dan bersifat fleksibel (tidak kaku). *CodeIgniter* mengizinkan para pengembang untuk menggunakan *framework* secara parsial atau secara keseluruhan. *CodeIgniter* merupakan sebuah *toolkit* yang ditujukan untuk orang yang ingin membangun aplikasi *web* dalam bahasa pemrograman PHP.

Beberapa keunggulan yang ditawarkan oleh *CodeIgniter* adalah sebagai berikut:

- *CodeIgniter* adalah *framework* yang bersifat free dan open-source.
- *CodeIgniter* memiliki ukuran yang kecil dibandingkan dengan *framework* lain. Setelah proses instalasi, *framework CodeIgniter* hanya berukuran kurang lebih 2MB (tanpa dokumentasi atau jika direktori *user_guide* dihapus). Dokumentasi *CodeIgniter* memiliki ukuran sekitar 6MB.
- Aplikasi yang dibuat menggunakan *CodeIgniter* bisa berjalan cepat.
- *CodeIgniter* menggunakan pola desain *Model-View-Controller* (MVC) sehingga satu file tidak terlalu berisi banyak kode. Hal ini menjadikan kode lebih mudah dibaca, dipahami, dan dipelihara di kemudian hari.
- *CodeIgniter* dapat diperluas sesuai dengan kebutuhan.
- *CodeIgniter* terdokumentasi dengan baik. Informasi tentang Pustaka kelas dan fungsi yang disediakan oleh *CodeIgniter* dapat diperoleh melalui dokumentasi yang disertakan di dalam paket distribusinya.

2.3.3 *Hypertext Markup Language* (HTML)

Menurut Kuswayatno (2012:14) “*Hypertext Markup Language (HTML)* merupakan salah satu varian dari SGML (*Standard Generalized Markup Language*), yaitu sebuah standar dari ISO (*International Organization Standarization*) untuk pertukaran dokumen secara elektronik “. HTML sendiri secara formal diumumkan sebagai RFC 1866. HTTP digunakan sebagai protokol dalam pertukaran dokumen. Dalam dokumen *web*, dokumen ini akan dieksekusi oleh browser, sehingga browser mampu menghasilkan suatu dokumen sesuai dengan keinginan yang mendesain *page*. Dokumen ini mempunyai kemampuan menampilkan gambar, suara, teks, maupun penyediaan link terhadap halaman *web* lainnya, baik dengan alamat yang sama maupun berbeda.

Hypertext Markup Language (HTML) adalah sebuah bahasa *markup* yang digunakan untuk membuat sebuah halaman *web*, menampilkan berbagai informasi didalam sebuah penjelajah *web Internet* dan *formatting hypertext* sederhana yang ditulis kedalam berkas format ASCII agar dapat menghasilkan tampilan wujud yg terintegrasi. Dengan kata lain, berkas yang dibuat dalam perangkat lunak pengolah kata dan disimpan kedalam format ASCII normal sehingga menjadi *home page* dengan perintah-perintah HTML. HTML saat ini merupakan standar internet yang di definisikan dan dikendalikan penggunaannya oleh *World Wide Web Consortium (W3C)*. HTML dibuat oleh kerjasama antara Caillau TIM dengan Berners-lee robert di CERN pada tahun 1989 (CERN adalah Lembaga Penelitian Fisika Energi Tinggi di Jenewa).

2.3.4 JQuery

Menurut Resig (2012:10) “*jQuery* adalah sebuah *library Javascript* yang sangat ringkas dan sederhana untuk memanipulasi komponen didalam HTML, menangani *even*, animasi, efek dan memproses interaksi *ajax*”. *jQuery* dirancang sedemikian rupa supaya membuat program menggunakan *Javascript* menjadi relatif mudah. *jQuery* ukurannya cukup kecil sehingga tidak memperlambat proses loading halaman *web* yang kita buat. *jQuery* mampu berjalan disemua browser-*cross browser*.

jQuery diluncurkan Januari 2012 oleh John Resig. *jQuery* merupakan *library* yang paling populer saat ini, karena kecanggihannya *jQuery* banyak digunakan oleh perusahaan besar seperti *Google, Dell, CBS, digg, Netflix, Bank of America, Mozilla, Drupal* dan sebagainya. Karakteristik *jQuery* yang gratis dan *open source*, maka kita dapat menggunakan *jQuery* dengan bebas untuk pengembangan *website* memudahkan kita. Plugin tambahan seperti *jQuery UI (User Inetrface)* semakin memudahkan dalam membuat *website* yang lebih cantik dan interaktif.

2.3.5 Cascading Style Sheet (CSS)

Nama CSS didapat dari fakta bahwa setiap deklarasi *style* yang berbeda dapat diletakan secara berurutan, yang kemudian membentuk hubungan ayah-anak (*parent-child*) pada setiap *style*. CSS sendiri merupakan sebuah teknologi internet yang direkomendasikan oleh World Wide Web Consortium atau W3C pada tahun 1996. Setelah CSS di standarisasikan, *Internet Explorer* dan *Netscape* melepas browser terbaru mereka yang telah sesuai atau paling tidak hampir mendekati dengan standar CSS.

Menurut Ardhana (2012a:108) *Cascading Style Sheet* atau biasa disebut (CSS) merupakan salah satu bahasa pemrograman *web* untuk mengendalikan beberapa komponen dalam sebuah *web* sehingga akan lebih terstruktur dan seragam. Sama halnya *style* dalam aplikasi pengolahan kata seperti *Microsoft Word* yang dapat mengatur beberapa *style*, misalnya *heading*, *subbab*, *bodytext*, *footer*, *images*, dan *style* lainnya untuk dapat digunakan bersama-sama dalam beberapa berkas (file). Pada CSS dipakai untuk memformat tampilan dalam halaman *web* yang dibuat dengan bahasa HTML dan XHTML.

2.3.6 *Macromedia Dreamweaver*

Menurut Nugroho (2008:10) “*Dreamweaver* adalah sebuah *software* HTML editor profesional yang digunakan untuk mendesain secara visual dan mengelola situs *web* maupun halaman *web*”. Apabila kita menyukai berurusan dengan kode-kode HTML secara manual atau lebih menyukai bekerja dengan lingkungan secara visual dalam melakukan editing. *Dreamweaver* membuatnya menjadi lebih mudah dengan menyediakan *tools* yang sangat berguna dalam peningkatan kemampuan dan pengalaman kita dalam mendesain *web*. Dalam hal ini *Dreamweaver 8* digunakan untuk mendesain *web*. *Dreamweaver 8* mengikutsertakan banyak *tools* untuk kode-kode dalam halaman *web* beserta fasilitas-fasilitasnya, antara lain : Referensi HTML,CSS dan *Javascript*, *Javascript debugger*, dan editor kode (tampilan kode dan *inspector*) yang mengizinkan kita mengedit kode HTML, dan dokumen teks lain secara langsung dalam *Dreamweaver*.

Teknologi *Dreamweaver Roundtrip HTML* mampu mengimpor dokumen HTML tanpa harus memformat ulang kode tersebut dan kita dapat juga 15 menggunakan *Dreamweaver* untuk membersihkan dan memformat ulang HTML bila kita menginginkannya. Selain itu *Dreamweaver* juga dilengkapi kemampuan manajemen situs, yang memudahkan kita mengelola keseluruhan elemen yang ada dalam situs. Kita juga dapat melakukan evaluasi situs dengan melakukan pengecekan *broken link*, kompatibilitas browser, maupun perkiraan waktu mengunduh halaman *web*.

2.4 Basis Data

2.4.1 Definisi Basis Data

Menurut Kristanto (2010:13) “Basis data adalah sekumpulan data yang saling terhubung secara logical, dirancang untuk memenuhi kebutuhan organisasi”. Sedangkan sistem basis data adalah sekumpulan aplikasi program yang berinteraksi dengan basis data melalui DBMS dan basis data itu sendiri.

Basis data adalah sekumpulan relasi data persisten yang secara logika terhubung dimana data tersebut merupakan deskripsi dari satu atau lebih aktivitas dari organisasi yang bersangkutan yang digunakan oleh aplikasi sistem perusahaan yang dibagi dan saling berhubungan yang dirancang demi memenuhi kebutuhan organisasi. Sedangkan sistem basis data dapat disimpulkan sebagai sekumpulan aplikasi program yang saling berinteraksi dengan basis data melalui DBMS dan basis data itu sendiri dan juga merupakan suatu sistem penyimpanan record yang sudah terkomputerisasi(Sastra, 2017).

2.4.2 *Data Base Management System (DBMS)*

Menurut Nugroho (2015:10) “*Data Base Management System (DBMS)* adalah suatu sistem perangkat lunak yang didesain untuk menangani pengelolaan dan penggunaan dari suatu kumpulan data, serta memungkinkan pengguna untuk mendefinisikan, membuat, memelihara, dan mengendalikan akses terhadap sistem basis data”. Fungsi-fungsi dasar yang harus didukung DBMS adalah:

a. Pendefinisian data (*data definition*)

DBMS harus dapat menerima pendefinisian data (skema eksternal, skema konseptual dan semua asosiasi pemetaan) dari sumber dan mengkonversikan ke dalam bentuk objek yang sesuai.

b. Manipulasi data (*data manipulation*)

DBMS harus dapat menangani permintaan untuk mengambil, memperbaharui atau menghapus data yang sudah ada di basis data, maupun menambah data kedalam basis data.

c. Optimalisasi dan eksekusi (*optimization and execution*)

Permintaan *Data Manipulation Language (DML)* harus diproses dikomponen pengoptimalisasi yang bertujuan untuk menentukan cara yang efisien untuk implementasi permintaan. Permintaan yang telah dioptimalisasi kemudian dieksekusi dibawah kendali *re-time manager*.

d. Keamanan dan integritas data (*data security and integrity*)

DBMS harus mengawasi permintaan pengguna dan menolak gangguan yang dapat membahayakan keamanan dan *integrity constraint* yang sudah ditentukan oleh *Database Administrator (DBA)*.

e. Perbaikan data dan konkurensi (*data recovery and concurrency*)

DBMS dapat juga disebut *Transaction Processing Monitor* (TPM) harus melakukan kendali perbaikan dan konkurensi.

f. Kamus data (*data dictionary*)

Kamus data berisi “data mengenai data” adalah definisi dari objek lain di sistem. Semua skema dan pemetaan, berbagai sistem keamanan, dan *integrity constraint* akan disimpan, baik dari sumber maupun bentuk objek didalam kamus data.

g. Kinerja (*performance*)

DBMS harus dapat mengerjakan semua tugas seefektif mungkin.

2.5 Barcode

2.5.1 Pengertian & sejarah

(Apache friends:2013) Perkembangan *Barcode* sendiri dimulai dari tahun 1932, saat Wallace Flint membuat sistem pemeriksaan barang di sebuah perusahaan retail yang kemudian diikuti oleh perusahaan industri. *Barcode* pertama kalinya diperkenalkan dan dipatenkan di Amerika oleh Norman Joseph Woodland dan Bernard Silver, mahasiswa *Drexel Institute of Technology* pada akhir 40-an. Implementasi *Barcode* dimungkinkan atas kerja keras dua orang insinyur yaitu Raymond Alexander dan Frank Stietz. Pada tahun 1948 sampai 1949 Bernard Silver dan Norman Joseph Woodland mengembangkan teknologi *Barcode* ini menjadi lebih baik. Sampai akhirnya di tahun 1952, mereka mendapatkan hak paten dari hasil penelitian tersebut. Penggunaan *Barcode* untuk keperluan komersial dimulai sejak tahun 1966.

Pengertian *Barcode* dapat diartikan sebagai kumpulan kode yang berbentuk garis, dimana masing-masing ketebalan setiap garis berbeda sesuai dengan isi kodenya.

Barcode termasuk dalam unit masukan (*input device*). Fungsi alat ini adalah untuk membaca suatu kode yang berbentuk kotak-kotak atau garis-garis tebal vertical yang kemudian diterjemahkan dalam bentuk angka-angka.

Kode-kode ini biasanya menempel pada produk-produk makanan, minuman, alat elektronik dan buku. Sekarang ini, setiap kasir di supermarket atau pasar swalayan di Indonesia untuk mengidentifikasi produk yang dijualnya dengan *Barcode* (Febriyanti & Agustina, 2018).

Barcode dibaca dengan menggunakan sebuah alat baca *Barcode* atau lebih dikenal dengan *Scanner Barcode*. Merk *Scanner Barcode* yang terkenal diantaranya DATALOGIC PSC, HHP, CHIPERLAB, ZEBEX, dan lain-lain. Seiring semakin bertambahnya penggunaan *Barcode*, kini *Barcode* tidak hanya bisa mewakili karakter angka saja tapi sudah meliputi seluruh kode ASCII (Simanjuntak & Arfan B, 2015).

Kebutuhan akan kombinasi kode yang lebih rumit itulah yang kemudian melahirkan inovasi baru berupa kode matriks dua dimensi (*2D Barcodes*) yang berupa kombinasi kode matriks bujur sangkar. *2D Barcode* ini diantaranya adalah *PDF Code*, *QRCode*, *Matrix Code* dan lain-lain. Dengan menggunakan *2D code* karakter yang bisa kita masukkan ke *Barcode* bisa semakin banyak, dengan *1D Barcode* biasanya kita hanya memasukkan kode 5-20 digit tetapi dengan *2D Barcode* kita bisa memasukkan sampai ratusan digit kode. Saya sendiri pernah

menangani pembuatan program di Toyota Tsusho Logistic Centre yang menggunakan 2D *Code* sampai 200 digit karakter.

(Roger C.Palmer : 2013) Terdapat beberapa standar kode dalam *Barcode* sesuai dengan kegunaan dan tujuan pemakaian *Barcode*, seperti pada daftar berikut:

- a. ***Uniform Product Code*** (UPC) : untuk checkout penjualan, persediaan, dan sebagainya pada toko retail.
- b. ***Code 39*** (*Code 3 of 9*) : identifikasi, inventarisasi, dan pengiriman pelacakan.
- c. **POSTNET** : kode pos *encoding* di US mail.
- d. ***European Article Number*** (EAN) : sebuah superset dari UPC yang memungkinkan digit ekstra untuk identifikasi negara.
- e. ***Japanese Article Number*** (JAN) : serupa dengan EAN, digunakan di Jepang.
- f. ***Bookland*** : berdasarkan nomor ISBN dan digunakan pada sampul buku.
- g. ***ISSN bar code*** : berdasarkan nomor ISSN, digunakan pada majalah di luar AS.
- h. ***Code 128*** : digunakan dalam preferensi untuk *Code 39* karena lebih kompak.
- i. ***Interleaved 2 of 5*** : digunakan dalam industri pelayaran dan gudang.
- j. ***Codabar*** : digunakan oleh *Federal Express*, di perpustakaan dan bank darah.

- k. **MICR** (*Magnetic Ink Character Recognition*) : sebuah font khusus yang digunakan untuk nomor di bagian bawah cek bank.
- l. **OCR-A** : format pengenalan karakter optik yang digunakan pada sampul buku, untuk nomor ISBN agar bisa dibaca oleh manusia.
- m. **OCR-B** : digunakan untuk mempermudah pembacaan *Barcode* versi UPC, EAN, JAN, *Bookland*, dan ISSN dan *Code 39*.
- n. **Maxicode** : digunakan oleh *United Parcel Service*.
- o. **PDF417** : suatu jenis *Barcode* 2-D baru yang dapat *encode* sampai 1108 *byte* informasi; dapat terkompresi seperti pada sebuah *portabel file data* (PDF).
- p. **UPC-E**, terdiri dari 7 digit, yaitu 6 digit data, 1 check digit : untuk bisnis retail skala kecil.
- q. **EAN-13 atau UPC-A** versi Eropa, terdiri dari 13 digit, yaitu 12 digit data, 1 check digit.
- r. **European Articles Numbering (EAN)-8**, terdiri dari 8 digit, yaitu 2 digit kode negara, 5 digit data, 1 check digit.
- s. **CODE 39**, sebagai simbolik yang paling populer di dunia *Barcode* non-retail, dengan variabel digit yang panjang. Namun saat ini *code 39* makin sedikit dipergunakan dan digantikan dengan *Code 128* yang lebih mudah dibaca oleh pemindai.

Barcode Product yang banyak di Indonesia adalah EAN 13, yaitu kode Ink *Barcode* dengan 13 digit. Dimana 3 kode awalnya merupakan kode negara Indonesia (899). Kemudian empat angka berikutnya menunjukkan kode

perusahaan. Selanjutnya lima angka secara berturut-turut merupakan kode produk (*Ink Barcode*) dan angka terakhir berupa validasi atau cek digit.

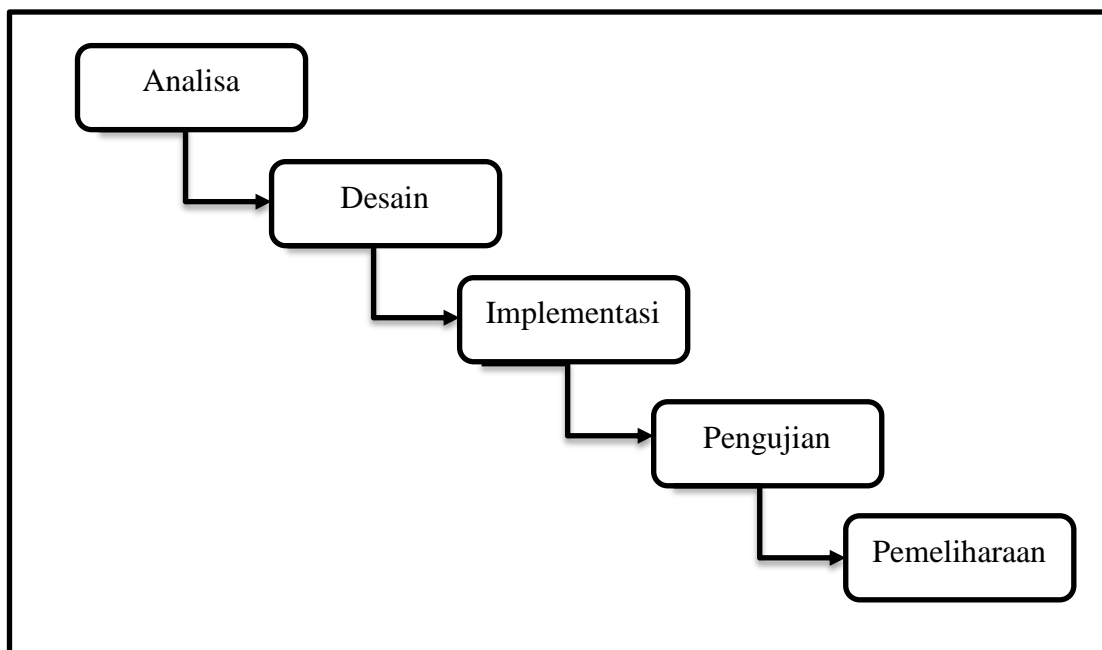
2.6 *Zxing*

ZXing (dibaca “*Zebra Crossing*”) adalah sebuah *library* pengolahan citra *Barcode* 1D/2D yang diterapkan dalam bahasa pemrograman *Java*. *Library* ini dikembangkan dengan fokus agar dapat melakukan pemindaian *Barcode* menggunakan kamera yang terpasang pada sebuah alat tanpa harus berkomunikasi dengan sebuah server. Akan tetapi *library* ini juga dapat digunakan untuk menyalin dan menerjemahkan sebuah *Barcode* pada desktop dan server. Pada versi terbarunya *ZXing* telah mendukung format-format berikut: UPC-A, UPC-E, EAN-8, EAN-13, *Code 39*, *Code 93*, *Code 128*, ITF, Codabar, RSS-14 (semua jenis), *RSS Expanded* (hampir semua jenis), *Bar code*, *Data Matrix*, *Aztec* ('beta' quality), dan PDF 417 ('alpha' quality).

2.7 Metode Perancangan Sistem

Metode perancangan yang digunakan pada perancangan sistem informasi laporan gaji pegawai borongan di PT. Otsuka Indonesia ini adalah metode pengembangan sistem SDLC (*System Development Life Cycle*) *Waterfall*. Dengan SDLC proses membangun sistem dibagi menjadi beberapa langkah dan pada sistem yang besar masing masing langkah dikerjakan oleh tim yang berbeda. SDLC sangat bermanfaat bukan hanya untuk proses produksi perangkat lunak, melainkan juga untuk proses perawatannya.

Model pengembangan yang digunakan adalah model *Waterfall* (air terjun). Model SDLC *Waterfall* menyediakan pendekatan alur hidup perangkat lunak secara sekuensial (urut) dimulai dari analisis, pengumpulan data, pengkodean sistem, pengujian dan tahap implementasi. Skema model pengembangan ini ditunjukkan pada Gambar 2.5



Gambar 2.1 Kerangka kerja pengembangan sistem informasi (SDLC).

Tahapan dari metode SDLC pada gambar 2.5 memetakan kegiatan – kegiatan perancangan dasar yaitu:

2.7.1 Analisis

Analisis kebutuhan merupakan tahap pertama yang menjadi dasar proses pembuatan selanjutnya. Kelancaran proses pembuatan *software* secara keseluruhan dan kelengkapan fitur *software* yang dihasilkan sangat tergantung pada hasil analisa kebutuhan ini. Untuk memperoleh informasi tentang proses bisnis dan kebutuhan perusahaan, umumnya tim pengembang melakukan

wawancara, diskusi dan survey. Hasil analisa kebutuhan yang tidak lengkap berpotensi menyebabkan beberapa permasalahan yang tidak diharapkan, antara lain: waktu pembuatan *software* menjadi lebih lama, proses dalam *software* tidak sesuai dengan proses bisnis dan *software* tidak dapat memenuhi semua kebutuhan perusahaan. Analisa Desain Implementasi Pengujian Pemeliharaan

2.7.2 Desain

Desain sistem merupakan tahap penyusunan proses, data, aliran proses dan hubungan antar data yang paling optimal untuk menjalankan proses bisnis dan memenuhi kebutuhan perusahaan sesuai dengan hasil analisa kebutuhan. Dokumentasi yang dihasilkan dari tahap desain sistem ini antara lain: *System Flow*, *Data Flow Diagram (DFD)* dan *Entity Relationship Diagram (ERD)* (Octafian, 2011).

2.7.3 Implementasi

Implementasi merupakan Penulisan kode program yaitu tahap penerjemahan desain sistem yang telah dibuat ke dalam bentuk perintah-perintah yang dimengerti komputer dengan menggunakan bahasa pemrograman, dan *database* tertentu di atas *platform* yang menjadi standar perusahaan. PHP untuk pembuatan *software* dalam kategori *Web Application* dan Java untuk pembuatan *software* yang dapat berjalan di atas beberapa *platform* sistem operasi / *multiplatform*. *Database* yang saat ini banyak dipergunakan antara lain: ORACLE dan *Microsoft SQL Server 2005* untuk kebutuhan penyimpanan data dalam jumlah besar.

2.7.4 Pengujian

Pengujian *software* dilakukan untuk memastikan bahwa *software* yang dibuat telah sesuai dengan desainnya dan semua fungsi dapat dipergunakan dengan baik tanpa ada kesalahan. Pengujian *software* biasanya dilakukan dalam dua tahap, yaitu: pengujian oleh internal tim pengembang, dan pengujian oleh pengguna di perusahaan.

2.7.5 Pemeliharaan

Pemeliharaan dilakukan ketika sistem informasi sudah dioperasikan. Pada tahapan ini dilakukan *monitoring* proses, evaluasi dan perubahan (perbaikan) bila diperlukan.

2.8 Teori Pendukung

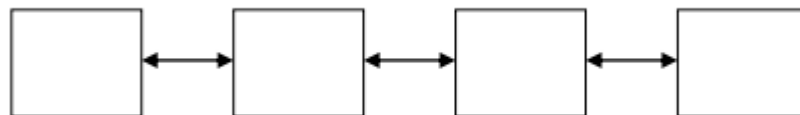
Dalam pembuatan suatu program, berhasil tidaknya suatu program tergantung dari persiapan sebelumnya, misalnya membuat Struktur Navigasi atau ERD. Untuk itu, dalam proses pembuatan program kali ini penulis membutuhkan beberapa peralatan pendukung (*tools system*) yang membantu mempermudah penulis dalam membuat suatu program. Adapun peralatan pendukung tersebut terdiri dari:

2.8.1 Struktur Navigasi

Menurut Sutisna (2008:26) “Struktur navigasi merupakan struktur dari sebuah program, dan merupakan bagian yang paling penting didalam pembuatan multimedia”. Struktur navigasi terbagi menjadi empat, yaitu:

a. Struktur Navigasi *Linear*

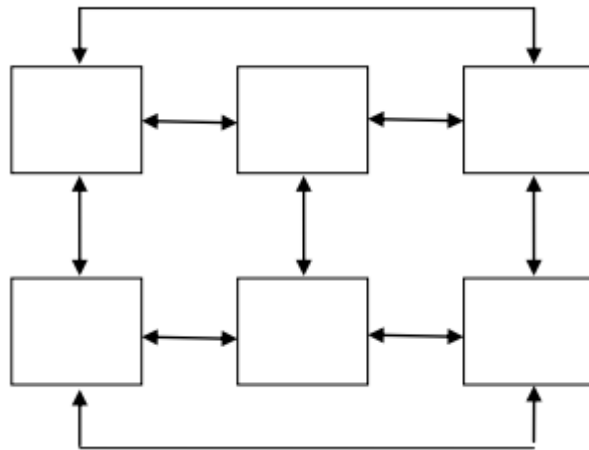
Merupakan struktur yang hanya mempunyai satu rangkaian cerita berturut. Pada umumnya dibuat untuk halaman *web* presentasi karena tidak terlalu menuntut keinteraksian tetapi perlu keindahan tampilan dan kemudahan menampilkan data sebagai informasi. Dalam struktur ini tidak boleh ada percabangan.



Gambar 2.2 Struktur Navigasi *Linear*. Sumber :Sutisna (2008:29)

b. Struktur Navigasi *Non Linear*

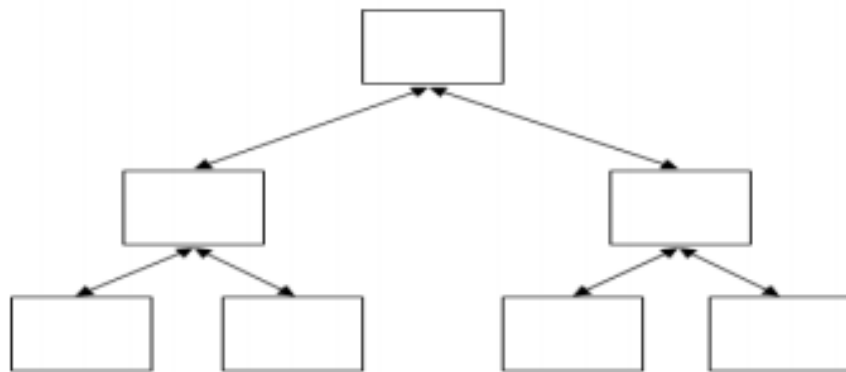
Merupakan pengembangan dari struktur navigasi *linear*. Pada struktur ini diperbolehkan mempunyai cabang, tapi tiap cabang mempunyai kedudukan yang sama. Tidak ada *master page* dan *slave page*.



Gambar 2.3 Struktur Navigasi *Non Linear*. Sumber :Sutisna (2008:31)

c. Struktur Navigasi *Hirarki*

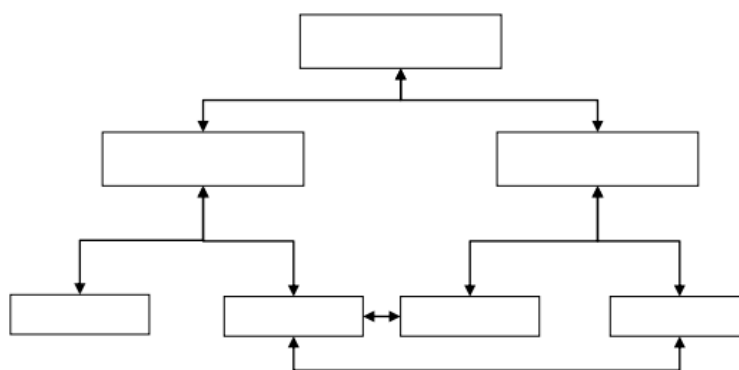
Merupakan struktur navigasi yang memerlukan percabangan untuk menampilkan data berdasarkan bagian tertentu. Tampil pada menu pertama disebut *master page*, tampilan pertama dicabangkan lagi dengan nama *slave page*.



Gambar 2.4 Struktur Navigasi *Hirarki*. Sumber: Sutisna (2008:33)

d. Struktur Navigasi Campuran

Struktur navigasi campuran (*composite*) merupakan gabungan dari struktur sebelumnya dan disebut juga struktur navigasi bebas, maksudnya adalah jika suatu tampilan membutuhkan percabangan maka dibuat percabangan. Struktur ini paling banyak digunakan dalam pembuatan aplikasi multimedia.



Gambar 2.5 Struktur Navigasi *Composite*. Sumber :Sutisna (2008:35)

2.8.2 *Entity Relationship Diagram*

1. Pengertian ERD

Menurut Brady dan Loonam (2010:13) “*Entity-Relationship Model (ERM)* merupakan abstrak dan konseptual representasi data”. *EntityRelationship* adalah salah satu metode pemodelan basis data yang digunakan untuk menghasilkan skema konseptual untuk jenis/model data semantik sistem. Dimana sistem seringkali memiliki basis data relasional, dan ketentuan bersifat *top-down*. Diagram untuk menggambarkan model *Entity Relationship* disebut *Entity Relationship Diagram (ERD)*.

2. Komponen ERD

Komponen dalam *Entity Relationship Diagram* adalah sebagai berikut:

a. Entitas

Adalah segala sesuatu yang dapat digambarkan oleh data. Entitas juga dapat diartikan sebagai individu yang mewakili sesuatu yang nyata (eksistensinya) dan dapat dibedakan dari sesuatu yang lain. Ada dua macam entitas yaitu entitas kuat dan entitas lemah. Entitas kuat merupakan entitas yang tidak memiliki ketergantungan dengan entitas lainnya. Contohnya entitas anggota. Sedangkan entitas lemah merupakan entitas yang kemunculannya tergantung pada keberadaan entitas lain dalam suatu relasi. Entitas digambarkan dengan persegi panjang.

b. Atribut

Atribut merupakan pendeskripsian karakteristik dari entitas. Atribut digambarkan dalam bentuk lingkaran atau elips. Atribut yang menjadi kunci entitas atau key diberi garis bawah.

c. *Relationship*

Relasi menunjukkan adanya hubungan diantara sejumlah entitas yang berasal dari himpunan entitas yang berbeda. Dalam hal ini digambarkan dengan garis lurus.

d. Derajat *Relationship*

Derajat *relationship* yang menjelaskan jumlah entitas yang berpartisipasi dalam suatu *relationship*. Dalam Derajat *Relationship* terbagi dari tiga tingkatan yakni :

- *Unary Degree* (Derajat Satu)

Adalah satu buah *relationship* menghubungkan satu buah *entity*.

- *Binary Degree* (Derajat Dua)

Adalah satu buah *relationship* yang menghubungkan dua buah *entity*.

- *Ternary Degree* (Derajat Tiga)

Adalah satu buah *relationship* menghubungkan tiga buah *entity*.

e. *Cardinality Ratio Constraint*

Menunjukkan jumlah maksimum entitas yang dapat berelasi dengan entitas pada himpunan entitas yang lain. Jenis-jenis *Cardinality* :

- *One To One* (1 : 1)

Yaitu perbandingan antara *entity* pertama dengan *entity* kedua berbanding satu berbanding satu.

- *One To Many* (1 : M)

Yaitu perbandingan antara *entity* pertama dengan *entity* kedua berbanding satu berbanding banyak.

- *Many To One* (M : 1)

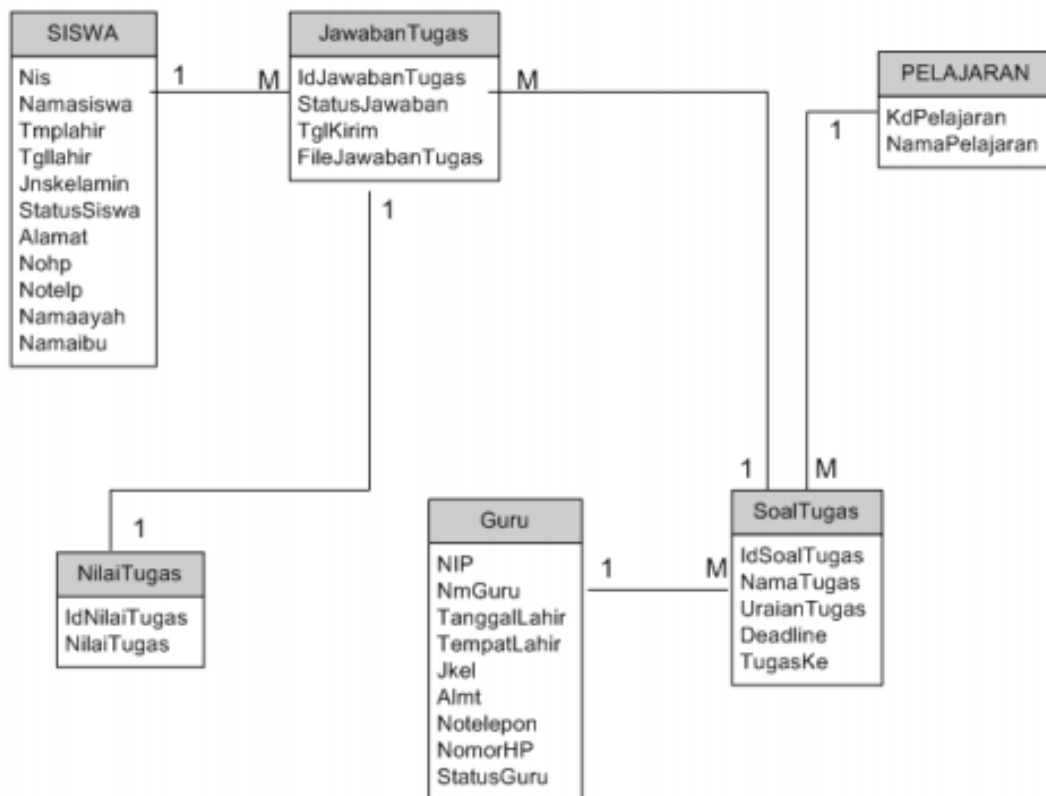
Yaitu perbandingan antara *entity* pertama dengan *entity* kedua berbanding banyak berbanding satu.

- *Many To Many* (M : M)

Yaitu perbandingan antara *entity* pertama dengan *entity* kedua berbanding banyak berbanding banyak

2.8.3 Logical Record Structure (LRS)

Menurut Purnomo (2008:15) "pengertian LRS adalah representasi dari struktur *record-record* pada tabel-tabel yang terbentuk dari hasil antar himpunan entitas yang digunakan untuk Menentukan kardinalitas, jumlah *table* dan *Foreign Key* (FK)". Berikut adalah contoh gambar LRS :



Gambar 2.6 Contoh LRS Sumber Purnomo (2008:15)

2.9 Sistem Informasi laporan gaji

2.9.1 Pengertian Sistem Informasi

Sistem adalah suatu bentuk integrasi antara satu komponen dengan komponen lainnya. karena sistem memiliki sasaran yang berbeda untuk setiap kasus yang terjadi yang ada didalam sistem tersebut(Siddiq & Pitrawati, 2020). Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolah transaksi, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan (Romzi, 2014).

Cara-cara yang untuk mengumpulkan, memasukkan dan mengelolah 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 di tetapkan. (Krismiaji, 2015). Sedangkan menurut Satyawan dkk, (2013) suatu sistem yang dibuat oleh manusia yang terdiri dari komponen-komponen dalam organisasi untuk mencapai suatu tujuan yaitu menyajikan informasi.

Sistem informasi mempunyai peranan penting dalam membantu menyediakan informasi untuk berbagai tingkatan 10 manajemen. Suatu sistem informasi dibuat untuk suatu keperluan tertentu, maka struktur dan cara kerja sistem informasi berbeda-beda bergantung pada macam keperluan dan permintaan yang harus terpenuhi.

Dari beberapa pengertian di atas dapat dipahami bahwa sistem informasi adalah sistem informasi mencakup sejumlah komponen (manusia, teknologi, informasi, prosedur kerja), sesuatu yang diproses (data menjadi informasi) dan dimaksudkan untuk mencapai suatu sasaran atas tujuan.

2.9.2 Tujuan dan Manfaat Sistem Informasi

Menurut Krismiaji (2015) tujuan umum penyusunan sistem informasi adalah:

- a. Untuk memperbaiki informasi yang diberikan oleh sistem dalam kualitas, ketepatan waktu atau struktur dari informasi tersebut.
- b. Untuk memperbaiki pengendalian akuntansi dan pengecekan intern, yang berarti memperbaiki daya andal informasi dan menyediakan catatan yang lengkap sebagai pertanggungjawaban dalam melindungi harta perusahaan.
- c. Untuk menurunkan biaya dalam menyelenggarakan catatan akuntansi.

Menurut Krismiaji (2015) Manfaat Sistem Informasi antara lain adalah sebagai berikut:

- a. Meningkatkan aksesibilitas data yang tersaji secara tepat waktu dan akurat bagi para pemakai, tanpa mengharuskan adanya prantara sistem informasi.
- b. Menjamin terjadinya kualitas dan keterampilan dalam memanfaatkan sistem informasi secara kritis.
- c. Mengembangkan proses perencanaan yang efektif.
- d. Mengidentifikasi kebutuhan-kebutuhan akan keterampilan pendukung sistem informasi.
- e. Menetapkan investasi yang akan diarahkan pada system informasi.
- f. Mengantisipasi dan memahami konsekuensi-konsekuensi ekonomis dari sistem informasi dan teknologi baru.
- g. Memperbaiki produktivitas dalam aplikasi pengembangan dan pemeliharaan sistem.

2.9.3 Fungsi Sistem Informasi

Informasi merupakan data yang telah diolah menjadi bentuk yang lebih berguna dan berarti bagi yang menerimanya. Di dalam dunia usaha saat ini yang penuh persaingan, informasi merupakan suatu yang berharga karena dengan penguasaan yang baik atas informasi tersebut perusahaan akan dapat mengambil atau memperoleh kesempatan yang lebih luas untuk berkembang. Dengan adanya informasi yang memadai suatu perusahaan akan dapat mengambil keputusan yang akan mendukung kemajuan perusahaan itu sendiri.

Menurut Krismiaji (2015) di dalam organisasi, sistem informasi berfungsi untuk:

- a. Mengumpulkan dan menyimpan aktifitas yang dilaksanakan di suatu organisasi, sumber daya yang dipengaruhi oleh aktifitas-aktifitas tersebut dan para pelaku aktifitas tersebut.
- b. Mengubah data menjadi informasi yang berguna bagi manajemen.
- c. Menyediakan pengendalian yang baik.

2.9.4 Pengertian Sistem Informasi laporan gaji

Untuk mengatasi adanya kesalahan dan penyimpangan dalam perhitungan dan pembayaran gaji maka perlu dibuat suatu sistem penggajian. Sistem akuntansi gaji juga dirancang oleh perusahaan untuk memberikan gambaran yang jelas mengenai gaji karyawan sehingga mudah dipahami dan mudah digunakan.

Menurut Mulyadi (2016) menyatakan bahwa sistem informasi laporan gaji dirancang untuk menangani transaksi gaji atau upah karyawan pembayarannya.

Sistem informasi akuntansi penggajian dapat digunakan oleh manajemen perusahaan untuk merencanakan dan mengendalikan operasi perusahaan. Menurut Krismiaji, (2015) Sistem informasi akuntansi adalah sistem yang memproses data dan transaksi guna menghasilkan informasi yang bermanfaat untuk merencanakan, mengendalikan, dan mengoperasikan bisnis. Sedangkan Sistem penggajian memegang peranan penting karena sistem ini akan menentukan berapa besar gaji yang semestinya akan diterima karyawan (Satyawan dkk, 2013).

Dari beberapa pengertian diatas dapat dipahami bahwa Sistem Informasi laporan gaji adalah keseluruhan prosedur dan teknik yang diperlukan untuk mengumpulkan data dan mengelolanya sehingga menjadi bahan informasi bagi pihak yang berkepentingan sehingga dapat memenuhi kebutuhan penggajian yang efektif.

2.9.5 Flowchart Penggajian

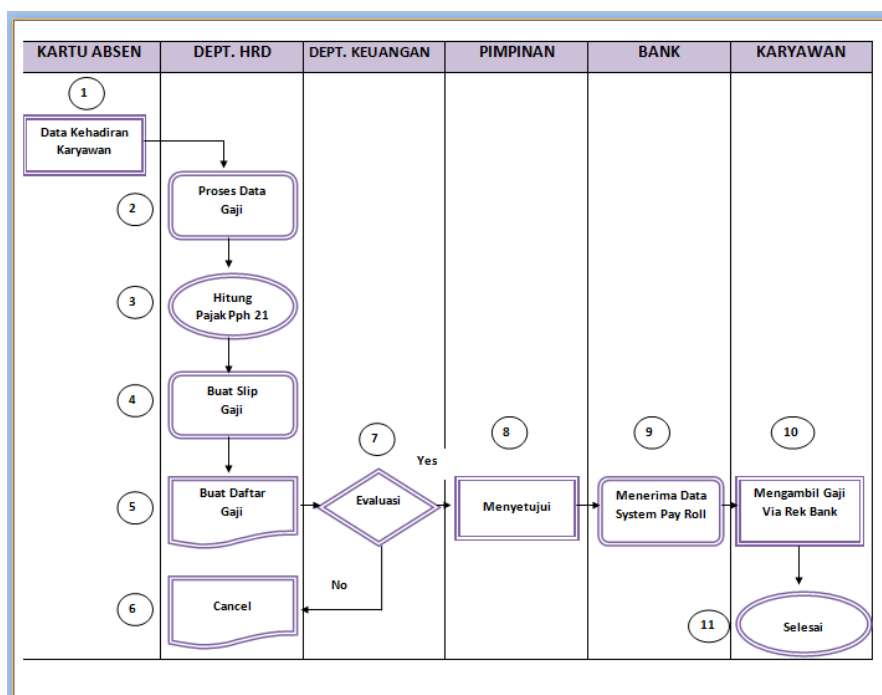
Menurut Mulyadi (2016) *Flowchart* adalah aliran dokumen dalam sistem tertentu, digunakan simbol-simbol dalam suatu bagan aliran dokumen (*Flowchart*). Dalam bagan alir, arus dokumen ini dapat diakui dengan melihat nomer dalam simbol dalam simbol penghubung pada halaman yang sama (*on-page connector*).

Penggunaan *Flowchart* lebih bermanfaat dibandingkan dengan uraian tertulis dalam menggambarkan suatu sistem. Menurut Mulyadi (2016) manfaat *Flowchart* tersebut sebagai berikut:

- a. Gambaran sistem menyeluruh lebih mudah diperoleh dengan menggunakan bagan alir.

- b. Perubahan sistem lebih mudah digambarkan dengan menggunakan bagan alir.
- c. Kelemahan-kelemahan dalam sistem dan identifikasi bidang-bidang yang melakukan perbaikan lebih mudah ditemukan dengan bagan alir.
- d. Dokumen sistem akuntansi dilakukan dengan menggunakan bagan alir.

Adapun *Flowchart* penggajian dapat dilihat dalam gambar berikut ini:



Gambar 2.7 *Flowchart* penggajian Menurut Mulyadi (2016).

Keterangan *Flowchart*:

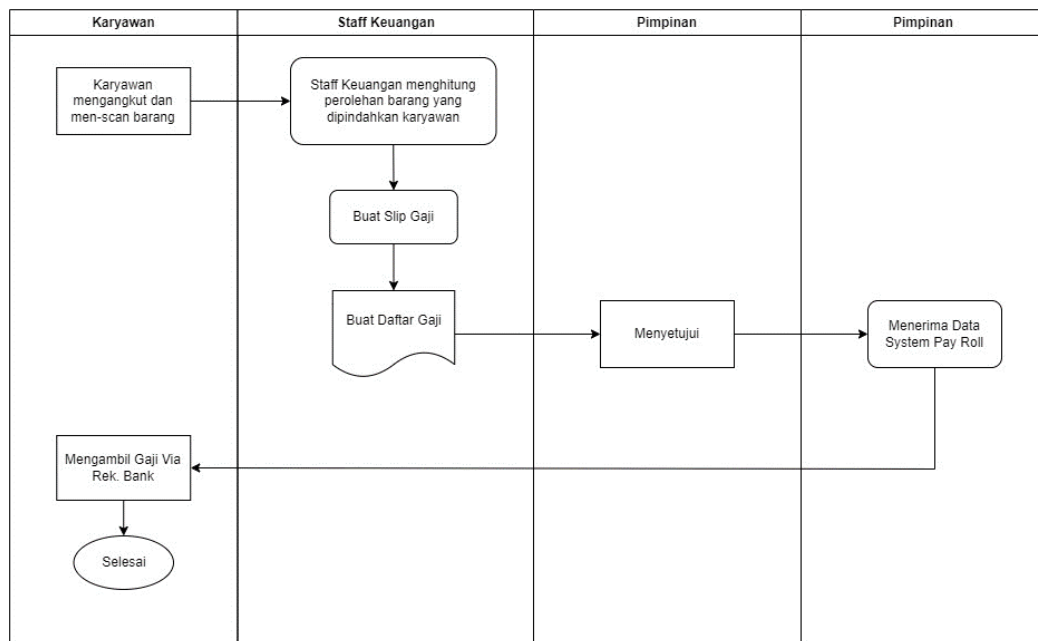
1. Data kehadiran seluruh karyawan yang sudah berupa rekapan per hari pada tanggal *cut off* sudah diperiksa ulang tentang kebenarannya dan siap dijadikan data penggajian.
2. Departemen HRD (bagian *payroll*) menerima data kehadiran yang sudah valid untuk diproses penggajiannya per orang.

3. Departemen HRD (bagian pajak pph 21) menghitung atau mengkoreksi pajak baik yang gajinya ada kenaikan atau ada perubahan status keluarga (tambah anak atau ada perubahan dari bujang menjadi kawin, dan lain-lain).
4. Departemen HRD (*bagian payroll*) setelah menerima revisi perhitungan pajak gaji dari bagian pajak, membuat slip gaji dan daftar gaji seluruh karyawan untuk dikoreksi dan dimintakan tanda tangan manager HRD.
5. Departemen keuangan menerima daftar gaji dan slip gaji karyawan departemen HRD untuk dikoreksi secara menyeluruh perhitungan gaji masing-masing karyawan maupun perhitungan pajak gajinya.
6. Apabila departemen keuangan menemukan ada kesalahan hitung atau salah ketik, harus segera mengembalikannya ke departemen HRD, atau cancel.
7. Apabila departemen keuangan evaluasinya tidak menemukan kesalahan pada daftar gaji atau slip gaji tersebut, maka wajib menandatangani dan membuat cek tunai sebesar jumlah gaji seluruh karyawan dan menyerahkannya pada pimpinan perusahaan.
8. Pimpinan perusahaan menerima dan menandatangani daftar gaji seluruh karyawan cek tunai untuk transfer gaji dibank yang ditunjuk.
9. Bank yang ditunjuk menerima daftar gaji dan cek transfer ke rekening pribadi masing-masing karyawan pada tanggal yang telah ditentukan.

10. Karyawan pada tanggal penggajian yang telah ditentukan, mengambil gajinya melalui ATM yang telah ditunjuk, dengan rentang waktu selama 24 jam per hari.

11. Selesai.

Gaji merupakan pembayaran atas penyerahan jasa yang dilakukan oleh para karyawan yang mempunyai jenjang jabatan manager, dan dibayarkan secara tetap per bulan. Sedangkan upah merupakan pembayaran atas penyerahan jasa yang dilakukan oleh karyawan pelaksana (buruh) umumnya dibayarkan berdasarkan hari kerja, jam kerja, atau jumlah satuan produk yang dihasilkan oleh karyawan (Pertwi, 2022). Berbeda dengan penggajian karyawan harian, penggajian karyawan borongan dihitung berdasarkan jumlah yang dikerjakan oleh karyawan tersebut dalam satu waktu dan dijumlahkan secara keseluruhan ketika waktunya penggajian. Semakin banyak pekerjaan yang diselesaikan oleh karyawan borongan, semakin banyak pula gaji yang akan didapatkan nanti. Berikut *Flowchart* penggajian karyawan borongan (Hendrawan M dkk, 2019).



Gambar 2.8 Flowchart Penggajian Karyawan Borongan menurut Hendrawan M dkk, 2019.

2.9.6 Metode SDLC

Model SDLC air terjun (*Waterfall*) atau sekuensial linier (*sequential linear*) atau alur hidup klasik (*classic life cycle*). SDLC atau *Software Development Life Cycle* atau sering disebut juga *System Development Life Cycle* adalah proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem-sistem perangkat lunak sebelumnya, berdasarkan *best practice* atau cara-cara yang sudah teruji baik (Firmansyah & Udi, 2017). Model ini menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau berurutan dimulai dari analisis, desain, pengkodean, pengujian, dan tahap pendukung (*support*). Metode *Waterfall* merupakan suatu metode dalam pengembangan *software* dimana pengerjaannya harus dilakukan secara berurutan

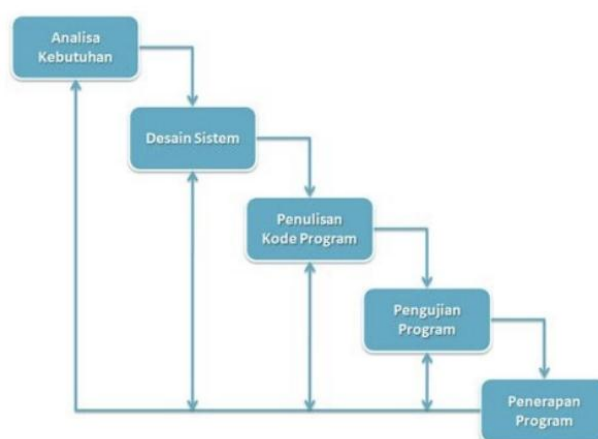
dari awal hingga akhir (Susanto dkk, 2021). SDLC memiliki alur cara kerja yang terdiri dari tahap-tahap: rencana (*planning*), analisis (*analysis*), desain (*design*), implementasi (*implementation*), uji coba (*testing*) dan pengelolaan (*maintenance*)(Pratama dkk, 2022).

Berikut adalah kelebihan dari metode *Waterfall*:

- 1) Memiliki proses yang terurut, sehingga pengerjaan dapat terjadwal dengan baik dan mudah.
- 2) Cocok untuk sistem dengan kompleksitas rendah (*predictable*).
- 3) Setiap proses yang dilakukan tidak dapat saling tumpang tindih.

Berikut adalah kekurangan dari metode *Waterfall*:

- 1) Waktu pengerjaan relatif lebih lama, karena harus menunggu tahap sebelumnya selesai.
- 2) Biaya yang dibutuhkan lebih mahal karena waktu pengembangan yang dibutuhkan lebih lama.
- 3) Model *Waterfall* ini kurang cocok untuk pengembangan proyek yang memiliki kompleksitas tinggi



Gambar 2.9 Metode SDLC *Waterfall* menurut Pratama dkk, 2022.

Berikut keterangan dari Metode SDLC *Waterfall*:

- 1) Analisa Kebutuhan, tahap analisis dilakukan untuk mengumpulkan data yang dibutuhkan dalam penelitian. Pada tahap ini bertujuan untuk memperoleh informasi mengenai harapan dari pengguna sistem atau aplikasi yang akan dikembangkan.
- 2) Desain Sistem, tahap desain dilakukan untuk membuat simulasi rancangan yang siap untuk diimplementasikan. Pada tahap ini akan dibuat rancangan sistem seperti arsitektur sistem.
- 3) Penulisan Kode Program, tahap pengkodean dilakukan pengembangan sistem informasi menggunakan bahasa pemrograman tertentu.
- 4) Pengujian Program, setelah pengkodean selesai maka dilakukan tahap pengujian terhadap sistem yang sudah dikembangkan.
- 5) Penerapan Program, setelah dilakukan pengetesan dirasakan sudah berhasil maka sistem informasi diterapkan pada pengguna untuk digunakan.