

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

Adapun penelitian terkait yang dijadikan sebagai referensi dalam penelitian ini adalah jurnal yang membahas tentang studi kasus atau yang berkenaan dengan penelitian:

Berdasarkan penelitian Fendy S, 2020 ISSN 2548-8368. Perkembangan teknologi saat ini, dalam kegiatan pencarian informasi dan pengambilan keputusan di Fakultas Teknologi Komunikasi dan Informatika Universitas Nasional belum efektif. Karena banyaknya data yang ada menyulitkan pengguna untuk menemukan data khusus yang dibutuhkan. Seperti mencari data mahasiswa berdasarkan angkatan, agama dan jenis kelamin. Oleh karena itu, dirancanglah Sistem Informasi Eksekutif Fakultas Teknologi Komunikasi dan Informatika Universitas Nasional berbasis *web* sehingga pengguna bisa menemukan data yang dibutuhkan. Hasil uji coba aplikasi untuk mencari data mahasiswa, sistem dapat menampilkan hasil pencarian berdasarkan angkatan, agama dan jenis kelamin dengan tingkat keakuratan 100%. Dalam pencarian data dosen sistem dapat menampilkan hasil pencarian berdasarkan pendidikan terakhir, keahlian dan penelitian dengan tingkat keakuratan 100%. Kemudian untuk pencarian data tenaga pengajar sistem dapat menampilkan hasil pencarian berdasarkan pendidikan terakhir dan keahlian dengan tingkat keakuratan 100%. Sistem juga dapat menampilkan jumlah mahasiswa

berdasarkan angkatan, jumlah dosen berdasarkan keahlian dan jumlah tenaga kependidikan berdasarkan pendidikan terakhir dalam bentuk grafik.

Penelitian dari Himas A, 2020 ISSN 2722-130X. Dalam menjalankan proses bisnisnya, PT Samudera Sarana Logistik memiliki KPI sebagai standarnya. Pada PT Samudera Sarana Logistik terdapat dua jenis KPI, antara lain Logistik (penjualan & persewaan *container office*), dan Pemeliharaan & Perbaikan. Dalam mengolah proses Logistik dan pemeliharaan & perbaikan, perusahaan ini telah menggunakan sebuah aplikasi yang bernama COINS (*Container Information System*), serta telah menggunakan *database MySQL* untuk menyimpan data-datanya. Namun belum ada sistem untuk meninjau hasil dari proses pelayanan pemeliharaan & perbaikan. Hal ini menyebabkan pihak eksekutif kesusahan dalam mengambil keputusan untuk meningkatkan pelayanan pemeliharaan & perbaikan tersebut. Oleh karena itu, dibutuhkan suatu Sistem Informasi Eksekutif (SIE) yang terdapat fitur *drill-down* dan analisa *what-if* berbasis *web* yang responsif untuk dapat membantu pihak eksekutif mengakses SIE di manapun dan kapan pun. Selain itu, dalam SIE ini juga terdapat fitur *activity monitoring* yang dapat memantau aktifitas bongkar, muat, dan reparasi secara *real-time*.

Penelitian dari Agung B. 2019 ISSN **2686-0775**. Komoditas teh adalah salah satu unit usaha yang membuat PTPN IV memiliki keunikan dibanding perkebunan lain di Sumatera Utara. Walaupun hanya memiliki 3 kebun teh, namun PTPN IV tetap diperhitungkan sebagai salah satu produsen teh yang kompetitif di Indonesia. Hal ini menjadi sebuah kebanggaan sekaligus tantangan. Untuk mendukung kinerja membuat rencana strategis dan

keputusan yang obyektif sebagai pemimpin eksekutif perlu didukung oleh sistem yang mampu memberikan informasi spesifik. Sistem informasi eksekutif adalah alternatif yang tepat untuk memecahkan masalah. Produk akhir dari pengembangan sistem ini adalah perangkat lunak sistem informasi eksekutif yang kemudian dipakai oleh pemimpin eksekutif dalam menganalisis data.

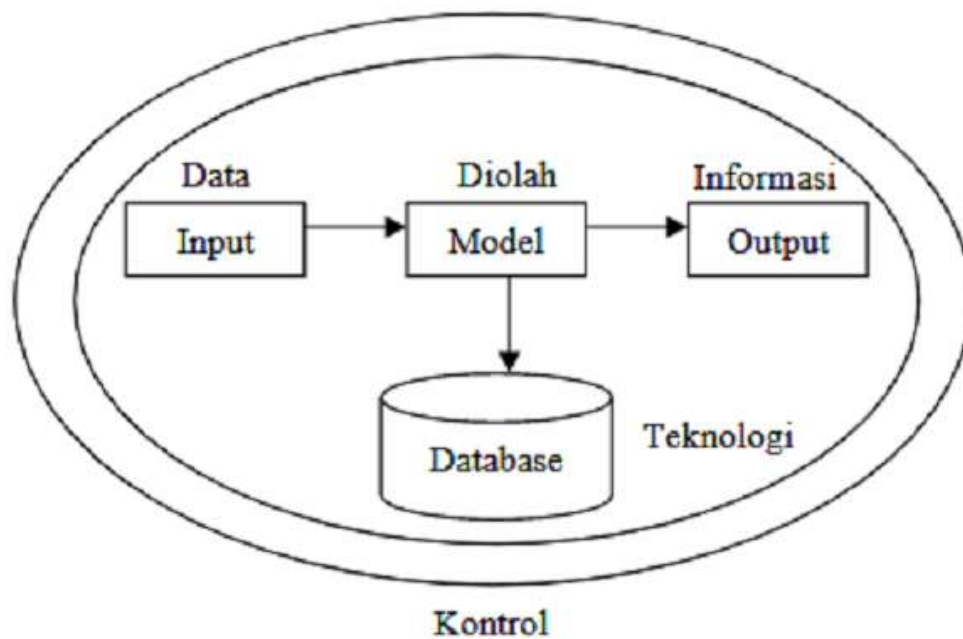
II.2. Landasan Teori

II.2.1. Sistem Informasi

Sistem informasi merupakan gabungan dari orang, data, proses, penyajian informasi dan teknologi informasi yang satu dan lainnya saling berinteraksi dan berintegrasi untuk mendukung operasi dalam bisnis guna mengatasi dan membuat keputusan untuk pihak manajemen dan user. Tujuan dari sistem informasi itu sendiri adalah menghasilkan suatu informasi yang berguna bagi para pemakainya.

Sistem Informasi mempunyai enam buah komponen yaitu:

1. Input: Komponen input merupakan data yang masuk dalam sistem informasi.
2. Output: Komponen output berupa informasi yang berguna bagi pemakainya.
3. Database: Kumpulan dari data yang saling berhubungan satu dengan yang lainnya.
4. Model: Bagaimana informasi diolah. Model yang diolah berupa model logika atau matematik.
5. Teknologi: Sarana yang digunakan untuk mengolah dan menyampaikan informasi.
6. Kontrol: Dengan komponen kontrol, Informasi yang dihasilkan harus mempunyai sifat yang akurat dan meminimalkan kesalahan.



Gambar II.1. Komponen Sistem Informasi

(Sumber: Fauzi O, 2022: 2)

II.2.2. Sistem Informasi Eksekutif

Sistem Informasi Eksekutif (*Executive Information System* atau EIS) terkadang disebut sebagai sistem pendukung eksekutif (*Executive Support System* atau ESS). Sistem ini merupakan sistem informasi yang menyediakan fasilitas yang fleksibel bagi manajer dan eksekutif dalam mengakses informasi eksternal dan internal yang berguna untuk mengidentifikasi masalah atau mengenali peluang. Untuk memenuhi fungsinya, SIE memiliki karakteristik sebagai berikut :

1. Dibuat secara spesifik untuk seorang eksekutif.
2. Akan digunakan oleh manajemen secara langsung.
3. Mampu melakukan proses ekstraksi, penyaringan, menyingkat dan melacak data.

4. Mengakses dan mengintegrasikan data internal dan eksternal.
5. Bersifat User Friendly atau mudah digunakan. (Syamsul B; 2018; 151).

Sistem informasi eksekutif sendiri merupakan ipse sistem informasi berbasis komputer yang ditujukan untuk memfasilitasi kebutuhan informasi yang berkaitan dengan tercapainya tujuan suatu organisasi bagi eksekutif. Dalam membangun SIE digunakan metode rekayasa siklus hidup sistem SIE yang terdiri dari beberapa tahapan. Adapun gambarnya dapat dilihat dibawah ini:



Gambar II.2. Siklus Sistem Informasi Eksekutif

Aristektur SIE terdiri dari 3 level [1], yaitu:

1. *Data Management*, yaitu data yang disimpan dalam *database*, data *warehouse*, dan lain-lain;
2. *Model Management, level* yang memuat proses pengambilan data dari sumber luar lainnya di luar *database* yang sudah ada dan mentransformasikannya untuk dimasukkan ke dalam *database* yang

digunakan dalam SIE. Proses ini dikenal dengan nama *Extract, Transform, Load* (ETL).

3. Visualisasi Data, yaitu penampilan informasi baik berbentuk grafik maupun narasi yang digunakan oleh eksekutif untuk mengambil keputusan. (Nyoman I, 2022: 235)

II.2.3. Penyaluran Barang / Distribusi

Distribusi atau penyaluran barang merupakan aktivitas prioritas bagi perusahaan yang bergerak di bidang distributor, sebagai distributor maka perusahaan sangat bergantung ada total persediaan atau stok barang yang akan di distribusikan ke pelanggan, seiring dengan distribusi dan teknologi yang semakin meningkat, maka kepuasan konsumen harus diperhatikan sehingga distributor di haruskan untuk mendistribusikan barang dengan baik untuk mengantisipasi kelebihan persediaan dan tidak terjadinya kekurangan stok yang dapat berdampak pada transaksi antara perusahaan ke konsumen. Salah satu metode yang digunakan untuk perencanaan distribusi barang adalah *Distribution Requirement Planning* (DRP). *DRP* sebagai sebuah sistem yang menentukan permintaan untuk persediaan pada pusat-pusat distribusi, menggabungkan permintaan historis dan sebagai input untuk sistem produksi dan material. *DRP* adalah metode yang dapat membantu perusahaan dalam mengatasi kendala aktivitas pendistribusian produk, dengan menggunakan metode *DRP* perusahaan dapat mengetahui jumlah permintaan konsumen, menentukan jumlah barang optimal yang akan di distribusikan pada konsumen dan waktu pemesanan barang yang tepat, sehingga diharapkan dengan adanya penjadwalan aktivitas distribusi yang baik dapat

mengoptimalkan biaya distribusi perusahaan dan meningkatkan kualitas pelayanan perusahaan dalam memenuhi kebutuhan konsumen dengan tepat waktu, tepat jumlah dan tepat mutu untuk meningkatkan keuntungan dan loyalitas konsumen perusahaan. (Rizky C, 2019: 341)

II.2.4. UML

Unified Modeling Language (UML) adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. *UML* merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem *Unified Modeling Language (UML)* adalah sebuah bahasa yang berdasarkan grafik atau gambar untuk memvisualisasi, menspesifikasikan, membangun, dan pendokumentasian dari sebuah sistem pengembangan *software* berbasis OO (*Object-Oriented*). *UML* sendiri juga memberikan standar penulisan sebuah sistem *blue print*, yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik, skema *database*, dan komponen-komponen yang diperlukan dalam sistem *software*. (Suendri, 2018: 2)

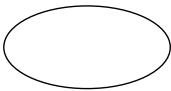
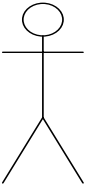
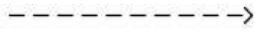
Diagram *Unified Modelling Language (UML)* antara lain sebagai berikut:

a. *Usecase Diagram*

Usecase diagram merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Usecase* mendeskripsikan sebuah interaksi antara satu kata atau lebih *actor* dengan sistem informasi yang akan dibuat. Dapat dikatakan *usecase* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-

fungsi tersebut. Berikut adalah simbol-simbol yang ada pada *Use Case Diagram*:

Tabel II.1. Use Case Diagram

NO	Simbol	Nama	Keterangan
1.		<i>Case</i>	Gambaran dari interaksi antara aktor dengan sistem
2.		<i>Actor</i>	Pengguna yang berinteraksi Dengan sistem. Aktor ditunjukkan dengan nama perannya dalam system
3.		<i>Association</i>	Menggambarkan interaksi aktor dan use case secara langsung.
4.		<i>Extend</i>	Menggambarkan fungsi sebuah use case yang dapat diperluas oleh use case lain, jika dibutuhkan
5.		<i>Generalization</i>	Relasi antara suatu aktor dengan aktor turunannya.
6.		<i>Uses / Include</i>	Menggambarkan fungsi dari sebuah use case yang hanya dapat dipenuhi dengan bantuan dari use case yang lainnya

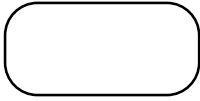
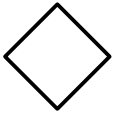
(Sumber : M Gilvy, 2021:573)

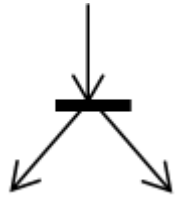
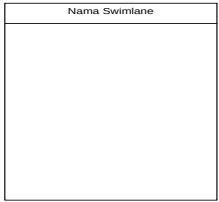
b. Diagram Aktivitas (*Activity Diagram*)

Activity diagram adalah teknik untuk menggambarkan logika *procedural* proses bisnis dan aliran kerja dalam banyak kasus. *Activity diagram* mempunyai peran seperti halnya *flowchart* akan tetapi perbedaannya dengan *flowchart*

adalah *activity diagram* bisa mendukung perilaku paralel sedangkan *flowchart* tidak. *Activity diagram* menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. *Activity diagram* merupakan *state diagram* khusus, dimana sebagian besar *state* adalah *action* dan sebagian besar transisi di-trigger oleh selesainya *state* sebelumnya (*internal processing*). Oleh karena itu *activity diagram* tidak menggambarkan *behaviour internal* sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum. (Minda S, 2018: 25) Berikut simbol-simbol yang sering digunakan dalam *activity diagram*:

Tabel II.2. Activity Diagram

NO	Simbol	Nama	Keterangan
1.		<i>Initial State</i>	<i>Initial State</i> aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2.		Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3.		<i>Decision</i>	<i>Decision</i> ialah percabangan dimana jika ada pilihan aktivitas lebih dari satu.

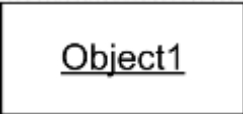
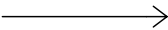
4.		<i>Fork</i>	Fork/percabangan, digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu
5.		<i>Join</i>	Asosiasi penggambungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
5.		<i>Final State</i>	<i>Final State</i> yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah akhir.
6.		<i>Swimlane</i>	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

(Sumber : Fitri A, 2018:27)

c. Diagram Urutan (*Sequence Diagram*)

Sequence Diagram menggambarkan kelakuan objek pada *usecase* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Berikut simbol-simbol yang sering digunakan dalam *sequence diagram*:

Tabel II.3. *Sequence Diagram*

NO	Simbol	Nama	Keterangan
1.		Aktor	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi. Jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan kata benda di awal frase nama aktor.
2.		<i>Lifeline</i>	Menyatakan kehidupan suatu objek.
3.		<i>Object</i>	Menyatakan objek yang berinteraksi dengan pesan.
4.		<i>Activation</i>	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya.
5.		<i>Message</i>	Menyatakan suatu objek membuat objek lain, arah panah mengarah pada objek yang dibuat.
6.		<i>Message Call</i>	Menyatakan suatu objek memanggil operasi / metode yang ada pada objek lain atau dirinya sendiri.

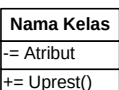
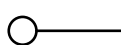
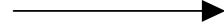
7.		<i>Recursive</i>	<i>Recursive</i> , menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.
----	---	------------------	---

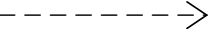
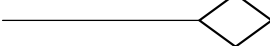
(Sumber : Fitri A, 2018:17)

d. Diagram Kelas (*Class Diagram*)

Class diagram adalah suatu diagram yang dapat menggambarkan variabel atau atribut suatu kelas pada sistem beserta metode yang dapat dilakukan oleh kelas tersebut. Metode (methods) merupakan proses sistem dalam mengelola data yang dimiliki sistem tersebut. (M Gilvy, 2021: 574) Berikut simbol-simbol yang sering digunakan dalam *class diagram*:

Tabel II.4. Simbol-Simbol *Class Diagram*

NO	Simbol	Nama	Keterangan
1.		Kelas	Kelas pada struktur sistem.
2.		<i>Interface</i>	Sama dengan konsep <i>interface</i> dalam pemograman berorientasi objek.
3.		<i>Association</i>	Relasi antarkelas dengan makna umum, asosiasi biayanya juga disertai dengan <i>multiplicity</i> .
4.		<i>directed association</i>	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai

			dengan <i>multiplicity</i> .
5.		<i>Generalization</i>	Relasi antarkelas dengan makna generalisasi spesialisasi (umum-khusus).
6.		<i>dependecy</i>	Relasi antarkelas dengan makna kebergantungan antarkelas.
7.		<i>aggregation</i>	Relasi antarkelas dengan makna semua bagian (<i>whole-part</i>).

(Sumber : M Gilvy, 2021:574)

II.2.5. *Dreamweaver*

Dreamweaver adalah suatu bentuk program editor *web* yang dibuat oleh macromedia”. Dengan program ini seorang programmer *web* dapat dengan mudah membuat dan mendesain *webnya*. *Dreamweaver* adalah editor yang komplit yang dapat digunakan untuk membuat animasi sederhana yang berbentuk layer. Dengan adanya program ini akan memudahkan mengetik *script-script* format *HTML*, *PHP*, *ASP* maupun bentuk program yang lainnya. (Joko S, 2019: 120).

Adobe Dreamweaver adalah sebuah *Hyper Text Markup Language* (*HTML*) editor professional untuk mendesain secara visual dan mengelola situs *website* beserta halamannya. *HTML* merupakan bahasa yang digunakan untuk menulis halaman *website*, dirancang untuk tidak tergantung pada suatu sistem operasi tertentu. Sementara pendapat lain menyatakan bahwa *adobe Dreamweaver* adalah sebuah program untuk membuat atau mengedit sebuah *website* yang dibuat oleh *Adobe System*. Kegunaan dari *Adobe Dreamweaver CS6* dapat mendesain

secara visual dan mengelola Web. WYSIWIG (*What You See Is What You Get*) HTML (*Hyper Text Mark-Up Language*) akan membuat halaman web dan website secara mudah dan cepat. Berdasarkan beberapa pendapat para ahli yang dikemukakan di atas dapat ditarik kesimpulan bahwa *Adobe Dreamweaver CS6* merupakan sebuah HTML editor professional untuk mendesain secara visual dan mengelola Web dan membuat halaman web dan website secara mudah dan cepat. (Hardiyanto A, 2020:6)

II.2.6. **PHP**

Server-side yang bersifat *open source*. *PHP* merupakan *script* yang terintegrasi dengan *HTML* dan berada pada *server* (*server-side HTML embedded scripting*). *PHP* adalah *script* yang digunakan untuk membuat halaman website yang dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat saat halaman itu diminta oleh *client*. Mekanisme ini menyebabkan informasi yang diterima *client* selalu yang terbaru *server-side* yang bersifat *open source*. *PHP* merupakan *script* yang terintegrasi dengan *HTML* dan berada pada *server* (*server-side HTML embedded scripting*). *PHP* adalah *script* yang digunakan untuk membuat Halaman website yang dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat saat halaman itu diminta oleh *client*. Mekanisme ini menyebabkan informasi yang diterima *client* selalu yang terbaru. (Didi S, 2020: 67).

PHP adalah salah satu bahasan pemrograman skrip yang dirancang untuk membangun aplikasi web. Ketika dipanggil dari web browser, program yang ditulis dengan PHP akan di-*parsing* di dalam web server oleh *interpreter* PHP dan diterjemahkan ke dalam dokumen HTML, yang selanjutnya akan

ditampilkan kembali *web server*. Karena pemrosesan program PHP dilakukan didalam lingkungan *web browser*, PHP dikatakan sebagai bahasa sisi *server* (*server-side*). Oleh sebab itu, seperti yang telah dikemukakan sebelumnya, kode PHP tidak akan terlihat pada saat *user* memilih perintah “View Source” pada *web browser* yang mereka gunakan. Penulisan scrip PHP diawali dengan tanda lebih kecil (<) dan diakhiri tanda lebih besar (>). Ada cara untuk menuliskan script PHP, yaitu:

```
<? Scrip_php ?>
```

```
<? Php Scrip_php?>
```

```
<% Script_php %>
```

```
<Script laguange= “ php” > Script_php</Script>
```

Pemisah antara instruksi adalah titik koma (;) dan untuk membuat atau menambahkan komentar penulisannya adalah: /* komentar */, # komentar # komentar. Cara penulisan dibedakan menjadi Embeddedscript dan non-Embedded script.

1) Embedded Script

Contoh:

```
<html>
```

```
<head>
```

```
<title> PHP dengan Embeddcd Script </title>
```

```
</head>
```

```
<body>
```

```
</html>
```

2) Non-Embedded Script

Contoh:

```
<?php
```

```
echo"<html>";
```

```
echo"<head>";
```

```
echo"<title> PHP dengan Embedded Script
```

```
</title>";
```

```
echo "<body>";
```

```
echo"<p> Web master </P>";
```

```
echo"</body>";
```

```
echo"</html>";
```

(sumber : Saed M, 2019: 47)

II.2.7. *MySql*

SQL (Structured Query Language) adalah sebuah bahasa *scripting* yang dipergunakan untuk mengolah *database*. *Database* besar seperti *Mysql*, *PostgreSQL* dan *SQL Server* sudah menggunakan *SQL* untuk mengolah *database* nya. *SQL* yang di pergunakan software *database* tersebut adalah sama kecuali sedikit perbedaan di beberapa tempat. Pada tugas akhir ini penulis menggunakan *MySQL* sebagai *platform* pengoperasian *SQL* ini. *MySQL* adalah salah satu jenis *database server* yang sangat terkenal. *MySQL* menggunakan bahasa *SQL* untuk mengakses *database* nya. Lisensi *Mysql* adalah *FOSS License Exception* dan ada

juga yang versi komersial nya. Tag *Mysql* adalah “*The World's most popular open source database*”. *MySQL* tersedia untuk beberapa *platform*, di antaranya adalah untuk versi *windows* dan versi *linux*. Untuk melakukan administrasi secara lebih mudah terhadap *Mysql*, anda dapat menggunakan *software* tertentu, di antara nya adalah *phpmyadmin* dan *mysql* (Rizky F, 2020; 130).