

BAB II TINJAUAN PUSTAKA

1.1 Tinjauan Pustaka

Berdasarkan bahan acuan dalam menulis laporan akhir, referensi menjadi bagian penting untuk memperlancar penulisan karya ilmiah. Referensi adalah sebuah informasi yang dijadikan sebagai rujukan atau acuan untuk mempertegas suatu pernyataan dalam penulisan karya ilmiah, seperti pada Tabel 2. 1.

Tabel 2. 1 Referensi Jurnal

NO	JUDUL JURNAL	PENULIS	TAHUN TERBIT	PENERBIT	LINK
1	Perancangan Sistem Informasi Manajemen Berbasis <i>Web</i> Studi Kasus: PT UNICORN INTERTRANZ	Ghavinkson Enstayn Abednego Kustanto, Hanna Prillysca Chernovita	2021	Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK)	https://jtiik.ub.ac.id
2	Aplikasi SIMADE (Sistem Informasi Manajemen Desa) Dalam Meningkatkan Pelayanan Administrasi di Kepenghuluan	Volvo Sihombing	2018	Jurnal Sistemasi	https://sistemasi.ftik.unisi.ac.id

	Bakti Makmur Kecamatan Bagan Sinembah Kab. Rokan Hilir Riau				
3	Analisis Implementasi Sistem Informasi Desa Terintegrasi di Kabupaten Pangandaran	Asep Nurdin Rosihan Anwar, Ii Sujai	2020	Jurnal Moderat	https://jurnal.unigal.ac.id
4	Analisis dan Perancangan Sistem Informasi Manajemen Penerima Bantuan Dana Covid – 19 Berbasis Dashboard (Study Kasus : Kantor Desa Rangai)	Wiji Asrurin, Jupriyadi	2021	Jurnal Teknologi dan Sistem Informasi (JTSI)	https://teknorat.ac.id
5	Analisis dan Perancangan Sistem Informasi Pendataan	M. Irsyad Rahmadani, Effiyaldi.	2019	Jurnal Manajemen Sistem Informasi	https://stikom-db.ac.id

	Keluarga Penerima Manfaat (KPM) Bantuan Pangan Non Tunai (BPNT) Pada Dinas Sosial Kota Jambi				
--	---	--	--	--	--

1.2 Landasan Teori

1.2.1 Aplikasi

Aplikasi adalah sebuah perangkat lunak yang berisi sebuah *coding* atau perintah yang dimana bisa diubah sesuai dengan keinginan (Syani, 2019).

Aplikasi adalah sebuah perangkat lunak yang dimana tujuannya adalah agar bisa melayani setiap aktivitas komputerisasi yang dilakukan oleh pengguna.

Aplikasi juga diartikan sebagai penggunaan atau penerapan suatu konsep yang menjadi pokok pembahasan atau sebagai program komputer yang dibuat untuk menolong manusia dalam melaksanakan tugas tertentu. Aplikasi *software* yang dirancang untuk penggunaan praktisi khusus, klasifikasi luas ini dapat dibagi menjadi 2 (dua) yaitu:

1. Aplikasi *software* spesialis, program dengan dokumentasi tergabung yang dirancang untuk menjalankan tugas tertentu.
2. Aplikasi paket, suatu program dengan dokumentasi tergabung yang dirancang untuk jenis masalah tertentu (Sari, 2017).

Dari kedua pengertian di atas dapat disimpulkan bahwa aplikasi adalah sekumpulan perintah atau kode yang disusun secara sistematis untuk menjalankan suatu perintah yang diberikan oleh manusia melalui komponen atau *hardware* komputer yang digunakan oleh manusia dalam menjalankan program aplikasi,

dengan demikian bisa membantu manusia untuk memberikan solusi dari apa yang diinginkan.

1.2.2 Sistem Informasi Manajemen

McLeod dan Schell dalam (Mulyani S, 2017) mengatakan *Management Information System* adalah sebuah sistem yang sudah terkomputerisasi yang melakukan pengolahan data agar bisa digunakan oleh orang yang membutuhkannya. Sistem informasi manajemen merupakan komputerisasi proses bisnis sebab adanya interaksi antar manusia dan komputer. Sistem informasi manajemen memiliki cakupan fungsi yang cukup luas termasuk dalam menganalisis data dan risiko hingga pengambilan keputusan.

1.2.3 Website

Website adalah kumpulan halaman-halaman. Yang digunakan untuk menampilkan informasi teks gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman (Arkhiansyah, 2018).

1.2.4 HTML

Kustiyahningsih dan Anamisa dalam (Sasongko, 2017), menjelaskan bahwa HTML kependekan dari *Hyper Text Markup Language* yang *file* teks 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 dikomputer *client (user)* sehingga isi formasinya dapat ditampilkan secara visual di komputer pengguna (*user*).

Hidayatulah dan Kawistara dalam (Ayu, dkk, 2018), menjelaskan tentang HTML sebagai berikut :

Hypertext Markup Language (HTML) adalah bahasa standard yang digunakan untuk menampilkan halaman *web*. Yang bisa dilakukan dengan HTML yaitu:

1. Mengatur tampilan dan halaman *web* dan isinya.
2. Membuat tabel dalam halaman *web*.
3. Mempublikasikan dalam halaman *web* secara *online*.
4. Membuat form yang bisa digunakan untuk menangani registrasi dan transaksi via *web*.

Dari penjelasan di atas dapat disimpulkan bahwa *Hyper Text Markup Language* (HTML) adalah bahasa standar yang merupakan teks murni yang berisi instruksi yang diterjemahkan oleh *browser* dan ditampilkan secara visual di komputer pengguna.

1.2.5 CSS

Menurut (Adi, 2020), CSS adalah kependekan dari *Cascading Style Sheet*. CSS merupakan salah satu kode pemrograman yang digunakan untuk mengatur tampilan dan menghiasi halaman *website* supaya menarik.

1.2.6 Bootstrap

Menurut (Kaban, 2019), 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.

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

1.2.7 Framework Codeigniter

Framework secara sederhana dapat diartikan kumpulan dari fungsi-fungsi/prosedur-prosedur dan *class-class* untuk tujuan tertentu yang sudah siap digunakan sehingga bisa lebih mempermudah dan mempercepat pekerjaan seorang pemrograman, tanpa harus membuat fungsi atau class dari awal.

Codeigniter adalah aplikasi *open source* berupa *framework* dengan model MVC (*Model, View, Controller*) untuk membangun *website* dinamis dengan 8 meggunakan PHP seperti pada Gambar 2. 1. *Codeigniter* memudahkan *developer* atau pengembang *web* untuk membuat aplikasi *web* dengan cepat dan mudah dibandingkan dengan membuat dari awal (Supono, 2018).



Gambar 2. 1 *Codeigniter*

1.2.8 Microsoft Visual Studio Code

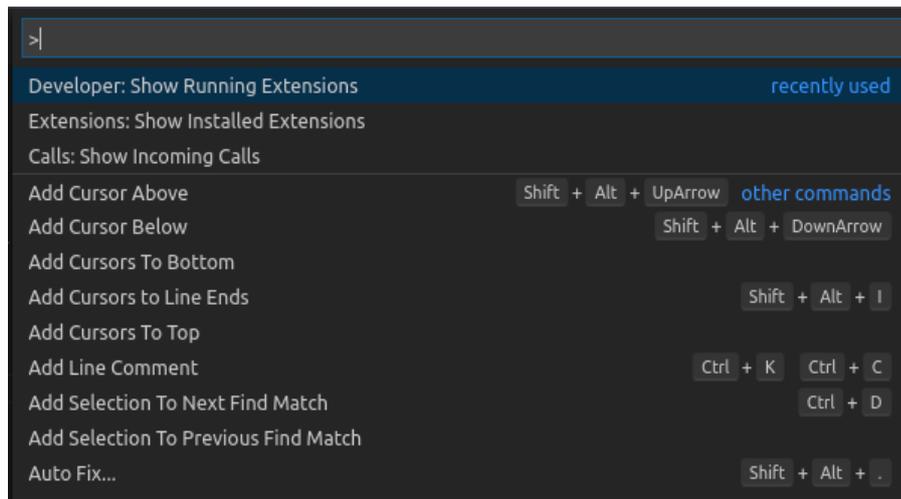
2.2.8.1 Definisi Microsoft Visual Studio Code

Visual Studio Code adalah kode *editor* yang ringan namun kuat yang berjalan di *desktop* Anda dan tersedia untuk Windows, macOS, dan Linux. Muncul dengan dukungan built-in untuk JavaScript, TypeScript dan Node.js dan memiliki ekosistem ekstensi yang kaya untuk bahasa lain (seperti C ++, C #, Java, Python, PHP, Go) dan *runtime* (seperti .NET dan Unity) (Visual Studio Code, 2021).

2.2.8.2 Struktur Text Editor Microsoft Visual Studio Code

1. *Command Palette*

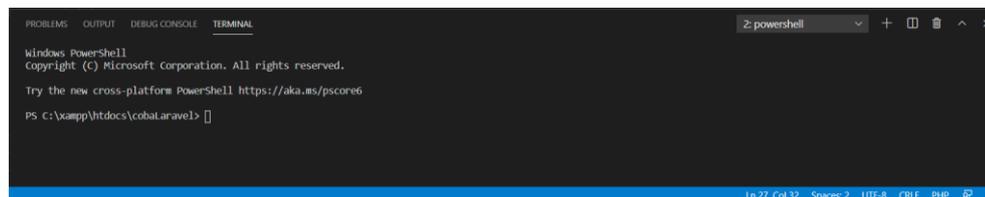
Command Palette menyediakan banyak akses perintah. Kita bisa memberikan perintah editor, membuka file, mencari file dan sebagainya dengan cepat dan mudah, Untuk membuka *Command Palette* tekan kombinasi **Ctrl+Shift+P** seperti pada Gambar 2. 2.



Gambar 2. 2 Tampilan Jendela Kerja Standar Microsoft Visual Studio

2. *Integrated Terminal*

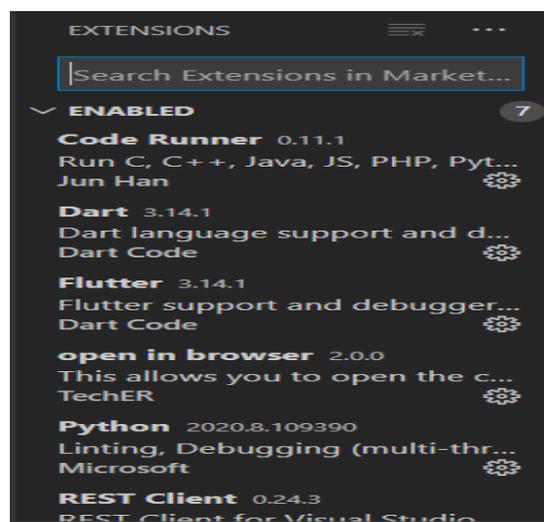
Salah satu kelebihan *Visual Studio Code* adalah kita bisa langsung mengeksekusi *script* di editor secara langsung di terminal tanpa harus membuka terminal tambahan, Terminal komputer ialah peranti perkakasan elektronik atau peranti perkakasan elektromekanik yang digunakan untuk memasukkan data ke dalam komputer (atau sistem pengkomputan), dengan data itu dipaparkan pada skrin monitor. Ia merupakan salah satu contoh antara muka manusia-mesin (HMI) seperti pada Gambar 2. 3.



Gambar 2. 3 Tampilan Terminal Microsoft Visual Studio

3. Extensions

Ekstensi adalah fungsi tambahan dalam yang berfungsi untuk memperluas kemampuan dari *editor* yang sdapat membantu pengembang dalam melakukan *programming*, Di paragraf awal sudah diberikan contoh sekilas kelebihan ekstensi untuk membantu pengguna. Untuk membuka fitur *extensions* tekan tombol *extensions* di sidebar kiri dan lakukan pencarian ekstensi yang diinginkan seperti pada Gambar 2. 4.

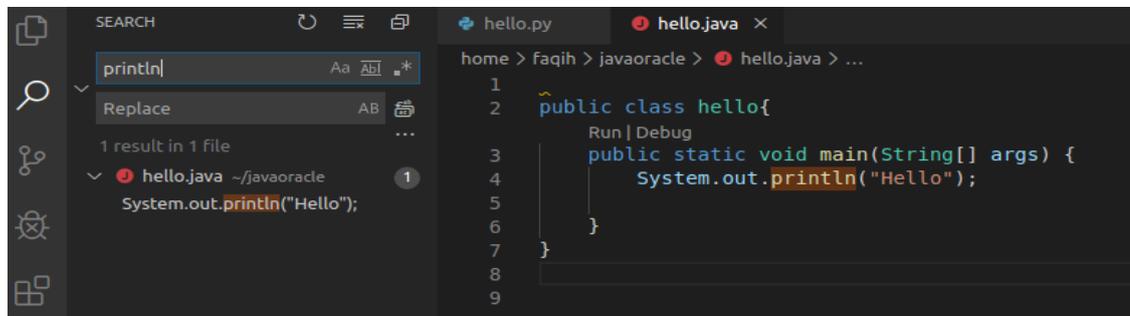


Gambar 2. 4 Tampilan *Extensions* atau Fitur Tambahan Microsoft Visual Studio

4. Search

Fitur *search Visual Studio Code* juga luar biasa cepatnya. Kemudahan yang diberikan selain kecepatan *query* pencarian data juga dia akan mencari sampai di level kontennya, misal saya membuat 2 file dengan nama `hello.py` dan `hello.java` dan ingin melakukan pencarian kode `println` di java, hasilnya, selain teks `println` di `hello.java` di *highlight*, di hasil pencarian juga ditampilkan kata `println` dan ditemukan di file apa saja beserta isinya. `hello.java` di *highlight*, di hasil pencarian

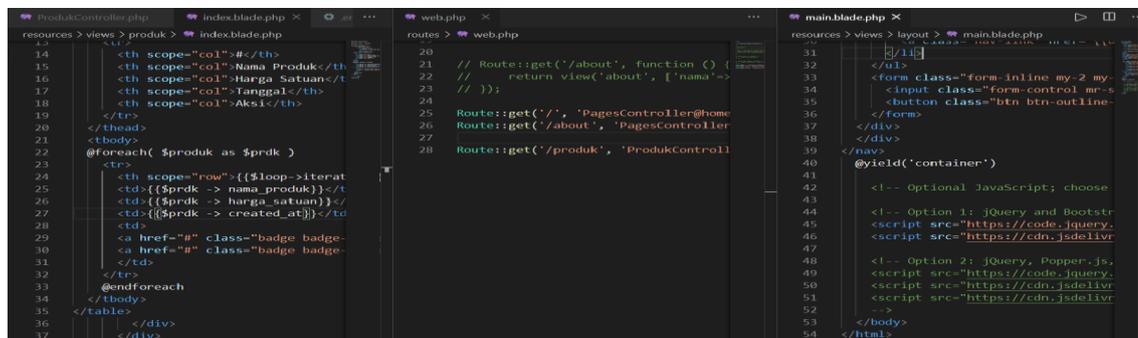
juga ditampilkan kata *println* dan ditemukan di file apa saja beserta isinya seperti pada Gambar 2. 5.



Gambar 2. 5 Tampilan *Tool Search* Microsoft Visual Studio

5. Grid Editor Layout

Kita juga mudah dalam manajemen *layout* Visual Studio Code. Kita dapat dengan mudah mengatur grup *editor* dalam tata letak apapun baik secara vertikal maupun horizontal, untuk masuk ke menu *Editor Layout*, langsung klik *View – Editor Layout* seperti pada Gambar 2. 6.

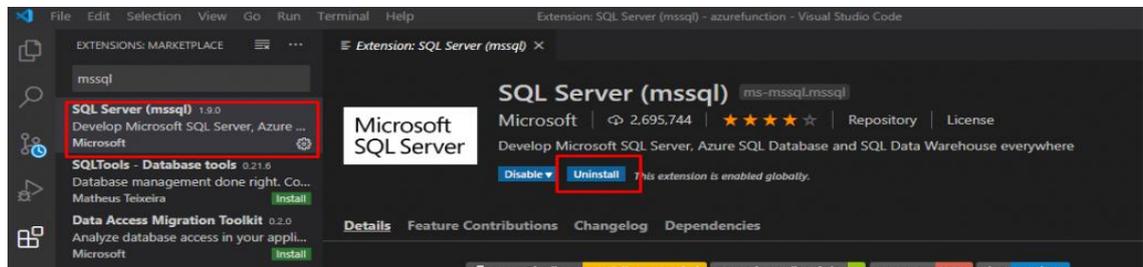


Gambar 2. 6 Tampilan *Grid Editor* Microsoft Visual Studio

6. Cloud Environment

Kita juga bisa melakukan merasakan lingkungan *cloud* melalui *Visual Studio Code* seperti membuat database, melakukan perintah *insert*, *update*, *delete* dan sebagainya di *cloud*, ini adalah contoh – contoh instalasi ekstensi untuk

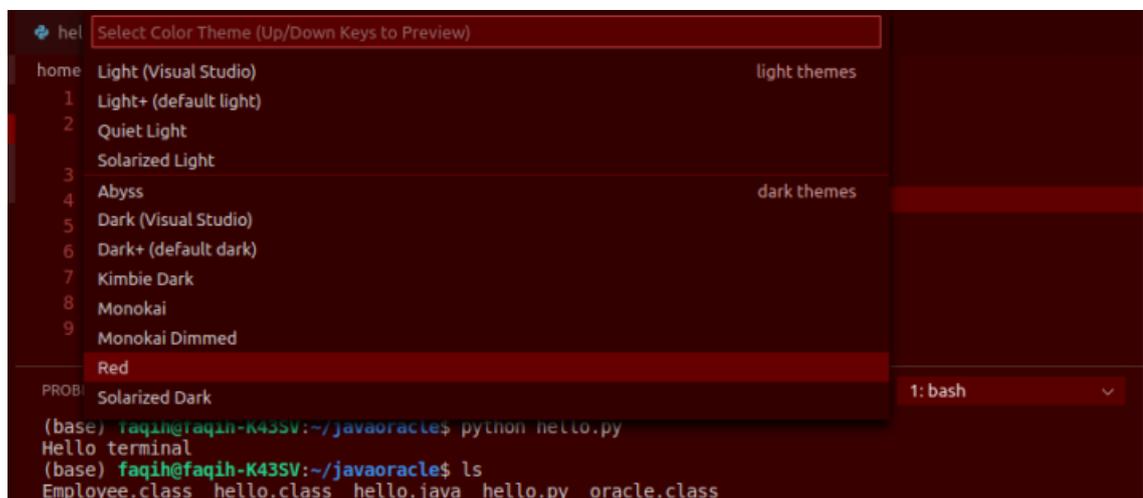
menghubungkan aplikasi Visual Studio Code dengan *platform Azure SQL Database* seperti pada Gambar 2. 7.



Gambar 2. 7 Tampilan *Cloud Environment* Microsoft Visual Studio

7. Color Theme

Color Themes memungkinkan kita untuk memodifikasi warna dalam antarmuka Visual Studio Code agar sesuai dengan selera, pilih menu **File > Preferences > Color Theme** selanjutnya cukup geser kursor keatas dan kebawah dengan keyboard untuk memilih tema yang diinginkan seperti pada Gambar 2. 8.



Gambar 2. 8 Tampilan *Setting Color Theme* Microsoft Visual Studio

2.2.9 PHP

Menurut Madcoms dalam (Ayu, dkk, 2018), PHP (*Hypertext Preprocessor*) adalah bahasa *script* yang dapat ditanamkan atau disisipkan ke dalam HTML. PHP banyak dipakai untuk membuat program situs *web* dinamis”.

PHP (*Hypertext Processor*) adalah bahasa pemrograman dengan skrip yang sederhana yang dapat ditanamkan, disisipkan atau digunakan ke dalam HTML. PHP biasanya digunakan untuk membuat *web* dinamis.

Nugroho dalam (Lavarino, 2016), menjelaskan tentang beberapa kelebihan dari PHP sebagai berikut :

1. Bahasa pemrograman PHP adalah sebuah bahasa *script* yang tidak melakukan sebuah kompilasi dalam penggunaannya.
2. PHP memiliki tingkat akses yang lebih cepat.
3. PHP memiliki tingkat *lifecycle* yang cepat sehingga selalu mengikuti perkembangan teknologi internet.

PHP juga mendukung akses ke beberapa *database* yang sudah ada baik yang bersifat *free/gratis* ataupun komersial. *Database* itu antara lain : MySQL, PostgreSQL, infomix, dan MicrosoftSQL Server. *Web server* yang mendukung PHP dapat ditemukan dimana mana dari mulai Apache, IIS, AOServer, phttp, Fhttp. PWS, Lighttpd hingga Xitami dengan konfigurasi yang relatif mudah.

2.2.10 MySQL

MySQL adalah RDBMS yang didistribusikan secara gratis di bawah lisensi GPL. Dimana setiap orang bebas untuk menggunakan MySQL, namun tidak boleh dijadikan produk turunan yang bersifat komersial. Karena merupakan turunan salah satu konsep utama dalam *database* sejak lama, yaitu SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoperasian *database*, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis. Keandalan suatu sistem *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 *database server* lainnya dalam *query* data. Hal ini terbukti untuk *query* yang dilakukan oleh *single user*, kecepatan *query MySQL* bisa sepuluh kali lebih cepat dari *PostgreSQL* dan lima kali lebih cepat dibandingkan *Interbase* (Standsyah, 2017). Berikut merupakan logo dari *MySQL*, dapat dilihat pada gambar 2. 9.



Gambar 2. 9 Logo MySQL

2.2.11 XAMPP

Menurut (Enterprise, 2018), “XAMPP merupakan *server* yang paling banyak digunakan untuk keperluan belajar PHP secara mandiri, terutama bagi *programmer* pemula” seperti pada Gambar 2. 10.

Menurut (Iqbal, 2019), “Xampp merupakan sebuah *software web server* apache yang didalamnya sudah tersedia *database server* MySQL dan support PHP *programming*”.

Berdasarkan pengertian diatas dapat disimpulkan bahwa XAMPP adalah perangkat lunak, yang mendukung banyak sistem operasi, dan memiliki fungsi sebagai *server* yang berdiri sendiri (*localhost*), yang terdiri atas program Apache HTTP Server, MySQL *database*, PHP dan Perl.



Gambar 2. 10 Tampilan XAMPP

2.2.12 *PHPMysqlAdmin*

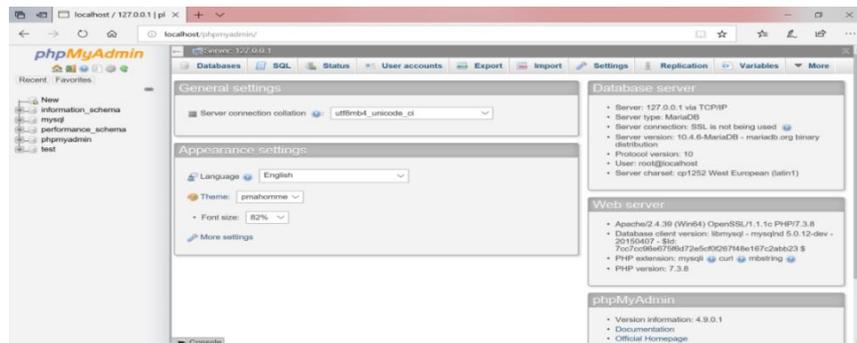
Dalam beberapa kasus, mempelajari cara mengelola *database* menggunakan *PhpMyAdmin* memiliki kelebihan. Sebagai contoh, banyak *web hosting* yang menyediakan panel kontrol dengan *interface PhpMyAdmin*. *PhpMyAdmin* adalah sebuah aplikasi atau perangkat lunak bebas (*opensource*) yang ditulis dalam bahasa pemrograman PHP yang digunakan untuk menangani administrasi *database MySQL* melalui jaringan lokal maupun internet (Standisyah, 2017). Aplikasi mendukung berbagai operasi *MySQL*, diantaranya mengelola basis data, tabel-tabel, bidang (*fields*), relasi (*relations*), indeks, pengguna (*users*), perijinan (*permissions*), dan lain-lain. Berikut merupakan logo dari *PhpMyAdmin*, seperti pada gambar 2. 11.



Gambar 2. 11 Logo PhpMyAdmin

1. Menu utama PhpMyAdmin

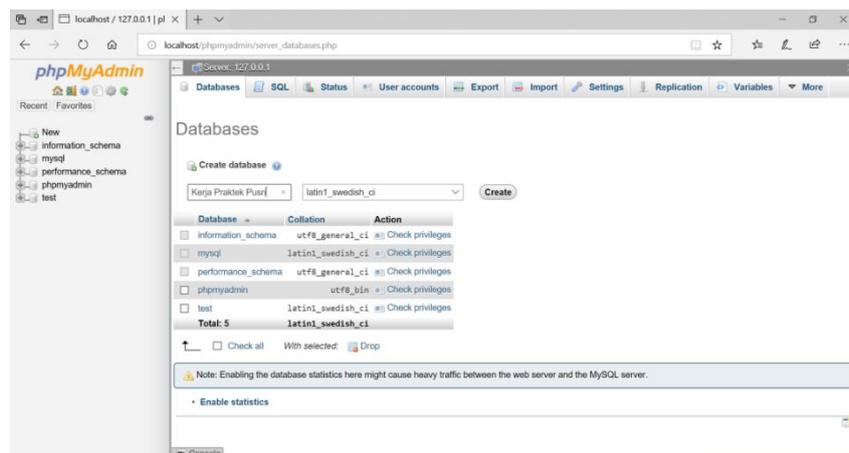
Menu utama *PhpMyAdmin* untuk membuat sebuah *database* seperti pada Gambar 2. 12.



Gambar 2. 12 Tampilan Menu Utama PhpMyAdmin

2. Membuat Database MySQL dengan PhpMyAdmin

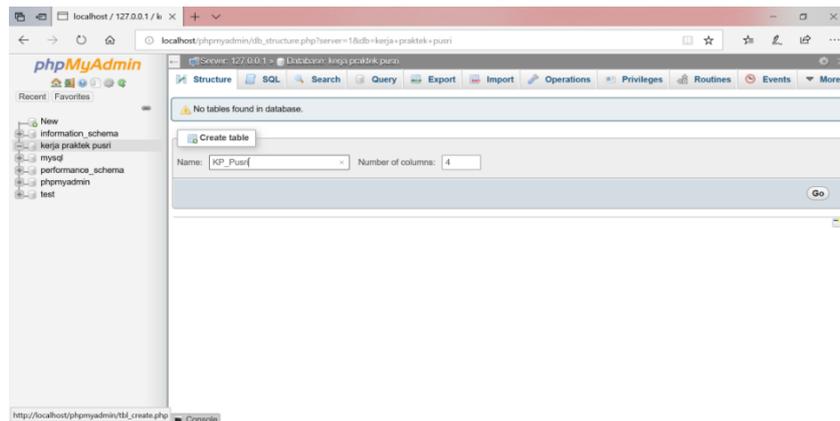
Untuk membuat *database* di *MySQL* akan lebih mudah jika kita menggunakan *PhpMyAdmin*, pada *browser* ketikkan <http://localhost/phpmyadmin/> yaitu menggunakan *PhpMyAdmin*. Ketika pertama kali membuat *database MySQL* dengan *PhpMyAdmin* beri nama *database* terlebih dahulu, kemudian klik *Create* seperti pada Gambar 2. 13.



Gambar 2. 13 Tampilan Halaman Pembuatan Database

3. Membuat *Table*

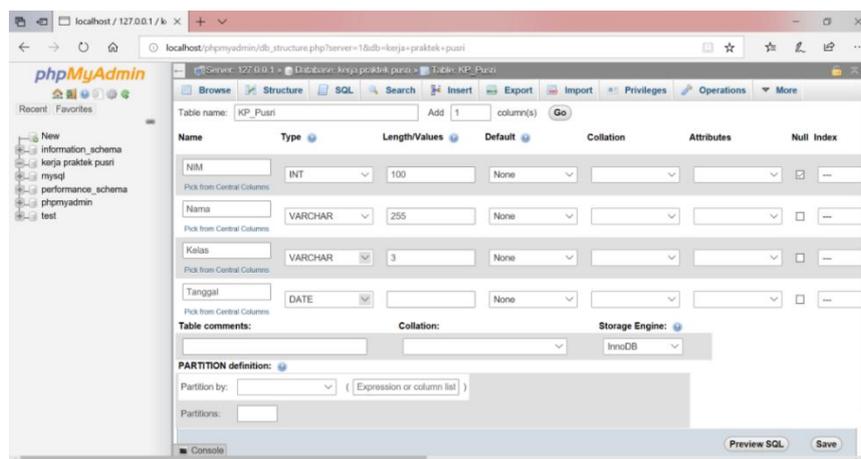
Setelah kita *create* atau membuat *database*, maka akan muncul halaman baru untuk membuat atau *create table*. Ketikkan nama *table* kemudian klik *Go* seperti pada Gambar 2. 14.



Gambar 2. 14 Tampilan Halaman Pembuatan Tabel

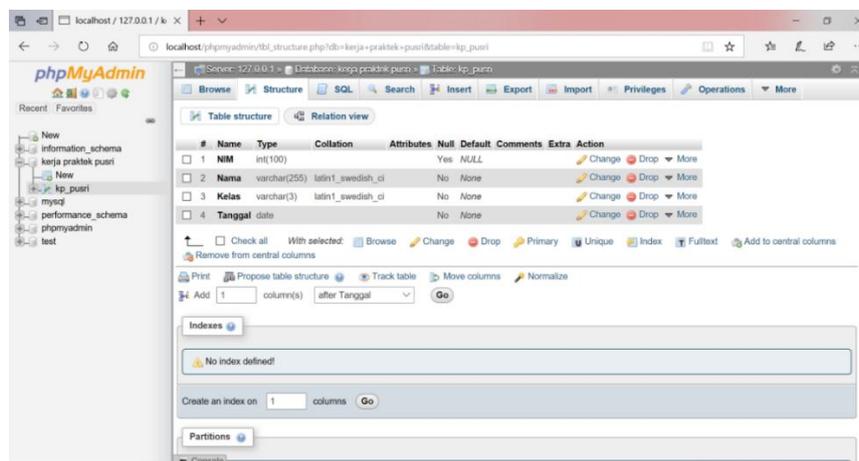
4. Membuat *Field*

Isi kolom atau *field* pada *table* yang telah dibuat sebelumnya, masukkan *Name*, *Type*, *Values/Length* sesuai kebutuhan, beri tanda pada *Null* sebagai *Primary Key* pada *Name* yang dipilih, lalu Klik *Save* seperti pada Gambar 2. 15.



Gambar 2. 15 Tampilan Halaman Pembuatan *Field* pada *Database*

Setelah berhasil *Save*, maka akan muncul halaman baru untuk melihat hasil *database* yang telah dibuat, untuk melihat *record* lalu klik *browse* seperti pada Gambar 2. 16.



Gambar 2. 16 Tampilan Halaman setelah di *Browse*

2.2.13 Database

Menurut (Abdulloh, 2018), *Database* atau basis data adalah kumpulan informasi yang disimpan dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi.

Menurut (Enterprise, 2017), *Database* adalah suatu aplikasi yang menyimpan sekumpulan data. Setiap *database* mempunyai perintah tertentu untuk membuat, mengakses, mengatur, mencari, dan menyalin data yang ada di dalamnya.

Maka dari dua pendapat tersebut dapat disimpulkan bahwa *database* atau basis data adalah kumpulan informasi yang tersimpan di dalam komputer secara sistematis dan dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut.

2.2.14 Data Tables

Data Tables berfungsi untuk menghasilkan tabel data yang dinamis, dimana data dapat langsung diurutkan berdasarkan kolom, selain itu dengan *Data Tables*

juga menyediakan formulir pencarian yang langsung mencari data semua kolom yang tampil tanpa perlu melakukan *query* dari *database* terlebih dahulu (Anonymous, 2018).

2.2.15 FPDF

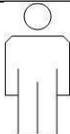
FPDF merupakan suatu kumpulan *class (library)* PHP yang memungkinkan penggunaanya dapat membuat/generate file PDF secara langsung dari PHP (*on-the-fly*) tanpa harus meng*install* atau mengaktifkan modul PHP tertentu (seperti PDFLib. **F** dari nama FPDF merupakan singkatan dari **Free**, jadi FPDF dapat di *download*, digunakan dan dikembangkan secara bebas (Fahmi, 2013).

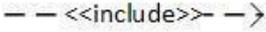
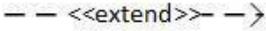
2.2.16 Use Case Diagram

Menurut Tohari dalam (Tabrani, 2019) *use case diagram* adalah rangkaian atau uraian sekelompok yang saling terkait dan membentuk sistem secara teratur yang dilakukan atau diawasi oleh sebuah aktor. Sedangkan menurut (Kurniawan, 2020) *use case diagram* merupakan diagram yang bekerja dengan cara mendeskripsikan tipikal interaksi antara *user* (pengguna) sebuah sistem dengan suatu sistem tersendiri melalui sebuah cerita bagaimana sebuah sistem dipakai.

Dengan demikian dapat disimpulkan bahwa pengertian *use case diagram* adalah rangkaian yang saling terkait yang menggambarkan hubungan interaksi antara sistem dan aktor. Berikut ini adalah simbol-simbol yang digunakan dalam *use case diagram*, serta keterangannya seperti yang dijelaskan pada tabel 2. 2.

Tabel 2. 2 Simbol *Use Case Diagram*

NO	SIMBOL	KETERANGAN
1		Aktor, mewakili perang orang, sistem lain, atau alat ketika berkomunikasi.

2		<i>Use case</i> , abstraksi dan interaksi antara sistem dan aktor.
3		<i>Association</i> , abstraksi dari penghubung antara aktor dengan <i>use case</i> .
4		<i>Generalization</i> , menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan <i>use case</i> .
5		Menunjukkan bahwa suatu <i>use case</i> seluruhnya merupakan fungsionalitas dari <i>use case</i> lainnya.
6		Menunjukkan bahwa suatu <i>use case</i> merupakan tambahan fungsionalitas dari <i>use case</i> lainnya jika suatu kondisi terpenuhi.

2.2.17 Flowchart

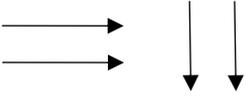
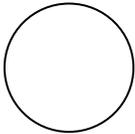
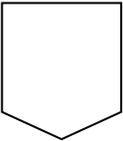
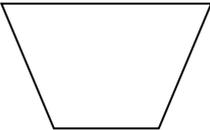
Menurut (Wibawanto, 2017) *Flowchart* adalah suatu bagan dengan simbol-simbol tertentu yang menggambarkan urutan proses secara mendetail dan hubungan antara suatu proses (instruksi) dengan proses lainnya dalam suatu program. Diagram alur dapat menunjukkan secara jelas, arus pengendalian suatu algoritma yakni bagaimana melaksanakan suatu rangkaian kegiatan secara logis dan sistematis.

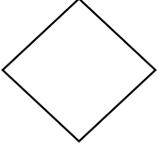
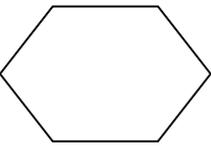
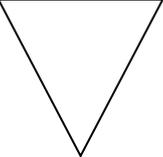
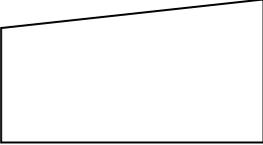
Simbol-simbol yang di pakai dalam *flowchart* dibagi menjadi 3 kelompok:

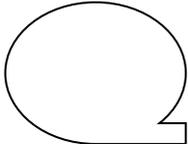
- 1) *Flow direction symbol*, digunakan untuk menghubungkan simbol satu dengan yang lain, disebut juga *connecting line*.
- 2) *Processing symbols*, menunjukkan jenis operasi pengolahan dalam suatu proses atau prosedur.
- 3) *Input/Output symbol*, menampilkan jenis peralatan yang digunakan sebagai media *input* atau *output*.

Berikut di bawah ini adalah simbol-simbol yang digunakan dalam *flowchart* disertai dengan keterangan fungsinya sebagaimana dijelaskan pada tabel 2. 3.

Tabel 2. 3 Simbol *Flowchart*

NO	SIMBOL	KETERANGAN
1		Simbol arus/ <i>flow</i> , berfungsi untuk menyatakan jalannya arus suatu proses
2		Simbol <i>connector</i> , berfungsi untuk menyatakan sambungan dari proses ke proses lainnya dalam halaman yang sama
3		Simbol <i>offline connector</i> , berfungsi untuk menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda
4		Simbol <i>process</i> , berfungsi untuk menyatakan suatu tindakan (proses) yang dilakukan oleh komputer
5		Simbol <i>manual</i> , berfungsi untuk menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer

6		<p>Simbol <i>decision</i>, berfungsi untuk menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban : ya/tidak</p>
7		<p>Simbol <i>terminal</i>, berfungsi untuk menyatakan permulaan atau akhir suatu program</p>
8		<p>Simbol <i>predefined process</i>, berfungsi untuk menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal</p>
9		<p>Simbol <i>keying operation</i>, berfungsi untuk menyatakan segala jenis operasi yang diproses dengan menggunakan suatu mesin yang mempunyai <i>keyboard</i></p>
10		<p>Simbol <i>offline-storage</i>, berfungsi untuk menunjukkan bahwa data dalam simbol ini akan disimpan ke suatu media tertentu</p>
11		<p>Simbol <i>manual input</i>, berfungsi untuk memasukkan data secara manual dengan menggunakan <i>online keyboard</i></p>

12		Simbol <i>input/output</i> , berfungsi untuk menyatakan proses <i>input</i> atau <i>output</i> tanpa tergantung jenis peralatannya
13		Simbol <i>magnetic tape</i> , berfungsi untuk menyatakan <i>input</i> berasal dari pita magnetis atau <i>output</i> disimpan ke pita magnetis
14		Simbol <i>disk storage</i> , berfungsi untuk menyatakan <i>input</i> berasal dari <i>disk</i> atau <i>output</i> disimpan ke <i>disk</i> .
15		Simbol <i>document</i> , berfungsi untuk mencetak keluaran dalam bentuk dokumen (melalui <i>printer</i>)
16		Simbol <i>punched card</i> , berfungsi untuk menyatakan <i>input</i> berasal dari kartu atau <i>output</i> ditulis ke kartu

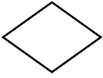
(Sumber : Rusmawan 2019)

2.2.18 ERD (*Entity Relationship Diagram*)

ERD adalah suatu diagram yang digunakan untuk memodelkan struktur data dan hubungan antar data. ERD berfungsi untuk menggambarkan relasi dari dua *file* atau dua tabel yang dapat digolongkan dalam tiga macam bentuk relasi, yaitu satu-satu, satu-banyak dan banyak-banyak (Novendri, dkk, 2019).

Berikut di bawah ini adalah simbol-simbol yang digunakan dalam ERD disertai dengan keterangan fungsinya sebagaimana dijelaskan pada tabel 2. 4.

Tabel 2. 4 ERD (*Entity Relationship Diagram*)

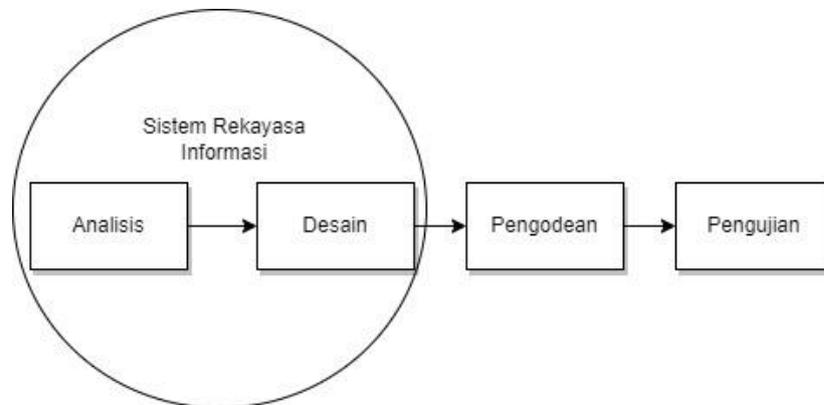
NO	SIMBOL	KETERANGAN
1		<p>Entitas (<i>Entity</i>) adalah objek data prinsip tentang informasi yang dikumpulkan. Suatu objek yang dapat didefinisikan lingkungan pemakaian dalam konteks sistem yang telah dibuat.</p>
2		<p><u>Relasi (<i>Relationship</i>)</u> Relasi adalah suatu asosiasi antara dua tabel atau lebih.</p> <ol style="list-style-type: none"> 1) Relasi satu ke satu (<i>One-to-One Relationship</i>) adalah suatu hubungan dimana entitas pertama hanya mempunyai 1 hubungan pada entitas kedua. Hubungan ini dinotasikan dengan 1 ke 1 atau 1:1. 2) Relasi satu ke banyak (<i>One-to-Many Relationship</i>) adalah suatu hubungan dimana satu entitas pertama bisa mempunyai banyak hubungan pada entitas kedua. Hubungan ini dinotasikan dengan 1 ke M atau 1:M 3) Relasi banyak ke satu (<i>Many-to-One Relationship</i>) adalah suatu

		<p>hubungan dimana banyak entitas pertama bisa mempunyai satu hubungan pada entitas kedua. Hubungan ini dinotasikan dengan M ke 1 atau M:1</p> <p>4) Relasi banyak ke banyak (Many-to-Many Relationship) adalah setiap entitas pertama dapat mempunyai banyak hubungan pada entitas yang kedua, begitu juga yang kedua bisa memiliki banyak hubungan pada entitas pertama. Hubungan ini dinotasikan dengan M ke M atau M:M</p>
3		Atribut adalah properti atau karakteristik yang dimiliki oleh suatu entitas data yang berfungsi mendeskripsikan karakter dari entitas.
4		Garis menunjukkan garis.

2.2.19 Metode Waterfall

Waterfall model merupakan sebuah *software development lifecycle* yang memiliki jenis *linear*. Memiliki beberapa *fase* seperti contohnya *requirement analysis* (analisis kebutuhan), *design* (perancangan), *coding* (pembuatan kode program), *testing* (pengujian) dan implementasi dimana umumnya satu *fase* tidak

diulang kembali dan berlanjut ke *fase* berikutnya sampai *fase* sebelumnya selesai (Prihandoyo, 2018) seperti pada Gambar 2. 17.



Gambar 2. 17 Metode *Waterfall*

Berikut ini adalah tahapan dari model waterfall yaitu:

1. Analisis

Melakukan analisis kebutuhan perangkat lunak, fungsi dan proses dari *web* yang dibuat, pengidentifikasian kendala dalam pembuatan *web*, menganalisis keandalan, kelemahan, dan teknologi yang dipakai

2. Desain

Desain perangkat lunak adalah proses beberapa tahapan langkah pada rancangan pembuatan program perangkat lunak meliputi struktur data, arsitektur perangkat lunak, representasi antarmuka dan prosedur pengodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahapan analisis kebutuhan ke representasi rancangan agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Pada tahap ini, hasil dari desain perangkat lunak yang telah ada didokumentasikan.

3. Pengkodean

Desain harus ditranslasikan kedalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai desain yang telah dibuat

pada tahap desain. Atau tahapan penulis membuat program dengan bahasa program seperti php, html, scc dan lain-lain.

4. Pengujian

Tahapan ini penulis melakukan pengujian terhadap program yang telah dibuat untuk mengetahui kekurangan dari program tersebut. Seperti validasi halaman *login*, apakah sesuai dengan harapan.

2.2.20 Metode *BlackBox Testing*

Black-Box Testing merupakan teknik pengujian perangkat lunak yang berfokus pada spesifikasi *fungsional* dari perangkat lunak. Pengujian aplikasi dengan metode ini memungkinkan pengembang *software* untuk membuat himpunan kondisi *input* yang akan melatih seluruh syarat-syarat *fungsional* suatu program (Jaya, 2018).

Keuntungan penggunaan metode *Blackbox Testing* adalah:

1. Penguji tidak perlu memiliki pengetahuan tentang bahasa pemrograman tertentu.
2. Pengujian dilakukan dari sudut pandang pengguna, ini membantu untuk mengungkapkan ambiguitas atau inkonsistensi dalam spesifikasi persyaratan.
3. *Programmer* dan *tester* keduanya saling bergantung satu sama lain.

