

BAB I. PENDAHULUAN

1.1 Latar Belakang

Lembaga Penyelenggara Pemilu yang bersifat nasional, tetap, dan mandiri yang bertugas melaksanakan Pemilu yang disebut Komisi Pemilihan Umum atau sering disingkat KPU. KPU Provinsi adalah Penyelenggara Pemilu di Provinsi dan KPU Kabupaten/Kota. saat menyelenggarakan Pemilu, KPU menjalankan tugasnya secara berkesinambungan dan juga bebas dari pengaruh dari pihak manapun yang berkaitan dengan pelaksanaan tugas dan wewenangnya (Tentang KPU – Komisi Pemilihan Umum Kota Batu, 2022).

KPU Kota Bandar Lampung adalah salah satu dari Lembaga KPU Kabupaten/Kota yang terletak di Kota Bandar Lampung dengan beralamatkan Jl. Pulau Sebesi No.90, Sukarame, Kecamatan Sukarame, Kota Bandar Lampung, Provinsi Lampung. KPU Kota Bandar Lampung memiliki empat divisi yaitu (1) Divisi Keuangan, Umum dan Logistik; (2) Divisi Teknis Penyelenggaraan Pemilu, Partisipasi dan Hubungan Masyarakat; (3) Divisi Perencanaan, Data dan Informasi; serta (4) Divisi Hukum dan Sumber Daya Manusia. Setiap divisi memiliki program kerja yang sesuai dengan tugasnya masing-masing seperti pada Divisi Hukum mempunyai tugas yaitu melakukan analisis dan penyiapan, pelaksanaan penyusunan dan pengkajian produk hukum, dokumentasi informasi hukum, pemberian advokasi dan pendapat hukum, fasilitasi penyelesaian sengketa Pemilu dan Pemilihan, pengelolaan sumber daya manusia di lingkungan KPU Kabupaten/Kota, serta penghitungan tunjangan kinerja pegawai (PKPU No. 14, 2020).

Pegawai diwajibkan melakukan absensi kehadiran pada sistem absensi *fingerprnt* yang diberikan oleh KPU Provinsi Lampung dengan cara menempelkan jari ke alat *fingerprnt*. Pegawai melakukan absensi masuk pada pukul 07.30 WIB dan absensi kepulangan pada pukul 16.00 WIB, serta toleransi keterlambatan sampai dengan pukul 08.29 WIB. berdasarkan ketentuan tersebut pegawai yang terlambat wajib mengganti waktu keterlambatan masuk kerja guna meningkatkan kedisiplinan dan juga sebagai sistem untuk pencatatan data yang baik dan detail yang terintegrasi. Absensi mempermudah perhitungan absen pegawai untuk

menentukan tunjangan kinerja sesuai dengan data absen yang telah tercatat pada sistem.

Tunjangan Kinerja diberikan dengan ketentuan berdasarkan Kelas Jabatan yang telah ditetapkan dengan Keputusan Sekretaris Jenderal KPU. Pemotongan tunjangan kinerja diberlakukan jika pegawai tidak masuk kerja, terlambat masuk kerja, pulang sebelum waktunya, tidak mengganti keterlambatan, tidak mengisi daftar hadir, tidak mengisi laporan kinerja elektronik/manual. Pegawai yang dijatuhi hukuman disiplin dan pegawai yang berada ditempat tugas kurang dari 7 jam 30 menit dalam sehari. Tunjangan kinerja merupakan sekumpulan prosedur yang memungkinkan perusahaan untuk menarik, menahan, memotivasi pegawai serta untuk mengendalikan biaya pembayaran gaji. Pemberian tunjangan kinerja disesuaikan dengan kebijakan tiap-tiap organisasi, dan merupakan kebijakan yang dianggap adil (KPT SJ No. 326, 2022).

Sistem absensi saat ini pada Kantor KPU Kota Bandar Lampung masih menggunakan sistem absensi yang diberikan oleh Kantor KPU Provinsi Lampung berupa sistem absensi *fingerprnt*. Data absen pegawai yang masuk akan tersimpan pada *database* Kantor KPU Provinsi Lampung namun KPU Kota Bandar Lampung tidak memiliki akses untuk mengolah data absensi tersebut, lalu data tersebut diterima dan diunduh oleh pegawai KPU Kota Bandar Lampung berupa *file* PDF. Kemudian penghitungan tunjangan kinerja pegawai dilakukan secara terpisah menggunakan *Microsoft Excel*, dengan cara memasukan data absensi pegawai satu persatu yang telah diterima untuk dihitung berdasarkan jam kerja dari absen kehadiran dan kepulangan. Hal tersebut dilakukan karena data absensi yang didapat berupa *file* PDF yang tidak terhubung secara langsung dengan proses perhitungan.

Sistem yang sudah dijelaskan diatas maka terdapat beberapa kekurangan yang menjadikan sistem tersebut kurang efektif untuk digunakan. Kekurangan sistem tersebut seperti *file* absensi yang diterima berupa *file* PDF sehingga tidak memiliki basis data untuk menyimpan data absensi, dan terjadinya redundansi data karena perhitungan tunjangan kinerja dilakukan menggunakan aplikasi terpisah yaitu *Microsoft Excel*. Kekurangan lainnya yang ada pada sistem tersebut yaitu membutuhkan waktu untuk proses perhitungan tunjangan kinerja yang dilakukan

secara satu persatu data pegawai dengan menyusun ulang data absensi dan menambahkan rumus untuk menghitung dan melakukan pembukuan laporan.

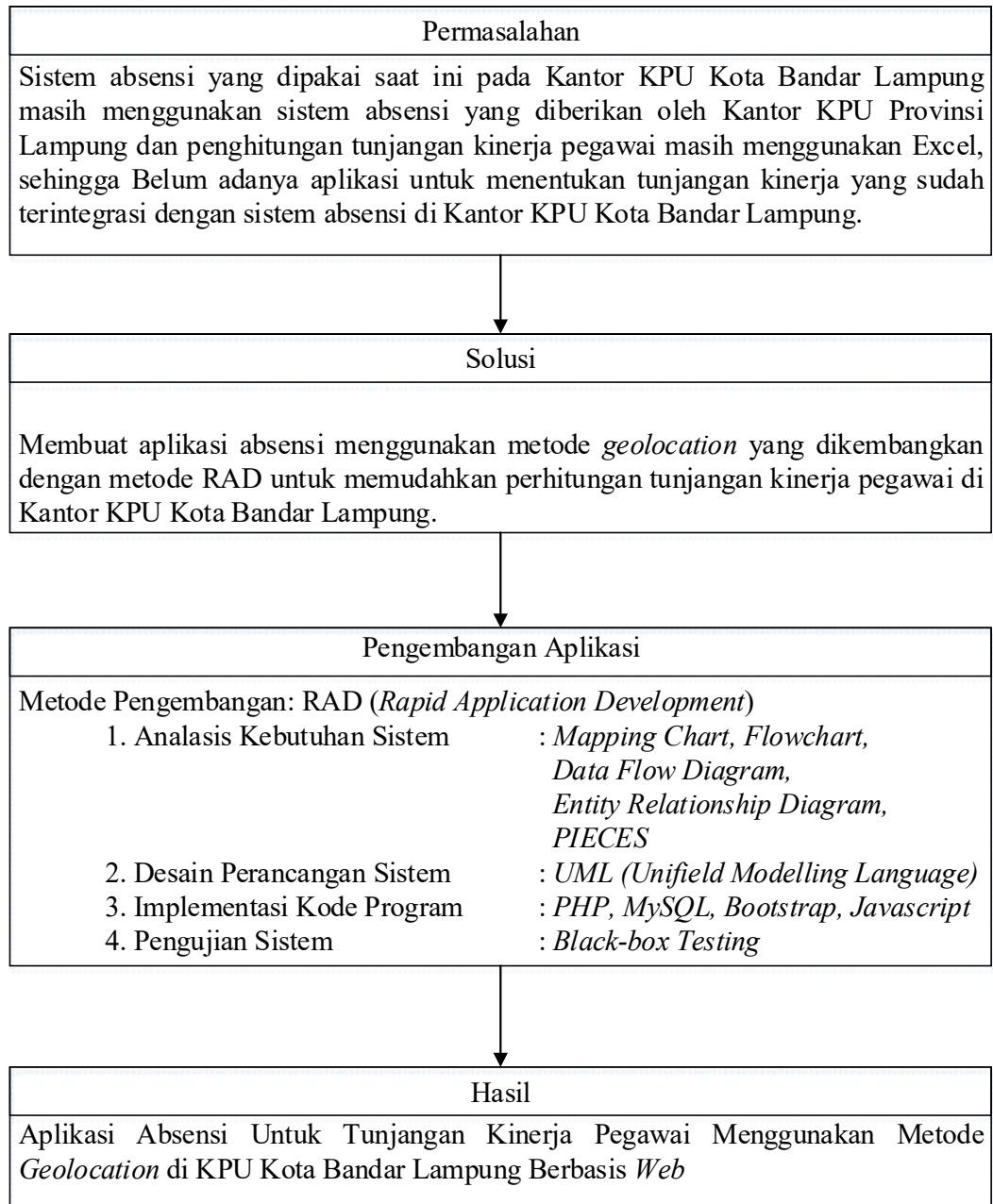
Permasalahan tersebut maka diperlukan adanya suatu upaya penggunaan teknologi yang tepat guna. Teknologi tersebut harus efisien dan efektif dalam proses absensi serta perhitungan tunjangan kinerja pegawai. Sistem yang akan dibuat dapat digunakan sebagai pengganti sistem lama, sehingga dibuatlah sebuah aplikasi tunjangan kinerja pegawai berdasarkan absensi menggunakan metode *geolocation* berbasis *web* agar dapat membantu Kantor KPU Kota Bandar Lampung untuk mencatat data absensi serta menghitung tunjangan kinerja pegawai yang secara otomatis dilakukan setiap pegawai melakukan absensi. *Geolocation* adalah identifikasi lokasi geografis suatu objek pada dunia nyata dan lebih spesifik dalam menentukan sebuah lokasi (misalnya alamat jalan) serta mengandung nilai *latitude* dan *longitude* (Sammir, 2021). Metode *geolocation* berjalan dimana saat pegawai melakukan absensi dapat terekam posisi perangkat yang digunakan serta menentukan maksimal jarak pegawai dari titik koordinat kantor sejauh lima puluh (50) meter agar tidak terjadi kecurangan absensi. Aplikasi ini juga menggunakan *framework Codeigniter* yaitu salah satu *framework* PHP yang paling *powerful* dengan sedikit *bug*. Selain itu, *framework* ini juga menggunakan konsep MVC (*Models-View-Controller*) yang dapat mempermudah pengelolaan sebuah aplikasi *website* (Prasena and Sama, 2020). *Framework* adalah sebuah kerangka kerja atau sekumpulan *script* yang dapat membantu memfasilitasi pengembangan dan pembuatan sebuah aplikasi *website* dengan memecahkan berbagai masalah pemrograman (Mandasari and Kaban, 2020).

1.2 Tujuan

Tujuan dari tugas akhir ini berdasarkan uraian latar belakang diatas yaitu dengan menghasilkan sebuah aplikasi yang berjudul “Aplikasi Absensi Untuk Tunjangan Kinerja Pegawai Menggunakan Metode *Geolocation* Di KPU Kota Bandar Lampung Berbasis *Web*”.

1.3 Kerangka Pemikiran

Berdasarkan latar belakang masalah yang ada dapat digambarkan suatu kerangka pemikiran seperti pada Gambar 1.



Gambar 1. Kerangka Pemikiran

1.4 Kontribusi

Beberapa pihak diberikan kontribusi dengan adanya Aplikasi Absensi Untuk Tunjangan Kinerja Pegawai Menggunakan Metode *Geolocation* Di KPU Kota Bandar Lampung Berbasis *Web* diharapkan mampu memberikan kontribusi kepada para pegawai Kantor KPU Kota Bandar Lampung. Berikut merupakan beberapa kontribusi yang dapat diberikan diantaranya:

1. Admin KPU Kota Bandar Lampung
 - a. Memudahkan dalam pengolahan data pegawai, data divisi, data kelas jabatan, data absensi, data tunjangan kinerja, serta data izin.
 - b. Memudahkan pengolahan laporan rekapitulasi absensi dan tunjangan kinerja pegawai
2. Pegawai KPU Kota Bandar Lampung
Memudahkan absensi kehadiran dan kepulangan pegawai, mengajukan izin serta memantau dan mengunduh data tunjangan kinerja untuk meningkatkan kualitas kinerja dan profesionalitas pegawai.
3. Politeknik Negeri Lampung
Penelitian ini diharapkan mampu menjadi referensi bagi penelitian selanjutnya dan bisa dikembangkan menjadi lebih sempurna.

BAB II. TINJAUAN PUSTAKA

2.1 Website

Kumpulan halaman yang dapat diakses melalui jalur koneksi *internet* dan dilihat oleh semua orang di seluruh dunia yang berisi informasi data digital baik berupa teks, gambar, animasi, suara dan video atau gabungan dari semuanya adalah pengertian dari halaman *website* Menurut Abdulloh (2018). Bahasa standar yang digunakan halaman *website* yaitu HTML. Skrip HTML ini akan diterjemahkan oleh *web browser* sehingga dapat ditampilkan dalam bentuk informasi yang dapat dibaca oleh semua orang.

2.2 Aplikasi

Aplikasi diuraikan Mahardika (2020) adalah program siap pakai atau suatu kelompok *file (form, class, rePort)* yang dapat dimanfaatkan sebagai media untuk menjalankan perintah-perintah dari pengguna untuk mendapatkan hasil yang lebih akurat sesuai dengan tujuan pengolahan data ataupun berbagai kegiatan tertentu yang saling terkait seperti pembuatan atau pengolahan dokumen, mengatur *windows*, permainan (*game*) dan sebagainya. Arti dari aplikasi yaitu pemecahan masalah yang menggunakan salah satu teknik pemrosesan data aplikasi dengan berpacu pada sebuah komputasi maupun pemrosesan data yang diinginkan. Secara umum aplikasi diartikan sebagai alat terapan yang difungsikan secara khusus dan terpadu sesuai kemampuan yang dimilikinya dan menjadi suatu perangkat komputer yang siap pakai bagi pengguna.

2.3 Tunjangan Kinerja

Tunjangan Kinerja adalah tunjangan yang diberikan kepada pegawai negeri sipil yang besarnya didasarkan pada hasil evaluasi jabatan dan capaian prestasi kerja pegawai negeri sipil. Tunjangan kinerja memegang peranan penting didalam kehidupan suatu organisasi diantaranya yaitu pemberian tunjangan pegawai penting, mengingat banyaknya kebutuhan pegawai dalam memenuhi tuntutan kehidupan sehari-harinya, besarnya tunjangan kinerja mencerminkan ukuran nilai karya mereka di antara pegawai itu sendiri, keluarga dan masyarakat, diharapkan tunjangan yang diterima mampu meningkatkan kesejahteraan hidup pegawai, dan

tunjangan yang diberikan kepada pegawai bertujuan untuk lebih memotivasi dalam meningkatkan kinerja pegawai dalam menyelesaikan pekerjaannya

Selain itu, kehadiran program tunjangan kinerja sangat penting bagi organisasi dalam memotivasi kerja pegawainya. Pemberian insentif dalam instansi, baik lembaga pemerintahan maupun lembaga swasta sangat bermanfaat untuk bisa meningkatkan kinerja pegawai, karena insentif dapat meningkatkan keyakinan pegawai bahwa prestasi yang tinggi akan menghasilkan imbalan, dan untuk dapat mengikuti segala perkembangan yang ada dan tercapainya tujuan organisasi maka perlu adanya suatu motivasi agar pegawai mampu bekerja dengan baik, dan salah satu motivasi itu dengan memenuhi keinginan-keinginan pegawai antara lain: gaji atau upah yang baik, pekerjaan yang aman, suasana kerja yang kondusif, penghargaan terhadap pekerjaan yang dilakukan, pimpinan yang adil dan bijaksana, pengarahan dan perintah yang wajar, atau dengan mengupayakan insentif yang besarnya proporsional dan juga bersifat progresif yang artinya sesuai dengan jenjang karir, karena insentif sangat diperlukan untuk memacu kinerja para pegawai agar selalu berada pada tingkat optimal sesuai kemampuan masing-masing (Mirasanti, 2020).

2.4 Absensi

Absensi (Triyono et al., 2018) adalah suatu pendataan kehadiran, bagian dari pelaporan aktivitas suatu institusi, suatu komponen institut itu sendiri yang berisi data-data kehadiran yang disusun dan diatur sedemikian rupa sehingga mudah untuk dicari dan dipergunakan apabila sewaktu-waktu di perlukan oleh pihak yang berkepentingan.

2.5 Geolocation

Geolocation merupakan suatu mekanisme identifikasi lokasi geografis pengguna perangkat komputer dengan menggunakan beberapa cara seperti *network routing address* dan *Global Positioning System (GPS)* (Sammir, 2021). *Geolocation API* memperbolehkan pengguna memberikan koordinat lokasinya kepada aplikasi *web browser*. *Geolocation API* dapat diakses menggunakan *navigator geolocation* yang mengharuskan *browser* meminta izin kepada pengguna untuk mengakses lokasi perangkat dan memberikannya kepada aplikasi *web browser*.

2.6 *Internet*

Menurut Putri (2020) *International Networking (Internet)* adalah jaringan komputer yang sangat besar, terdiri dari jutaan perangkat komputer yang terhubung melalui protokol tertentu untuk bertukar informasi antar komputer tersebut. Semua komputer yang terhubung ke *internet* saling bertukar informasi menggunakan protokol yang sama, yaitu TCP/IP (*Transmission Control Protocol/Internet Protocol*). *Internet* menyediakan akses ke layanan telekomunikasi dari sumber informasi ke jutaan pengguna di seluruh dunia.

2.7 *Web Browser*

Browser adalah program aplikasi yang menerjemahkan kode HTML dan menampilkannya di halaman *website*. *Web browser* dapat diartikan sebagai aplikasi dengan fungsi mengambil, menyajikan, dan menjelajahi. Sumber informasi di *World Wide Web* diidentifikasi dengan *Uniform Resource Identifier (URI)* yang merujuk ke halaman *web*. *Hyperlink* memungkinkan pengguna untuk dengan mudah menavigasi *browser* ke sumber daya terkait. *Web browser* dapat digunakan untuk mengakses informasi yang disediakan oleh *server* di jaringan lokal dalam *file* sistem. Ada 3 jenis *web browser* yang umum digunakan, antara lain *Internet Explorer*, *Mozilla Firefox* dan *Opera* (Putra et al., 2021).

2.8 *Database*

Database adalah himpunan kelompok data yang saling berhubungan yang diorganisasikan sedemikian rupa sehingga dapat dimanfaatkan kembali dengan cepat dan mudah (Sitinjak et al., 2020). *Database* merupakan sistem penyimpanan yang menyimpan kumpulan informasi yang disusun sehingga mudah untuk diakses.

2.8.1. DBMS

DBMS adalah singkatan dari *Database Management System* yaitu kumpulan dari *database* itu sendiri, *software database* dan kamus data yang dioperasikan oleh kelompok pengguna atau *developer* aplikasi dan dikelola oleh administrator *database*. DBMS membantu dalam pemeliharaan dan pengolahan kumpulan data dalam jumlah besar (Junaedi et al., 2020).

2.8.2. MySQL

MySQL merupakan *software database open source* yang sering digunakan untuk mengolah basis data yang menggunakan bahasa SQL (Sitinjak et al., 2020). MySQL merupakan tipe data relasional yang artinya MySQL menyimpan datanya dalam bentuk tabel-tabel yang saling berhubungan. Keuntungan menyimpan data di *database* adalah kemudahannya dalam penyimpanan dan menampilkan data karena dalam bentuk tabel. MySQL adalah salah satu aplikasi DBMS yang sudah sangat banyak digunakan para pemrogram aplikasi *web*. Kelebihan dari MySQL adalah gratis, handal, selalu di-*update* dan banyak forum yang memfasilitasi para pengguna jika memiliki kendala. MySQL juga menjadi DBMS yang sering di *bundling* dengan *web server* sehingga proses instalasinya jadi lebih mudah. Dapat ditarik kesimpulan bahwa MySQL merupakan aplikasi yang digunakan untuk mengolah basis data yang banyak digunakan untuk membangun aplikasi yang menggunakan *database*).

2.9 Bahasa Pemrograman

Bahasa Pemrograman, atau biasa dikenal dengan bahasa komputer atau bahasa pemrograman komputer, adalah instruksi standar untuk memberikan instruksi yang dimasukkan oleh pengguna ke dalam komputer sehingga komputer dapat berfungsi sesuai arahan dari pengguna. Bahasa pemrograman ini adalah seperangkat aturan sintaksis dan semantik yang digunakan untuk mendefinisikan program komputer. Bahasa ini memungkinkan pemrogram untuk menentukan dengan tepat data apa yang akan diproses oleh komputer, bagaimana data itu akan disimpan/ditransmisikan, dan tindakan apa yang harus dilakukan dalam situasi lain secara bersamaan.

Bahasa pemrograman (*programming language*) adalah instruksi standar untuk menginstruksikan komputer untuk melakukan fungsi tertentu. Bahasa pemrograman ini adalah seperangkat aturan sintaksis dan semantik yang digunakan untuk mendefinisikan program komputer. Bahasa ini memungkinkan pemrogram untuk menentukan dengan tepat data apa yang akan diproses oleh komputer, bagaimana data itu akan disimpan/ditransmisikan, dan tindakan apa yang harus dilakukan dalam situasi lain secara bersamaan. Bergantung pada seberapa dekat Anda dengan komputer, bahasa pemrograman meliputi:

1. Bahasa Mesin, yang memberikan instruksi ke komputer dalam kode bahasa biner, misalnya: 011001011100110
2. Bahasa tingkat rendah, juga dikenal sebagai bahasa rakitan (*Assembly*), memberikan instruksi ke komputer menggunakan kode pendek (kode memori), misalnya: *engine_code* | MOV, SUB, CMP, JMP, JGE, JL, LOOP, dll.
3. Bahasa antara, yaitu bahasa komputer yang menggunakan campuran instruksi verbal manusia (lihat contoh bahasa tingkat tinggi di bawah) dan instruksi simbolik, misalnya {, }, ?, Andlt ; dan lt ;, >, > dan ||, dst.
4. Bahasa tingkat tinggi, adalah bahasa komputer yang menggunakan instruksi-instruksi yang berasal dari unsur-unsur bahasa manusia, misalnya *start*, *end*, *if*, *for*, *while* dan, *or*, dst.

Komputer dapat memahami bahasa manusia, yang membutuhkan program *compiler* atau *interpreter*. Kebanyakan bahasa pemrograman tergolong bahasa tingkat tinggi, hanya bahasa C yang tergolong bahasa tingkat menengah dan bahasa *assembly* sebagai bahasa tingkat rendah (Saragih, 2018).

2.9.1. PHP

PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman yang berjalan di *server web* dan bertindak sebagai pengolah data di *server*. Data yang dikirimkan oleh klien pengguna akan diproses dan disimpan di *database server web* dan dapat ditampilkan kembali saat diakses. Untuk menjalankan kode program PHP, *file* perlu diunggah ke *server*. Download adalah proses mentransfer data atau *file* dari komputer klien ke *server web* (Khairun & Mubarak, 2019).

Pembuatan halaman *web* dinamis yang dapat dengan mudah diperbarui dari *browser* setiap saat, diperlukan suatu program yang dapat mengolah data dari klien atau *server* itu sendiri sehingga disajikan dengan mudah dan nyaman di *web browser*. PHP adalah salah satu program yang dapat berjalan di *server* dan cukup andal.

PHP bekerja dalam dokumen HTML (*Hypertext Markup Language*) untuk dapat menghasilkan konten halaman *web* sesuai permintaan sehingga kita dapat mengubah *website* kita menjadi aplikasi *web* dan bukan hanya kumpulan halaman statis yang jarang diperbarui.

Awalnya, PHP dirancang untuk berintegrasi dengan *server web Apache*. Namun belakangan ini PHP juga dapat bekerja dengan *web server* seperti PWS (*Personal Web Server*), IIS (*Internet Information Server*) dan *Xitami*. Perbedaan PHP dengan bahasa pemrograman lainnya adalah adanya *tag* pendefinisian yang diawali dengan "<?" atau "<?php" dan diakhiri dengan ">," sehingga kita bebas menempatkan *script* PHP dimana saja di dalam dokumen HTML yang telah kita buat.

2.9.2. HTML

Menurut Mariko (2019) *Hypertext Markup Language* atau biasa disingkat HTML adalah bahasa pemrograman standar yang digunakan untuk menampilkan konten pada halaman *web*. Fungsi-fungsi yang dapat dilakukan dengan bahasa pemrograman HTML adalah:

1. Mengatur dan mendesain tampilan konten *website*,
2. Membuat tabel di *website*,
3. Mempublikasikan halaman *web* secara online,
4. Membuat formulir yang dapat dimasukkan dan mengelola pendaftaran dan transaksi melalui *website*,
5. Menampilkan area gambar di *browser*.

Penanda perintah dalam bahasa pemrograman *Hypertext Markup Language* (HTML) disebut *signaling*. *Tag* digunakan untuk mendefinisikan tampilan dan nuansa dokumen HTML. *Tag* HTML digunakan untuk menentukan bahwa konten *file* adalah dokumen. Elemen *head* adalah judul dokumen HTML. Elemen *head* digunakan untuk mengatur identitas *file*, sedangkan *tag body* digunakan untuk menentukan apa yang akan ditampilkan pada halaman *web*.

2.9.3. JavaScript

JavaScript adalah bahasa pemrograman berbentuk kumpulan *script* yang berjalan pada suatu dokumen HTML. *JavaScript* dapat menyempurnakan tampilan dan sistem pada halaman *web-based application* yang dikembangkan (Mariko, 2019). Adapun karakteristik dari bahasa pemrograman *JavaScript* adalah:

1. Bahasa pemrograman berjenis *high-level programming*;
2. Bersifat *client-side*;
3. Berorientasi pada objek, dan

4. Bersifat *loosely typed*.

Aplikasi atau *tools* standar yang digunakan dalam mengimplementasikan *JavaScript* adalah:

1. *Software text-editor*: seperti Notepad++, Adobe Dreamweaver dan NetBeans;
2. Web browser, seperti: Mozilla Firefox, Google Chrome, Internet Explorer, OperaMini, Safari, dan lain sebagainya;
3. HTML.

Kelemahan dari bahasa pemrograman *JavaScript* ini sendiri adalah tidak didukung oleh *browser* versi lama.

2.9.4. CSS (*Cascading Style Sheet*)

Cascading Style Sheet (CSS) adalah aturan untuk mengatur beberapa elemen dalam sebuah halaman *web* agar lebih terstruktur dan seragam. CSS bukanlah bahasa pemrograman. Demikian pula, *styles* dalam aplikasi pengolah kata seperti *Microsoft Word* dapat mengatur beberapa *style*, seperti *heading*, subbab, *bodytext*, *footer*, *image*, dan lain-lain untuk digunakan bersama-sama dalam beberapa *file*. Umumnya, CSS digunakan untuk memformat tampilan halaman *web* yang dibuat dengan bahasa HTML dan XHTML. CSS dapat mengontrol ukuran gambar, warna teks tubuh, warna tabel, ukuran batas, warna batas, warna *hyperlink*, warna *mouse*, spasi paragraf, spasi teks, kiri, kanan, atas, bawah, dan pengaturan lainnya. CSS adalah bahasa *style sheet* yang digunakan untuk mengontrol tampilan dokumen yang memungkinkan kita untuk menampilkan halaman yang sama dengan format yang berbeda (Nurjaman & Yasin, 2020).

2.9.5. *Bootstrap*

Bootstrap merupakan sebuah *library framework* CSS yang dibuat khusus untuk pengembangan situs *web frontend*. *Bootstrap* adalah salah satu *framework* HTML, CSS, dan *javascript* paling populer di kalangan pengembang *web* (Nurjaman & Yasin, 2020). Saat ini, hampir semua pengembang *web* telah menggunakan *bootstrap* untuk membuat *rendering* UI lebih mudah dan sangat cepat karena hanya perlu menambahkan beberapa *layer*, misalnya membuat tombol, *grid*, navigasi dan lain-lain. *Bootstrap* adalah *framework* untuk membuat desain *web* yang responsif. Artinya, layar *web* yang dibuat oleh *bootstrap* akan menyesuaikan dengan ukuran layar *browser* yang digunakan di desktop, tablet atau

mobile device. Fitur ini dapat dihidupkan atau dimatikan sesuai dengan keinginan sehingga bisa membuat *website* yang hanya untuk tampilan desktop, dan ketika ditampilkan oleh *mobile device*, tampilan dari *web* yang dibuat tidak bisa beradaptasi dengan layar. *Bootstrap* juga bisa membuat *web* dinamis atau statis. *Bootstrap* telah menyediakan kumpulan komponen *class interface* dasar yang dirancang sedemikian rupa untuk membuat tampilan yang menarik, bersih, dan ringan. *Bootstrap* juga memiliki fitur *grid* yang mengatur tata letak pada halaman *web* yang dapat digunakan dengan sangat mudah dan cepat. Penggunaan *bootstrap* akan memiliki kemampuan untuk mengembangkan tampilan *website* menggunakan *bootstrap*, yaitu dengan memodifikasi tampilan *bootstrap* dengan menambahkan *class* dan CSS sendiri.




2.10 Perancangan Sistem



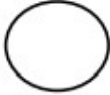
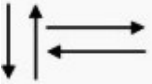
Perancangan sistem adalah tahap analisis pengembangan sistem seperti definisi, perencanaan, dan fabrikasi untuk menggambarkan bagaimana suatu sistem dikandung. Langkah-langkah perancangan sistem meliputi *mapping chart*, *flowchart*, *data flow diagram*, dan *entity relationship diagram*.

2.10.1. Mapping Chart

Mapping Chart merupakan bagan alir menampilkan laporan dan aliran formulir termasuk salinan. *Flowchart* dari dokumen ini gunakan notasi yang sama dengan sistem debit digunakan dalam diagram dokumen (tabel pemetaan) seperti pada Tabel 1.

Tabel 1. Simbol-Simbol *Mapping Chart*

Simbol	Nama	Keterangan
(1)	(2)	(3)
	<i>Document</i>	Menyatakan dokumen <i>input</i> atau <i>output</i>
	Kegiatan Manual	Menyatakan kegiatan secara manual
	<i>Process</i>	Menyatakan suatu proses yang sedang dilakukan

(1)	(2)	(3)
	<i>Keyboard</i>	Menunjukkan <i>input</i> dengan <i>keyboard</i>
	<i>Harddisk/Penyimpanan</i>	Untuk penyimpanan data
	<i>Connector</i>	Penghubung suatu proses pada halaman yang sama dan berbeda
	Garis Alir	Menyatakan jalannya arus suatu proses

Sumber: (Liksha, 2020)




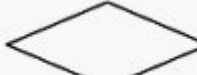






2.10.1. Flowchart

Flowchart adalah gambar diagram langkah-langkah dan urutan prosedur sebuah program. *Flowchart* adalah grafik menunjukkan aliran internal program atau prosedur sistem dengan cara yang wajar. Grafik atas terutama digunakan untuk alat bantu komunikasi dan untuk dokumentasi. Ada lima jenis *flowchart*, terdiri dari:

1. Bagan Alir Sistem (*System Flowchart*) adalah grafik yang menunjukkan arus kerja sistem secara keseluruhan.
2. Bagan Alir Dokumen (*Document Flowchart*) Juga dikenal sebagai bagan alir formulir. adalah diagram alir yang menunjukkan arus laporan dan formulir, termasuk salinannya.
3. Bagan Alir Skematik (*Schematic Flowchart*) adalah diagram alur deskriptif prosedur dalam sistem menggunakan notasi diagram alur sistem dan gambar komputer dan perangkat lain yang digunakan oleh sistem.
4. Diagram Alir Program (*Program Flowchart*) adalah tabel penjelasan rinci tahapan proses program.
5. Diagram alir proses (*Process Flowchart*) adalah diagram alur yang banyak digunakan dalam teknik industri untuk menggambarkan proses dalam satu prosedur.

Simbol-Simbol *Flowchart* beserta penjelasannya lengkap dapat dilihat pada Tabel 2.

Tabel 2. Simbol-Simbol *Flowchart*

Simbol	Nama	Keterangan
	Terminal	Menyatakan permulaan atau akhir program
	<i>Input / Output</i>	Menyatakan <i>input</i> dan <i>output</i>
	<i>Process</i>	Menyatakan suatu proses yang sedang dilakukan
	<i>Decision</i>	Menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban ya/tidak
	<i>Connector</i>	Menyatakan sambungan dari proses ke proses lainnya dalam satu halaman yang sama
	<i>Offline Connector</i>	Menyatakan sambungan dari proses ke proses lainya dalam halaman yang berbeda
	<i>Predefined Proses</i>	Menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal
	<i>Punch Tape</i>	Menyatakan <i>input</i> dan <i>output</i> menggunakan pita kertas penghubung
	<i>Document</i>	Mencetak keluaran dalam bentuk dokumen
	<i>Flow</i>	Menyatakan jalannya arus suatu proses

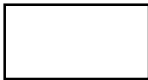

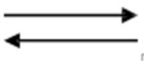

Sumber: (Liksha, 2020).

2.10.2. Data Flow Diagram

Data flow diagram adalah teknik grafis untuk mewakili aliran informasi dan transformasi yang diterapkan pada data saat bergerak dari *input* ke *output*. *Data flow diagram* (DFD) adalah alat pemodelan yang memungkinkan profesional sistem untuk menggambarkan sistem sebagai jaringan proses fungsional yang dihubungkan oleh aliran data manual atau terkomputerisasi. DFD ini sering disebut dengan *bubble chart*, *bubble diagram*, *process model*, *flowchart*, atau *function*

model. DFD datang dalam dua bentuk: DFD fisik (*Physical Data Flow Diagram*) dan DFD logis (*Logical Data Flow Diagram*). DFD fisik lebih fokus pada bagaimana proses sistem diimplementasikan, sedangkan DFD logis lebih fokus pada proses dalam sistem.

Tabel 3. Simbol-Simbol *Data Flow Diagram*


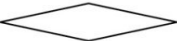


Simbol	Nama	Keterangan
	Entitas Eksternal	Dapat berupa orang atau unit terkait yang berinteraksi dengan sistem langsung akan tetapi diluar sistem
	Proses	Sebuah proses yang melakukan transformasi data atau komponen fisik.
	Alir Data	Mengalirkan data yang mengarah khusus dari sumber ke tujuan
	<i>Data Store</i>	Tempat penyimpanan data atau tempat data di <i>refer</i> oleh proses.

Sumber: (Soufitri, 2019)

2.10.3. *Entity Relationship Diagram*

Entity Relationship Diagram atau umumnya dikenal sebagai diagram ER adalah representasi grafis yang menggambarkan desain konseptual dari basisdata yang akan dibangun. *Entity Relationship Model* digunakan untuk mengubah data yang ada di dunia nyata dalam bentuk simbolik sebagai perangkat konseptual menjadi diagram data yang dikenal sebagai *diagram Entity-Relationship*.

Tabel 4. Simbol-simbol *Entity Relationship Diagram*

Simbol	Nama	Keterangan
	Entitas	Menyatakan kumpulan objek yang dapat diidentifikasi secara unik
	Relasi	Menyatakan hubungan antar entitas
	Atribut	Menyatakan suatu
	penghubung	Menyatakan hubungan antar entitas dengan atributnya dan hubungan entitas dengan relasinya

Sumber: (Syukron, 2019)

2.11 *Rapid Application Development (RAD)*

Rapid Application Development atau disingkat RAD adalah model proses pengembangan perangkat lunak sekuensial linier yang menekankan pada siklus pengembangan yang sangat singkat. Tahapan RAD terdiri dari tiga tahap yaitu:

1. Perencanaan Kebutuhan (*Requirements Planning*) merupakan tahap awal dalam pengembangan sistem, dimana pada tahap inilah identifikasi masalah dan pengumpulan data yang diperoleh dari pengguna dilakukan dengan tujuan untuk mengidentifikasi menentukan maksud atau tujuan akhir dari sistem dan informasi yang diinginkan. kebutuhan. Pada tahap ini, keterlibatan kedua belah pihak sangat penting dalam menentukan kebutuhan untuk mengembangkan suatu sistem.
2. *Workshop* Perancangan (*Design Workshop*) merupakan fase desain dan perbaikan yang dapat digambarkan sebagai *workshop*. Analis dan pemrogram dapat bekerja untuk membuat dan menyajikan kepada pengguna representasi visual dari desain dan model kerja. Lokakarya desain ini dapat memakan waktu beberapa hari tergantung pada ukuran aplikasi yang akan dikembangkan. Selama lokakarya desain, pengguna menanggapi prototipe yang ada, dan penganalisis menyempurnakan modul yang dirancang berdasarkan umpan balik pengguna.
3. Implementasi (*Implementation*) adalah langkah di mana *programmer* menerapkan desain sistem yang disetujui pada langkah sebelumnya. Sebelum mengimplementasikan sistem, proses pengujian program dilakukan terlebih dahulu untuk mendeteksi *bug* yang tersisa pada sistem yang dikembangkan. Pada tahap ini, biasanya untuk memberikan umpan balik pada sistem yang telah dibuat dan menerima persetujuannya (Mandasari & Kaban, 2020).

2.12 *Framework*

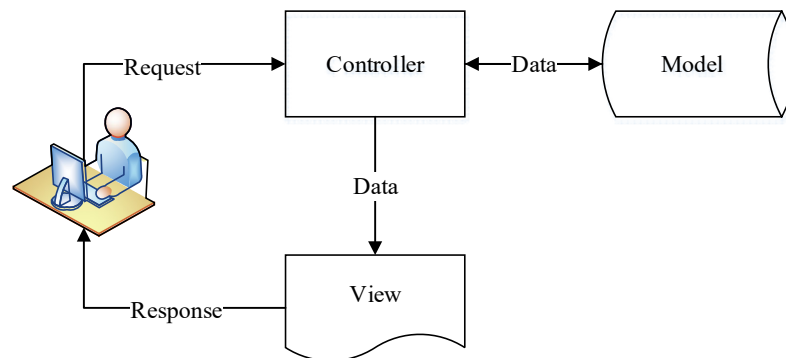
Framework adalah sebuah kerangka kerja atau sekumpulan *script* yang dapat membantu memfasilitasi pengembangan dan pembuatan sebuah aplikasi *website* dengan memecahkan berbagai masalah pemrograman (Safitri et al., 2021). Definisi lain mengatakan bahwa kerangka kerja adalah kumpulan komponen dan variabel, fungsi atau prosedur dan kelas dengan tujuan tertentu sehingga dapat menyederhanakan dan mempercepat pembacaan kode, perencanaan, pembuatan

situs *web*, percobaan tanpa harus membuat fungsi atau kelas dari awal. Desain aplikasi situs *web* untuk kemudahan dan pemeliharaan.

Ada banyak *framework* yang bisa digunakan, salah satunya adalah *CodeIgniter*. *CodeIgniter* adalah kerangka kerja yang dibuat dalam bahasa PHP, yang dapat digunakan untuk membuat halaman *web* dengan cepat. *Framework* ini banyak digunakan karena merupakan salah satu *framework open source* yang dikembangkan oleh komunitas sehingga cukup mudah dipelajari dan merupakan salah satu *framework* PHP yang paling *powerful* dengan sedikit *bug*. Selain itu, *framework* ini juga menggunakan konsep MVC (*Models-View-Controller*) yang dapat mempermudah pengelolaan sebuah aplikasi *website*.

2.13 Konsep *Model-View-Controller*

Model-View-Controller (MVC) adalah konsep yang diperkenalkan oleh pendiri *Smalltalk* (Trygve Reenskaug) untuk mengubah satu jenis datagram jaringan menjadi jenis data lain dengan pemrosesan (*model*), pengontrol proses manipulasi (*controller*), dan tampilan (*view*) yang berbeda untuk ditampilkan di antarmuka pengguna (Wijaya and Christian, 2019). Berdasarkan penjelasan yang ada dapat digambarkan suatu konsep *Model View Controller* seperti pada Gambar 2.



Gambar 2. Konsep *Model View Controller*

berikut adalah penjelasan lengkap dari gambar di atas alur dari *Model View Controller*:

1. *Model*

Model adalah bagian dari kode program manajemen basis data, isi *model* adalah bagian-bagian (fungsi) yang berhubungan langsung dengan basis data untuk

pengelolaan data seperti entri data, pembaruan data, penghapusan data dan sejenisnya, tetapi tidak dapat dihubungkan secara langsung ke *view*.

2. *view*

pada bagian tampilan (*view*) ini berisi semua detail implementasi UI. *view* adalah bagian dari kode program yang mengontrol tampilan halaman *web*. Tampilan biasanya berupa *file* skrip HTML. Tampilan juga digunakan untuk menampilkan data dan *input* pengguna, sehingga *view* adalah tampilan halaman *web*.

3. *Controller*

Metode pemrosesan (*controller*) adalah bagian yang berguna untuk menghubungkan antara *model* dan *view*. *Controller* berisi perintah yang bertanggung jawab untuk memproses suatu data dan mengirimkannya ke halaman *web*. *Controller* berfungsi untuk menerima *request* dan data dari *user* kemudian menentukan apa yang harus diproses oleh aplikasi.

2.14 *CodeIgniter*

Codeigniter adalah *framework* yang dirilis oleh *Ellis Lab* pada tanggal 28 Februari 2006 dengan versi stabil 2.1.0, hingga Oktober 2011 diambil alih oleh *British Columbia Institute of Technology* (BCIT) untuk dikembangkan hingga sekarang. Menggunakan *framework Codeigniter* dalam pengembangan aplikasi sudah ada *framework* untuk bekerja dalam pemrograman PHP. *Codeigniter* menyediakan sejumlah perpustakaan yang diperlukan untuk melakukan tugas-tugas umum dan didukung oleh struktur dan antarmuka yang sederhana untuk mengakses koleksi perpustakaan (Prasena & Sama, 2020). Beberapa struktur umum yang digunakan antara lain:

1. *Index.php* menjadi pengontrol awal yang menginisiasi sumber daya yang dibutuhkan untuk menjalankan *Codeigniter*.
2. *Router* bekerja untuk memeriksa permintaan HTTP untuk menentukan kelangkaan permintaan.
3. *File cache* dikirim langsung ke *browser* melewati eksekusi sistem normal.
4. Untuk keamanan, sebelum panggilan suatu aplikasi *controller* apa pun, permintaan HTTP yang dikirim pengguna akan difilter.

2.15 XAMPP

XAMPP (Suprianto et al., 2018) merupakan perangkat lunak gratis yang mendukung banyak sistem operasi dan merupakan kumpulan program. Fungsinya adalah sebagai *server* mandiri (*localhost*), yang terdiri dari program *Apache* HTTP *Server*, *database* MySQL, dan penerjemah bahasa yang ditulis dalam bahasa pemrograman PHP dan Perl. Nama XAMPP adalah singkatan dari X (empat sistem operasi), Apache, MySQL, PHP, dan Perl. Program ini tersedia di bawah *GNU General Public License* dan merupakan *web server* gratis yang mudah digunakan yang dapat melayani halaman *web* dinamis.

2.16 Sublime Text

Sublime Text (Setiawan & Masya, 2020) merupakan aplikasi teks editor dan kode yang dapat berjalan di berbagai *platform* sistem operasi menggunakan teknologi *Python API*. Pembuatan aplikasi ini terinspirasi dari aplikasi *Vim* yang sangat fleksibel dan *powerful*. Fungsionalitas aplikasi ini dapat diperluas menggunakan *sublime-packages*. *Sublime Text* merupakan bukan aplikasi *open source* dan juga aplikasi yang dapat digunakan dan diunduh secara gratis, namun beberapa fitur pengembangan fitur (*packages*) dalam aplikasi ini merupakan hasil eksplorasi dan mendapat dukungan penuh dari masyarakat dan memiliki linsensi aplikasi gratis.

2.17 Penelitian Terkait

Penelitian terkait yang digunakan penulis sebagai refrensi dalam menyusun tugas akhir ini disajikan pada Tabel 5.

Tabel 5. Penelitian Terkait

No.	Judul	Penulis	Tujuan	Hasil
1	Aplikasi Perhitungan Tunjangan Kinerja Pegawai Berbasis <i>Web</i> Dengan <i>Codeigniter</i> Studi Kasus Dinas Tenaga	Edi Murwanto (2018)	Merancang aplikasi perhitungan tunjangan kinerja dengan memanfaatkan <i>Codeigniter</i> sebagai <i>framework</i> dalam aplikasi	Hasil pengujian keseluruhan menunjukkan bahwa aplikasi tersebut dapat berfungsi dengan baik dan berguna bagi Kantor Dinas Tenaga Kerja

	Kerja Dan Transmigrasi Provinsi Lampung		perhitungan tunjangan kinerja serta memakai bahasa pemrograman PHP untuk membangun aplikasi berbasis <i>web</i>	dan Transmigrasi Provinsi Lampung
2	Aplikasi Absensi Dengan Metode Geolocation Berbasis Web (Studi Kasus: PT. Codepolitan Integrasi Indonesia)	Shandy Tresnawati dan Alfian Pratama (2021)	Merancang aplikasi absensi dengan memanfaatkan metode <i>geolocation</i> dalam aplikasi absensi serta memakai nilai <i>latitude</i> , <i>longitude</i> dan <i>radius</i> untuk perhitungan jarak dan lokasi pengguna	Hasil pengujian keseluruhan menunjukkan bahwa aplikasi tersebut dapat berfungsi dengan baik dan berguna bagi PT. Codepolitan Integrasi Indonesia
3	Pengembangan Aplikasi <i>Geolocation</i> Untuk Monitoring Lokasi Mahasiswa Selama Pandemi Berbasis <i>Android</i> Menggunakan Metode <i>Waterfall</i> (Studi Kasus: STMIK Insan Pembangunan)	Nana Supiana (2022)	Membuat sebuah perangkat lunak sebagai media dalam membatasi kegiatan mahasiswa selama <i>pandemic</i> yang berada di area publik	Hasil penelitian penulis mengambil kesimpulan aplikasi tersebut untuk mempermudah proses <i>tracking</i> lokasi mahasiswa selama berada diluar lingkungan kampus dan memperoleh informasi lokasi mahasiswa berada selama melakukan kegiatan di area terbuka