

BAB I

PENDAHULUAN

1.1 Latar Belakang

Penerapan teknologi informasi saat ini memiliki banyak manfaat salah satunya di instansi/perusahaan yang digunakan untuk menjalankan aktivitas-aktivitas penting, hampir seluruh kegiatan yang terjadi tidak lepas dengan teknologi informasi yang ada. Teknologi informasi menempati peranan utama dalam kehidupan masyarakat sekarang ini dan perkembangannya pun sangat pesat sekali, dan kita dapat mengolah dan mendapatkan informasi dengan cepat, tepat dan akurat. Saat ini sudah menjadi kebutuhan yang tidak dapat terelakkan bahwa sistem komputerisasi dapat memberi kemudahan dalam mencari informasi yang diinginkan. (Muhammad Al Khusnul Rizki et al., 2021)

Salah satu aspek dari perkembangan Sistem Informasi adalah pada aktifitas pengolahan data dan penyampaian informasi. Aktifitas yang semula dilakukan secara manual perlahan-lahan mulai beralih ke perangkat elektronik seperti komputer dan *handphone*. Sistem informasi yaitu suatu entitas yang saling berinteraksi dari beberapa jaringan dan berusaha untuk mencapai tujuan yang sama (Yusri et al., 2022)

Salah satu proses yang dapat mendorong suatu pelaksanaan fungsi dari manajemen suatu Lembaga atau organisasi yaitu keberadaan dan kelancaran kegiatan pegawai atau karyawan dalam melakukan suatu kegiatan di Lembaga atau organisasi tersebut. Sistem kepegawaian di suatu instansi berfungsi juga untuk mendukung pelaksanaan suatu manajemen dalam lingkup Lembaga atau organisasi. Memanfaatkan informasi yang disuguhkan oleh singkatnya waktu dalam pemrosesannya dan ketelitian dalam kebenaran informasi yang dikeluarkan atau di hasilkan. Hal berikut berada dalam penggunaan perangkat keras komputer (*hardware*), perangkat lunak (*software*), dan koneksi *internet* sebagai sarana untuk mengelola data informasi.

Terkait dengan keterangan di atas di Dinas Perhubungan jika proses untuk pengelolaan suatu data pegawai atau karyawan masih menggunakan sistem manual

pasti dalam suatu penyimpanan, pencarian, dan penyajian informasi untuk data pegawai atau karyawan tidak dapat diperoleh dengan cepat dan tepat. Seiring dalam perkembangan ilmu teknologi informasi dan komunikasi, penerapan informasi yang cepat dan tepat diperlukan untuk meningkatkan proses efektifitas kinerja dalam sebuah Lembaga atau perusahaan, banyak hal yang dapat dimanfaatkan oleh suatu instansi seperti dalam kegiatan administrasi yang berjalan lebih efektif dan efisien, serta lebih mudah untuk mencapai tujuan suatu sistem informasi berbasis *web* yang dapat mengolah dan menyuguhkan informasi kepegawaian secara cepat dan akurat.

Dengan dibuatnya sistem informasi berbasis *web* diharapkan mampu digunakan kalangan internal Dinas Perhubungan Kota Bandar Lampung dalam mengelola data pegawai dan diharapkan dengan dibuatnya aplikasi ini Dinas Perhubungan Kota Bandar Lampung dapat bekerja lebih efektif dan efisien agar mampu menunjang Lembaga ini menjadi lebih baik.

Berdasarkan latar belakang yang telah dijelaskan dan suatu masalah diatas, penulis akan merancang suatu sistem informasi kepegawaian yang mampu menjawab masalah yang dihadapi oleh Dinas Perhubungan Kota Bandar Lampung. Berdasarkan uraian diatas maka dirancang sebuah sistem informasi yang diajukan sebagai tugas akhir dengan judul “Perancangan Sistem Informasi Kepegawaian Berbasis *Web* Pada Dinas Perhubungan Kota Bandar Lampung”.

1.2 Tujuan

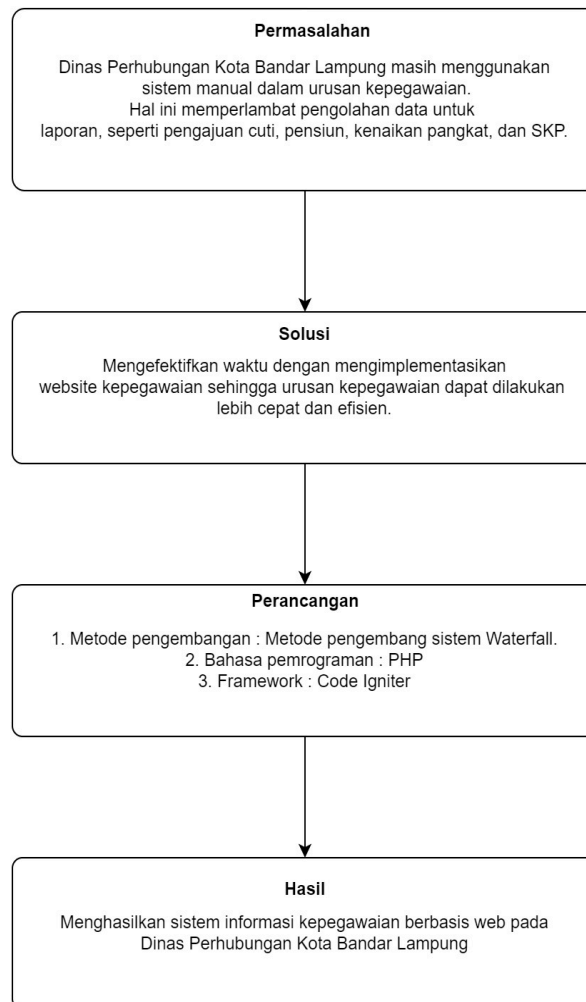
Berdasarkan masalah penjelasan diatas, tujuan tugas akhir ini yaitu :

- a. Membantu sistem kepegawaian yang ada di Dinas Perhubungan Kota Bandar Lampung
- b. Mempermudah dalam proses pengolahan data pegawai dengan mudah.
- c. Menghasilkan sistem informasi berbasis *online* yang mampu membantu kinerja pegawai.

1.3 Kerangka Pemikiran

Dalam era modern ini, kebutuhan akan efisiensi dan akurasi dalam pengelolaan data kepegawaian menjadi semakin mendesak, terutama bagi instansi pemerintah seperti Dinas Perhubungan Kota Bandar Lampung. Dengan *volume* data kepegawaian yang terus meningkat, risiko kesalahan dan kekurangan dalam pengelolaan manual menjadi hal yang sangat memungkinkan. Untuk mengatasi

tantangan ini, dirancanglah sebuah sistem informasi kepegawaian yang bertujuan untuk mengotomatisasi dan mempermudah proses pengelolaan data kepegawaian, termasuk pelaporan data pegawai, pengelolaan cuti, pemantauan pensiun pegawai, dan peningkatan pangkat. Penelitian ini menggunakan metode deskriptif dan tindakan (aksi) untuk memahami secara mendalam kebutuhan dan proses yang terlibat dalam manajemen kepegawaian, sementara metode pengembangan *Waterfall* menjadi landasan untuk tahapan pengembangan sistem. Melalui penerapan alat bantu seperti use case, diagram aktivitas, diagram kelas, *prototyping*, dan *coding*, diharapkan sistem informasi kepegawaian ini dapat menjadi solusi yang efektif dan andal bagi Dinas Perhubungan Kota Bandar Lampung dalam mengelola data kepegawaian mereka.



Gambar 1. Kerangka Pemikiran

1.4 Kontribusi

Pembuatan *website* kepegawaian bagi Dinas Perhubungan Kota Bandar Lampung dapat berkontribusi pada beberapa pihak yang terkait, di antaranya adalah:

- a. Dinas Perhubungan: Memudahkan pengelolaan data kepegawaian dan perencanaan sumber daya manusia.
- b. Pegawai Dinas Perhubungan: Mempermudah akses informasi terkait jadwal, cuti, dan pelatihan.

BAB II

TINJAUAN PUSTAKA

2.1 Sejarah Singkat Dinas Perhubungan Kota Bandar Lampung

Bentuk awal Departemen Perhubungan lahir dalam kancah perjuangan adalah gabungan antara Departemen Perhubungan dan Departemen Pekerjaan Umum, yang dipimpin oleh seorang Menteri Abikusno Tjokrosuyoso. Namun hal ini tidak berlangsung lama, karena Departemen Perhubungan dan Departemen Pekerjaan Umum tidak lagi dijabat oleh orang yang sama dengan merangkap tugas seperti sebelumnya. Urusan perhubungan dan pekerjaan umum kini berada di bawah dua pejabat yang berbeda yaitu Kementerian Perhubungan dipimpin oleh Ir. Abdulkarim dan Kementerian Pekerjaan Umum di bawah pimpinan Ir. Putuhena. Sesuai dengan nama yang disandangnya, Departemen Perhubungan mengurus masalah perhubungan.

Keinginan Belanda untuk berkuasa System5 di Indonesia sangat jelas terlihat ketika mereka melancarkan agresi militernya yang kedua pada tanggal 19 Desember 1948. Dalam agresinya tersebut, Belanda berhasil menguasai Yogyakarta dan menangkap Presiden Soekarno dan Wakil Presiden Hatta. Dalam kondisi darurat ini, Dinas Telegrap sebagai salah satu Jawatan dalam Departemen Perhubungan berhasil menjalankan tugasnya yang sangat berdampak penting bagi kelangsungan tegaknya Indonesia saat itu. Dinas Telegrap berhasil mengirim berita terakhir ke Bukittinggi yang ditujukan kepada Mr. Sjafruddin Prawiranegara dari Presiden Soekarno yang isinya memberi wewenang untuk membentuk suatu pemerintahan darurat. Selanjutnya dibentuklah Kabinet Darurat dengan Mr. Sjafruddin Prawiranegara sebagai Perdana Menteri dan Ir. Indratjaja sebagai Menteri Perhubungan dan merangkap sebagai Menteri Kemakmuran. Sejak awal kemerdekaan hingga pengakuan kedaulatan Belanda atas RIS tahun 1949, Departemen Perhubungan memiliki wewenang untuk mengatur perhubungan laut, udara, darat, perkeretaapian serta pos, telegraf, dan telekomunikasi dan masing-masing sektor tersebut diurus oleh jawatan-jawatannya sendiri yang berada di bawah struktur organisasi Departemen Perhubungan.

Dinas Perhubungan Kota Bandar Lampung merupakan unsur pelaksanaan otonomi daerah yang melaksanakan urusan pemerintah daerah yang dipimpin oleh seorang Kepala Dinas yang berkedudukan dibawah dan bertanggung jawab kepada walikota melalui sekretaris daerah. Dinas perhubungan Kota Bandar Lampung mempunyai tugas pokok melaksanakan urusan pemerintahan daerah dibidang perhubungan darat, perhubungan laut, pos dan telekomunikasi berdasarkan asas otonomi dan tugas pembantuan. Dalam melaksanakan tugas pokok Dinas Perhubungan Kota Bandar Lampung mempunyai fungsi sebagai berikut :

- a. Perumusan kebijakan teknis dibidang perhubungan darat, perhubungan laut, pos dan telekomunikasi
- b. Penyelenggaraan urusan pemerintah dan layanan umum sesuai dengan lingkup tugasnya
- c. Pembinaan dan pelaksanaan tugas sesuai dengan lingkup tugasnya.
- d. Pelaksanaan tugas lain yang diperintahkan oleh walikota sesuai dengan tugas dan fungsinya

2.2 Struktur Organisasi

Berdasarkan Peraturan Walikota Bandar Lampung Nomor 08 Tahun 2009 tentang Tugas, Fungsi Dan Tata Kerja Dinas Perhubungan Kota Bandar Lampung, maka Dinas Perhubungan Kota Bandar Lampung mempunyai tugas pokok melaksanakan urusan pemerintahan daerah dibidang perhubungan darat, perhubungan laut, pos dan telekomunikasi berdasarkan asas otonomi. Adapun berdasarkan pasal 5 Peraturan Walikota nomor 08 Tahun 2009 tentang susunan organisasi Dinas Perhubungan Kota Bandar Lampung yaitu terdiri dari:

- a) Kepala Dinas
- b) Sekretariat, membawahi ;
 1. Sub bagian penyusunan program, monitoring dan evaluasi
 2. Sub bagian umum dan kepegawaian
 3. Sub bagian keuangan
- c) Bidang lalu lintas jalan, membawahi :
 1. Seksi manajemen dan rekayasa lalu lintas jalan
 2. Seksi keselamatan lalu lintas jalan
 3. Seksi pengendalian dan operasional lalu lintas jalan

- d) Bidang Angkutan Jalan, membawahi ;
 1. Seksi angkutan orang
 2. Seksi angkutan barang
 3. Seksi angkutan khusus
- e) Bidang Teknik, membawahi ;
 1. Seksi system sarana
 2. Seksi system prasarana
 3. Seksi karoseri dan perbengkelan
- f) Bidang Perhubungan Laut, membawahi ;
 1. Seksi angkutan laut
 2. Seksi system laut
 3. Seksi keselamatan pelayaran
- g) Unit Pelaksana Teknis
- h) Kelompok Jabatan Fungsional

Berikut adalah struktur organisasi Dinas Perhubungan kota Bandar Lampung tersaji pada Gambar 2.



Gambar 2. Struktur Organisasi

2.3 Pengertian Sistem Informasi

Menurut (Mustakini, 2009), bahwa system dapat didefinisikan dengan pendekatan prosedur dan dengan pendekatan komponen. “Dengan pendekatan

prosedur, system dapat didefinisikan sebagai system dari prosedur-prosedur yang mempunyai tujuan tertentu”.

Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Kejadian-kejadian adalah sesuatu yang terjadi pada saat yang tertentu. Di dalam dunia bisnis, kejadian-kejadian nyata yang sering terjadi adalah perubahan dari suatu nilai yang disebut dengan transaksi.

Informasi dapat didefinisikan sebagai data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya. Sumber dari informasi adalah data. Informasi dapat diperoleh dari system informasi (*information systems*) atau disebut juga dengan *processing systems* atau *processing systems* atau *information-generating systems*.

Sistem Informasi adalah suatu system 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 diperlukan.

2.4 Model Pendekatan Sistem

Menurut (Sommerville, 2011) menyimpulkan bahwa Pendekatan sistematis yang digunakan dalam rekayasa piranti lunak biasa juga disebut dengan proses perangkat lunak. Proses perangkat lunak adalah sebuah aktivitas terurut yang menuju kepada produksi dari pembuatan produk rekayasa piranti lunak. Ada 4 aktivitas umum yang mendasar pada semua proses rekayasa piranti lunak. Aktivitas ini adalah sebagai berikut:

- a. *Software specification*, system pengguna dan perekayasa menentukan perangkat lunak yang akan dibuat dan dibatasi pada proyek tersebut.
- b. *Software development*, system perangkat lunak tersebut dirancang dan *deprogram*.
- c. *Software validation*, system perangkat lunak di cek apakah sudah memenuhi apa yang dibutuhkan oleh pengguna.
- d. *Software evolution*, system perangkat lunak diubah, diperbaiki untuk mengatasi perubahan pengguna dan mengikuti perkembangan zaman.

2.5 Model Pengembangan Sistem

Menurut (A. S. & Shalahuddin, 2013) “Model SDLC air terjun (*waterfall*) sering juga disebut model sekuensial linier atau alur hidup klasik”. Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara terurut (sekuensial) dimulai dari *system*, desain, pengkodean, pengujian dan tahap pendukung. Penjelasan dari tahap-tahap *waterfall model* adalah sebagai berikut:

2.5.1. Analisis Kebutuhan Perangkat Lunak

Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk di dokumentasikan.

2.5.2. Design

Desain perangkat lunak adalah proses multilangkah yang *system* pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengkodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari *system* analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu di dokumentasikan.

2.5.3. Pembuatan Kode Program

Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program *system* sesuai dengan desain yang telah dibuat pada tahap desain.

2.5.4. Pengujian (*Testing*)

Pengujian *system* pada perangkat lunak dari segi logika dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

2.6 UML (*Unified Modelling Language*)

Menurut (Grace Gata, 2013), UML adalah *system* spesifikasi standar yang dipergunakan untuk mendokumentasikan. Menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi dalam mengembangkan *system* berorientasi objek dan juga merupakan alat untuk mendukung pengembangan

10ystem. Berdasarkan pendapat yang dikemukakan diatas dapat disimpulkan bahwa, *Unified Modelling Language* (UML) adalah sebuah 10ystem standar *industry* untuk memvisualisasikan, menspesifikasikan, membangun dan pendokumentasian dari sebuah 10ystem pengembangan perangkat lunak berbasis objek (*Object Oriented Programming*)”. Definisi diagram-diagram UML (*Unified Modeling Language*)


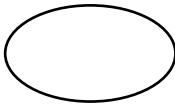

Berikut ini adalah definisi mengenai 6 diagram UML yaitu:

2.6.1. Use Case Diagram


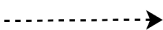


Menurut (Grace Gata, 2013) menyatakan bahwa “*Use case diagram* merupakan pemodelan untuk kelakuan (*behavior*) 10ystem informasi yang akan dibuat. *Use Case* mendeskripsikan sebuah interaksi antara satu atau lebih 10yste dengan *system* informasi yang akan dibuat”.

Use Case Diagram bersifat statis, diagram ini memperlihatkan himpunan use case dan 10yste-aktor (suatu jenis khusus dari kelas). Diagram ini sangat penting untuk mengorganisasi dan memodelkan perilaku dari suatu 10ystem yang dibutuhkan serta diharapkan pengguna (Nugroho, 2005). Simbol-simbol yang digunakan dalam *Use Case Diagram* dijelaskan pada Tabel 1.

Tabel 1. *Use Case Diagram*

Simbol	Nama	Keterangan
(1)	(2)	(3)
	<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i>
	<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan 10ystem yang menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i>
	<i>System</i>	Menspesifikasikan paket tampilan di 10ystem secara terbatas

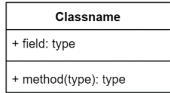
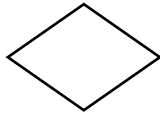


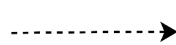
Tabel 1 (Lanjutan)

Simbol	Nama	Keterangan
(1)	(2)	(3)
	<i>Generalization</i>	Hubungan dimana objek anak (<i>Descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>Ancestor</i>)
	<i>Clude</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit
	<i>Tend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan
	Association	Menghuungkan antara objek satu dengan objek lainnya

2.6.2. *Class Diagram*

Menurut (Grace Gata, 2013) menyatakan bahwa " *Class diagram* merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem ". *Class* adalah sebuah spesifikasi yang akan menghasilkan objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut / properti) suatu sistem. *Class diagram* menggambarkan struktur dan deskripsi kelas, *package* beserta hubungan satu sama lai (Dharwiyanti & Romi Satria Wahono, 2003) n. Simbol-simbol yang digunakan dalam *class diagram* yaitu pada Tabel 2.

Tabel 2. *Class Diagram*




Simbol	Nama	Keterangan
(1)	(2)	(3)
	<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama
	<i>N-ary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek
	<i>Association</i>	Hubungan statis antar <i>class</i> yang menggambarkan <i>class</i> yang memiliki atribut berupa <i>class</i> lain atau <i>class</i> yang harus mengetahui eksistensi <i>class</i> lain
	<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>)
	<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya

2.6.3. *Sequence Diagram*

Menurut (Grace Gata, 2013) menyatakan bahwa "*Sequence diagram* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek". *Sequence diagram* menggambarkan interaksi antara objek di dalam dan disekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu, menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respon dari sebuah event untuk menghasilkan output tertentu. *Collaboration* diagram juga menggambarkan interaksi antara objek seperti

sequence diagram, akan tetapi lebih menekankan pada masing-masing objek dan bukan pada waktu penyampaian *message*. Setiap *message* memiliki *sequence number*, dimana *message* dari level tertinggi memiliki nomor 1 (Dharwiyanti & Romi Satria Wahono, 2003). Simbol-simbol yang digunakan dalam *sequence diagram* dijelaskan pada Tabel 3.

Tabel 3. *Sequence Diagram*


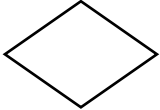



Simbol	Nama	keterangan
(1)	(2)	(3)
	<i>Object</i> dan <i>lifeline</i>	Orang, tempat, benda, kejadian atau konsep yang ada dalam dunia nyata yang penting bagi suatu aplikasi yang saling berinteraksi
	<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktivitas yang terjadi
	<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i>

2.6.4. *Activity Diagram*

Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan oleh aktor, jadi aktivitas yang dapat dilakukan oleh sistem. Menurut (Booch et al., 2005) berpendapat bahwa, "*An activity diagram is essentially a flowchart, showing flow of control from activity to activity*", *activity diagram* secara esensial mirip dengan *flowchart* atau diagram alur yang menunjukkan aliran kendali dari sebuah aktivitas ke aktivitas lainnya. Dalam *activity diagram* terdapat aksi atau aktivitas, *activity*

nodes, flows atau aliran, dan objek. Simbol-simbol yang dipakai dalam *activity diagram* pada Tabel 4.

Tabel 4. *Activity Diagram*

Simbol	Nama	Keterangan
(1)	(2)	(3)
	<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
	<i>Decision</i>	pilihan untuk pengambilan keputusan
	<i>Initial Node</i>	Titik awal
	<i>Activity Final Node</i>	Titik akhir
	<i>Fork</i>	Menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu

2.7 *Frame Work CodeIgniter*

CodeIgniter adalah sebuah *Framework* PHP yang dapat membantu mempercepat *developer* dalam pengembangan aplikasi *web* berbasis PHP dibandingkan jika menulis semua kode program dari awal. *CodeIgniter* menyediakan banyak *library* untuk mengerjakan tugas-tugas yang umumnya ada pada sebuah aplikasi berbasis *web*. Selain itu, struktur dan susunan logis dari *CodeIgniter* membuat aplikasi yang dibuat menjadi semakin teratur dan rapi. Dengan demikian, dapat fokus pada fitur-fitur apa yang dibutuhkan aplikasi dengan membuat kode program seminimal mungkin. Alasan penggunaan *CodeIgniter* *Framework* merupakan *Framework* PHP *open source* untuk pengembang *web* yang mendukung pola MVC. Objek utama dari *Framework* ini untuk menghadirkan cara yang simpel dan efisien untuk menyelesaikan proyek pengembangan *web*. Menjadikan proyek pengembangan *web* lebih mudah dan cepat.

Ada beberapa kelebihan *CodeIgniter* (CI) dibandingkan dengan *Framework* PHP lain. Salah satu kelebihan *CodeIgniter* (CI) yaitu berukuran kecil, ukuran *CodeIgniter* yang kecil merupakan keunggulan tersendiri. Dibanding *Framework* lainnya yang berukuran besar, serta membutuhkan resource yang besar pula untuk berjalan. Pada *CodeIgniter*, bisa diatur agar sistem memproses *library* yang dibutuhkan saja, sehingga dapat berjalan ringan dan cepat.

2.8 Konsep Dasar Website

Merurut (Kustiyahningsih & Anamisa, 2011) menyatakan bahwa "pada dasarnya *website* dapat didefinisikan sebagai menampilkan informasi apapun yang ada dibelahan dunia tidak peduli seberapa jauhnya. Dengan *web*, seseorang bisa mendapatkan informasi yang dibutuhkan dengan mudah, cepat dan murah". *Web* juga menjadi sarana yang disukai karena sifatnya yang mendunia (*world wide web*) dan mejadi ajang tukar pendapat bagi penggunanya.

2.8.1. Internet

Menurut (Kustiyahningsih & Anamisa, 2011) memberikan batasan bahwa "*internet* adalah sebuah jaringan komputer dunia, semua berbicara dengan bahasa yang sama. Banyak keuntungan yang didapatkan dijaringan komputer, diantaranya produktivitas dan efisien". Jaringan komputer menurut area atau lokasi dapat dibagi menjadi empat yaitu:

- a. *Local Area Network* (LAN), yaitu jaringan komputer dimana komputer komputer yang terhubung masih dalam satu area.
- b. *Wide Area Nerwork* (WAN), yaitu koneksi antara LAN-LAN yang berbeda lokasi atau area. Ciri utamanya adalah memiliki *bandwidth* yang terbatas karena disesuaikan dengan fungsi harga, adanya *problem delay* antar stasiun.
- c. *Metropolitan Area Nerwork* (MAN), yaitu sama seperti LAN hanya saja lebih luas areanya semisal dalam satu kota atau daerah dengan mecapai 50 km.
- d. *Internet*, yaitu kepanjangan dari *interconnection networking* atau juga yang telah menjadi *intermational networking* merupakan suatu jaringan yang menghubungkan komputer diseluruh dunia tanpa dibatasi oleh jumlah unit menjadi satu jaringan yang bisa saling mengakses.

Dengan *internet* tersebut, satu komputer dapat berkomunikasi secara langsung dengan komputer lain diberbagai belahan dunia. Selain itu masih ada satu istilah lagi yang tentunya tidak asing lagi "*intranet*", secara fisik *intranet* adalah gabungan dari LAN dan *internet* dimana dalam satu LAN disediakan fasilitas-fasilitas seperti di *internet* dan tentunya terhubung langsung ke *internet*.

2.8.2. URL dan HTTP

Menurut (Kustiyahningsih & Anamisa, 2011) menyatakan bahwa pada dasarnya URL dapat didefinisikan "sebagai konsep nama *file* standar yang diperluas dengan jaringan untuk menentukan lokasi informasi pada *web server*. Nama file ini tidak hanya menunjukkan direktori dan nama *file*-nya, tetapi juga nama mesinnya dalam jaringan". URL (*Universal Resource Locator*) dapat disediakan (ada atau diakses) dengan berbagai metode, dan bisa jadi bukan sekedar *file* karena URL dapat menunjukkan *query-query*, dokumen yang disimpan didalam database, hasil dari suatu perintah *finger* atau *archive* atau apapun yang berkaitan dengan data hasil proses.

2.8.3. Web Browser

Menurut (Kustiyahningsih & Anamisa, 2011) memberikan batasan bahwa "*Software* yang digunakan untuk menampilkan informasi dari *web server*. *Software* ini telah dikembangkan dengan menggunakan *graphical user interface*, sehingga pemakai dapat dengan mudah melakukan *point click* untuk pindah antar dokumen.

2.8.4. Web Server

Menurut (Kustiyahningsih & Anamisa, 2011) meberikan batasan bahwa: Komputer yang digunakan untuk menyimpan dokumen-dokumen *web*, komputer ini akan melayani permintaan dokumen *web* dari kliennya". *Web browser* seperti *explorer* atau *navigator* berkomunikasi melalui jaringan (termasuk jaringan internet) dengan *web server* menggunakan *HTTP*, Contoh *web server* adalah:

a. Apache

Merupakan *web server* yang paling populer dan memiliki ranking pertama dalam persentase penggunaannya. *Apache* bisa digunakan diberbagai platform OS (*operating system*). Contoh: *Linux*, *Windows*, dan lain-lain.

b. IIS (*Internet Information Services*).

Digunakan di sistem operasi *windows* NT dan *windows* 2002.

c. PWS (*Personal Web Server*).

Digunakan di sistem operasi *windows* 9x.

d. HTML (*Hypertext Markup Language*).

Menurut (Kustiyahningsih & Anamisa, 2011) menyatakan bahwa "pada dasarnya HTML dapat didefinisikan sebagai *file text* murni yang dapat dibuat dengan editor teks sembarang, dokumen ini dikenal sebagai *web page*". *File-file* HTML ini berisi instruksi-instruksi yang kemudian diterjemahkan oleh browser yang ada di komputer client (*user*) sehingga isi informasinya dapat ditampilkan secara visual di komputer pengguna (*user*). HTML dikenal sebagai standard bahasa yang digunakan untuk menampilkan dokumen *web*.

e. CSS (*Cascading Style Sheet*).

Menurut b menyatakan bahwa "Pada dasarnya CSS dapat didefinisikan sebagai kumpulan kode-kode yang berurutan dan saling berhubungan untuk mengatur format atau tampilan suatu halaman HTML".

f. PHP (*Hypertext Preprocessor*).

Menurut (Saputra, 2011) menyatakan bahwa "Pada dasarnya PHP dapat didefinisikan sebagai suatu bahasa pemrograman yang difungsikan untuk membangun suatu *website* dinamis". PHP menyatu dengan kode HTML maksudnya adalah beda kondisi. HTML digunakan sebagai pembangun atau pondasi dari kerangka *web layout*, sedangkan PHP difungsikan sebagai prosesnya sehingga dengan adanya PHP tersebut, sebuah *web* akan sangat mudah di *maintenance*.

Menurut karangan (Arief, 2011) PHP (*Perl Hypertext Preprocessor*) adalah bahasa *server-side scripting* yang menyatu dengan HTML untuk membuat halaman *web* yang dinamis. Karena PHP merupakan *server-side scripting* maka sintaks dan perintah-perintah PHP akan dieksekusi di server kemudian hasilnya dikirimkan ke browser dalam format HTML. dengan demikian kode program yang ditulis dalam PHP tidak akan terlihat oleh user sehingga keamanan halaman *web* yang dinamis, yaitu

halaman *web* yang dapat membentuk suatu tampilan berdasarkan permintaan terkini, seperti menampilkan isi basis data kehalaman *web*. PHP dapat dibangun sebagai modul pada *web server Apache* dan sebagai binary yang dapat berjalan sebagai CGI (*Common Gateway Interface*). PHP termasuk dalam *Open Source Product*, sehingga *source code* PHP dapat diubah dan di distribusikan secara bebas. PHP juga mampu lintas Platform. Artinya PHP dapat berjalan dibanyak sistem operasi yang beredar saat ini, diantaranya Sistem Operasi *Microsoft Windows* (semua versi), *Linux*, *Mac OS*, *Solaris*.

2.8.5. Sublime text 3

Sublime Text 3 adalah sebuah software yang dikembangkan oleh Jon Skinner. Beliau merupakan seorang *programmer* dari Australia. *Sublime Text 3* merupakan aplikasi *text editor* untuk menulis kode. Banyak sejumlah bahasa program yang ada pada aplikasi ini. Diantaranya PHP, CSS, C, C++, HTML, ASP, Java, dan sebagainya. Tentu saja ini bisa lebih memudahkan pekerjaan pengguna saat membuat sebuah program. Langkah-langkah menggunakan *Sublime Text 3* sebagai teks editornya, di bawah ini terdapat tutorial tentang membuat *File PHP* di *Sublime Text 3*.

2.8.6. MySQL

MySQL merupakan *software* sistem manajemen *database (Database Management System –DBMS)* yang paling populer dikalangan pemrograman *web*, terutama di lingkungan *Linux* dengan menggunakan *script PHP* dan *Perl* yang digunakan untuk membangun aplikasi *web* yang menggunakan *database* sebagai sumber dan pengelola datanya. MySQL dan PHP dianggap sebagai pasangan *software* pengembangan aplikasi *web* yang ideal dan sering digunakan untuk membangun aplikasi berbasis *web*, umumnya pengembangan aplikasinya menggunakan bahasa pemrograman *script PHP*. MySQL juga merupakan *database* yang digunakan oleh situs-situs terkemuka di *internet* untuk menyimpan datanya (Komala, 2011).

MySQL dikembangkan oleh sebuah perusahaan Swedia bernama MYSQLAB yang pada saat itu bernama TcX Data Konsult AB sekitar tahun 1994-1995, namun cikal bakal kodenya sudah ada sejak 1979. Awalnya TcX membuat

MySQL dengan tujuan mengembangkan aplikasi *web* untuk *klien*. Kepopuleran MySQL antara lain karena MySQL menggunakan SQL sebagai bahasa dasar untuk mengakses databasenya sehingga mudah untuk digunakan, kinerja *query* cepat, dan mencukupi untuk kebutuhan database perusahaan-perusahaan skala menengah kecil.

Keandalan suatu *system database* (DBMS) dapat diketahui dari cara kerja *optimizer*-nya dalam melakukan proses perintah-perintah SQL, yang dibuat oleh user maupun program-program aplikasinya. Sebagai *database server*, MySQL dapat dikatakan lebih unggul dibandingkan dengan *database server* yang lainnya dalam *query data*.

2.8.7. Penelitian Terkait

(Miftah et al., 2014) Sistem Informasi Kepegawaian dan Gaji MTSN 2 Simo Kabupaten Boyolali". Aplikasi Kepegawaian dan Penggajian ini dibuat dengan tujuan agar dapat meminimalisasi kesalahan yang salah satunya dapat disebabkan karena *human error* dari menghemat waktu dalam pendataan pegawai serta pengolahan gaji pegawai.

Sistem Informasi Kepegawaian dan Penggajian merupakan fungsi penting yang menjadi tanggung jawab Manajemen Sumber Daya Manusia. Fungsi utamanya adalah memberikan kompensasi untuk pegawai berupa gaji sebagai ganti kontribusi mereka terhadap organisasi/instansi. Informasi Pegawai dan Penggajian merupakan salah satu proses dalam organisasi yang rentan terhadap masalah.

Pengolahan data yang lambat dapat mengakibatkan lambatnya penyajian informasi kepegawaian sehingga dimungkinkan pula terjadi keterlambatan pembayaran gaji. Ditambah lagi jika terjadi kesalahan perhitungan dapat menjadikan informasi menjadi tidak akurat. Hal ini menjadikan sistem penggajian perlu didukung dengan sistem informasi yang baik. Penelitian ini dalam rangka melakukan pengembangan sistem terhadap sistem pengolahan data penggajian pada MTsN 2 Simo Kabupaten Boyolali meliputi: analisis sistem, desain sistem serta implementasi sistem. Hasil dari penelitian ini adalah dengan diimplementasikan Sistem Informasi Kepegawaian dan Gaji MTsN 2 Simo Kab. Boyolali dapat meningkatkan efektifitas, kemudahan dan kecepatan dalam proses pengelolaan dan pelaporan data sub bagian kepegawaian dan keuangan MTsN 2 Simo Kab. Boyolali.

(Purnamasari, 2017) Pada Unit Pelaksana Teknis Taman Kanak-Kanak dan Sekolah Dasar Kecamatan Pringkuku, data kepegawaian merupakan sarana penting. Informasi - Informasi didalamnya mulai dari pangkat atau golongan, status, penggajian dan lain - lainnya harus dikelola dengan tepat. Pada saat ini, prosedur yang diterapkan pada pengolahan data pegawai dan penggajian pada Unit Pelaksana Teknis Taman Kanak-Kanak dan Sekolah Dasar Kecamatan Pringkuku masih menggunakan metode konvensional. Metode ini membutuhkan waktu yang lama dalam pengolahan data pegawai dan penggajian. Dengan adanya hal tersebut, Unit Pelaksana Teknis Taman Kanak-Kanak dan Sekolah Dasar Kecamatan Pringkuku dirasa perlu untuk merubah metode pengelolaan data pegawai yang mereka gunakan saat ini, yaitu metode konvensional menjadi metode terkomputerisasi dan otomatis. Adapun metode penelitian yang digunakan untuk menyelesaikan berbagai permasalahan yang terjadi adalah pustaka, observasi, wawancara, analisis data dan *system*, perancangan sistem, pembuatan program, pengujian program dan implementasi program. Hasil yang diharapkan dari penelitian ini adalah menghasilkan Sistem Informasi yang dapat mengolah data pegawai dan penggajian yang efektif dan efisien.

(Ristian et al., 2012) Pada saat dilakukannya penelitian ini, sistem informasi kepegawaian pada Balai Besar Teknologi Energi (B2TE) Balai Pengkajian Penerapan Teknologi (BPPT) Serpong dilakukan secara semi manual menggunakan aplikasi *Microsoft Access*, *Microsoft Excel* dan *Microsoft Word*. Penelitian ini berusaha merancang dan membangun Sistem Informasi Kepegawajan Berbasis *Web* dengan menggunakan pendekatan terstruktur, yaitu metode SDLC (*System Development Life Cycle*).

Sistem informasi ini terdiri dari komponen input data pegawai, semua resumeportofolio yang dimiliki oleh pegawai, permintaan cuti, data kehadiran pegawai data proposal pelatihan, pegawai pensiunan, promosi dan mutasi pegawai Berdasarkan hasil pengujian, dapat disimpulkan bahwa *system* informasi telah berjalan sesuai tujuan pembuatannya.