BUKU PINTAR APLIKASI AI PROMOSI MEDIA SOSIAL DENGAN PHP & MYSQL





KATA PENGANTAR

PHP adalah singkatan dari PHP: Hypertext Preprocessor yang pada awalnya merupakan singkatan dari Personal Home Page, merupakan bahasa pemrograman script server-side yang didesain untuk pengembangan web. Selain itu, PHP juga bisa digunakan sebagai bahasa pemrograman umum. PHP disebut bahasa pemrograman server side karena PHP diproses pada komputer server. Hal ini berbeda dibandingkan dengan bahasa pemrograman client-side seperti JavaScript yang diproses pada web browser (client).

SQL adalah Structured Query Language yaitu suatu bahasa khusus yang digunakan untuk membuat dan mengolah database. Dengan SQL atau bisa juga disebut query kita dapat memanipulasi atau mengedit database sesuai yang kita kehendaki. seperti menjalankan query untuk mengambil data, menambah data, memperbarui data dan menghapus data.

STUKTUR PENULISAN BUKU

Bab 1 Pendahuluan

Bab 1 berisi tentang pengenalan My SQL, CodeIgniter (CI) , XAMPP dan PHP (PHP Hypertext Preprocessor) .

Bab 2 Instalasi CI dan web server XAMPP

Bab 2 berisi langkah – langkah menginstal CodeIgniter dan XAMPP pada Windows.

Bab 3 Mengenal database MYSQL beserta Fiture

Bab 3 menjelaskan pemrograman database meliputi cara mempersiapkan database dan tabel serta melakukan penambahan, penghapusan, perubahan dan menampilkan data.

Bab 4 Integrasi koneksi database MYSQL dengan CodeIgniter

Bab 4 Berisi tentang pembuatan database, membuat tabel, dan lainnya.

Bab 5 Membuat Aplikasi

Bab 5 menjelaskan tentang pembuatan sistem pengambil keputusan promosi media sosial menggunakan CodeIgniter.

UNTUK PARA PEMBACA

Didalam penulisan buku ini, kami berusaha melakukan pengkajian dan penelitian serta berusaha menyampaikan materi secara lengkap dan terstruktur. Tentunya dalam setiap karya tidak ada yang sempurna, maka mungkin buku ini kurang memenuhi kebutuhan pengguna atau pembaca atau mengalami kesulitan setelah membaca buku ini.

BAB I

PENDAHULUAN

1.1 Pengertian Dasar

CODEIGNITER

merupakan aplikasi open source (sumber terbuka) yang berupa kerangka kerja PHP dengan model MVC (Model, View, Controller) untuk membangun website dinamis dengan menggunakan bahasa pemrograman PHP. Codelgniter memudahkan pengembang web untuk membuat aplikasi web dengan cepat mudah dibandingkan dengan membuatnya dari awal. Codelgniter dirilis pertama kali pada 28 Februari 2006. Versi stabil terakhir adalah versi 3.0.6.

XAMPP

adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program.

Fungsinya adalah sebagai server yang berdiri sendiri (localhost), yang terdiri atas program Apache HTTP Server, MySQLdatabase, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMPP merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP dan Perl. Program ini tersedia dalam GNU General Public License dan bebas, merupakan web server yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis. Untuk mendapatkanya dapat mendownload langsung dari web resminya.

MySQL

adalah sebuah perangkat lunak sistem manajemen basis data SQL (bahasa Inggris: *database management system*) atau DBMS yang multialur, multipengguna, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis di bawah

lisensi GNU General Public License (GPL), tetapi mereka juga menjual di bawah lisensi komersial untuk kasus-kasus di mana penggunaannya tidak cocok dengan penggunaan GPL.

PHP

singkatan *PHP: Hypertext Preprocessor* adalah bahasa skrip yang dapat ditanamkan atau disisipkan ke dalam HTML. PHP banyak dipakai untuk memprogram situs web dinamis. PHP dapat digunakan untuk membangun sebuah CMS.

Pada awalnya PHP merupakan kependekan dari Personal Home Page (Situs personal). PHP pertama kali dibuat oleh Rasmus Lerdorfpada tahun 1995. Pada waktu itu PHP masih bernama Form Interpreted (FI), yang wujudnya berupa sekumpulan skrip yang digunakan untuk mengolah data formulir dari web.

Selanjutnya Rasmus merilis kode sumber tersebut untuk umum dan menamakannya PHP/FI.

Dengan perilisan kode sumber ini menjadi sumber terbuka, maka banyak pemrogram yang tertarik untuk ikut mengembangkan PHP.

Pada November 1997, dirilis PHP/FI 2.0. Pada rilis ini, interpreter PHP sudah diimplementasikan dalam program C. Dalam rilis ini disertakan juga modul-modul ekstensi yang meningkatkan kemampuan PHP/FI secara signifikan.

Pada tahun 1997, sebuah perusahaan bernama Zend menulis ulang interpreter PHP menjadi lebih bersih, lebih baik, dan lebih cepat. Kemudian pada Juni 1998, perusahaan tersebut merilis interpreter baru untuk PHP dan meresmikan rilis tersebut sebagai PHP 3.0 dan singkatan PHP diubah menjadi akronim berulang PHP: Hypertext Preprocessing.

Media sosial

Media sosial adalah sebuah media online, dengan para penggunanya bisa dengan mudah berpartisipasi, berbagi, dan menciptakan isi meliputi blog, jejaring sosial, wiki, forum dan dunia virtual. Blog, jejaring sosial dan wiki merupakan bentuk media sosial yang paling umum digunakan oleh masyarakat di seluruh dunia. Pendapat lain mengatakan bahwa media

sosial adalah media online yang mendukung interaksi sosial dan media sosial menggunakan teknologi berbasis web yang mengubah komunikasi menjadi dialog interaktif.

Media sosial juga merupakan alat promosi bisnis yang efektif karena dapat diakses oleh siapa saja, sehingga jaringan promosi bisa lebih luas. Media sosial menjadi bagian yang sangat diperlukan oleh pemasaran bagi banyak perusahaan dan merupakan salah satu cara terbaik untuk menjangkau pelanggan dan klien. Media sosial sperti blog, facebook, twitter, dab youtube memiliki sejumlah manfaat bagi perusahaan dan lebih cepat dari media konvensional seperti media cetak dan iklan TV, brosur dan selebaran.

BAB II

INSTALASI CI DAN XAMPP

Instalasi CodeIgniter

Seperti pada awal pembahasan bahwa codeigniter ini adalah sebuah framework yang sangat sederhana, sehingga cara install codeigniter sendiri tidak terlalu sulit bahkan sangat mudah, dimana kita hanya memerlukan sebuah web server pada komputer kita, berikut ini caranya:

Langkah 1

Instal web server (Xampp) pada Windows PC/laptop.

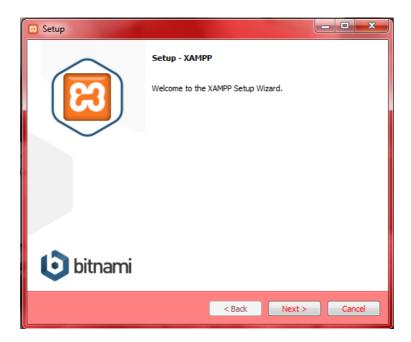
Cara instal Xampp:

- **1.** Download Xampp di web browser Anda (Bisa menggunakan Xampp yang sama atau versi terbaru).
- **2.** Kemudian klik Xampp hasil download,lalu klik yes untuk install Xampp di windows Anda.
- **3.** Klik oke dan ikuti langkah langkah dibawah ini.

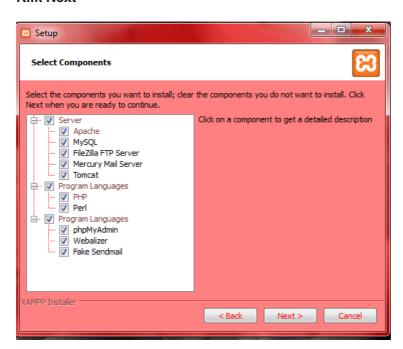
Klik ok



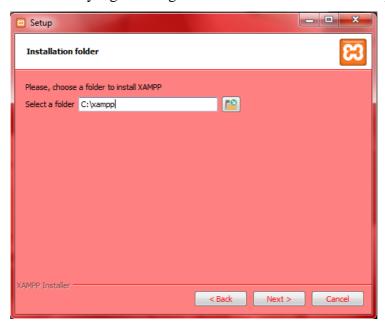
Klik Next



Klik Next



4. Pilih folder yang akan digunakan untuk meletakkan Xampp. Kemudian klik next.



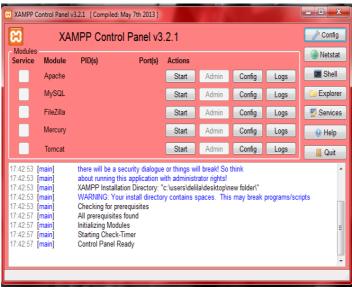
5. Tunggu beberapa waktu untuk menyelesaikan proses instal.





6. Setelah selesai, kemudian klik finish. Xampp sudah dapat digunakan.



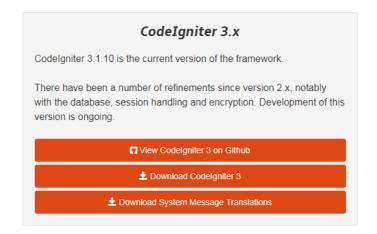


Langkah 2

Download CodeIgniter pada browser Anda.

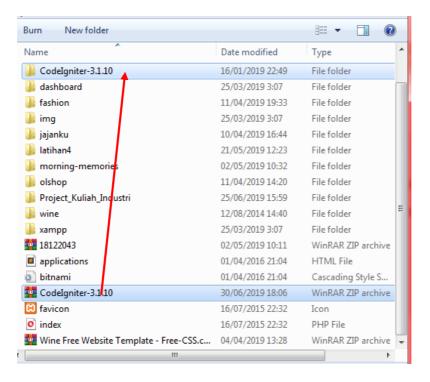


Codelgniter comes in three flavors: Codelgniter 3 (cur



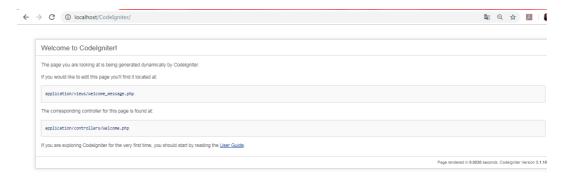
Setelah selesai download, kemudian:

- 1. Pindahkan file zip CodeIgniter Ke dalam folder xampp ->htdocs.
- 2. Kemudian ekstrak file zip didalam folder htdocs



Langkah 3

Setelah selesai meng-ekstrak file kemudian buka browser anda, lalu cek melalui link http://localhost/CodeIgniter/ (CodeIgniter Adalah nama file yang sudah diekstrak sebelumnya, bisa diganti dengan nama project lainnya).



Jika sudah seperti gambar diatas maka CodeIgniter berhasil di download dan sudah dapat digunakan.

BAB III

MENGENAL DATABASE MYSQL DAN FITURE MYSQL

1. Pengertian Database

Basis data (database) adalah kumpulan data yang di simpan secara sistematis di dalam komputer yang dapat di olah atau di manipulasi menggunakan perangkat lunak (program aplikasi) untuk menghasilkan informasih dan mengolah data. Pendefenisian basis data meliputi spesifikasi berupa tipe data, stuktur data dan juga batasan-batasan pada data yang kemudian di simpa di dalam aplikasi yang dapat menyimpan data tersebut.

Basis data sala satu aspek pendataan di dalam suatu informasi kerena berfungsi sebagai penyimpanan data yang akan di olah lebih lanjut ke depannya, di dalam suatu system pemograman basis data sangat penting kerena dapat mengordinasi data menghindari suatu hubungan antara data yang tidak jelas dan juga update yang rumit.

Proses memasukan data dan mengambil data dari penyimpanan data memerlukan perangkat lunak yang di sebut dengan system menajemen basis data (database management system DBMS). DBMS merupakan system perangkat lunak yang memungkinkan pengguna basis data (database user) untuk mengontrol dan mengakses data secara praktik dan efesien.

Ada beberapa fungsi yang harus ditangani DBMS seperti pendefinisian data, menangani permintaan pengguna untuk mengakses data, memeriksa sekuriti dan integriti data yang didefinisikan oleh DBA (Database Administrator), menangani kegagalan dalam pengaksesan data yang disebabkan oleh kerusakan sistem maupun media penyimpanan (disk) dan juga menangani unjuk kerja semua fungsi secara efisien.

Tujuan utama DBMS adalah untuk memberikan tinjauan abstrak data kepada pengguna. Jadi sistem menyembunyikan informasi tentang bagaimana data disimpan, dipelihara dan juga bisa diakses secara efisien. Pertimbangan efisien di sini adalah rancangan struktur data yang kompleks tetapi masih bisa digunakan oleh pengguna awam tanpa mengetahui kompleksitas strukturnya.

System manajemen basis data umumnya menyediakan interface grafis yang terbatas dan umumnya hanya bisa di gunakan oleh programe professional. System manajemen basis

data tidak cocok untuk pengguna awam yang tidak mempunyai pengetahuan khusus tentang basis data program aplikasi merupakan perangkat lunak yang di gunakan sebagai media untuk menjembatani antar pengguna awam dengan basis data.

2. Pengertian MYSQL

MySQL adalah sistem manajemen basis data relasional (RDBMS) sumber terbuka. Pada tahun 2013, MySQL merupakan RDBMS kedua yang paling banyak digunakan di dunia dan yang pertama untuk RDBMS sumber terbuka. MySQL didasarkan pada SQL (Select Query language) yaitu bahasa standar untuk berkomunikasi dengan sistem manajemen basis data relasional.

MySQL dibuat (ditulis) menggunakan bahasa pemrograman C dan C++ serta dapat berjalan di semua platform seperti Windows, Linux dan Unix. MySQL merupakan database pilihan untuk beberapa bahasa pemrograman web seperti PHP, Ruby on Rails dan Python. MySQL juga menjadi pilihan untuk beberapa aplikasi web sistem pengelolaan konten (content management system | CMS) sumber terbuka seperti Joomla, Wordpress dan Drupal.

Selain itu, MySQL juga menjadi salah satu komponen penting dari web service solution stack LAMP (Linux, Apache, MySQL and PHP) yaitu platform pengembangan web sumber terbuka dimana Linux sebagai sistem operasi, Apache sebagai Web Server, MySQL sebagai RDBMS dan PHP sebagai bahasa skrip berorientasi obyek (object-oriented scripting language).

3. Mengenal fiture database MYSQL

- ✓ Ditulis dengan menggunakan bahasa C dan C++
- ✓ Berkerja dalam berbagai platform

- ✓ Menyediakan mesin penyimpan (storage engine) baik untuk data transactional atau nontransactional.
- ✓ Mengeksekusi pernyataan Join dengan cepat
- ✓ Menyediakan banyak tipe data
- ✓ Mempunyai banyak fungsi (function)
- ✓ Sistem Privilege dan password sangat fleksibel dan aman dan dapat digunakan untuk verivikasi.
- ✓ Keamanan password dilakukan dengan enkripsi pada semua lalu-lintas password ketika terkoneksi ke sebuah server
- ✓ Mendukung untuk penyimpanan data yang besar
- ✓ Client dapat terkoneksi ke server MySQL menggunakan beberapa protokol jaringan
- ✓ Mempunyai librari konektivitas yang beragam
- ✓ Mempunyai banyak tool .

4. Tools yang dapat digunakan untuk membuat koneksi dengan menggunakan MYSQL

Koneksi ke Server MySQL dilakukan dengan MySQL Client (tools), tools tersebut telah tersedia didalam MySQL itu sendiri. Untuk melakukan koneksi ke Server MySQL dapat dilakukan dengan cara yang berbeda, tergantung pada MySQL Client yang digunakan. Berikut berbagai macam MySQL Client untuk administrasi Server MySQL:

- ✓ MySQL Command Line Client.
- ✓ MySQL-Front, merupakan front-end MySQL berbasis Windows.
- ✓ PHPMyAdmin, merupakan front-end MySQL berbasis web.
- ✓ SQLYog.
- ✓ MySQL Administrator dan MySQL Query Browser.

Dari berbagai macam tools tersebut, yang paling mudah digunakan untuk **belajar database MySQL** maupun membuat database MySQL adalah PHP MyAdmin. PHP MyAdmin merupakan front-end MySQL berbasis web, dibuat dengan menggunakan PHP. PHP MyAdmin banyak digunakan di hampir semua penyedia hosting di internet. PHP MyAdmin mendukung

berbagai fitur administrasi MySQL termasuk manipulasi database, tabel, index dan dapat mengeksport data ke dalam berbagai format data.

5. Cara Kerja MYSQL

MySQL didasarkan pada model client-server. Inti dari MySQL adalah server MySQL, yang menangani semua instruksi (atau perintah) basis data. Server MySQL tersedia sebagai program terpisah untuk digunakan dalam lingkungan jaringan klienserver dan sebagai perpustakaan yang dapat disematkan (atau ditautkan) ke dalam aplikasi yang terpisah.

MySQL beroperasi bersama dengan beberapa program utility yang mendukung administrasi database MySQL. Perintah dikirim ke MySQLServer melalui klien MySQL, yang diinstal pada komputer.

MySQL pada awalnya dikembangkan untuk menangani basis data besar dengan cepat. Meskipun MySQL biasanya diinstal pada satu engine saja, ia dapat mengirim database ke beberapa lokasi, karena pengguna dapat mengaksesnya melalui aklien MySQL yang berbeda. Antarmuka ini mengirimkan pernyataan SQL ke server dan kemudian menampilkan hasilnya.

6. Mengenal SQL

Jika telah mengenal Database MySQL, selanjutnya harus mengetahui apa itu SQL (Structured Query Language). SQL atau para programer sering menyebutnya query adalah suatu bahasa (language) yang digunakan untuk mengakses database. Secara umum, terdapat 3 jenis perintah SQL, yaitu :

1. DDL (Data Definition Language)

DDL merupakan perintah SQL yang berhubungan dengan pendefinisian suatu struktur database, yaitu database dan table. Beberapa perintah dasar yang termasuk DDL ini antara lain:

- ✓ CREATE
- ✓ ALTER
- ✓ RENAME
- ✓ DROP

2. DML (Data Manipulation Language)

DML merupakan perintah SQL yang berhubungan dengan manipulasi atau pengolahan data. Perintah SQL yang termasuk dalam DML antara lain:

- ✓ SELECT
- √ INSERT
- ✓ UPDATE
- ✓ DELETE

3. DCL (Data Control Language)

DCL merupakan perintah SQL yang berhubungan dengan manipulasi user dan hak akses (priviledges). Perintah SQL yang termasuk dalam DCL antara lain:

- √ GRANT
- ✓ REVOKE
- 7. Langkah langkah pembuatan CRUD (Create, Read, Update, Delete) dengan MYSQL menggunakan Command Prompt (cmd).
 - Pengertian Command Prompt (cmd)

Command Prompt (cmd.exe) adalah baris perintah penerjemah (baris perintah yang dieksekusi ke sistem operasi yang disediakan oleh Microsoft pada: OS/2. Windows CE, dan. keluarga sistem operasi Windows NT (termasuk Windows 2000, Windows XP, Windows Vista, Windows Server 2003 dan Windows Server 2008), (Wikipedia).

• Istilah – istilah yang digunakan dalam CMD

ASSOC > Menampilkan atau mengubah asosiasi ekstensi file.

AT > Menjadwalkan perintah dan program agar berjalan di komputer.

ATTRIB > Menampilkan atau mengubah atribut file.

BREAK > Mengeset atau menghilangkan pengecekan extended CTRL+C.
 CACLS > Menampilkan atau mengubah access control lists (ACLs) dai files.
 CALL > Memanggil sebuah program batch dari program batch yang lain.

CD > Menampilkan nama atau mengubah direktori sekarang.
 CHCP > Menampilkan atau mengubah active code page number.
 CHDIR > Menampilkan nama atau mengubah direktori sekarang.
 CHKDSK > Memeriksa sebuah disket/harddisk dan menampilkan laporan

status.

CHKNTFS > Menampilkan atau mengubah pemeriksaan disket pada saat

booting.

CLS > Menghapus layar.

CMD > Menjalankan interpreter command Windows yang baru.

COLOR > Mengubah warna foreground dan background pada command

prompt.

COMP > Membandingkan isi dari dua atau beberapa file.

COMPACT > Menampilkan atau mengubah kompresi file pada partisi NTFS. **CONVERT** > Mengkonversi FAT ke NTFS. Anda tidak dapat mengkonversi

drive yang sedang aktif.

COPY > Menyalin satu atau beberapa file ke lokasi lain.

DATE > Menampilkan atau mengubah tanggal.DEL > Menghapus satu atau beberapa file.

DIR > Menampilkan daftar file dan subdirektori dalam sebuah direktori.

DISKCOMP > Membandingkan isi dari dua buah disket.DISKCOPY > Menyalin isi dari satu disket ke disket lain.

DOSKEY > Mngedit baris perintah, memanggil kembali perintah Windows,

dan membuat macro.

ECHO > Menampilkan pesan, atau mengubah command echoing on atau

off.

ENDLOCAL > Mengakhiri localization dari environment changes in a batch file.

ERASE > Menghapus satu atau beberapa fie. **EXIT** > Keluar dari program CMD.EXE.

FC > Membandingkan 2 atau beberapa file, dan Menampilkan

perbedaan file-file tersebut.

FIND > Mencari string teks dalam sebuah file atau beberapa file.

FINDSTR > Mencari string dalam file.

FOR > Menjalankan perintah yang spesifik untuk masing-masing file

dalam sekumpulan file.

FORMAT > Mem-Formats sebuah hardisk/disket untuk digunakan dalam

Windows.

FTYPE > Menampilkan atau mengubah tipe file yang digunakan dalam

asosiasi ekstensi file.

GOTO > Mengarahkan Windows command interpreter ke baris yang

berlabel dalam sebuah program batch.

GRAFTABL > Membolehkan Windows untuk menampilkan sekumpulan

karakter extended dalam mode grafik.

HELP > Menyediakan informasi Help untuk perintah-perintah Windows.

IF > Menjalankan proses kondisi dalam program batch.

LABEL > Membuat, mengubah, atau menghapus volume label dari sebuah

disket/harddisk.

MD > Membuat direktori.

MKDIR > Membuat direktori.

MODE > Mengkonfigurasi system device.

MORE > Menampilkan hasil perlayar..

MOVE > Memindahkan satu atau beberapa file dari satu direktori ke

direktori yang lain.

PATH > Menampilkan ata mengeset search path untuk executable files. **PAUSE** > Menunda pemrosesan dari sebuah batch file dan Menampilkan

pesan.

POPD > Mengembalikan nilai sebelumnya dari direktori sekarang yang

disimpan oleh PUSH.

PRINT > Mencetak file teks.

PROMPT > Mengubah command prompt Windows.

PUSHD > Menyimpan direktori sekarang kemudian mengubahnya.

RD > Menghapus direktori.

RECOVER > Mengembalikan informasi yang masih dapat dibaca dari sebuah

disket/harddisk yang sudah bad/rusak.

REM > Merekam kemontar dalam batch files atau CONFIG.SYS.

REN > Mengubah nama file.
RENAME > Mengubah nama file.
REPLACE > Menggantikan file.
RMDIR > Menghapus direktori.

SET > Menampilkan, mengatur, atau menghapus Windows environment

variables.

SETLOCAL > Memulai localization dari environment changes dalam sebuah

batch file.

SHIFT > Menggeser posisi parameters yang replacable dalam batch files.

SORT > Mensortir input.

START > Memulai jendela terpisah untuk menjalankan perintah atau

program spesifik.

SUBST > Meng-asosiasikan sebuah path dengan drive letter.

TIME > Menampilkan atau mengatur waktu sistem.

TITLE > Mengatur judul jendela untuk sesi CMD.EXE.

TREE > Menampilkan secara grafis struktur direktori dari sebuah drive

atau path.

TYPE > Menampilkan isi dari sebuah file eks.

VER > Menampilkan versi windows.

VERIFY > Memberitahu Windows untuk mem-verifikasi file anda telah

disimpan secara benar dalam harddisk/disket.

VOL > Menampilkan nomor serial dan volume label dari sebuah

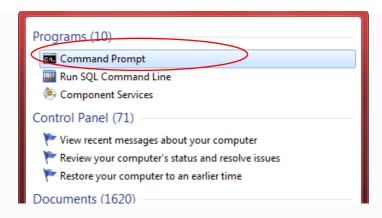
harddisk/disket

XCOPY > Menyalin file dan pohon direktori.

1. Membuat Database

Langkah membuat database dengan menggunakan Command Prompt.

- Buka Cmd melalui start pada windows anda, lalu ketikkan di bagian pencarian dengan kata kunci command prompt atau bisa juga dengan menggunakan kata kunci cmd, kemudian open.



- Kemudian akan muncul tampilan seperti di bawah ini



- Selanjutnya ketikkan kode untuk memulai membuat database.

```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.
C:\Users\Delila>cd..
C:\Users>cd..
C:\>cd..
C:\>cd..
C:\>cd xampp
C:\xampp>cd mysql
C:\xampp\mysql>cd bin
C:\xampp\mysql\bin>mysql -u root
```

Jika sudah sesuai, maka akan muncul tampilan seperti berikut

```
C:\xampp\mysql\bin>mysql -u root
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 90
Server version: 10.1.13-MariaDB mariadb.org binary distribution
Copyright (c) 2000, 2016, Oracle, MariaDB Corporation Ab and others.
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
MariaDB [(none)]>
```

- Kemudian ketikkan code berikut untuk membuat database

create database latihan_crud;

Kemudian enter, lalu ketikkan code *show databases;* untuk menampilkan database yang telah dibuat sebelumnya.

2. Membuat Tabel

Setelah membuat database, ketikkan code *use latihan_crud;* untuk menunjukkan bahwa database yang akan kita gunakan adalah **latihan_crud.**

Kemudian untuk membuat table, ketikkan code berikut:

```
create table data_siswa(
->no int(3) PRIMARY KEY,
->nama varchar(50),
->alamat varchar(60),
->gol_dar varchar(2),
->jenis_kelamin varchar(1),
```

```
->umur int(2));
```

data_siswa merupakan nama tabel yang akan kita buat.

Setelah kita membuat tabel, ketikkan code *desc data_siswa*; untuk menampilkan hasil tabel yang sudah kita buat. Maka hasilnya seperti berikut:

```
MariaDB [latihan_crud]> desc data_siswa;
                                      Nu11
 Field
                                               Key
                     Туре
                                                       Default
                                                       NULL
NULL
                      int(3)
                                                PRI
 no
                     varchar(50)
varchar(60)
 nama
                                       YES
YES
YES
YES
 alamat
                     varchar(2
                                                       NULL
  jenis_kelamin
  umur
                      int(2)
  rows in set (0.00 sec)
```

3. Memasukkan/input data ke dalam tabel

Untuk memasukkan data ke dalam tabel, ketikkan dengan format *insert into data_siswa* values('no','nama','alamat','gol dar','jenis kelamin','umur');

Contoh: insert into data mahasiswa values('1', 'fatma', 'jogja', '0', 'P', '20');

```
MariaDB [latihan_crud]> insert into data_siswa values('1','fatma','jogja','0','F
','20');
Query OK, 1 row affected (0.15 sec)
```

Ketikkan code tersebut sesuai jumlah data yang akan di masukkan ke dalam tabel.

Setelah selesai kemudian ketik select*from data_siswa;

Untuk menampilkan isi/data yang telah kita input kedalam tabel.

no i nama		gol_dar	jenis_kelamin	umur
1 fatma 2 fajar 3 ahmad 4 dwi 5 syahril 6 Ning	jakarta tegal kalteng NTT	A AB B A		20 21 21 22 23 23 23

4. Update (memperbaharui data)

Untuk memperbaharui data didalam tabel, ketikkan code berikut

update data_siswa set nama field='data_baru' where nama field="data lama";

Contoh: update data siswa set nama='delila' where nama='fatma";

- ->nama field adalah judul data didalam tabel (no/nama/alamat/dll).
- ->data_lama adalah data/ isi dari nama field (nama) sebelum diganti.
- ->data_baru adalah data/isi dari nama field (nama) terbaru atau data baru.

tabel lama (yang akan kita ganti adalah nama).

Marial	MariaDB [latihan_crud]> select*from data_siswa;						
no	nama	alamat	gol_dar	jenis_kelamin	นพนา		
2 3 4 5	fajar ahmad dwi syahril	jogja jakarta tegal kalteng NTT Jogja	A AB B A	L L L L	20 21 21 22 22 23 23		
6 rows in set (0.00 sec)							

tabel baru (nama yang sudah kita ganti ,fatma -> delila)

5. Delete (Menghapus Data)

Untuk menghapus data didalam tabel, maka code nya adalah sebagai berikut:

delete from data siswa where no='6';

- ->data_siswa adalah menunjukkan nama tabel
- ->no='6' adalah perwakilan pemanggilan satu data field yang akan dihapus.

Data yang akan kita hapus adalah field 'no' data ke '6' yaitu

No='6',nama='ning', alamat='jogja',gol dar='A',jenis kelamin='L',umur='23'.

Jadi data no 6 akan terhapus dari tabel.

tabel lama

MariaDB [latihan_crud]> select*from data_siswa;						
no	nama	alamat	gol_dar	jenis_kelamin	umur	
2 1 3 1 4 1 5	delila fajar ahmad dwi syahril Ning	jakarta tegal kalteng NTT	A AB B A	L L L	20 21 21 22 22 23 23	
6 rows	s in set (0					

tabel baru

```
MariaDB [latihan_crud]> delete from data_siswa where no='6';
Query OK, 1 row affected (0.12 sec)

MariaDB [latihan_crud]> select*from data_siswa;

no nama alamat gol_dar jenis_kelamin umur

1 delila jogja 0 P 20
2 fajar jakarta A L 21
3 ahmad tegal AB L 21
4 dwi kalteng B L 22
5 syahril NTT A L 23

5 rows in set (0.00 sec)
```

6. Drop (Menghapus Field)

Berbeda dengan delete, drop adalah perintah yang digunakan untuk menghapus field.

Seperti contoh kita akan menghapus salah satu field dari tabel data_siswa,

no/nama/alamat/gol_dar/jenis_kelamin/umur.

Sebagai contoh, kita akan menghapus Field 'umur'.

Code nya adalah sebagai berikut:

alter table data_siswa drop umur;

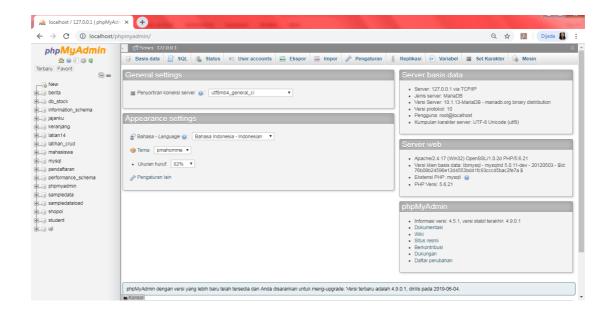
tabel lama

Marial)B [latihar	_crud]> se	elect*from	data_siswa;	.
l no	nama	alamat	gol_dar	jenis_kelamin	umur
2 3 4	fajar ahmad dwi	jogja jakarta tegal kalteng NTT	A AB B	L L	20 21 21 21 22 23
rows	in set (0.00 sec)			•

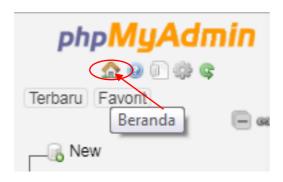
tabel baru

7. PHP MYADMIN

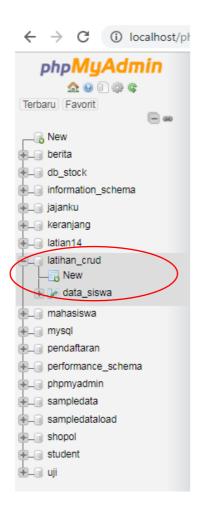
Setelah selesai melakukan latihan menggunakan beberapa perintah untuk CRUD (Cread Read Update dan Delete) didalam mysql dengan menggunakan Comman Prompt, maka kita akan cek database yang telah kita buat dengan mengetikkan link di browser Anda *localhost/phpmyadmin/*. Maka akan keluar tampilan berikut:



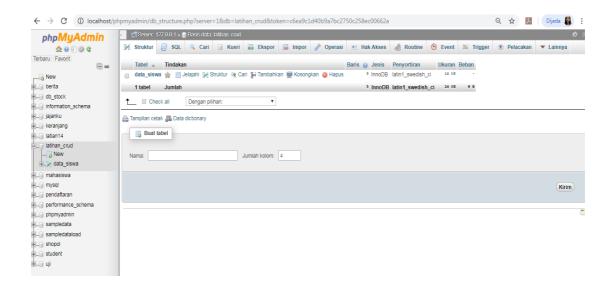
Kemudian klik icon beranda dibagian atas dibawah tulisan *phpmyadmin*



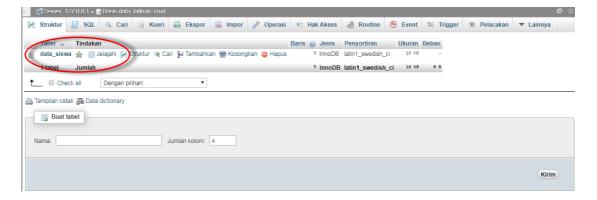
Selanjutnya klik database latihan_crud



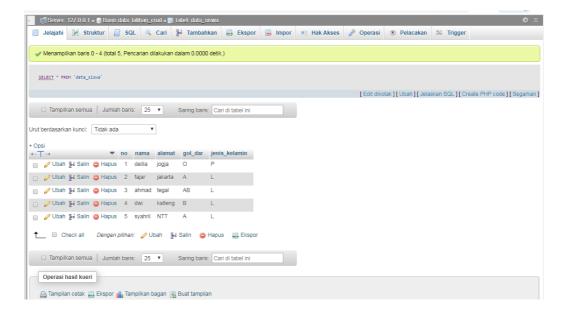
Maka tampilannya adalah seperti berikut ini:



Kemudian klik tabel **data_siswa** untuk menampilkan isi dari tabel yang sudah kita buat di Cmd sebelumnya.



Tampilan tabel data_siswa



BAB IV

INTEGRASI KONEKSI DATABASE MYSQL DENGAN CODEIGNITER

 Selanjutnya untuk menghubungkan codeigniter dengan database buka file config database codeigniter yang terletak di application/config/database.php
 application/config/database.php

```
$active_group = 'default';
     $query_builder = TRUE;
76 $db['default'] = array(
         'dsn' => '',
         'hostname' => 'localhost',
         'username' => 'root',
         'password' => '',
         'database' => 'wp',
         'dbdriver' => 'mysqli',
         'dbprefix' => '',
         'pconnect' => FALSE,
         'db_debug' => (ENVIRONMENT !== 'production'),
         'cache_on' => FALSE,
         'cachedir' => '',
         'char_set' => 'utf8',
         'dbcollat' => 'utf8_general_ci',
         'swap_pre' => '',
         'encrypt' => FALSE,
         'compress' => FALSE,
         'stricton' => FALSE,
'failover' => array(),
         'save queries' => TRUE
```

2. Pada file config database.php ini anda dapat melakukan pengaturan database anda dengan memasukkan hostname,username database, password dan nama database yang anda gunakan. pada tutorial ini username database adalah root, passwordnya kosong,

host nya localhost dan nama database yang di gunakan adalah database dengan nama malasngoding.

```
'hostname' => 'localhost',
'username' => 'root',
'password' => '',
'database' => 'wp',
```

pada password dapat di kosongkan karena saya tidak menggunakan password. anda dapat menyesuaikan nya sendiri.

setelah selesai melakukan pengaturan database, langkah selanjutnya yang juga penting adalah memanggil library database codeigniter. karena untuk penggunaan yang berhubungan dengan database kita perlu membuka atau mengaktifkan library database milik si codeigniter ini. cara nya seperti memanggil library codeigniter yang lain. anda bisa memanggil library codeigniter dari function construct() pada controller, pada method atau pada autoload codeigniter. jadi kebetulan di sini saya akan membuka atau memanggil library database codeigniter pada config autoload codeigniter yang terletak pada application/config/autoload.php

application/config/autoload.php

cari pengaturan yang mengarah ke load library codeigniter seperti berikut:

```
$
sautoload['libraries'] = array('database','session');
/*
```

Dan codeigniter pun berhasil di hubungkan dengan database mysql. Bagaimana cara memeriksanya sudah terhubung atau belum. Jadi pada tutorial ini saya akan menampilkan data dari table user yang sudah di buat tadi untuk sekalian berkenalan dengan model <u>codeigniter</u>.

Pengenalan Model Pada CodeIgniter

Selanjutnya untuk menampilkan data dari table akun tadi buat sebuah model dengan nama terserah anda. di sini saya membuat sebuah model dengan nama Model_app.php. menghubungkan codeigniter dengan database mysql

$application/models/Model_App.php$

Table akun

Perhatikan syntax di atas. Saat membuat class model Model_app penulisannya di anjurkan harus di awali dengan huruf besar. Kemudian meng-extend model Model_app yang sudah di buat dengan CI_Model. Kemudian saya membuat sebuah function email() pada model Model_App ini. function email() ini saya buat untuk mengambil

data pada table akun. Syntax \$this->db->query() di gunakan untuk mengambil data dari database. Dan nama table yang ingin di ambil data nya letakkan dalam parameternya sehingga menjadi

Table kriteria

function id kriteria

```
public function id_kriteria($a)
{
    $query = $this->db->query("SELECT * FROM kriteria WHERE email='$a' AND waktu=(SELECT max(waktu) FROM kriteria WHERE email='$a');");
    if ($query->num_rows()>0) {
        $id = $query->row_array();
        $id = $query->row_array();
        $id+;
        return 'C'.$id;
    } else {
        return 'C';
}
```

Kemudian meng-extend model Model_app yang sudah di buat dengan CI_Model. Kemudian membuat sebuah function id_kriteria () pada model Model_App ini. function id_kriteria() ini di buat untuk mengambil data pada table kriteria. Syntax \$this>db->query() di gunakan untuk mengambil data dari database. Dan nama table yang ingin di ambil data nya letakkan dalam parameternya.

function Tambah Kriteria

Kemudian meng-extend model Model_app yang sudah di buat dengan CI_Model. Kemudian membuat sebuah function tambah_kriteria() pada model Model_App ini. Function tambah_kriteria() ini di buat untuk mengambil data pada table kriteria. Syntax \$this->db->query() di gunakan untuk mengambil data dari database. Dan nama table yang ingin di ambil data nya letakkan dalam parameternya.

Function edit kriteria

Kemudian meng-extend model Model_app yang sudah di buat dengan CI_Model.

Kemudian membuat sebuah function edit_kriteria() pada model Model_App ini.

Function edit_kriteria () ini di buat untuk mengambil data pada table kriteria. Syntax \$this->db->query() di gunakan untuk mengambil data dari database dan nama table yang ingin di ambil data nya letakkan dalam parameternya.

function hapus_kriteria

Kemudian meng-extend model Model_app yang sudah di buat dengan CI_Model. Kemudian membuat sebuah function hapus_kriteria () pada model Model_App ini. Function hapus_kriteria () ini di buat untuk mengambil data pada table kriteria. syntax \$this->db->query() di gunakan untuk mengambil data dari database dan nama table yang ingin di ambil data nya letakkan dalam parameternya.

function data_kriteria

```
public function data_kriteria($a)
{
    $query = $this->db->query("SELECT * FROM kriteria WHERE email='$a' ORDER BY waktu;");
    return $query->result_array();
}
```

Kemudian meng-extend model Model_app yang sudah di buat dengan CI_Model.

Kemudian membuat sebuah function data_kriteria () pada model Model_App ini.

Function data_kriteria () ini di buat untuk mengambil data pada table kriteria. Syntax

\$this->db->query() di gunakan untuk mengambil data dari database dan nama table yang ingin di ambil data nya letakkan dalam parameternya.

Table sub_kriteria

function tambah_sub_kriteria

Kemudian meng-extend model Model_app yang sudah di buat dengan CI_Model. Kemudian membuat sebuah function tambah_sub_kriteria () pada model Model_App ini. Function tambah_sub_kriteria () ini di buat untuk mengambil data pada table sub_kriteria. Syntax \$this->db->query() di gunakan untuk mengambil data dari database dan nama table yang ingin di ambil data nya letakkan dalam parameternya.

function edit_sub_kriteria

Kemudian meng-extend model Model_app yang sudah di buat dengan CI_Model. Kemudian membuat sebuah function edit_sub_kriteria () pada model Model_App ini. Function edit_sub_kriteria () ini di buat untuk mengambil data pada table sub_kriteria. Syntax \$this->db->query() di gunakan untuk mengambil data dari database dan nama table yang ingin di ambil data nya letakkan dalam parameternya.

function data_sub_kriteria

Kemudian meng-extend model Model_app yang sudah di buat dengan CI_Model. Kemudian membuat sebuah function data_sub_kriteria () pada model Model_App ini. Function data_sub_kriteria () ini di buat untuk mengambil data pada table sub_kriteria. Syntax \$this->db->query() di gunakan untuk mengambil data dari database dan nama table yang ingin di ambil data nya letakkan dalam parameternya.

function hapus_sub_kriteria

Kemudian meng-extend model Model_app yang sudah di buat dengan CI_Model. Kemudian membuat sebuah function hapus_sub_kriteria () pada model Model_App ini. Function hapus_sub_kriteria () ini di buat untuk mengambil data pada table sub_kriteria. Syntax \$this->db->query() di gunakan untuk mengambil data dari database dan nama table yang ingin di ambil data nya letakkan dalam parameternya.

Table Alternatif

function id alternatif

```
public function id_alternatif($a)
{
    $query = $this->db->query("SELECT * FROM alternatif WHERE email='$a' AND waktu=(SELECT max(waktu) FROM alternatif WHERE email='$a');");
    if ($query->num_rows()>0) {
        $id = $query->row_array();
        $id = $ubstr($id['id'], 1);
        $id++;
        return 'A'.$id;
    } else {
        return 'A1';
    }
}
```

Kemudian meng-extend model Model_app yang sudah di buat dengan CI_Model. Kemudian membuat sebuah function id_alternatif () pada model Model_App ini. Function d_alternatif () ini di buat untuk mengambil data pada table Alternatif. Syntax \$this->db->query() di gunakan untuk mengambil data dari database dan nama table yang ingin di ambil data nya letakkan dalam parameternya.

function tambah alternatif

Kemudian meng-extend model Model_app yang sudah di buat dengan CI_Model. Kemudian membuat sebuah function tambah_alternatif () pada model Model_App ini. Function tambah_alternati () ini di buat untuk mengambil data pada table Alternatif. Syntax \$this->db->query() di gunakan untuk mengambil data dari database dan nama table yang ingin di ambil data nya letakkan dalam parameternya.

function edit alternatif

Kemudian meng-extend model Model_app yang sudah di buat dengan CI_Model.

Kemudian membuat sebuah function edit_alternatif () pada model Model_App ini.

Function edit_alternati () ini di buat untuk mengambil data pada table Alternatif.

Syntax \$this->db->query() di gunakan untuk mengambil data dari database dan nama table yang ingin di ambil data nya letakkan dalam parameternya

function hapus alternatif

Kemudian meng-extend model Model_app yang sudah di buat dengan CI_Model.

Kemudian membuat sebuah function hapus_alternatif () pada model Model_App ini.

Function hapus _alternati () ini di buat untuk mengambil data pada table Alternatif.

Syntax \$this->db->query() di gunakan untuk mengambil data dari database dan nama table yang ingin di ambil data nya letakkan dalam parameternya.

function data alternatif

```
public function data_alternatif($a)
{
    $query = $this->db->query("SELECT * FROM alternatif WHERE email='$a' ORDER BY waktu;");
    return $query->result_array();
}
```

Kemudian meng-extend model Model_app yang sudah di buat dengan CI_Model. Kemudian membuat sebuah function data_alternatif () pada model Model_App ini. Function data_alternati () ini di buat untuk mengambil data pada table Alternatif. Syntax \$this->db->query() di gunakan untuk mengambil data dari database dan nama table yang ingin di ambil data nya letakkan dalam parameternya

Table Alternatif

Membuat keputusan kriteria

```
public function pangkat($a,$b)
{
    $query = $this->db->query("SELECT id, bobot/total AS nilai, jenis FROM kriteria JOIN (SELECT sum(bobot) AS total FROM kriteria WHERE email='$a')
    return $query->row_array();
}

public function hasi1($a,$b)
{
    $query = $this->db->query("SELECT * FROM alternatif WHERE email='$a' AND id='$b';");
    return $query->row_array();
}

public function untuk_tombol($a,$b)
{
    $query = $this->db->query("SELECT * FROM pencocokan_kriteria WHERE email='$a' AND a='$b';");
    return $query->num_rows();
}

public function untuk_option($a,$b,$c)
{
    $query = $this->db->query("SELECT * FROM pencocokan_kriteria WHERE email='$a' AND a='$b' AND c='$c';");
    return $query->row_array();
}

public function untuk_cek($a,$b,$c)
{
    $query = $this->db->query("SELECT * FROM pencocokan_kriteria WHERE email='$a' AND a='$b' AND c='$c';");
    return $query->row_array();
}
```

Kemudian meng-extend model Model_app yang sudah di buat dengan CI_Model. Kemudian membuat beberapa function pada model Model_App ini. Pada function yang saya buat untuk mengambil data pada table Alternatif, dan pencocokan kriteria. Syntax \$this->db->query() di gunakan untuk mengambil data dari database dan nama table yang ingin di ambil data nya letakkan dalam parameternya

Table Akun Admin

Table Akun Admin digunakan untuk login menggunakan username dan password.

```
public function hapus_akun($a,$b)
   $cek = $this->db->query("SELECT * FROM akun WHERE email='$a';")->row_array();
   $cek_pass = password_verify($b,$cek['password']);
       $query = $this->db->simple_query("DELETE FROM akun WHERE email='$a';");
       if ($query) {
           $this->db->simple_query("DELETE FROM alternatif WHERE email='$a';");
           echo '<script'
               alert("Terima Kasih Telah Menggunakan Layanan Kami.");
               document.location.href="'.base_url('logout').'";
       } else {
           $this->session->set_flashdata('notif','<div class="alert alert-danger alert-dismissable">
                               <button type="button" class="close" data-dismiss="alert" aria-hidden="true">></button>
                               <h4><i class="icon fa fa-ban"></i> Error!.</h4>
                               Silahkan coba lagi.
           redirect('app/pengaturan');
       $this->session->set_flashdata('notif','<div class="alert alert-danger alert-dismissable">
                                    <h4><i class="icon fa fa-ban"></i> Password Salah!.</h4>
                                   Silahkan coba lagi.
                               </div>');
       redirect('app/pengaturan');
```

kemudian meng-extend model Model_app yang sudah di buat dengan CI_Model. kemudian saya membuat beberapa function pada model Model_App ini. Pada function yang saya buat untuk mengambil data pada table akun. syntax \$this->db->query() di gunakan untuk mengambil data dari database. dan nama table yang ingin di ambil data nya letakkan dalam parameternya

```
:lass Model_auth extends CI_Model {
   public function buat akun($a)
      $cek_email = $this->db->query("SELECT * FROM akun WHERE email='{$a['email']}';")->num_rows();
$cek_username = $this->db->query("SELECT * FROM akun WHERE username='{$a['username']}';")->num_rows();
      if ($cek_email>0 and $cek_username>0) {
          $this->session->set_flashdata('notif','Email dan Username Sudah Terdaftar!');
          redirect('daftar');
      } elseif ($cek_email>0) {
          $this->session->set_flashdata('notif','Email Sudah Terdaftar!');
          redirect('daftar');
          $this->session->set_flashdata('notif','Username Sudah Terdaftar!');
          redirect('daftar');
      } elseif ($a('password']!=$a['passwordlagi']) {
   $this->session->set_flashdata('notif','Password Tidak Sama!');
          redirect('daftar');
      } else {
          $pass = password_hash($a['password'],PASSWORD_DEFAULT);
          $query = $this->db->simple_query("INSERT INTO akun VALUES ('{$a['email']}','{$a['username']}','$pass');");
              $this->session->set_flashdata('notif','Berhasil mendaftar. Silahkan login.');
              redirect('daftar');
              $this->session->set_flashdata('notif','Error! Silahkan coba lagi.');
              redirect('daftar');
```

```
public function login($a)
   $cek_email = $this->db->query("SELECT * FROM akun WHERE email='{$a['email']}';");
   $cek_username = $this->db->query("SELECT * FROM akun WHERE username='{$a['email']}';");
   if ($cek email->num rows()>0) {
      $pass = $cek email->row array();
      $cek_pass = password_verify($a['password'],$pass['password']);
      if ($cek pass) {
          $this->session->set_userdata('user',$a['email']);
          redirect():
          $this->session->set_flashdata('notif','Password Salah!');
   } elseif ($cek_username->num_rows()>0) {
       $pass = $cek_username->row_array();
       $cek_pass = password_verify($a['password'],$pass['password']);
       if ($cek_pass) {
          $this->session->set_userdata('user',$a['email']);
          redirect();
       } else {
          $this->session->set_flashdata('notif','Password Salah!');
          redirect();
      $this->session->set_flashdata('notif','Username atau Email Tidak Terdaftar!');
      redirect();
```

Kemudian meng-extend model Model_app yang sudah di buat dengan CI_Model. Kemudian membuat beberapa function pada model Model_Auth ini. Pada function yang di buat untuk mengambil data pada table akun. syntax \$this->db->query() di gunakan untuk mengambil data dari database dan nama table yang ingin di ambil data nya letakkan dalam parameternya.

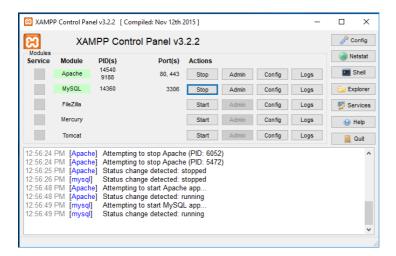
BAB V

MEMBUAT APLIKASI

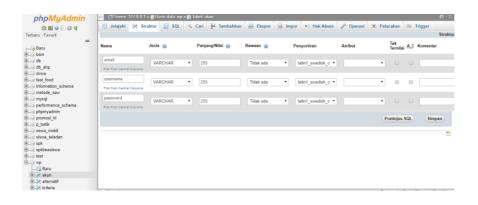
Sampailah kita ke bab terakhir, kali ini saya akan menjelaskan langkah demi langkah untuk menmbuat aplikasi ini di komputer anda, siapkan kopi dan cemilan yang banyak ya gan karna materi ini agar lumayan berat hehe... yuk langsung saja.

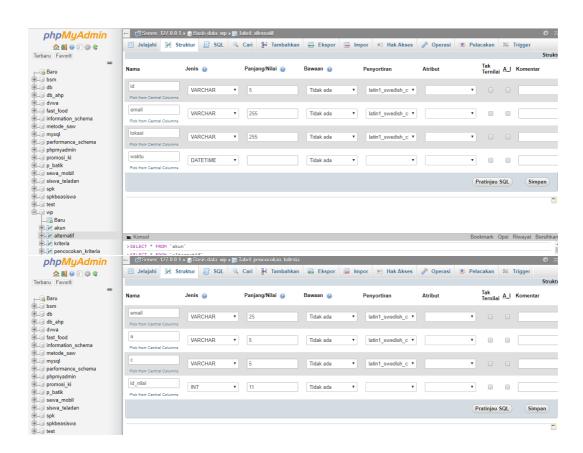
Langkah 1. Membuat Database

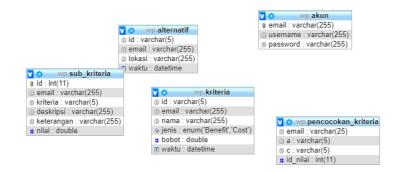
Setelah kita mengikuti langkah-langkah dibab sebelumnya pastikan lagi computer anda sudah terinstal software **XAMPP** dan **TEXT EDITOR** tentunya. Lalu buka aplikasi xampp dicomputer anda, setelah terbuka hidupkan apache web server dan mysql sebagai jembatan. Kemudian buka browser pribadi anda dan ketik https://localhost/phpmyadmin/

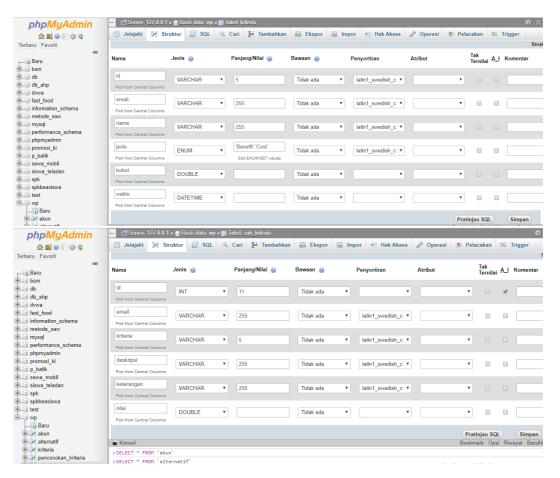


Maka kita akan masuk ke privat server seperti gambar dibawah









Langkah 2. Mengatur Config di CI

Kali ini saya akan membas bagaimana mengatur cofigurasi codeigniter agar lebih dinamis, mari kita langsung praktek saja.

waktu pertama kali saya menggunakan codeiniter, pada saat saya mau mengatur configurasi URL saya selalu menggunakan cara seperti ini,

Di folder config.php kita akan menemukan syntax seperti ini,

```
$config['base_url'] = 'http://localhost/nama_class';
```

Penamaan setelah localhost tergantung nama database yang anda buat, kali ini nama database yang saya buat adalah 'wp' bisa dilihat gambar dibawah ini

Langkah 3. Mengatur Cnfigurasi Autoload di CI

Pada tutorial kali ini kita akan belajar bagaimana cara **Mengatur Konfigurasi Autoload Pada Codeigniter**, dengan kita mengtur konfigurasi autoload kita bisa menggunakan package, libraries dan helpper yang suda di sediakan oleh codeigniter untuk project kita.

File ini berada di dalam folder application/config/autoload.php, secara default, akan menentukan apa saja yang secara otomatis akan di panggil atau di-load saat system berjalan. Untuk menjaga agar framework codeigniter tetap ringan dan cepat, maka default system hanya memanggil resource seminimal mungkin. Jika anda sering menggunakan library, driver, helper, language, model, ataupun file konfigurasi buatan sendiri, maka akan lebih baik untuk meletakkannya di dalam file konfigurasi autoload.php ini daripada melakukan pemanggilan secara berulang-ulang di dalam controller anda. Berikut ini adalah cara bagaimana kita melakukan pemanggilan otomatis dalam file konfigurasi autoload.php

Untuk memperkenalkan library kedalam Codeigniter kita menggunakan konfigurasi *Autoloads*, mari kita buka Autoloads.php yang terdapat di dalam directory yang sama dengan database.php. Di dalam autoloads terdapat tujuh konfigurasi yang dapat kita atur.

- Packages, digunakan untuk menambahkan package yang berisi custom resource ke dalam codeigniter.
- 2. Libraries, digunakan untuk menambahkan library kedalam codeigniter.
- Drivers, digunakan untuk menambahkan drivers, drivers adalah libraries yang memiliki parent class dan child class didalamnya.
- 4. Helpers, digunakan untuk menambahkan helpers yang diperuntungkan dalam membantu setiap task dan berisi function.
- Config, digunakan untuk menambahakan preferensi konfigurasi, dapat berasal dari config.php atau konfigurasi custom buatan sendiri.

- Language, digunakan untuk menambahkan file bahasa bahasa untuk meninternasionalisasikan sebuah web.
- 7. Model, digunakan jika kita ingin menambahkan model alternative kedealam controller.

Contoh cara autoload driver

Helper, Model dan Libraries

variabel ini digunakan untuk memmuat daftar helper yang akan dipanggil secara otomatis oleh system codeigniter. Contohpemanggilah helper adalah sebagai berikut:

```
$autoload['language'] = array();
    Auto-load Models
$autoload['model'] = array('model_auth','model_app');
🦈 autoload.php 🗴

    6. Language files
    7. Models

       $autoload['packages'] = array();
        application/libraries/ directory, with the addition of the 'database' library, which is somewhat of a special case.
       $autoload['libraries'] = array('database', 'session');
```

These classes are located in system/libraries/ or in your application/libraries/ directory, but are also placed ins

Langkah 3. Konek Database ke CI

kita akan membuat sebuah database dulu untuk contoh saja. agar kita bisa belajar membuat koneksi database dengan mysql. karena jika database nya belum di buat jadi php nya mau di hubungkan kemana? hehe.. pertama buatlah sebuah database dengan nama 'wp'. Setelah anda membuat database seperti yang saya instruksikan, selanjutnya kita akan mengkoneksikan database php dan mysql. Carny buatlah sebum file php dental nama koneksi.php kemudian simpan file tersebut pada direcroti localhost anda.

```
$active_group = 'default';
$query_builder = TRUE;

$db['default'] = array(
    'dsn' => '',
    'hostname' => 'localhost',
    'username' => 'root',
    'password' => '',
    'database' => 'wp',
    'dbdriver' => 'mysqli',
    'dbprefix' => '',
    'pconnect' => FALSE,
    'db_debug' => (ENVIRONMENT !== 'production'),
    'cache_on' => FALSE,
    'cachedir' => '',
    'char_set' => 'utf8',
    'dbcollat' => 'utf8_general_ci',
    'swap_pre' => '',
    'encrypt' => FALSE,
    'compress' => FALSE,
    'stricton' => FALSE,
    'failover' => array(),
    'save_queries' => TRUE
);
```

seperti penjelasan di atas. pada function mysql_connect() isikan nama host anda, username mysql, dan password mysql anda. password mysql saya di sini dikosongkan saja. jadi tidak saya isi dan username mysql saya root(default). dan nama host localhost. karena saya menggunakan localhost. mysql_connect adalah fungsi php untuk menghubungkan file php dengan host.dan memberikan settingan nama host, username dan password mysql.

Langkah 4. Membuat Model_app di Folder Model

Dalam pembuatan Web App, model direpresentasikan sebagai struktur data. Model berisi class yang mengandung function yang membantu menjalankan fungsi CRUD (Create, Read, Update dan Delete) terhadap suatu database.

Silahkan Kamu buat file php dengan nama mymodel.php dalam directory application/model, dan masukan kode berikut kedalamnya:

```
lic function tambah_kriteria($a,$b,$c)
 redirect('app');
   redirect('app');
                                                         Activate Windows
public function hapus_kriteria($a,$b)
 \label{lem:continuous} $$ query = $  his->db->simple_query("DELETE FROM kriteria WHERE email='$a' AND id='$b';"); if ($query) { } 
   $this->session->set_flashdata('notif','<div class="alert alert-success alert-dismissable">
              redirect('app');
 $query = $this->db->query("SELECT * FROM kriteria WHERE email='$a' ORDER BY waktu;");
return $query->result_array();
public function tambah_sub_kriteria($a,$b)
 $query = $this->db->simple_query("INSERT INTO sub_kriteria VALUES ('','$a','{$b['id']}','{$b['deskripsi']}','{$b['keterangan']}',($b['nilai']));");
if ($query) {
   redirect('app');
                                                         Activate Windows
```

```
$query = $this->do->query("SELECT * FROM sub_kriteria WHERE kriteria='$a' AND email='$b' ORDER BY nilai DESC;");
return $query->result_array();
$query = $this->db->query("SELECT * FROM alternatif MHERE email='$a' AND waktu=(SELECT max(waktu) FROM alternatif MHERE email='$a');");
if ($query->num_rows()>0) {
   $id = $query->num_rows();
   $id = substr($id'1al'), 1);
   $id=+;
   return 'A'.$id;
redirect('app');
```

```
$query = $this->db->query("SELECT * FROM alternatif WHERE email='$a' ORDER BY waktu;");
return $query->result_array();
public function tambah_pencocokan($a,$b,$c,$d)
    $query = $this->db->simple_query("INSERT INTO pencocokan_kriteria VALUES ('$a','$b','$c',$d);");
return $query;
public function edit_pencocokan($a,$b,$c,$d)
    $query = $this->db->simple_query("UPDATE pencocokan_kriteria SET id_nilai=$d WHERE email='$a' AND a='$b' AND c='$c';");
return $query;
public function cek_data_pencocokan_kriteria($a)
    $query = $this->db->query("SELECT * FROM pencocokan_kriteria WHERE email='$a';");
return $query->row_array();
public function data_pencocokan_kriteria($a,$b,$c)
    $query = $this->query("SELECT * FROM pencocokan_kriteria JOIN sub_kriteria WHERE pencocokan_kriteria.id_nilai=sub_kriteria.id AND pencocokan_k
return $query->row_array();
    $query = $this->db->query("SELECT id, bobot/total AS nilai, jenis FROM kriteria JOIN (SELECT sum(bobot) AS total FROM kriteria HMERE email='$a') ACTIVATE WINDOWS
public function pangkat($a,$b)
    Squery = Sthis->db->query("SELECT id, bobot/total AS nilai, jenis FROM kriteria JOIN (SELECT sum(bobot) AS total FROM kriteria NHERE email='$a') AS return Squery->row_array();
public function hasil($a,$b)
    $query = $this->db->query("SELECT * FROM alternatif WHERE email='$a' AND id='$b';");
return $query->row_array();
    $query = $this->db->query("SELECT * FROM pencocokan_kriteria WHERE email='$a' AND a='$b';");
return $query->num_rows();
public function untuk option($a,$b,$c)
   $query = $this->db->query("SELECT * FROM pencocokan_kriteria WHERE email='$a' AND a='$b' AND c='$c';");
return $query->row_array();
public function untuk_cek($a,$b,$c)
    $query = $this->db->query("SELECT * FROM pencocokan_kriteria WHERE email='$a' AND a='$b' AND c='$c';");
return $query->num_rows();
//-----
public function ganti_password($a,$b)
    $cek = $this->db->query("SELECT * FROM akun WHERE email='$a';")->row_array();
$cek_passlama = password_verify($b['passlama'],$cek['password']);
if ($cek_passlama) {
   if ($b['passbaru']=s$b['passvbaru']) {
      $passbaru = password_hash($b['passbaru'],PASSHORD_DEFAULT);
      $query = $this->db->simple_query("UPDATE akun SET password='$passbaru'
   if ($ouery) {
```

```
public function hapus_akun($a,$b)
{
     $cek = $this->db->query("SELECT * FROM akun WHERE email='$a';")->row_array();
$cek_pass = password_verify($b,$cek['password']);
if ($cek_pass) {
    $query = $this->db->simple_query("DELETE FROM akun WHERE email='$a';");
    if ($range) {
        redirect('app/pengaturan');
defined('BASEPATH') OR exit('No direct script access allowed');
class Model_app extends CI_Model {
      $username = $this->db->query("SELECT * FROM akun WHERE username='$a';");
      Susername = $this->db->query("Stitle!"
if ($username->num_nows()>8) {
    $email = $username->row_array();
    return $email['email'];
} else {
    return $a;
      $query = $this->db->query("SELECT * FROM kriteria WHERE email='$a' AND waktu=(SELECT max(waktu) FROM kriteria WHERE email='$a');");
if ($query->num_rows()>0) {
    $id = $query->row_array();
    $id = $query->row_array();
    $id = $ubstr($id['id'], 1);
    $id+:;
    return 'C'.$id;
} else {
   public function tambah_kriteria($a,$b,$c)
      </div>');
                                                                                                            Activate Wind
```

Langkah 5. Membat Controller Kriteria di CI

Kali ini saya akan menjelaskan tentang cara membuat controller pada codeigniter, pengaturan dasar dan cara menggunakan controller pada codeigniter, seperti yang sudah saya bayas di babbab sebelumnya bahwa codeigniter menggunakan metode MVC untuk menciptakan kode tau syntax yang clean. Diasumsikan untuk mengerti konsep OPP terlebih dahulu sebelum mulai dan mempelajari codeigniter. Karna codeigniter dibangun denna menggunakan konsep OPP (Object Oriented Programing).

Controller sesuai dengan artinya yang diartikan ke Bahasa indonesia adalah pengontrol atau pengatur. Yang dimaksud tersebut adalah controller yang berperan sebagai pengatur dari aksi pada aplikasi yang dibangun. Sepetti misalnya codeigniter controller yang berperan paling penting dari mulai mengirimkan parameter mengirimkan inputan, mengatur view dan model mengaktifkan tau memanggil library dan helper codeigniter dan masih banyak lagi peran controller untuk membangun sebuah aplikasi.

Secara default saat pertama kali menginstall codeigniter dan menjalankannya, controller codeigniter yang pertama kali di jalankan adalah controller_wellcome, saat anda menjalankan nya pada browser maka yang pertama kali di jalankan adalah controller welcome. hal ini dapat di lihat pada pengaturan routes codeigniter yang menetapkan controller welcome sebagai controller default yang di jalankan. buka project controller yang sudah anda letakkan pada localhost sesuai dengan tutorial di bab-bab sebelumnya. buka file pengaturan routes.php yang terletak di application/config/routes.php

Beirut adalah contoh peynambahan controller kriteria pada aplikasi yang kami buat :

```
blic function tambahsubkriteria()
     $email = $this->model_app->email($this->session->userdata('user'));
     $data = $this->input->post(array('id','deskripsi','keterangan','nilai'));
     $this->model_app->tambah_sub_kriteria($email,$data);
public function editsubkriteria($id)
     $data = $this->input->post(array('deskripsi', 'keterangan', 'nilai'));
     $this->model app->edit sub kriteria($id,$data);
public function hapussubkriteria($id)
     $this->model_app->hapus_sub_kriteria($id);
public function tambahalternatif()
     $email = $this->model_app->email($this->session->userdata('user'));
$data = $this->input->post('lokasi');
$id = $this->model_app->id_alternatif($email);
     $this->model_app->tambah_alternatif($id,$email,$data);
public function editalternatif($id)
     $email = $this->model_app->email($this->session->userdata('user'));
$data = $this->input->post('lokasi');
     $this->model_app->edit_alternatif($id,$email,$data);
public function hapusalternatif($id)
     $email = $this->model_app->email($this->session->userdata('user'));
     $this->model_app->hapus_alternatif($email,$id);
public function pencocokankriteria()
    $email = $this->model_app->email($this->session->userdata('user'));
    $a = $this->input->post('a');
$c = $this->input->post('c');
$nilai = $this->input->post('nilai');
    $i = 0;
    foreach ($nilai as $key) {
   $this->model_app->tambah_pencocokan($email,$a,$c[$i],$key);
         $i++;
    redirect('app/keputusan');
public function editpencocokankriteria()
    $email = $this->model_app->email($this->session->userdata('user'));
    $a = $this->input->post('a');
$c = $this->input->post('c');
$nilai = $this->input->post('nilai');
    foreach ($nilai as $key) {
    $cek = $this->model_app->untuk_cek($email,$a,$c[$i]);
         if ($cek==0) {
   $this->model_app->tambah_pencocokan($email,$a,$c[$i],$key);
         } else {
   $this->model_app->edit_pencocokan($email,$a,$c[$i],$key);
    redirect('app/keputusan');
public function gantipassword()
    $email = $this->model_app->email($this->session->userdata('user'));
$data = $this->input->post(array('passlama','passbaru','passvbaru'));
    $this->model_app->ganti_password($email,$data);
public function hapusakun()
{
    $email = $this->model_app->email($this->session->userdata('user'));
$data = $this->input->post('password');
```

Membuat view content_pengaturan di ci

```
div class="box-" div class="box-" div class="box-" div class="box-bader with-bonder" div class="box-bader" div class="box-bader" div class="box-bader" div class="box div
```

Membuat view content_keputusan di ci

```
| Pengsturan | (h1) | Pengsturan | (h1) | (h
```

```
ass="box">
class="box-header with-border">
class="box-title">Tabel Pencocokan Kriteria</h3:
     th:Alternatif-/th)

(2h)p formach (Skriteria as Skey): ?>

(th)<?= Skey['id'] ?>/th)

(th)Assi()th)
                                                                   Sdata_pencocokan = $this->model_app->data_pencocokan_kriteria($email,$keys['id'],$key['id']);
echo $data_pencocokan['nilai'];
                             <?php $cek_tombol = $this->model_app->untuk_tombol($email,$keys['id']); ?>
                                         | Class | Prince | American | Prince | Class | Prince | P
                                                       . Nobil 50
Class="modil fpde" [de"]=tr(?) $iepy['[de'] ?)* tabindex="1" role="dislog" aria-labelledby="myModallabel" aria-hiddens="true")
(div class="modil-dislog")
(div class="modil-basder")
(div class="modil-basder")
(div class="modil-basder")
(div class="modil-basder")
(div class="modil-title" [de"]myModallabel"-Percorolan=/Mod
(div class="modil-title" [de"]myModallabel"-Percorolan=/Mod
(div)
                                                                                                                                                                                                  | Comparison | Com
                                                                                                                                 sse"modal_content*?
class="modal_content*?
cbstlmodal_content*?
cbstlmodal_content*?
cbstlmon_type="button" class="close" data-dismiss="modal" aria-hidden="true")%times;</buttor
cbstlmss="modal_citile" id="myModallabel")Pencocotanr()hi
</pre>
                                                                                                                                                                                   ?php if (Sab briterial=NULL): >
chput type='text" name'n' value='cls Skeys('id'] >> hidden
chput type='text" name'n' value='cls Skeys('id'] >> hidden
chiput type='text" name'n' value='cls Skeys('id'] >> hidden
cdiv class='form.group'
clabel for='cls Skeys('id'] >> \class='form.corterl' id='cls Skeys('id'] >> "required
coption value=""--PAIN---(reption
class='form.corterl' id='cls Skeys('id'] >> "required
coption value=""--PAIN---(reption
class='cls Skeys('id') -> \class='cls Skeys('id') -> \closs='cls 
                                                                                                                                                                                                                                  <?php
$s_option = $this->model_app->untuk_option($email,$keys['id'],$subs_kriteria['kriteria']);
                                                                                                                                                                                                              | Option value='()- Subs_kriteria['id'] >> 'cpdp if(Subs_kriteria['id']=-$s_option['id_mliai'])(echo 'selected')) />>> Subs_kriteria['d'] echo 'selected')
```

```
| Spi = array_search(max($n), $n);
| $asx = $id[$i];
| "(tbody)
| "(tables)
| "(divcl:-/.box-body --)
| "(divcl:-/.box-bader with-border')
| "(divcl:-/.box-bad
```

```
cologomen's' class 'text-center'. Knoong /td )

cologomen's' class 'text-center'. Cl
```

Membuat view content_data di ci

```
| Bit | Data | Chass | Content - | Data | Chass | Chas
```

```
| data |
```

Membuat akun content daftar di ci

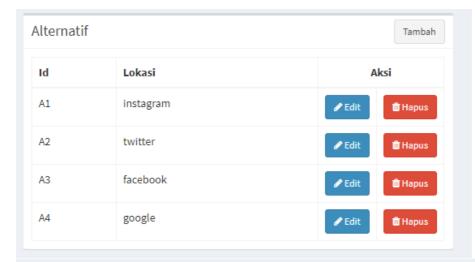
Membuat akun content_register di ci

```
| class="box-box-solid" | class="box-header with-border" | class="box-title">
| class="box-title">| class="box-title">| class="box-body">
| class="box-body">
| class="box-body">
| class="box-body">|
| class="form_open('ceklogin') ?>
| class="form_control-feedback"></span>
| class="form_open('ceklogin') ?>
| class="form_control-feedback"></span>
| class="form_open('ceklogin') ?>
```

Membuat akun content_akun di ci

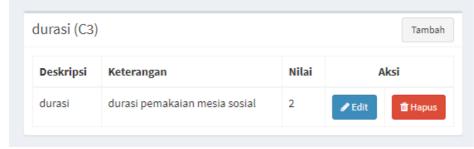
Tampilan web

Tampilan ini di gunakan untuk melihat, menghapus dan edit Kriteria





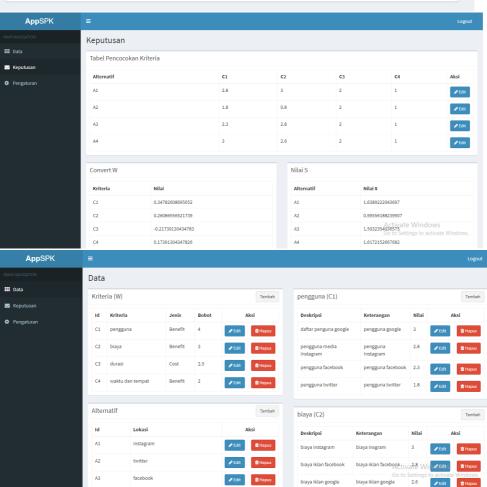






Activate Windows





Kriteria (W)

Tambah

Id	Kriteria	Jenis	Bobot	Aksi
C1	pengguna	Benefit	4	
C2	biaya	Benefit	3	
C3	durasi	Cost	2.5	
C4	waktu dan tempat	Benefit	2	

Nilai S

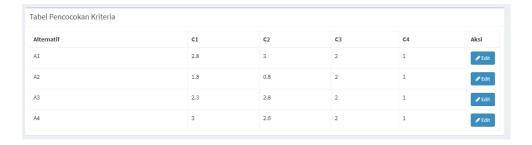
Alternatif	Nilai S
A1	1.6389222043697
A2	0.99556188239907
A3	1.5032354036575
A4	1.6172152667682

Hasil Keputusan

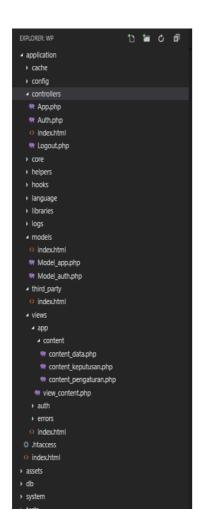
Hasil perhitungan menggunakan metode WP. Alternatif terbaik adalah ${\bf A1}$ yaitu ${\bf instagram.}$

Nilai V

Alternatif	Nilai V	
A1	0.28478554032621	
A2	0.17299273135191	
A3	0.26120807047868	
A4	0.28101365784321	

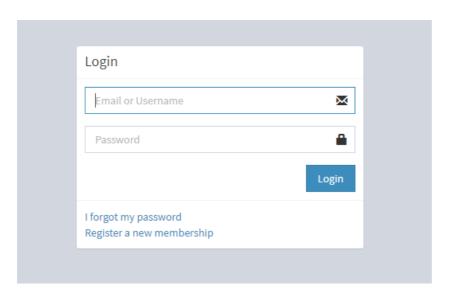


Tampilan folder prodect ci



Tampilan login

Digunakan admin untuk login



PROFIL PENULIS



Putri Taqwa Prasetyaningrum, S.T.,M.T. adalah alumni lulusan dari Teknik Informatika Universitas Atma Jaya Yogyakarta. Ia juga bekerja sebagai pengajar program studi Sistem Informasi Perguruan Tinggi swasta di Yogyakarta. Ia banyak melakukan penelitian di bidang Enterprise Information System, Data mining dan Artificial Intelligence. Di luar kesibukannya sebagai dosen, ia terlibat secara aktif dalam organisasi Muhammadiyah.



firdaus alfajar. Adalah mahasiswa Teknik Informatika Universitas mercubuna yogyakarta. Ia juga bekerja sebagai programmer,. Di luar kesibukannya sebagai mahasiswa, ia terlibat secara aktif dalam organisasi Suatu organisasi Mahasiswa.



Delila Fatmawati dalah mahasiswa Sistem informasi Universitas mercubuna yogyakarta. Ia juga bekerja sebagai penyusyun buku di project ini. Di luar kesibukannya sebagai sebagai karyawan swasta.



Akhmad Muzaki. Adalah mahasiswa teknik informatika di Universitas Mercu Buana Yogyakarta. Sebagai seorang mahasiswa aktif ia juga bekerja sebagai barista di sebuah kedai kopi, ia tidak banyak melakukan penelitian dan juga tidak terlibat dengan organisasi manapun.

Contac: twitter @_calonsuamimu ig

@zakiiku

email: akhmadmuzaki713@gmail.com



Dwi Agus Sutrisno. Adalah mahasiswa Sistem Informasi Universitas Mercubuana Yogyakarta. Ia juga bekerja sebagai analis sistem, website. Di luar kesibukannya sebagai mahasiswa, ia terlibat aktif dalam suatu organisasi mahasiswa



Muhammad Syahril adalah mahasiswa Teknik Informatika di Universitas Mercu Buana Yogyakarta. Sebagai seorang mahasiswa aktif, ia tidak banyak melakukan penelitian dan juga tidak terlibat dengan Organisasi manapun.

Contac: ig syahrilg97