



BAB II

TINJAUAN PUSTAKA

2.1 Teori Judul

2.1.1 Pengertian Aplikasi

Peraturan Pemerintah Nomor 7 Tahun 1977, “Kenaikan Gaji Berkala adalah kenaikan gaji yang diberikan kepada pegawai negeri yang telah mencapai masa kerja golongan yang ditentukan untuk kenaikan gaji berkala yaitu setiap 2 (dua) tahun sekali dan apabila telah memenuhi persyaratan berdasarkan peraturan perundang –undangan yang berlaku”.

2.1.2 Pengertian Gaji

Menurut Kamus Besar Bahasa Indonesia (KBBI), kata gaji dapat diartikan sebagai upah dari pekerjaan yang dilakukan yang dibayarkan dalam waktu yang tetap. Selain itu, dapat juga diartikan sebagai balas jasa yang diterima oleh perusahaan dalam bentuk uang berdasarkan waktu tertentu. Dari definisi tersebut, dapat dilihat bahwa salah satu ciri dari gaji adalah dibayarkan dalam waktu tertentu atau secara periodik dan tidak tentative.

Menurut Mulyadi (2001:377), “Gaji adalah pembayaran atas penyerahan jasa yang dilakukan oleh karyawan baik yang mempunyai jabatan maupun karyawan pelaksana.”

Menurut G. Sugiyarso & F. Winarni (2005), “Gaji merupakan sejumlah pembayaran kepada pegawai yang diberi tugas administratif dan manajemen yang biasanya ditetapkan secara bulanan”.

Dari beberapa definisi di atas, maka dapat disimpulkan bahwa gaji merupakan upah yang dibayarkan dalam waktu yang tetap sebagai tanda jasa atas pekerjaan yang sudah dilakukan, dan pemberian upah tersebut dilakukan dari orang yang memiliki posisi jabatan yang lebih tinggi dalam suatu organisasi atau perusahaan. Jumlah gaji yang diterima oleh seseorang tidak tetap begitu saja, dapat mengalami kenaikan dalam masa kerjanya.

2.1.3 Penegertian Website

Menurut Agus Hariyanto (2015), “Website adalah web dapat diartikan sebagai kumpulan halaman yang menampilkan informasi data teks, data gambar, data animasi, suara, video dan gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, dimana masing-masing dihubungkan dengan jaringan-jaringan halaman (*hyperlink*)”.

2.1.4.1 Pengertian Simple Additive Weighting (SAW)

Menurut Heny Pratiwi (2016:136) “Metode *Simple Additive Weighting* (SAW) adalah metode penjumlahan bobot dari kriteria setiap objek-objek yang berbeda dan memiliki kesempatan yang sama pada semua kriteria yang dimiliki.”

Adapun algoritma penyelesaian metode ini adalah sebagai berikut;

1. Mendefinisikan terlebih dahulu kriteria-kriteria yang akan di jadikan sebagai tolak ukur penyelesaian masalah.
2. Menormalisasi setiap nilai alternatif pada setiap atribut dengan cara mengitung nilai rating kinerja.

$$r_{ij} = \begin{cases} \frac{x_{ij}}{\text{Max}_i x_{ij}} & \text{jika } j \text{ adalah atribut keuntungan (benefit)} \\ \frac{\text{Min}_i x_{ij}}{x_{ij}} & \text{jika } j \text{ adalah atribut biaya (cost)} \end{cases}$$

Keterangan:

Max Xij = Nilai terbesar dari setiap kriteria i.

Min Xij = Nilai terkecil dari setiap kriteria i.

Xij = Nilai atribut yang dimiliki dari setiap kriteria.

Benefit = Jika nilai terbesar adalah nilai terbaik.

Cost = Jika nilai terkecil adalah terbaik.

3. Mengitung nilai bobot preferensi pada setiap alternative.

$$V_i = \sum_{j=1}^n W_j r_{ij}$$

Keterangan:

V_i = Rangkaian untuk setiap alternative.

W_j = Nilai bobot rangkaian (dari setiap kriteria)

R_{ij} = Nilai rating kinerja ternormalisasi.

4. Melakukan perankingan.

2.1 Teori Khusus

2.2.1 *Rapid Application Development (RAD)*

Menurut Kendall (2010), *Application Development (RAD)* adalah suatu pendekatan berorientasi objek terhadap pengembangan sistem yang mencakup suatu metode pengembangan serta perangkat-perangkat lunak. *Application Development (RAD)* bertujuan mempersingkat waktu yang biasanya diperlukan dalam siklus hidup pengembangan sistem tradisional antara perancangan dan penerapan suatu sistem informasi.

Terdapat tiga fase dalam *Application Development (RAD)* yang melibatkan penganalisis dan pengguna dalam tahap penilaian, perancangan, dan penerapan. Adapun ketiga fase tersebut adalah *requirements planning* (perencanaan syarat-syarat), *RAD design workshop* (*workshop* desain RAD), dan *Implementation* (implementasi). Sesuai dengan metodologi *Application Development (RAD)* menurut Kendall (2010), berikut ini adalah tahap-tahap pengembangan aplikasi dari tiap-tiap fase pengembangan aplikasi.

1. *Requirements Planning* (Perencanaan Syarat-Syarat)

Dalam fase ini, pengguna dan penganalisis bertemu untuk mengidentifikasi tujuan-tujuan aplikasi atau sistem serta untuk mengidentifikasi syarat-syarat informasi yang ditimbulkan dari tujuan-tujuan tersebut. Orientasi dalam fase ini adalah menyelesaikan masalah-masalah perusahaan.

2. *RAD Design Workshop* (*Workshop* Desain RAD)

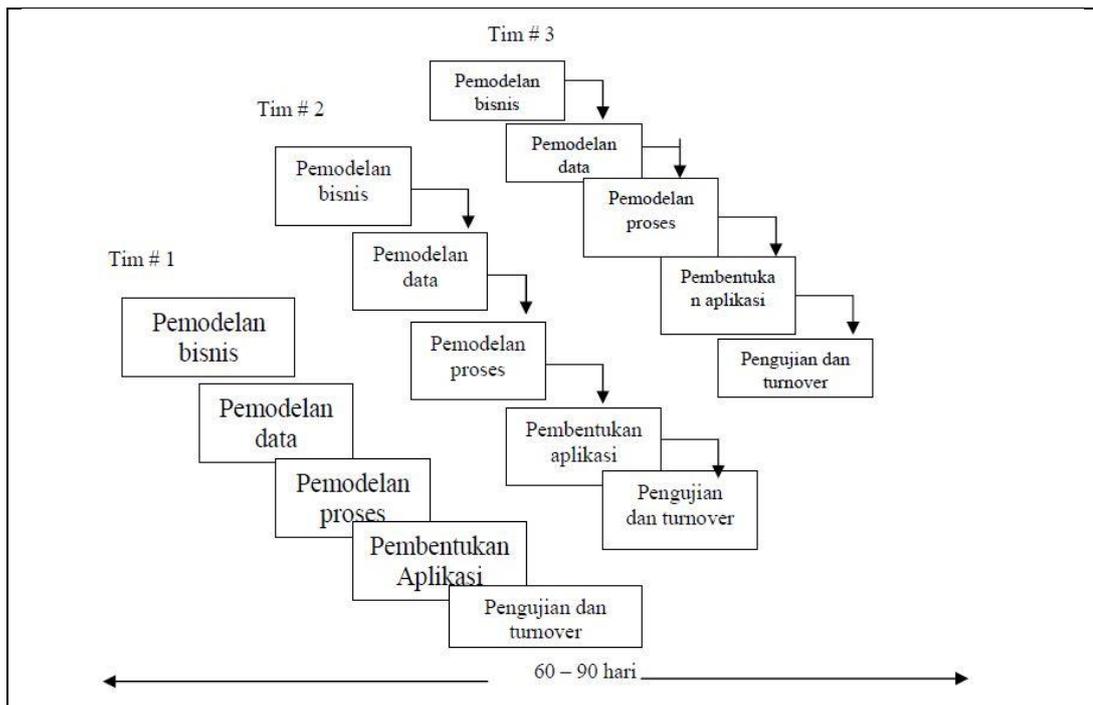
Fase ini adalah fase untuk merancang dan memperbaiki yang bisa digambarkan sebagai *workshop*. Penganalisis dan pemrogram dapat bekerja membangun dan menunjukkan representasi visual desain dan pola kerja kepada pengguna. *Workshop* desain ini dapat dilakukan selama beberapa hari tergantung dari ukuran aplikasi yang akan dikembangkan. Selama *workshop* desain RAD, pengguna merespon prototipe yang ada dan

penganalisis memperbaiki modul-modul yang dirancang berdasarkan respon pengguna. Apabila seorang pengembangnya merupakan pengembang atau pengguna yang berpengalaman, Kendall menilai bahwa usaha kreatif ini dapat mendorong pengembangan sampai pada tingkat terakselerasi.

3. *Implementation* (Implementasi)

Pada fase implementasi ini, penganalisis bekerja dengan para pengguna secara intens selama *workshop* dan merancang aspek-aspek bisnis dan nonteknis perusahaan. Segera setelah aspek-aspek ini disetujui dan sistem-sistem dibangun dan disaring, sistem-sistem baru atau bagian dari sistem diujicoba dan kemudian diperkenalkan kepada organisasi.

Berikut adalah gambar model *RAD*:



Gambar 1.1 Model *RAD*

Sumber : Kendall (2010)

1. Pemodelan Bisnis

Pemodelan yang dilakukan untuk memodelkan fungsi bisnis untuk mengetahui informasi apa yang terkait proses bisnis, informasi apa saja

yang harus dibuat, siapa yang harus membuat informasi itu, bagaimana alur informasi itu, proses apa saja yang terkait informasi itu.

2. Pemodelan Data

Memodelkan data apa saja yang dibutuhkan berdasarkan pemodelan bisnis dan mendefinisikan atribut-atributnya beserta relasinya dengan data-data yang lain.

3. Pemodelan Proses

Mengimplementasikan fungsi bisnis yang sudah didefinisikan terkait dengan pendefinisian data.

4. Pembentukan Aplikasi

Mengimplementasikan pemodelan proses dan data menjadi program. Model *RAD* sangat menganjurkan pemakaian komponen yang sudah ada jika dimungkinkan.

5. Pengujian dan Turnover

Menguji komponen-komponen yang dibuat. Jika sudah teruji maka tim pengembang komponen dapat beranjak untuk mengembangkan komponen berikutnya.

Berikut ini adalah beberapa kelebihan *Rapid Application Development* menurut (Marakas, 2006):

1. Penghematan waktu dalam keseluruhan fase proyek dapat dicapai.
2. Mengurangi seluruh kebutuhan yang berkaitan dengan biaya proyek dan sumberdaya manusia.
3. Sangat membantu pengembangan aplikasi yang berfokus pada waktu penyelesaian proyek.
4. Perubahan desain sistem dapat lebih berpengaruh dengan cepat dibandingkan dengan pendekatan *SDLC* tradisional.
5. Sudut pandang user disajikan dalam sistem akhir baik melalui fungsi-fungsi sistem atau antarmuka pengguna.
6. *RAD* menciptakan rasa kepemilikan yang kuat di antara seluruh pemangku kebijakan proyek.

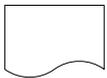
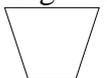
2.2.2 Flowchart

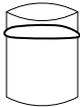
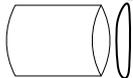
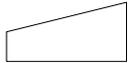
Bagan alir (*Flowchart*) adalah bagan (*Chart*) yang menunjukkan alir (*Flow*) di dalam program atau prosedur sistem secara logika. Bagan alir digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi serta pada waktu akan menggambarkan suatu bagan alir (Jogiyanto, 2005:795). Ada lima macam bagan alir yakni terdiri dari bagan alir sistem (*Systems flowchart*), bagan alir dokumen (*Document flowchart*), bagan alir skematik (*Scematic flowchart*), bagan alir program (*Program Flowchart*), bagan alir proses (*Process flowchart*) yang saya gunakan dalam penelitian ini adalah bagan alir sistem (*Systems flowchart*).

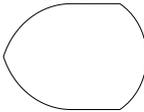
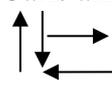
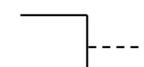
Bagan alir sistem (*Systems flowchart*) merupakan bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem. Bagan ini menjelaskan urutan-urutan dari prosedur-prosedur yang ada didalam sistem, bagan alir sistem menunjukan apa yang dikerjakan di sistem, bagan alir sistem digambar dengan menggunakan simbol-simbol (Jogiyanto, 2005:796).

Berikut simbol bagan alir sistem (*systems flowchart*) dapat dilihat pada Tabel 2.1 :

Tabel 2.1 Simbol *Systems Flowchart*

No	Simbol	Keterangan
1	Dokumen 	Menunjukkan dokumen <i>input</i> dan <i>output</i> baik proses manual, mekanil atau <i>computer</i>
2	Kegiatan Manual 	Menunjukkan pekerjaan manual
3	Simpanan Offline 	File non-komputer yang diarsipurut angka (<i>numerical</i>)
4	Simpanan Offline 	File non-komputer yang diarsipurut angka (<i>alphabetical</i>)
5	Simpanan Offline 	File non-komputer yang diarsipurut angka (<i>cronological</i>)

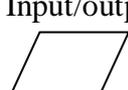
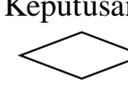
6	Kartu plong 	Menunjukkan <i>input/output</i> yang menggunakan kartu plong
7	Proses 	Menunjukkan proses dari operasi program komputer
8	Operasi luar 	Menunjukkan operasi yang dilakukan diluar proses operasi komputer
9	Pengurutan offline 	Menunjukkan proses pengurutan data diluar proses komputer
10	Pita magnetik 	Menunjukkan <i>input/ output</i> menggunakan pita magnetik
11	Hard disk 	Menunjukkan <i>input/ output</i> menggunakan hard disk
12	Diskette 	Menunjukkan <i>input/ output</i> menggunakan diskette
13	Drum magnetik 	Menunjukkan <i>input/ output</i> menggunakan Drum magnetic
14	Pita kertas berlubang 	Menunjukkan <i>input /output</i> menggunakan Pita kertas berlubang
15	Keyboard 	Menunjukkan <i>input/ output</i> menggunakan on-line keyboard

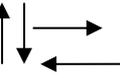
16	<p>Display</p> 	Menunjukkan output yang tampil di komputer
17	<p>Pita kontrol</p> 	Menunjukkan penggunaan pita kontrol dalam <i>batch control total</i> untuk pencocokan di proses <i>batch processing</i>
18	<p>Hubungan komunikasi</p> 	Menunjukkan proses transmisi data melalui channel komunikasi
19	<p>Garis alir</p> 	Menunjukkan arus proses
20	<p>Penjelasan</p> 	Penjelasan dari suatu proses
21	<p>Penghubung</p> 	Menunjukkan penghubung ke halaman yang masi sama atau ke halaman lain

Sumber: Jogiyanto (2005:796)

Bagan alir program (*program flowchart*) merupakan bagan yang menjelaskan secara rinci langkah-langkah dari proses program (Jogiyanto, 2005:802). Berikut simbol bagan alir program (*program flowchart*) dapat dilihat pada Tabel 2.2:

Tabel 2.2 Simbol Program Flowchart

No	Simbol	Keterangan	Simbol	Keterangan
1.	<p>Input/output</p> 	Simbol input / output digunakan untuk mewakili data input output	<p>Keputusan</p> 	Simbol keputusan digunakan untuk suatu penyeleksian kondisi di dalam program

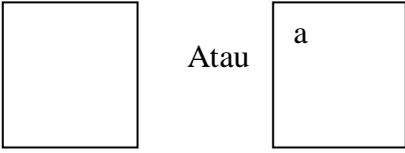
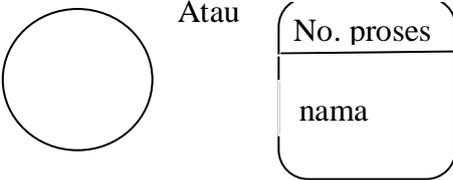
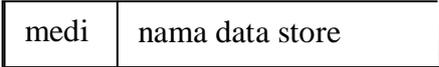
2.	Proses 	Simbol proses digunakan untuk mewakili proses	Proses terdefinisi 	Simbol proses terdefinisi digunakan untuk menunjukkan suatu operasi yang rinciannya ditunjukkan di tempat lain
3.	Garis alir 	Simbol garis alir (<i>flow lines simbol</i>) digunakan untuk menunjukkan arus dari proses	persiapan 	Simbol persiapan digunakan untuk member nilai awal suatu besaran
4.	Penghubung 	Simbol Penghubung menunjukkan sambungan dari bagan alir yang terputus di halaman yang masih sama atau di halaman lainnya.	Titik terminal 	Titik terminal digunakan untuk menunjukkan awal dan akhir dari suatu proses

Sumber: Jogiyanto (2005:802)

2.2.2 DFD (*Data Flow Diagram*)

Diagram yang menggunakan notasi-notasi ini untuk menggambarkan arus dari data sistem sekarang dikenal dengan nama diagram arus data (*data flow diagram* atau DFD). DFD merupakan alat yang digunakan pada metodologi pengembangan sistem yang terstruktur (*structured Analysis and design*). DFD merupakan alat yang cukup populer sekarang ini, karena dapat menggambarkan arus data di dalam sistem dengan terstruktur dan jelas. Lebih lanjut DFD juga merupakan dokumentasi dari sistem yang baik (Jogiyanto, 2005:700). Berikut simbol digunakan di DFD untuk maksud mewakili dapat dilihat pada Tabel 2.3:

Tabel 2.3 Simbol *Data Flow Diagram*

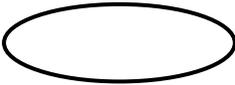
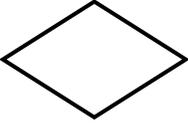
No	Simbol	Keterangan
1.		<i>External entity</i> (kesatuan luar) : sistem akan menerima input dan menghasilkan output kepada lingkungan luarnya.
2.		<i>Data flow</i> (arus data) : menunjukkan arus dari data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem
3.		<i>Process</i> (proses) : kegiatan atau kerja yang dilakukan oleh orang, mesin atau komputer dari hasil suatu arus data yang masuk kedalam proses untuk dihasilkan arus data yang akan keluar dari proses
4.		<i>Data store</i> (simpanan data) : merupakan simpanan dari data

Sumber: Jogiyanto (2005:700-707)

2.2.3 ERD (*Entity Relational Diagram*)

ERD adalah notasi yang digunakan untuk melakukan aktivitas pemodelan data (Pressman, 2014:353). ERD adalah gambar atau diagram yang menunjukkan informasi dibuat, disimpan dan digunakan dalam sistem bisnis (Fatta, 2007:121). Berikut simbol-simbol ERD (Fatta, 2007:124):

Tabel 2.4 Simbol *Entity Relational Diagram*

No	Simbol	Keterangan
1.		Entitas : Orang, tempat, atau benda memiliki nama tunggal
2.		Attribut : Property dari entitas harus digunakan oleh minimal 1 proses bisnis dipecah dalam detail
3.		Relationship: Menunjukkan hubungan antar 2 entitas, dideskripsikan dengan kata kerja.

Sumber: Fatta (2007:124)

2.2 Teori Program

2.3.1 Pengertian HTML (*Hypertext Markup Language*)

Kaban (2019:4), HTML merupakan singkatan dari *Hyper Text Markup Language* yaitu bahasa standar web yang dikelola penggunaannya oleh W3C (*World Wide Web Consortim*) berupa tag-tag yang menyusun setiap elemen dari website.

Hidayatullah dan kawistara (2017:15), Hyper Text Markup Language (HTML) adalah bahasa standard yang di gunakan untuk menampilkan halaman web. Sedangkan menurut Naista (2016:2), HyperText Markup Language atau lebih singkatnya HTML adalah bahasa markup yang digunakan untuk membuat sebuah halaman web.

Dari pendapat tersebut dapat disimpulkan bahwa HTML (*Hyper Text Markup Language*) adalah bahasa komputer yang digunakan untuk membuat dan menampilkan halaman web.

2.3.2 Pengertian PHP (*Hypertext Preprocessor*)

Winarno, dkk. (2013:1) menjelaskan, *PHP* adalah bahasa pemrograman web bersifat *server side*, yang tujuannya untuk menghasilkan skrip yang akan di-generate dalam kode *HTML* yang merupakan bahasa standar web.

Menurut Kaban (2019:127), PHP yaitu bahasa pemograman web yang dapat disisipkan dalam skrip HTML dan bekerja disisi server. Sedangkan Jubilee Enterprise (2017:1) berpendapat, *PHP* merupakan bahasa pemrograman berjenis *server-side* yang digunakan untuk membuat aplikasi berbasis *website*.

Berdasarkan penjelasan diatas maka php adalah bahasa pemrograman dengan skrip HTML yang bekerja disisi server.

Menurut Sutaji (2012: 2), aturan penulisan script PHP adalah sebagai berikut:

- a. *Embedded Script*, yaitu dengan cara meletakkan tag PHP diantara tag-tag HTML. Contohnya:

```
<html>
<body>
<?php echo "Belajar"; ?>
</body>
</html>
```

- b. *Non Embedded Script*, yaitu dengan cara semua script HTML diletakkan dalam *script* PHP. Contohnya :

```
<?php
Echo "<html>";
Echo "<body>";
Echo "<?php echo "Belajar"; ?> ";
Echo "</body>";
Echo "</html>";
```

PHP memiliki variabel yang digunakan sebagai tempat penyimpanan data sementara. Data tersebut akan hilang setelah program selesai dieksekusi. Berikut aturan penggunaan nama variable (Sutaji, 2012: 3):

1. Diawali dengan karakter \$.
2. Bersifat case sensitive, jadi \$Var berbeda dengan \$var atau \$VAR.
3. Karakter pertama harus berupa huruf atau garis bawah (_).
4. Karakter berikutnya boleh huruf, angka atau garis bawah (_).
5. Berikut adalah beberapa contoh penulisan dan pendeklarasian variabel:
 - a) \$namadepan = "Ali Shodikin";
 - b) \$harga_barang = 3500;
 - c) \$nilai3 = 3.45;

2.3.3 Pengertian MySQL

Solichin (2010:8) menjelaskan, *MySQL* adalah sebuah perangkat lunak sistem manajemen basis data *SQL* atau DBMS yang *multi thread, multi user*, dengan sekitar 6 juta instalasi diseluruh dunia. sedangkan Wahana Komputer (2010:5) juga berpendapat bahwa *MySQL* adalah program database yang mampu mengirim dan menerima data dengan sangat cepat dan *multi user*.

Penulis menyimpulkan pengertian *MySQL* berdasarkan pendapat tersebut yaitu perangkat lunak database yang memungkinkan pengiriman dan penerimaan data dengan cepat.

SQL dibagi menjadi tiga bentuk query, yaitu (Sutaji, 2012: 44) :

a. DDL (*Data Definition Language*)

DDL adalah sebuah Metode Query SQL yang berguna untuk mendefinisikan data pada sebuah database. Berikut *query* yang termasuk DDL:

1. *CREATE* digunakan untuk melakukan pembuatan tabel dan database.
2. *DROP* digunakan untuk melakukan penghapusan tabel maupun database.
3. *ALTER* digunakan untuk melakukan perubahan struktur tabel yang telah dibuat, baik menambah Field (*add*), mengganti nama Field (*change*) ataupun menamakannya kembali (*rename*), serta menghapus (*drop*).

b. DML (*Data Manipulation Language*)

DML adalah sebuah metode query yang dapat digunakan apabila DDL telah terjadi, sehingga fungsi dari query ini adalah untuk melakukan pemanipulasian database yang telah ada atau yang telah dibuat sebelumnya.

Berikut query yang termasuk DML:

1. *SELECT* digunakan untuk menampilkan data pada tabel.
2. *INSERT* digunakan untuk melakukan penginputan pemasukan data pada tabel database.
3. *UPDATE* digunakan untuk melakukan perubahan atau peremajaan terhadap data yang ada pada tabel.
4. *DELETE* digunakan untuk melakukan penghapusan data pada tabel.

c. DCL (*Data Control Language*)

DCL adalah sebuah metode Query yang dapat digunakan untuk mengendalikan eksekusi perintah. Biasanya berhubungan dengan pengaturan hak akses. Berikut query yang termasuk DCL:

1. *GRANT* : Digunakan untuk memberikan hak akses (*privilage*) kepada user tertentu.
2. *REVOKE* : Digunakan untuk mencabut hak akses dari user tertentu.

2.3.4 Pengertian phpMyAdmin

Menurut Firdaus (2007:15), *PhpMyAdmin* adalah suatu program *open source* yang berbasis web yang dibuat menggunakan aplikasi PHP. Program ini digunakan untuk mengakses database *MySQL*. Program ini mempermudah dan mempersingkat kerja penggunanya. Dengan kelebihanannya, para pengguna awam tidak harus paham sintak-sintak *SQL* dalam pembuatan database dan table.

Menurut Haqi dan Heri (2019), *PhpMyAdmin* adalah perangkat lunak bebas yang ditulis dalam bahasa pemrograman PHP yang digunakan untuk menangani administrasi *MySQL* melalui *World Wide Web* (WWW).

Berdasarkan penjelasan diatas maka penulis menyimpulkan bahwa *PhpMyAdmin* adalah perangkat lunak *open source* yang digunakan untuk menggunakan *MySQL*.

2.3.5 Pengertian Xampp

Komputer Wahana (2014:72), XAMPP merupakan singkatan dari X (empat sistem operasi apapun), Apache, *MySQL*, PHP, dan Perl. XAMPP adalah tool yang menyediakan paket perangkat lunak dalam satu buah paket. Dalam paket XAMPP sudah terdapat Apache (*web server*), *MySQL* (*database*), PHP (*server side scripting*), Perl, FTP Server, *PhpMyAdmin* dan berbagai pustaka bantu lainnya.

Menurut Haqi dan Heri (2019), XAMPP adalah perangkat lunak bebas (*free software*) yang mendukung banyak system operasi. Merupakan kompilasi dari berbagai program.

Dari pendapat tersebut dapat disimpulkan bahwa XAMPP adalah salah satu paket installer yang berisi Apache yang merupakan web server tempat

menyimpan file-file yang dibutuhkan website, dan PhpMyAdmin sebagai aplikasi yang digunakan untuk perancangan database MySQL.

2.3.6 Pengertian *Bootstrap*

Menurut Kaban (2019:261), *Bootstrap* merupakan salah satu *framework* CSS yang paling populer dari sekian banyak *framework* CSS yang ada. *Bootstrap* memungkinkan desain sebuah web menjadi reponsif sehingga dapat dilihat dari berbagai macam ukuran device dengan tampilan tetap menarik.

Menurut Alatas (2013), *Bootstrap* merupakan *framework* untuk membangun desain web secara resposif, artinya tampilan web yang dibuat oleh *bootstrap* akan menyesuaikan ukuran *layer* dan *browser* yang kita gunakan baik di desktop, tablet maupun *mobile device*. Dengan *bootstrap* kita juga bisa membangun web dinamis ataupun statis.

Berdasarkan pengertian yang dikemukakan diatas dapat ditarik kesimpulan bahwa *Bootstrap* merupakan *framework* yang digunakan untuk membangun sebuah web dengan tampilan yang menyesuaikan *browser*.

2.3.7 Pengertian *JavaScript*

Menurut Sigit (2011:1), *JavaScript* merupakan bahasa *Scripting* yang bekerja disisi *Client/Browser* sehingga *website* bisa lebih interaktif.

Menurut Flanagan (2011:1), *JavaScript* adalah bahasa pemrograman web. Mayoritas situs web modern menggunakan *JavaScript*, termasuk semua *browser web* modern di desktop, *game consoles*, *tablet*, dan *smartphone*.

Dari beberapa pengertian diatas dapat disimpulkan bahwa *JavaScript* adalah bahas pemrograman berbasis browser dengan kode yang langsung ditulis ke dalam HTML.

2.3 Referensi Penelitian Terdahulu

Penelitian ini menggunkan 5 (lima) jurnal dari penelitian terdahulu yang dilakukann oleh beberapa orang yaitu :

Berdasarkan penelitian yang dilakukan oleh Nasrun Marpaung pada tahun 2018 dengan judul Penerapan Metode *Simple Additive Weighting* pada Sistem Pendukung Keputusan untuk Menentukan Kenaikan Gaji Karyawan, Penelitian ini



menghasilkan suatu data sebagai solusi alternative keputusan dalam penentuan kenaikan gaji karyawan pada AKBID As-Syifa Kisaran menggunakan metode *Simple Additive Weighting* (SAW).

Pada jurnal yang di lakukan oleh Muhamad Meky Frindo pada tahun 2018 dengan judul Sistem Penunjang Keputusan Dalam Evaluasi Kinerja Karyawan Menggunakan Metode SAW pada PT.SIERAD PRODUCE, Hasil penghitungan dalam penelitian ini penerapan metode SAW untuk evaluasi kinerja karyawan dengan menggunakan data sampling sebanyak 50 alternatif dan kriteria sebanyak lima kriteria metode SAW dapat digunakan untuk mengevaluasi kinerja karyawan.

Menurut jurnal penelitian dengan judul Penerapan Metode *Simple Additive Weighting* (SAW) Dalam Sistem Pendukung Keputusan Promosi Kenaikan Jabatan pada tahun 2016 oleh Frieyadie, penelitian ini terdapat tiga kriteria yang meliputi masa kerja, penilaian kinerja, dan perilaku. Kriteria tersebut di gunakan untuk perhitungan metode *Simple Additive Weighting* (SAW) untuk mengukur dan menilai siapa yang akan terseleksi untuk mendapatkan promosi kenaikan jabatan.

Berdasarkan jurnal penelitian Petrus Sokibi, Apriyanto Noer Setiawan pada tahun 2018 yang berjudul Sistem Pendukung Keputusan Penilaian Kinerja Karyawan PT Harjamukti Jaya Mandiri Menggunakan Metode *Simple Additive Weighting*, penelitian ini bertujuan untuk mempermudah divisi Staff Manager PT. Harjamukti Jaya Mandiri untuk menilai karyawan dengan kinerja yang baik secara adil, realistis, valid dan relevan.

Lain halnya penelitian yang dilakukan oleh Tri Yani Akhirina, Sultan Mohammad Arif pada tahun 2017 dalam jurnal yang berjudul Analisis Sistem Pendukung Keputusan Kenaikan Gaji Menggunakan Metode *Simple Additive Weighting* (SAW) pada PT XXX, Tujuan dari sistem yang dibangun untuk memudahkan pihak manajemen dalam menentukan persentase kenaikan gaji sesuai dengan tingkat prestasi kerja tenaga kerja yang bersangkutan dengan menggunakan sistem terkomputersisasi maka proses pengambilan keputusan dapat lebih obyektif dan dapat memberikan hasil yang lebih akurat.