

MODUL

WEB PROGRAMMING III



Disusun oleh:

MOHAMMAD SYAMSUL AZIS

**SEKOLAH TINGGI MANAJEMEN INFORMATIKA
DAN KOMPUTER NUSA MANDIRI
PROGRAM STUDI SISTEM INFORMASI
JAKARTA
2020**

KATA PENGANTAR

Segala puji syukur kepada Allah Yang Maha Esa atas berkat rahmat-Nya penulisan modul Mata Kuliah Web Programming III dapat diselesaikan dengan baik. Modul ini disusun untuk memenuhi kebutuhan mahasiswa pada mata kuliah Web Programming III yang disajikan dalam bentuk praktikum dan diharapkan dapat membekali mahasiswa dalam memahami Pembuatan Website dengan menggunakan Template CSS dari Template Bootstrap dan Framework dari Codeigniter 4.

Modul Web Programming III hanya ruang lingkup tentang halaman Administrator (Back-End) saja. Di akhir perkuliahan mahasiswa diharapkan mampu mengimplementasikannya dalam bentuk final project yang harus dipresentasikan sebagai syarat kelulusan mata kuliah Web Programming III. Teknik penyajiannya dilakukan secara terpadu dan sistematis.

Seperti layaknya sebuah modul, maka pembahasan dimulai dengan menjelaskan target pembelajaran yang hendak dicapai. Dengan demikian pengguna modul ini secara mandiri dapat mengukur tingkat ketuntasan yang dicapainya.

Penulis menyadari sepenuhnya bahwa modul ini tentu masih memiliki beberapa kekurangan. Untuk itu penulis dengan lapang dada menerima masukan dan kritik yang membangun dari berbagai pihak demi kesempurnaannya di masa mendatang. Semoga modul ini dapat bermanfaat bagi para pembacanya. Aamiin.

Jakarta, 20 Februari 2020

Mohammad Syamsul Azis

DAFTAR ISI

| | |
|---|-----|
| HALAMAN SAMPUL..... | i |
| KATA PENGANTAR..... | ii |
| DAFTAR ISI..... | iii |
| PERTEMUAN 1 PENJELASAN KONTRAK KULIAH DAN TUGAS..... | 1 |
| 1.1. Penjelasan Kontrak Kuliah dan Penegasan Tugas..... | 1 |
| PERTEMUAN 2 INSTALASI CODEIGNITER 4 DAN APLIKASI PENDUKUNG.. | 2 |
| 2.1. Instalasi XAMPP..... | 2 |
| 2.2. Instalasi Github..... | 3 |
| 2.3. Instalasi Composer..... | 4 |
| 2.4. Instalasi Codeigniter 4..... | 5 |
| PERTEMUAN 3 PEMBUATAN DATABASE..... | 31 |
| 3.1. Membuat Database Pustaka..... | 31 |
| PERTEMUAN 4 MVC DAN INSTALASI PLUGINS VISUAL STUDIO CODE..... | 37 |
| 4.1. MVC..... | 38 |
| 4.2. Instalasi Plugins Visual Studio Code..... | 39 |
| PERTEMUAN 5 PEMBUATAN LAYOUT..... | 44 |
| 5.1. Pembuatan Layout..... | 45 |
| PERTEMUAN 6 IMPELEMENTASI MODEL DAN CONTROLLER..... | 60 |
| 6.1. Model dan Controller..... | 62 |
| PERTEMUAN 7 REVIEW MATERI..... | 91 |
| PERTEMUAN 8 UJIAN AKHIR SEMESTER..... | 92 |
| PERTEMUAN 9 PAGINATION DAN SEARCHING..... | 93 |
| 9.1. Pagination..... | 106 |
| 9.2. Searching..... | 108 |
| PERTEMUAN 10 INSTALASI LIBRARY FORM LOGIN..... | 105 |
| 10.1. Update Composer..... | 106 |
| 10.2. Instalasi Myth/Auth..... | 108 |
| PERTEMUAN 11 IMPLEMENTASI FORM LOGIN DAN FORM REGISTRASI... | 116 |

| | |
|--|-----|
| 11.1. Implementasi Form Login dan Form Registrasi..... | 116 |
| PERTEMUAN 12 PEMBUATAN LOGOUT..... | 122 |
| 12.1. Pembuatan Menu Logout..... | 122 |
| PERTEMUAN 13-15 PRESENTASI PROJECT | 123 |
| PERTEMUAN 16 UJIAN AKHIR SEMESTER..... | 124 |
| DAFTAR PUSTAKA..... | 125 |

PERTEMUAN 1

PENJELASAN KONTRAK KULIAH DAN TUGAS

Deskripsi Pembelajaran

Pada bab ini dijelaskan bagaimana proses pembelajaran berlangsung setiap pertemuannya yaitu mahasiswa diarahkan untuk mandiri dengan diberikan tugas individu di setiap pertemuan yang mana tugas tersebut berupa mengimplementasikan dan mengerjakan tiap materi dan latihan yang ada pada pertemuan berikutnya. Sehingga pada saatnya pertemuan tersebut berlangsung, mahasiswa sudah mengetahui materi yang ada dan bisa fokus berdiskusi pada bagian materi yang masih kurang paham.

Tujuan Pembelajaran

Pada bab ini diharapkan mahasiswa dapat memahami kontrak perkuliahan yang harus diikuti dan jenis tugas yang akan diberikan selama satu semester ke depan.

1.1. Penjelasan Kontrak Perkuliahan dan Tugas

A. Tugas Projek Kelompok

1. Tugas project diadakan untuk memperoleh nilai dan pengganti UTS dan UAS. Tugas ini dikerjakan secara kelompok dengan 1 kelompok sebanyak 5 mahasiswa atau disesuaikan dengan jumlah mahasiswa dalam satu kelas.
2. Bentuk tugas projek adalah mencari aplikasi web nativ (tidak menggunakan framework) yang sudah jadi kemudian di transformasikan menjadi aplikasi web framework CodeIgniter 4 dan membuat analisa kebutuhan user dan sistem dari web tersebut.
3. Aplikasi web nativ harus ditunjukan ke dosen pengampu mata kuliah, sebagai bukti bahwa aplikasi web nativ memang ada dan dapat dijalankan.
4. Aplikasi web nativ diperbolehkan didapat dari berbagai sumber.

B. Tugas Mingguan (Tugas Individu)

1. Bentuk tugas mingguan adalah mengimplementasikan materi tiap pertemuan yang ada di modul dimulai dari materi pertemuan ke 2 sampai materi pertemuan terakhir.
2. Tugas mingguan dikerjakan dari minggu sebelumnya lalu disetorkan di minggu berikutnya.

PERTEMUAN 2

INSATALASI CODEIGNITER 4 DAN APLIKASI PENDUKUNG

Deskripsi Pembelajaran

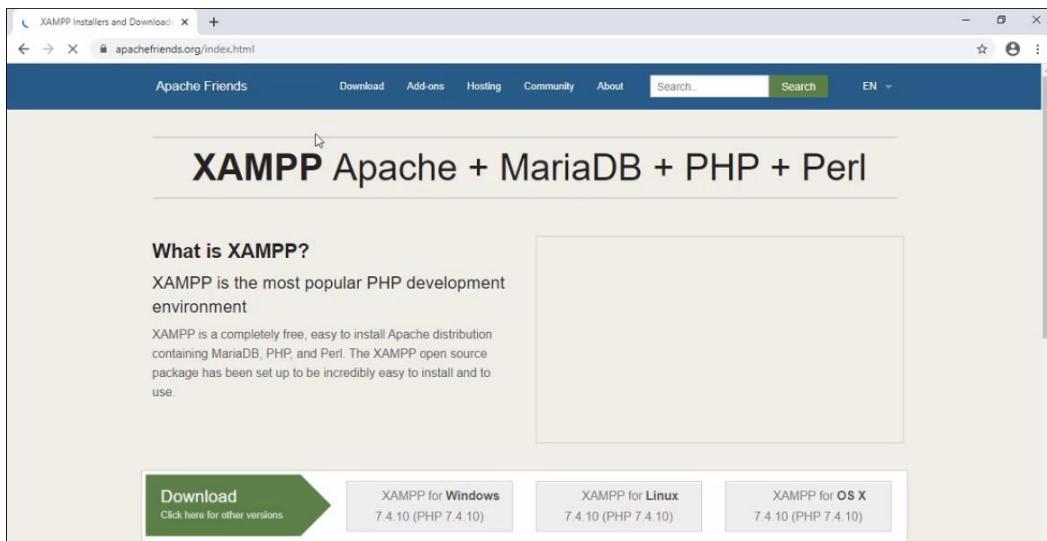
Pada bab ini akan dibahas tentang bagaimana cara download/mengundung dan memasang/instalasi framework codeigniter 4, xampp, git hub, dan composer. Dan mengenal struktur folder dari framework codeigniter 4 sendiri.

Tujuan Pembelajaran

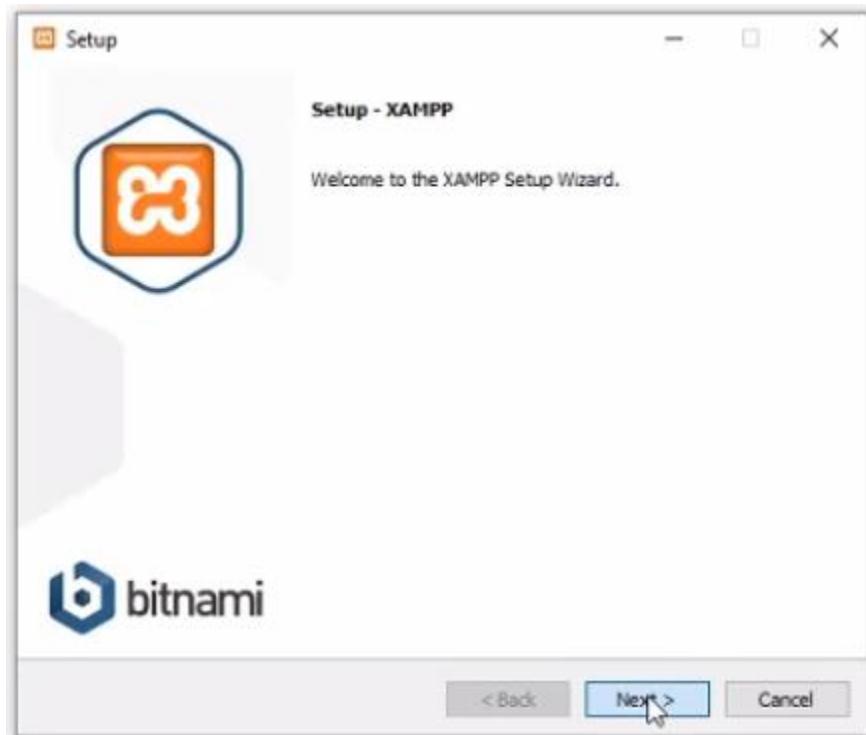
Pada bab ini diharapkan mahasiswa dapat memahami materi-materi fundamental yang diperlukan sebelum mempelajari framework codeigniter, yaitu bagaimana cara melakukan instalasi perangkat-perangkat lunak yang diperlukan untuk dapat bekerja dengan codeigniter seperti PHP, web server, database server, dan framework codeigniter itu sendiri. Mahasiswa juga diharapkan dapat memahami pola desain atau arsitektur dari Model-View-Controller (MVC), yang merupakan syarat mutlak dalam proses pengembangan aplikasi menggunakan codeigniter 4 ini.

2.1. Instalasi XAMPP

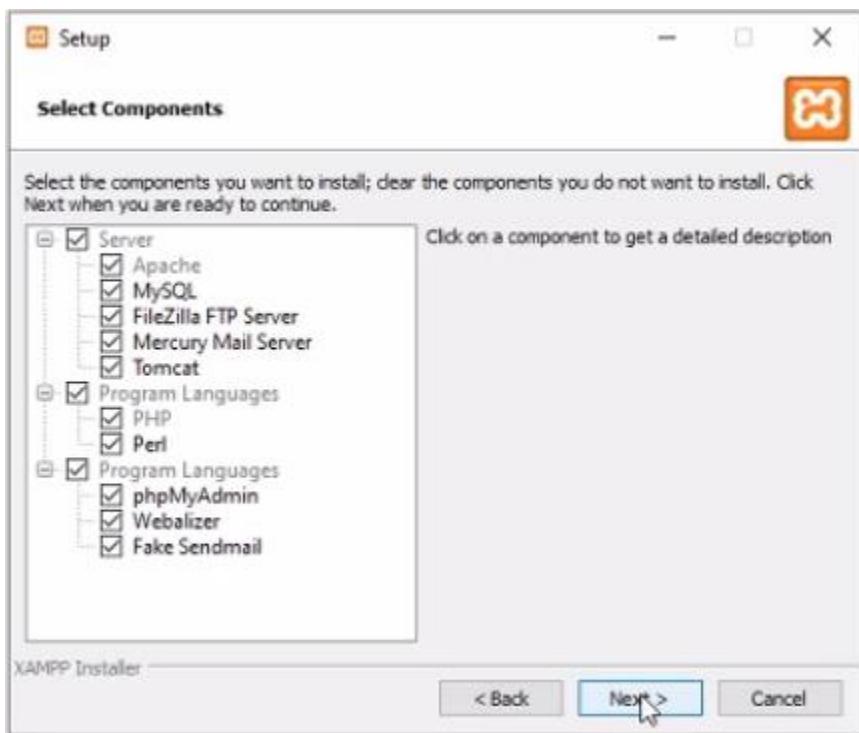
Xampp adalah aplikasi server yang nantinya digunakan untuk membuat database jenis MySQL. Selain MySQL yang digunakan di XAMPP ini juga digunakan fungsi dari Apache. Yang keduanya dapat diaktifkan melalui control panel xamp. Untuk mendukung codeigniter 4 dibutuh xampp versi terbaru minimal versi 7. Jika computer belum terinstalasi software XAMPP ini, maka silahkan download aplikasi xampp-nya di web resmi yakni di url: apachefriends.org Setelah itu pilih yang XAMPP for Windows jika menggunakan sistem operasi computer windows, jika bukan windows bisa disesuaikan dengan sistem operasi yang sudah terpasang. Untuk lebih jelasnya bisa ikuti tahapan-tahapanya di halaman berikutnya.



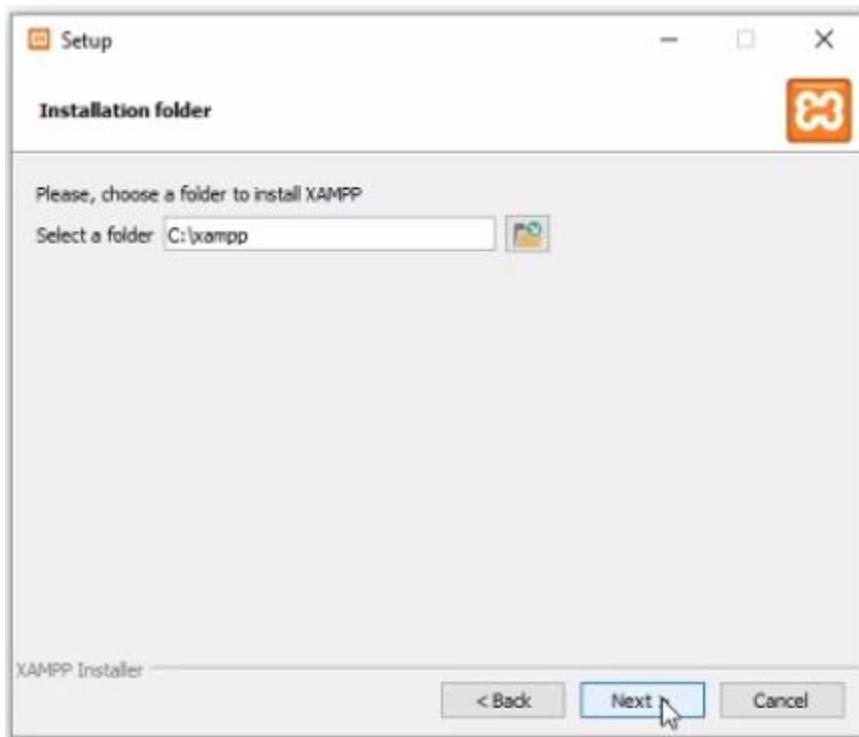
Setelah berhasil diunduh langsung bisa diinstall pada komputer dengan cara double click atau klik dua kali aplikasi XAMPP tersebut. Maka akan tampil seperti gambar di bawah ini:



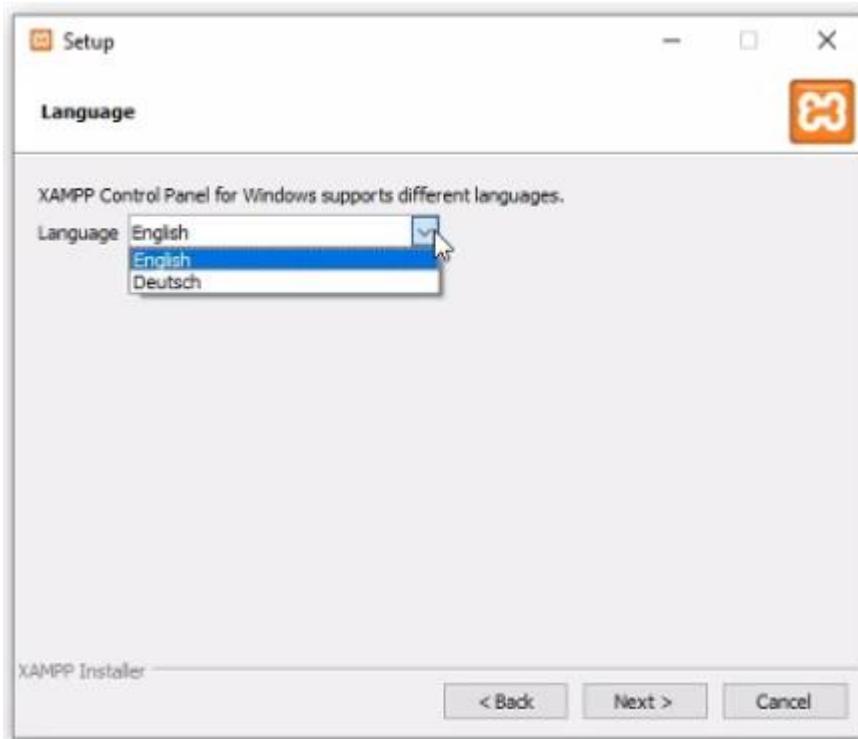
Lanjutkan dengan tekan tombol Next.



Tekan Next kembali.



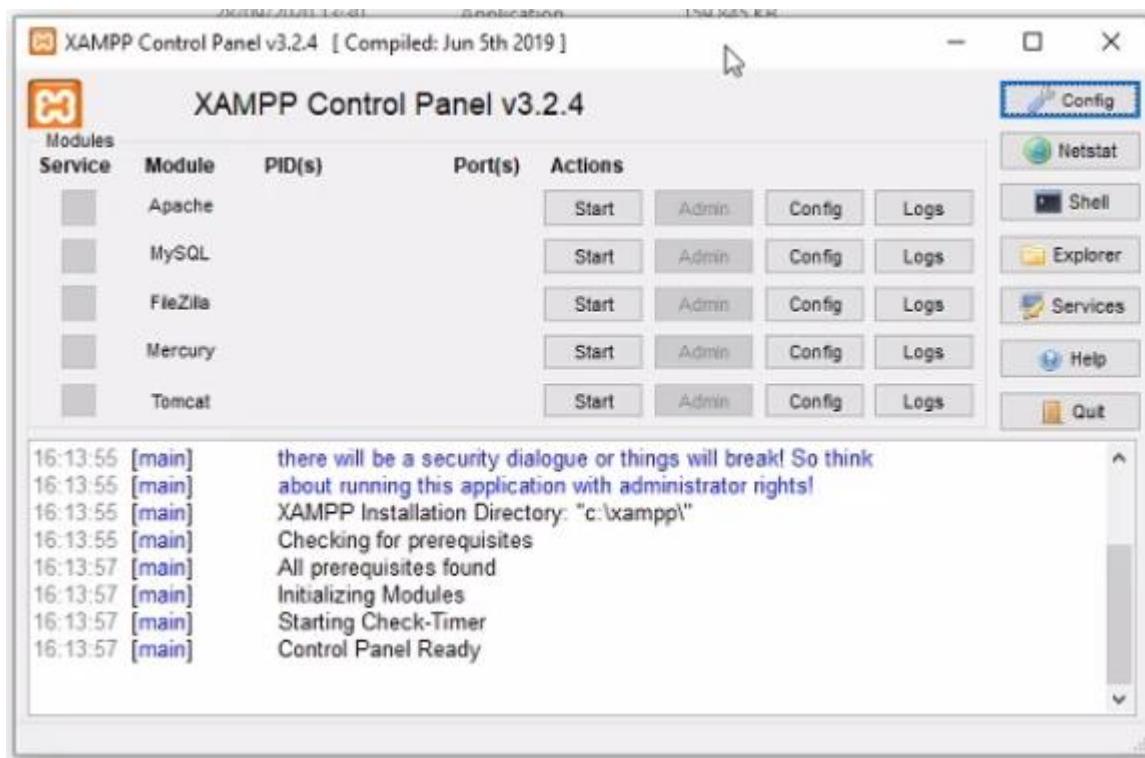
Untuk folder install posisi di disk C , lalu tekan tombol Next kembali.



Pilih bahasanya English, kemudian tekan tombol Next lagi. Terus pilih tombol next hingga tampil tampilan proses loading seperti gambar di bawah ini:



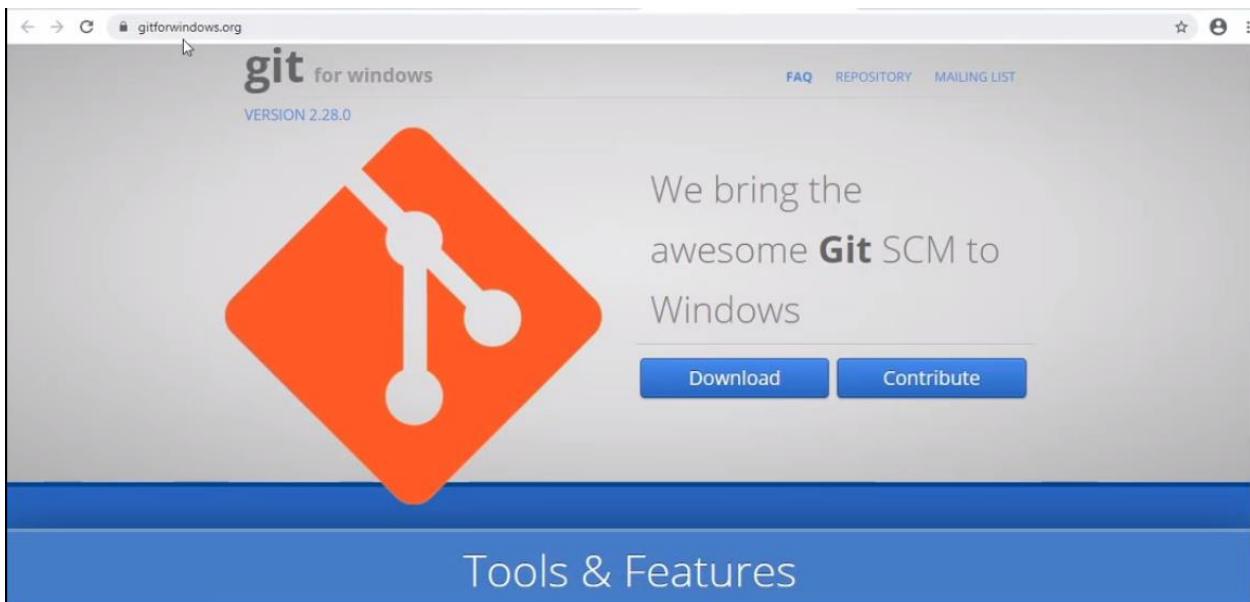
Jika prosesnya instalasinya sudah selesai maka akan tampil tampilan sebagai berikut:



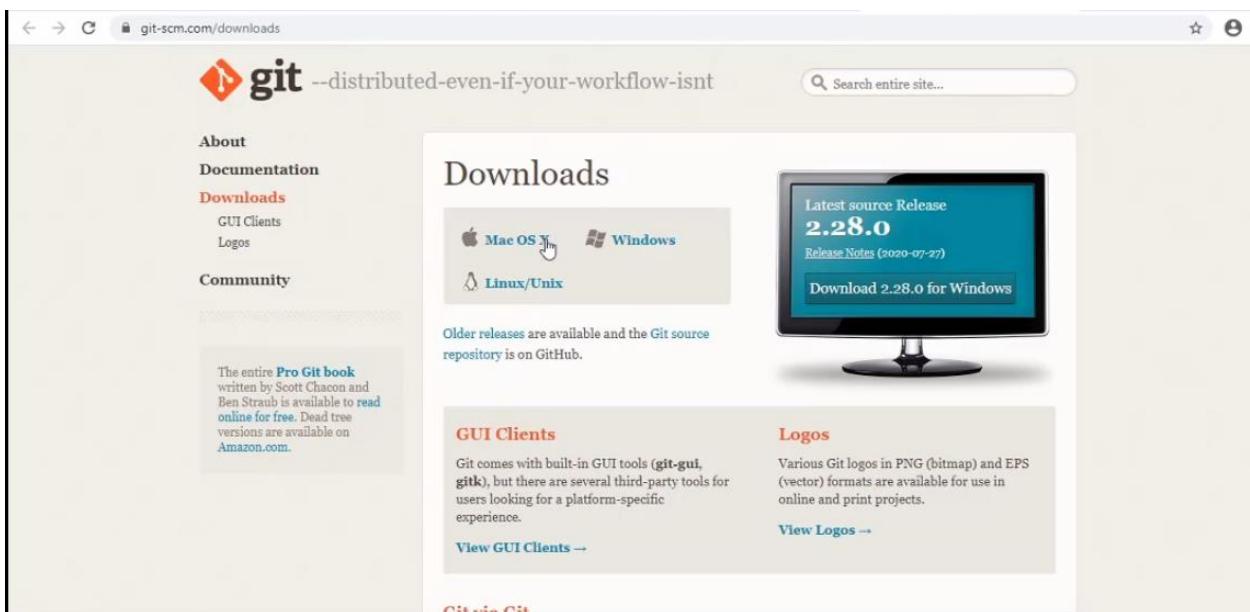
Tampilan di atas disebutnya XAMPP Control Panel. XAMPP Control Panel inilah yang nantinya akan sering dibuka untuk mengaktifkan Apache dan MySQL. Jadi hanya dua aksi yang dibutuhkan dalam pembuatan web ini, yaitu Apache dan MySQL saja, yakni dengan klik start pada kedua tombol Apache dan MySQL-nya. Dengan begitu Apache dan MySQL sudah dapat berfungsi.

2.2. Instalasi Git Bash

Kunjungi web resmi dari git bash seperti gambar di bawah ini. Lalu download.



Kemudian pilih dengan menyesuaikan sistem operasi yang sudah terpasang di komputer.



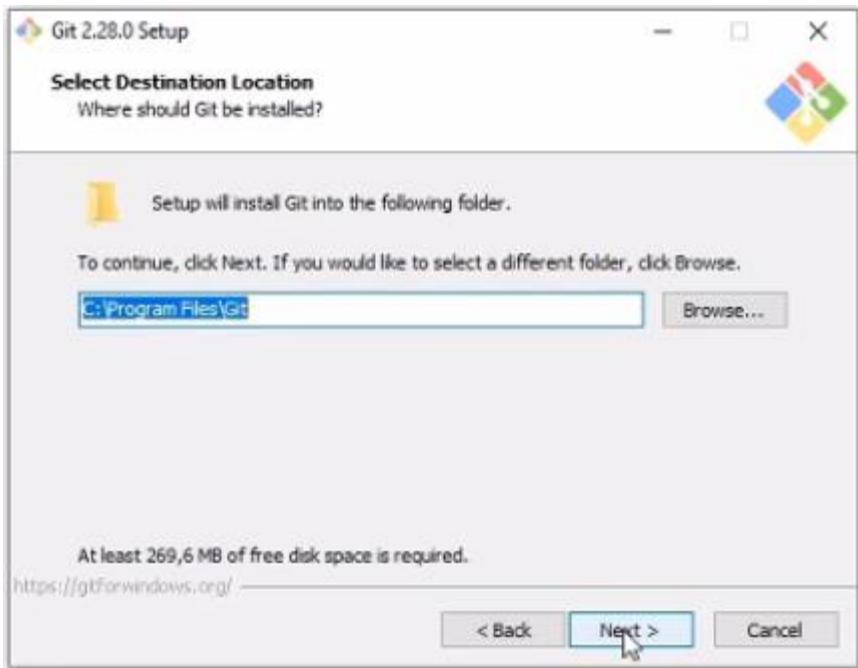
Setelah berhasil diunduh, langsung install aplikasi dengan mengikuti tahapan sebagai berikut:



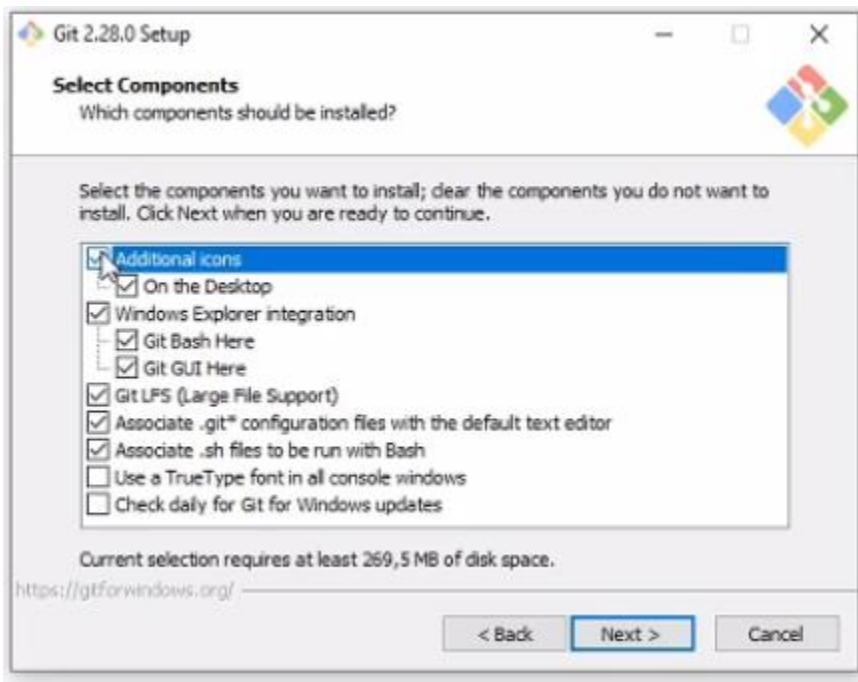
Ketika mentahan aplikasi Git Bash dibuka maka akan tampil jendela seperti diatas lalu klik run. Perlu diketahui untuk instalasi git bash ini dibutuhkan internet. Karena ada beberapa part git bash membutuhkan koneksi jaringan/internet. Jika tidak tersedia jaringan internet maka instalasi akan gagal.



Tampilan selanjutnya pilih Next.



Pastikan penyimpanan ada pada di disk C di dalam folder program files. Selain disk C pun boleh asal disk tersebut terdapat folder program files di dalamnya.



Checklist pada additional icons jika memang icon dari aplikasi git bash-nya ingin ditampilkan di dekstop komputer. Kemudian sampai akhir proses intalasi klik next saja.



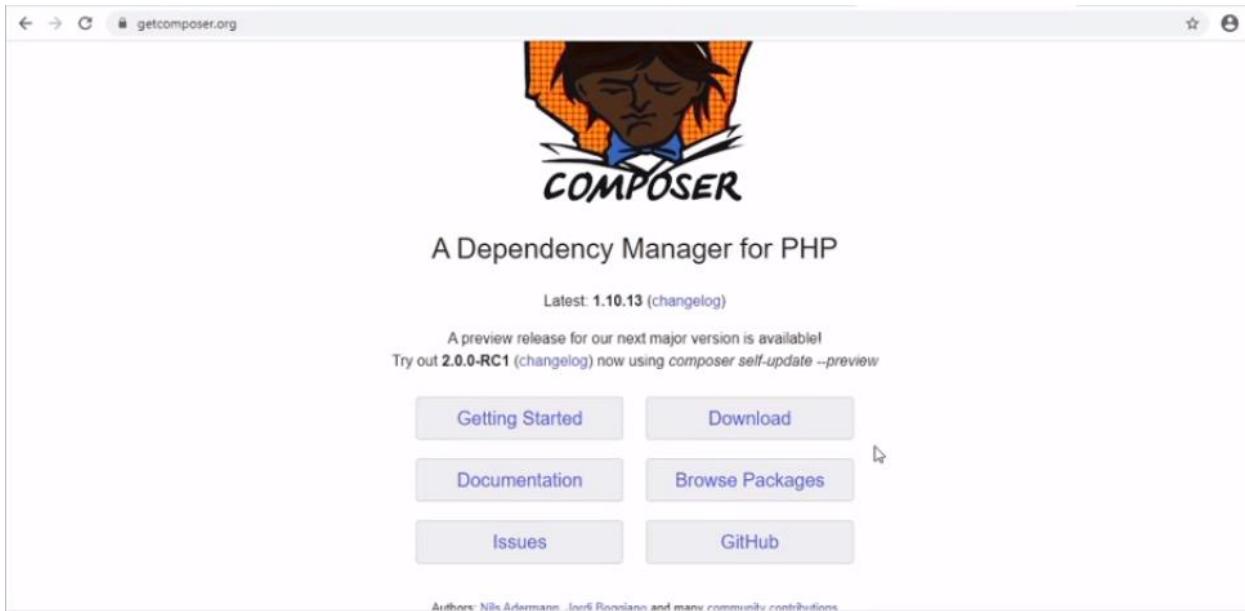
Apabila proses intalasinya berakhir tampilan terakhir seperti gambar di atas. Jika aplikasinya langsung ingin dijalankan bisa checklist pada check box Launch Git Bash. Tetapi jika tidak bisa di unchecklist saja.

Adapun tampilan git bash yang sudah berhasil diinstalasi bisa dilihat pada gambar di bawah ini:

A screenshot of a terminal window titled "MINGW64:/c/Users/user". The window shows a command prompt with the user's name and computer name: "user@DESKTOP-78MT4FJ MINGW64 ~". Below the title bar is a black command line area where the user can type commands. A small vertical scroll bar is visible on the right side of the terminal window.

2.3. Instalasi Composer

Silahkan masuk ke web resmi composer, untuk urlnya bisa dilihat pada gambar dibawah ini:

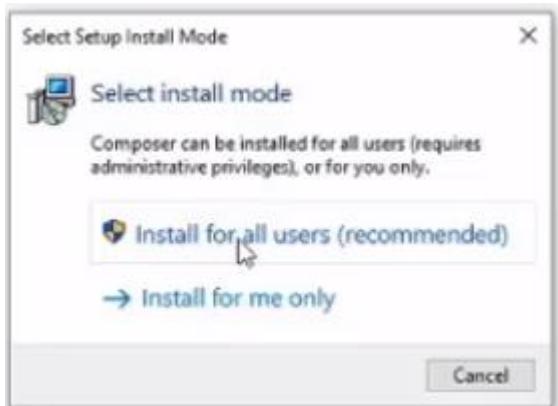


Kemudian pilih donwload.

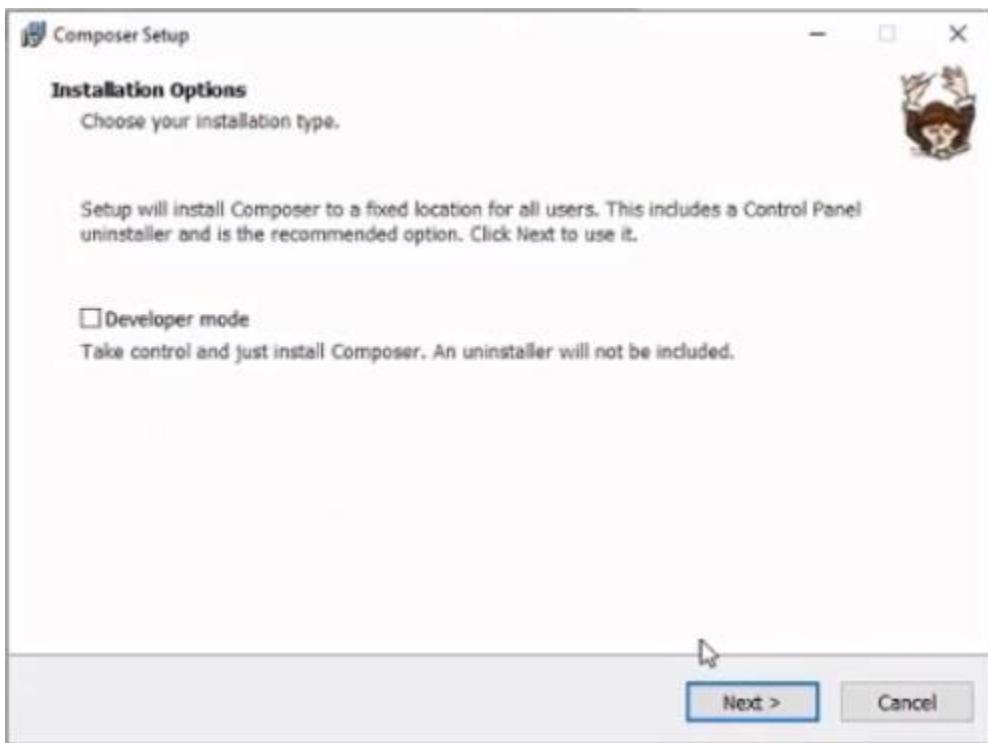
A screenshot of a web browser displaying the "Download Composer" page at getcomposer.org/download/. The page title is "Download Composer" with "Latest: v1.10.13" in parentheses. It features two main sections: "Windows Installer" and "Command-line installation". The "Windows Installer" section contains a note about requiring PHP to be installed and provides a link to download the "Composer-Setup.exe" installer. The "Command-line installation" section provides instructions for quickly installing Composer via terminal and includes a code block showing the command to run:

Setelah tampilan seperti di atas pilih yang Composer-Setup.exe.

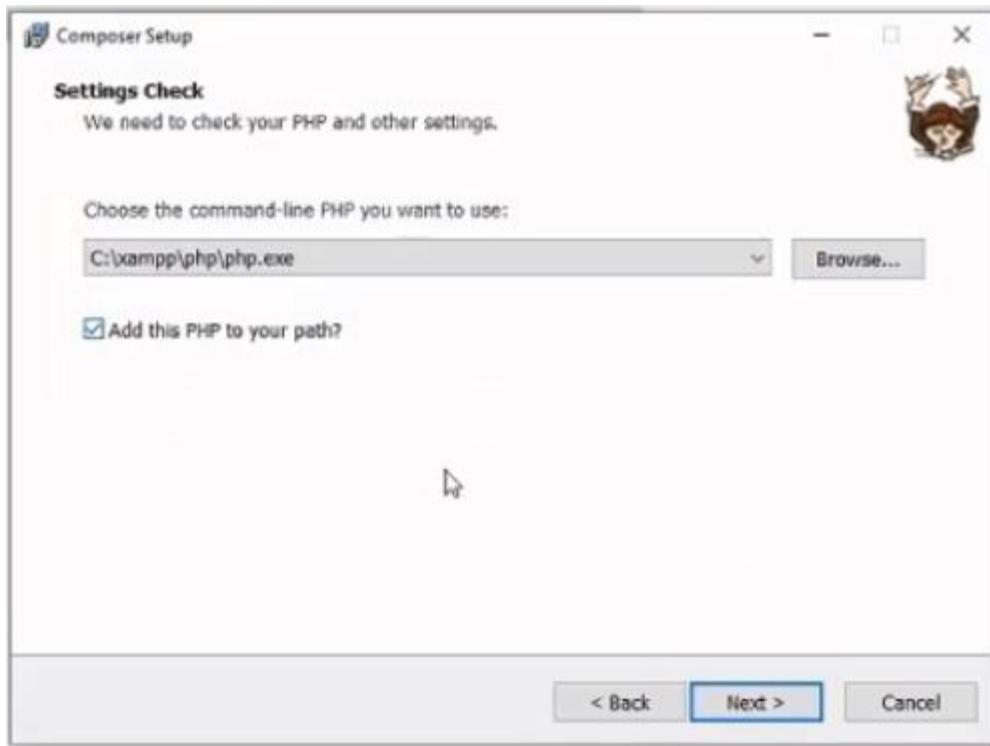
Saat install composer akan tampil tampilan di bawah ini:



Kemudian pilih Install for all users (recommended)



Lalu pilih Next.



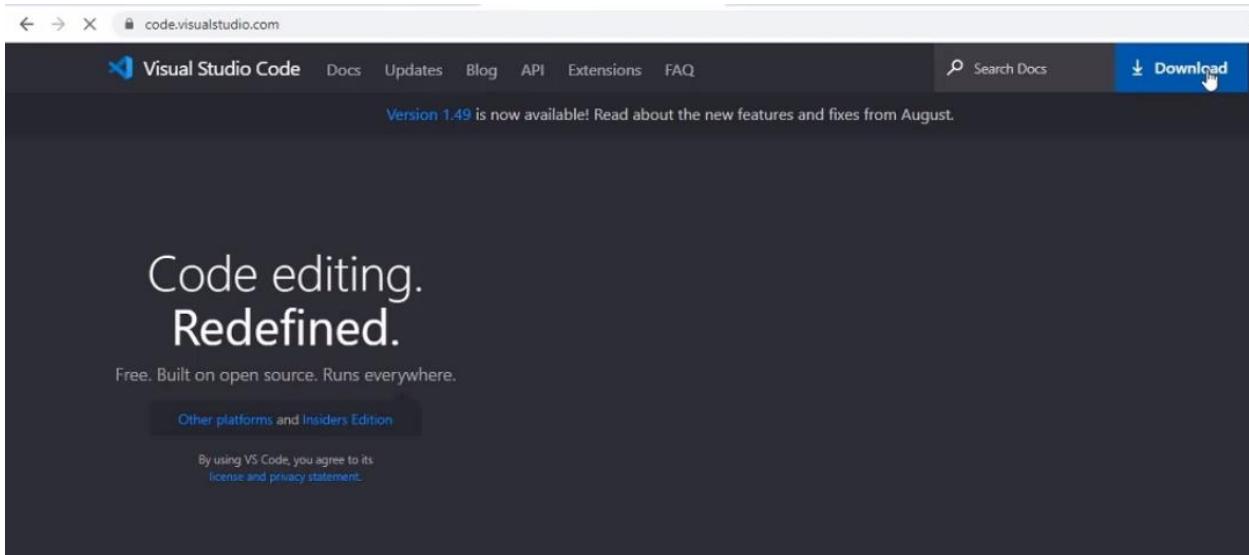
Seperti instalasi sebelumnya gunakan disk C untuk folder penyimpanannya dan checklist check box add this PHP your path. Kemudian pilih Next. Pastikan sebelum instalasi composer XAMPP sudah terinstall terlebih dahulu dikarenakan composer membutuhkan php.exe. Dan jangan lupa koneksi jaringan internet saat instalasi composer ini.



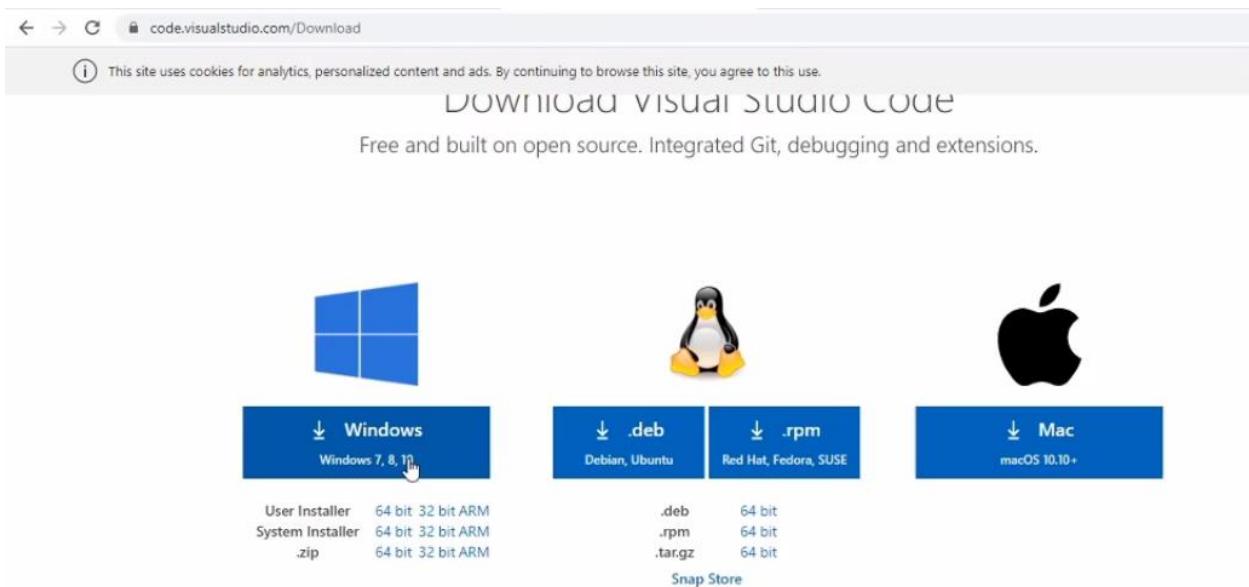
Kemudian next lagi hingga ketemu tombol finish seperti ini. Dan composer pun selesai diinstall.

2.4. Instalasi Visual Studio Code

Visual studio code disini dibutuhkan sebagai code editor. Selain visual studio code, ada atom dan juga sublime yang bisa digunakan code editor. Untuk kali ini digunakan visual studio code untuk editor yang memang sering digunakan oleh programmer saat ini dalam membuat web menggunakan codeigniter 4. Untuk tahap instalasi visual studio code dimulai setelah mengunduh mentahannya terlebih dahulu di web resmi. Untuk alamat websitenya bisa cek pada gambar di bawah ini:

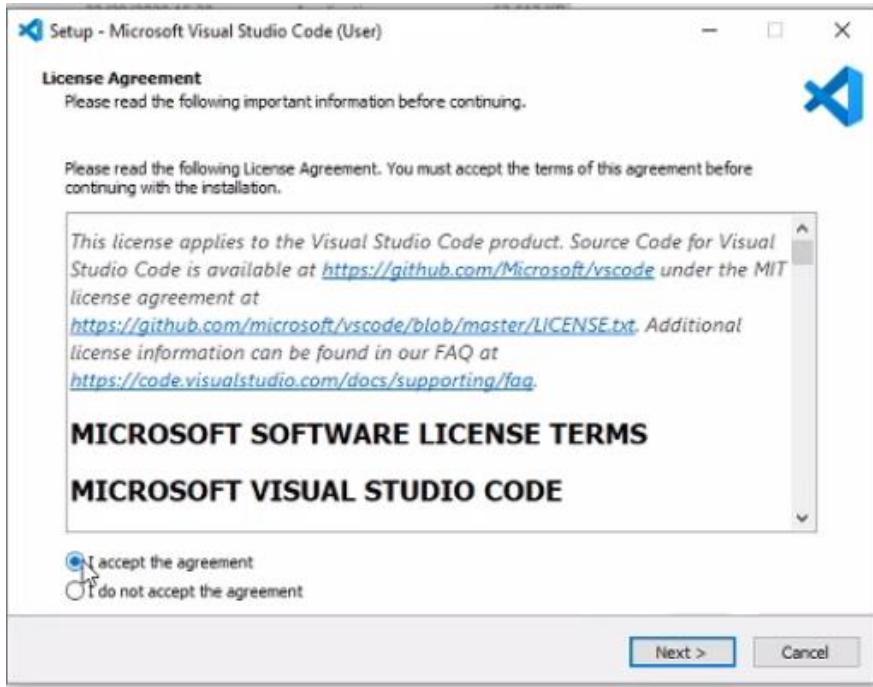


Setelah berhasil ke web resminya bisa langsung klik download.



Pilih sistem operasinya yang sesuai dengan yang sudah terpasang di komputer.

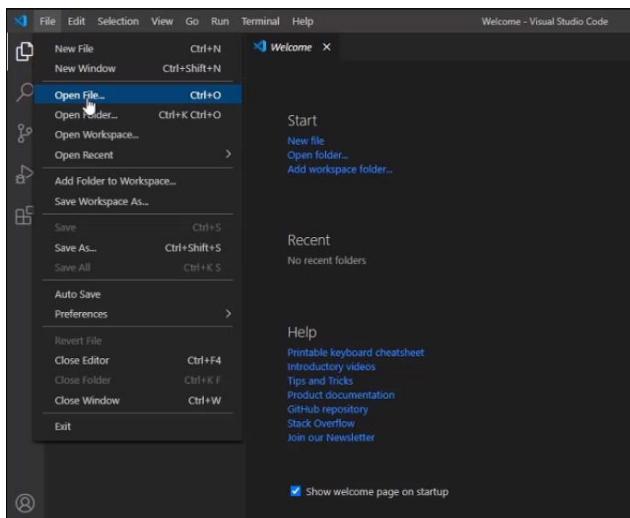
Saat instalasi awal, akan muncul tampilan sebagai berikut:



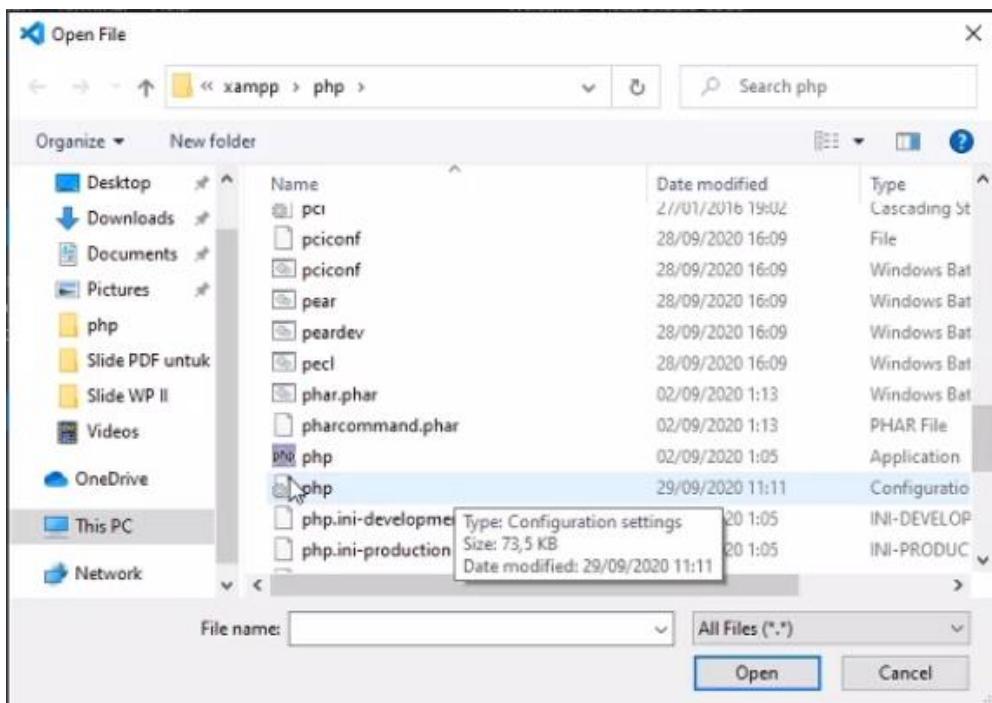
Pilih I accept the agreement, kemudian pilih Next hingga ketemu dengan tombol finish.

2.5. Instalasi Codeigniter 4

Sebelum melakukan instalasi codeigniter 4 nya ada tahapan yang harus dilakukan terlebih dahulu agar framework ini dapat digunakan pada komputer tanpa kendala. Tahapannya sebagai berikut:



Buka visual studio code, lalu pilih file -> open file.

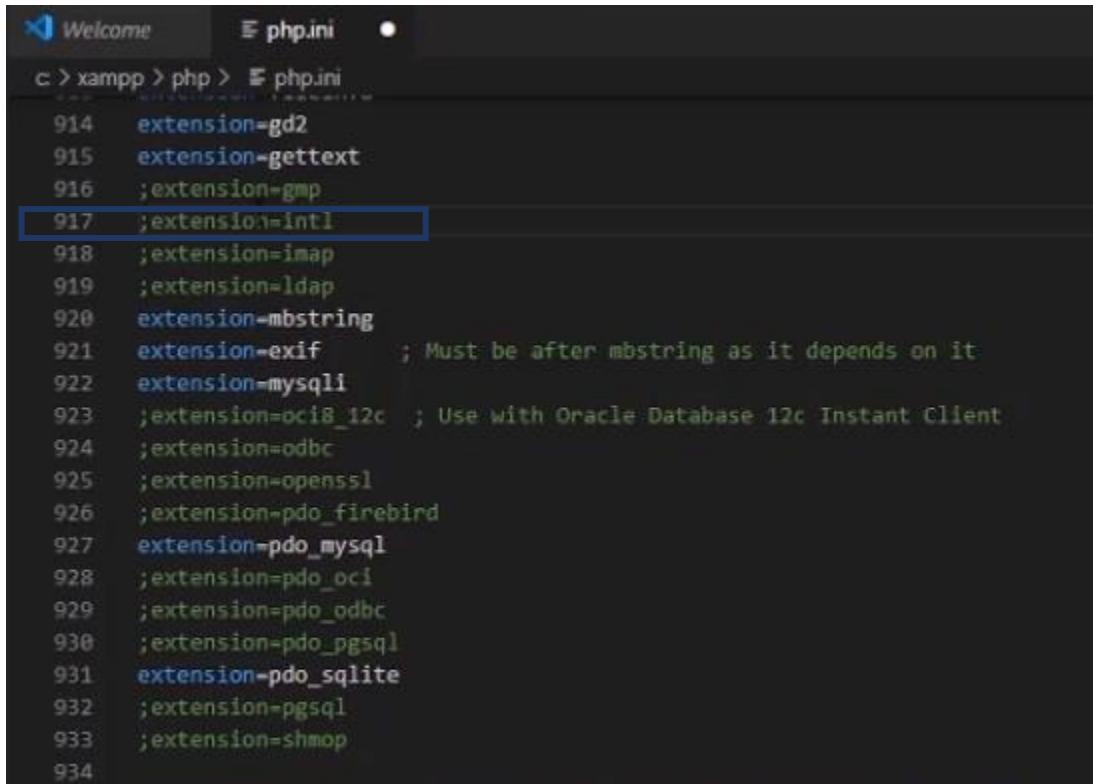


Kemudian masuk ke disk C -> xampp -> php. Di dalam folder php cari file dengan nama php yang memiliki format konfigurasi seperti yang ditunjuk seperti gambar di atas.
Setelah berhasil dibuka di visual studio code tampilannya akan seperti ini:

```
c:\xampp\php> php.ini
914 extension=gd2
915 extension=gettext
916 ;extension=gmp
917 extension=intl
918 ;extension=imap
919 ;extension=ldap
920 extension=mbstring
921 extension=exif ; Must be after mbstring as it depends on it
922 extension=mysqli
923 ;extension=oci8_12c ; Use with Oracle Database 12c Instant Client
924 ;extension=odbc
925 ;extension=openssl
926 ;extension=pdo_firebird
927 extension=pdo_mysql
928 ;extension=pdo_oci
929 ;extension=pdo_odbc
930 ;extension=pdo_pgsql
931 extension=pdo_sqlite
932 ;extension=pgsql
933 ;extension=shmop
934
935 ; The MIBS data available in the PHP distribution must be installed.
936 ; See http://www.php.net/manual/en/snmp.installation.php
937 ;extension=snmp
```

Nama filenya adalah php.ini.

Kemudian hilangkan tanda ; (titik koma) di depan script extension=intl letaknya ada pada baris ke 917 untuk lebih jelasnya bisa lihat gambar di bawah ini:



```
c:\xampp>php> php.ini
914 extension=gd2
915 extension=gettext
916 ;extension=gmp
917 ;extension=intl
918 ;extension=imap
919 ;extension=ldap
920 extension=mbstring
921 extension=exif ; Must be after mbstring as it depends on it
922 extension=mysqli
923 ;extension=oci8_12c ; Use with Oracle Database 12c Instant Client
924 ;extension=odbc
925 ;extension=openssl
926 ;extension=pdo_firebird
927 extension=pdo_mysql
928 ;extension=pdo_oci
929 ;extension=pdo_odbc
930 ;extension=pdo_pgsql
931 extension=pdo_sqlite
932 ;extension=pgsql
933 ;extension=shmop
934
```

Jika sudah berubah maka script tersebut akan berubah seperti gambar di bawah ini:

```
Welcome      php.ini      X
c > xampp > php > php.ini
914 extension=gd2
915 extension=gettext
916 ;extension=gmp
917 extension=intl
918 ;extension=imap
919 ;extension=ldap
920 extension=mbstring
921 extension=exif      ; Must be after mbstring as it depends on it
922 extension=mysqli
923 ;extension=oci8_12c  ; Use with Oracle Database 12c Instant Client
924 ;extension=odbc
925 ;extension=openssl
926 ;extension=pdo_firebird
927 extension=pdo_mysql
928 ;extension=pdo_oci
929 ;extension=pdo_odbc
930 ;extension=pdo_pgsql
931 extension=pdo_sqlite
932 ;extension=pgsql
933 ;extension=shmop
```

Setelah konfigurasi php.ini sudah berhasil lalu tahapan selanjutnya masuk ke web resmi daripada codeigniternya.

The screenshot shows the official CodeIgniter website at codeigniter.com. The page features a large orange header bar with the CodeIgniter logo and navigation links for Home, Download, Documentation, Community, and Contribute. Below the header is a large banner with a black flame icon on the left and the text "CodeIgniter Rocks" in white. A brief description follows: "CodeIgniter is a powerful PHP framework with a very small footprint, built for developers who need a simple and elegant toolkit to create full-featured web applications." Below the banner are four call-to-action boxes: "Download" (with a download icon), "Read the Manual" (with a book icon), "View the Forums" (with a user icon), and "On Github" (with a GitHub icon). At the bottom of the page, there's a note: "The latest is Version v4.0.4".

Kemudian pilih Menu Bar Documentation.

The screenshot shows the official CodeIgniter website. At the top, there's a navigation bar with links for Home, Download, Documentation (which is highlighted), Community, and Contribute. Below the navigation bar is a large orange header section featuring the CodeIgniter logo (a stylized flame icon) and the text "CodeIgniter Rocks". A sub-header below it reads: "CodeIgniter is a powerful PHP framework with a very small footprint, built for developers who need a simple and elegant toolkit to create full-featured web applications." There are also social media links for GitHub, Star count (3,045), Fork count (1,162), Twitter Follow (23.3K followers), and a download link. Below this, there are four buttons: "Download" (with a download icon), "Read the Manual" (with a book icon), "View the Forums" (with a user icon), and "On Github" (with a GitHub icon). The "Download" button has a tooltip indicating "The latest is Version v4.0.4".

Lalu pilih button Online UG (User Guide)

The screenshot shows a comparison between the documentation for CodeIgniter 4 and CodeIgniter 3. Both sections have a title and a brief description. Below each title are three buttons: "Online UG" (highlighted with a mouse cursor), "Offline UG", and "API Docs".

| CodeIgniter Version | Description | Buttons |
|---------------------|--|---|
| CodeIgniter 4 | CodeIgniter 4 is the upcoming version of the framework, licensed under the MIT License . | Online UG (highlighted), Offline UG, API Docs |
| CodeIgniter 3 | CodeIgniter 3 is the current version of the framework, licensed under the MIT License . | Online UG, Offline UG, Epub UG |

Pilih installation pada sidebar.

CodeIgniter

Search docs

Welcome to CodeIgniter4

Installation

Build Your First Application

CodeIgniter4 Overview

General Topics

Controllers and Routing

Building Responses

Working With Databases

Modeling Data

Managing Databases

Library Reference

Helpers

Testing

Command Line Usage

Extending CodeIgniter

Docs » CodeIgniter4 User Guide

CodeIgniter4 User Guide

- Welcome to CodeIgniter4
 - Welcome to CodeIgniter4
 - Server Requirements
 - Credits
 - PSR Compliance

Getting Started

- Installation
 - Manual Installation
 - Composer Installation
 - Running Your App
 - Upgrading From a Previous Version
 - Troubleshooting
 - CodeIgniter Repositories

Tahap selanjutnya pilih yang composer installation karena sebelumnya kita sudah instalasi yang composernya.

Welcome to CodeIgniter4

Installation

Manual Installation

Composer Installation

Running Your App

Upgrading From a Previous Version

Troubleshooting

CodeIgniter Repositories

Build Your First Application

CodeIgniter4 Overview

General Topics

Controllers and Routing

Building Responses

Working With Databases

Modeling Data

Managing Databases

Library Reference

Helpers

Docs » Installation

Installation

CodeIgniter4 can be installed in a number of different ways: manually, using [Composer](#), or using [Git](#). Which is right for you?

- If you would like the simple "download & go" install that CodeIgniter3 is known for, choose the manual installation.
- If you plan to add third party packages to your project, or want to keep CodeIgniter up to date easily, we recommend the Composer installation.

- Manual Installation
- Composer Installation
- Running Your App
- Upgrading From a Previous Version
 - Upgrading from 3.x to 4x
- Troubleshooting
- CodeIgniter Repositories

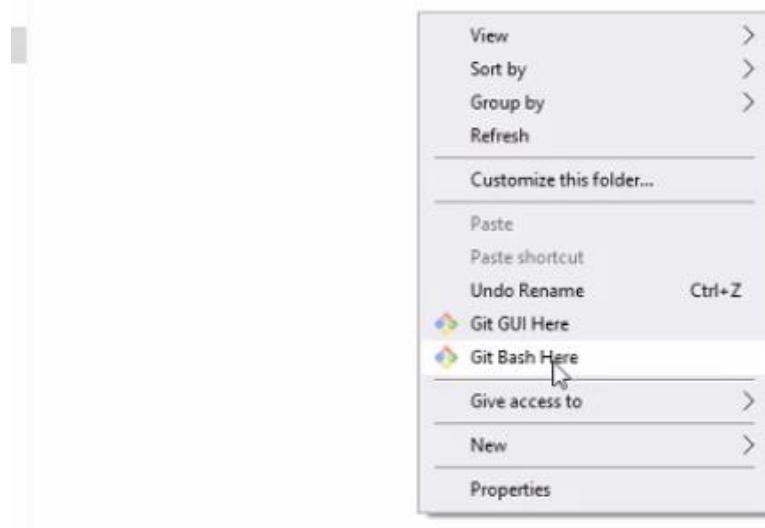
Lalu salin script yang diblock biru seperti yang ada pada gambar di bawah ini

The screenshot shows the 'Installation' sidebar with 'Composer Installation' selected. The main content area displays the command `composer create-project codeigniter4/appstarter project-root`. A note explains that this command creates a 'project-root' folder. Another note says if you omit 'project-root', it creates an 'appstarter' folder. A third note shows a sample command: `composer create-project codeigniter4/appstarter --no-dev`.

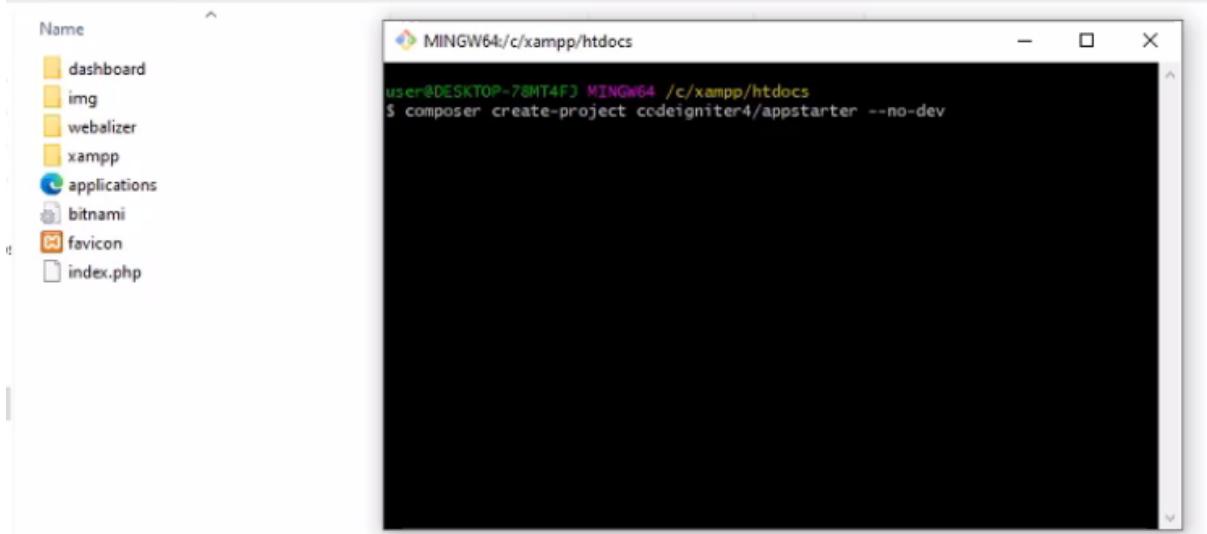
Setelah disalin buka git bash pada disk C -> xampp -> htdocs dan klik kanan lalu pilih Git Bash Here

This PC > Local Disk (C:) > xampp > htdocs

| Name | Date modified | Type | Size |
|--------------|------------------|----------------------|-------|
| dashboard | 28/09/2020 16:03 | File folder | |
| img | 28/09/2020 16:03 | File folder | |
| webalizer | 28/09/2020 16:03 | File folder | |
| xampp | 28/09/2020 16:03 | File folder | |
| applications | 27/08/2019 21:02 | Microsoft Edge H... | 4 KB |
| bitnami | 27/08/2019 21:02 | Cascading Style S... | 1 KB |
| favicon | 16/07/2015 22:32 | Icon | 31 KB |
| index.php | 16/07/2015 22:32 | PHP File | 1 KB |

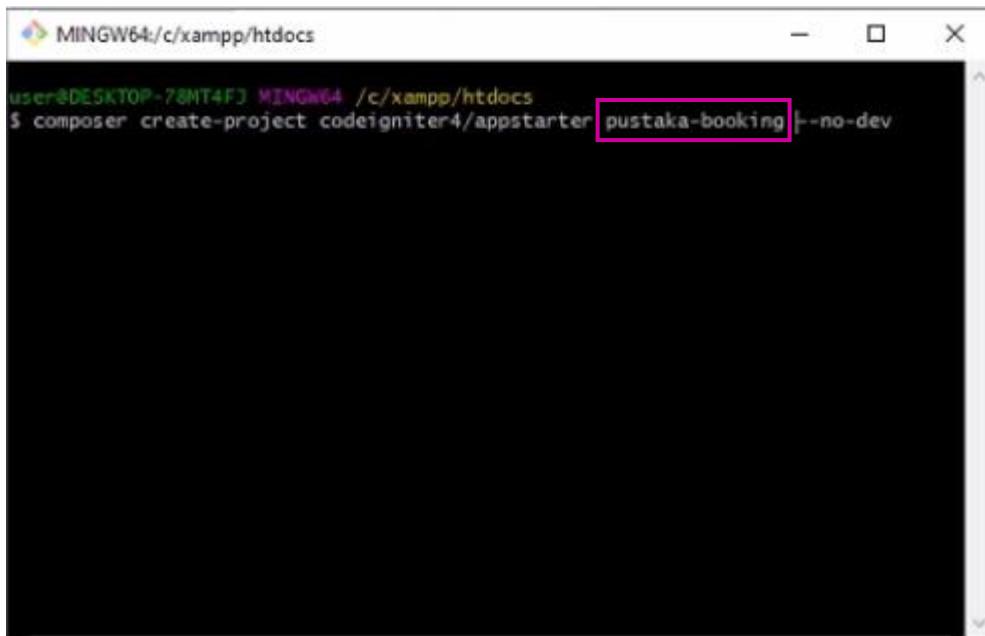


Lalu paste (tempel) hasil script yang sudah disalin sebelumnya pada git bash. Maka akan seperti gambar di bawah ini:



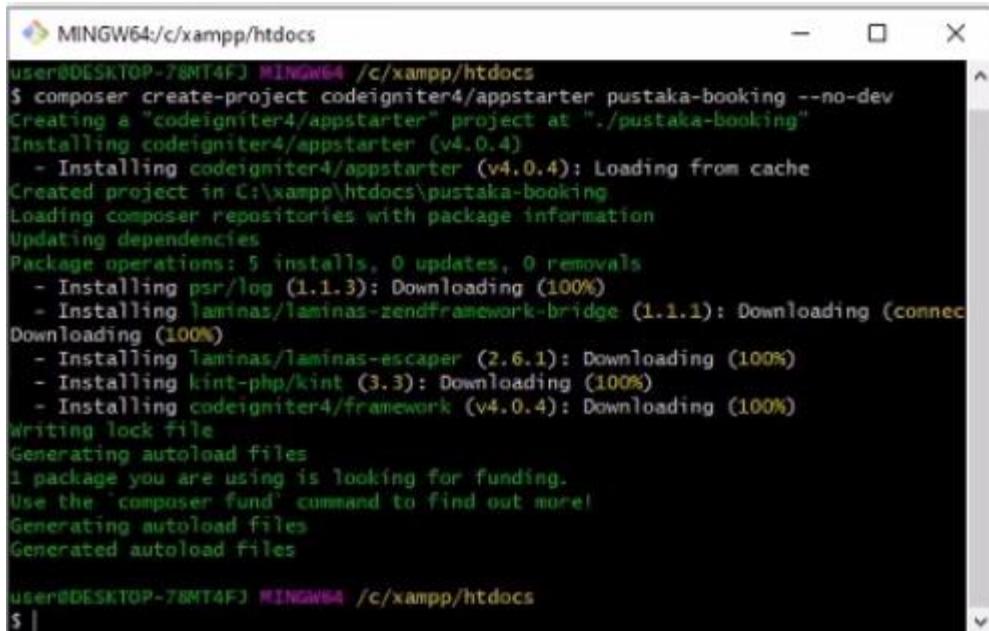
```
MINGW64:/c/xampp/htdocs
user@DESKTOP-78NT4F3 MINGW64 /c/xampp/htdocs
$ composer create-project codeigniter4/appstarter --no-dev
```

Tambahkan nama folder project yang diinginkan seperti gambar di bawah ini:



```
MINGW64:/c/xampp/htdocs
user@DESKTOP-78NT4F3 MINGW64 /c/xampp/htdocs
$ composer create-project codeigniter4/appstarter pustaka-booking --no-dev
```

Setelah selesai menambahkan nama folder projectnya tinggal tekan enter untuk mengeksekusi jalannya instalasi. Tunggu prosesnya hingga selesai. Jika berhasil maka tampilannya sebagai berikut:

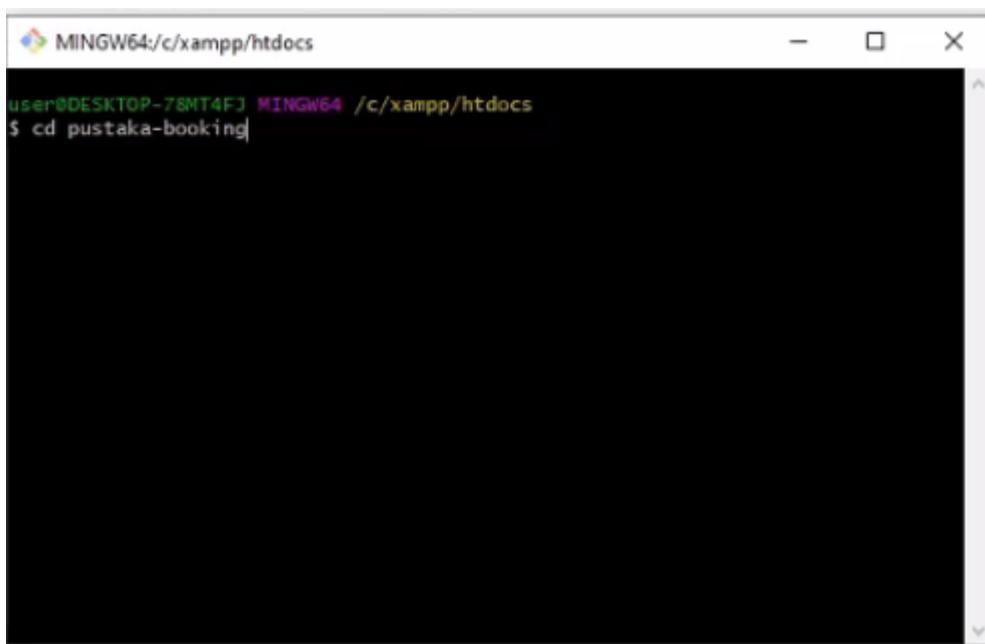


```
MINGW64:/c/xampp/htdocs
$ composer create-project codeigniter4/appstarter pustaka-booking --no-dev
Creating a "codeigniter4/appstarter" project at "./pustaka-booking"
Installing codeigniter4/appstarter (v4.0.4)
- Installing codeigniter4/appstarter (v4.0.4): Loading from cache
Created project in C:\xampp\htdocs\pustaka-booking
Loading composer repositories with package information
Updating dependencies
Package operations: 5 installs, 0 updates, 0 removals
- Installing psr/log (1.1.3): Downloading (100%)
- Installing laminas/laminas-zendframework-bridge (1.1.1): Downloading (connec
Downloaded (100%)
- Installing laminas/laminas-escaper (2.6.1): Downloading (100%)
- Installing kint-php/kint (3.3): Downloading (100%)
- Installing codeigniter4/framework (v4.0.4): Downloading (100%)
Writing lock file
Generating autoload files
1 package you are using is looking for funding.
Use the 'composer fund' command to find out more!
Generating autoload files
Generated autoload files

user@DESKTOP-78NT4FJ MINGW64 /c/xampp/htdocs
$ |
```

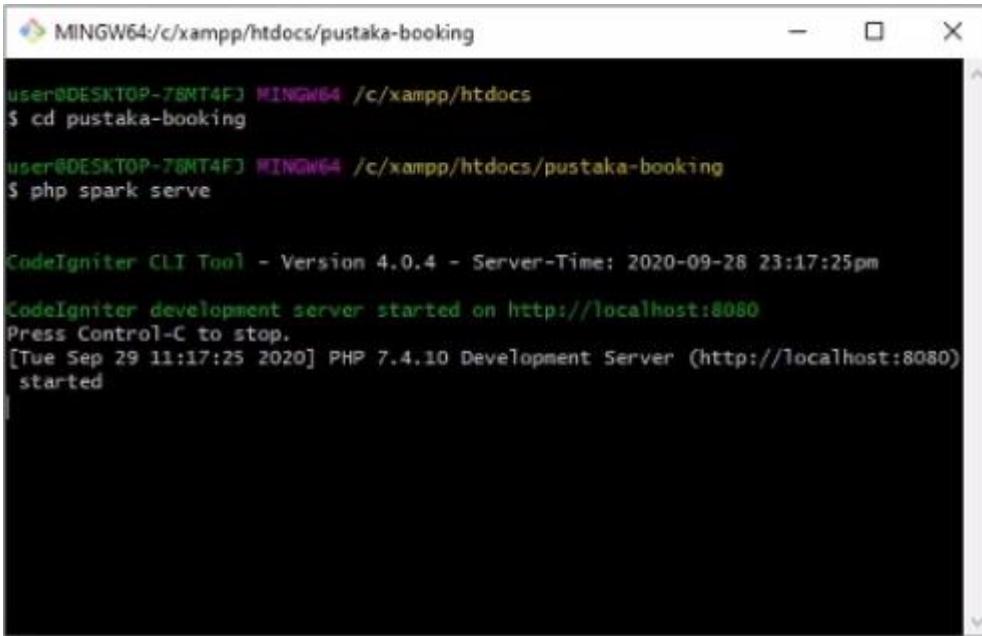
Agar tampilan di screen git bashnya kembali bersih, ketikkan perintah clear lalu enter.

Kemudian untuk mengaktifkan server codeigniter 4 tahapannya sebagai berikut:



```
MINGW64:/c/xampp/htdocs
$ cd pustaka-booking
```

Masuk terlebih dahulu ke dalam folder project yang sudah dibuat dengan cara mengetikkan koding pada gambar di atas. Kemudian enter.

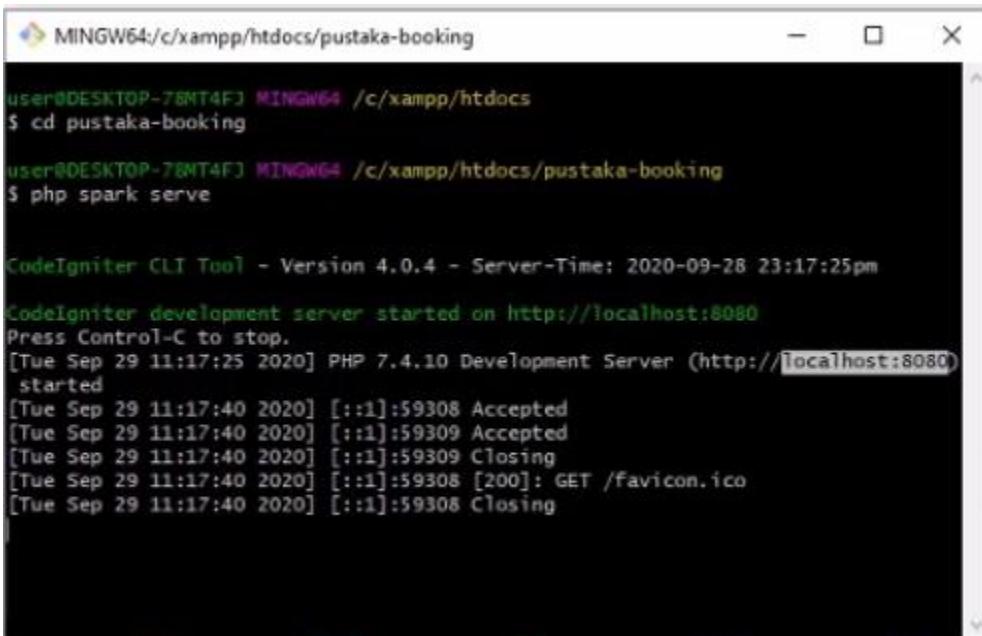


```
MINGW64:/c/xampp/htdocs/pustaka-booking
user@DESKTOP-78NT4FJ MINGW64 /c/xampp/htdocs
$ cd pustaka-booking

user@DESKTOP-78NT4FJ MINGW64 /c/xampp/htdocs/pustaka-booking
$ php spark serve

CodeIgniter CLI Tool - Version 4.0.4 - Server-Time: 2020-09-28 23:17:25pm
CodeIgniter development server started on http://localhost:8080
Press Control-C to stop.
[Tue Sep 29 11:17:25 2020] PHP 7.4.10 Development Server (http://localhost:8080)
started
```

Lanjutkan dengan mengaktifkan server php pada codeigniternya dengan mengetikkan koding php spark serve seperti gambar di atas.

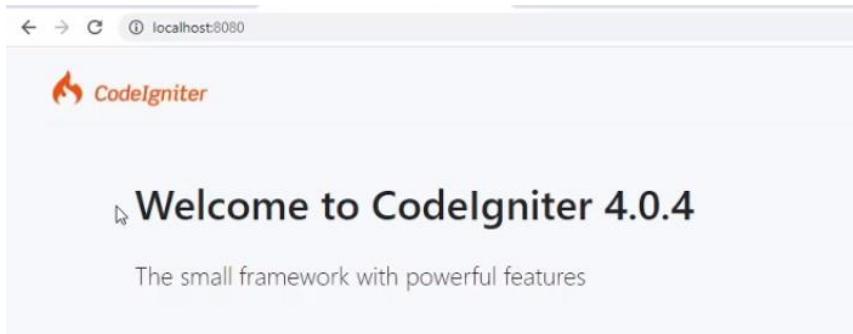


```
MINGW64:/c/xampp/htdocs/pustaka-booking
user@DESKTOP-78NT4FJ MINGW64 /c/xampp/htdocs
$ cd pustaka-booking

user@DESKTOP-78NT4FJ MINGW64 /c/xampp/htdocs/pustaka-booking
$ php spark serve

CodeIgniter CLI Tool - Version 4.0.4 - Server-Time: 2020-09-28 23:17:25pm
CodeIgniter development server started on http://localhost:8080
Press Control-C to stop.
[Tue Sep 29 11:17:25 2020] PHP 7.4.10 Development Server (http://localhost:8080)
started
[Tue Sep 29 11:17:40 2020] [::1]:59308 Accepted
[Tue Sep 29 11:17:40 2020] [::1]:59309 Accepted
[Tue Sep 29 11:17:40 2020] [::1]:59309 Closing
[Tue Sep 29 11:17:40 2020] [::1]:59308 [200]: GET /favicon.ico
[Tue Sep 29 11:17:40 2020] [::1]:59308 Closing
```

Salin pada bagian localhost:8080 untuk dicek dijalankan di browser apakah framework codeigniternya sudah berjalan atau belum.



About this page

The page you are looking at is being generated dynamically by CodeIgniter.

If you would like to edit this page you will find it located at:

`app/Views/welcome_message.php`

The corresponding controller for this page can be found at:

Jika sudah dipaste di url browser maka tampilannya seperti gambar di atas. Artinya framework codeigniter sudah bisa digunakan dan dalam keadaan aktif.

PERTEMUAN 3

PEMBUATAN DATABASE

Deskripsi Pembelajaran

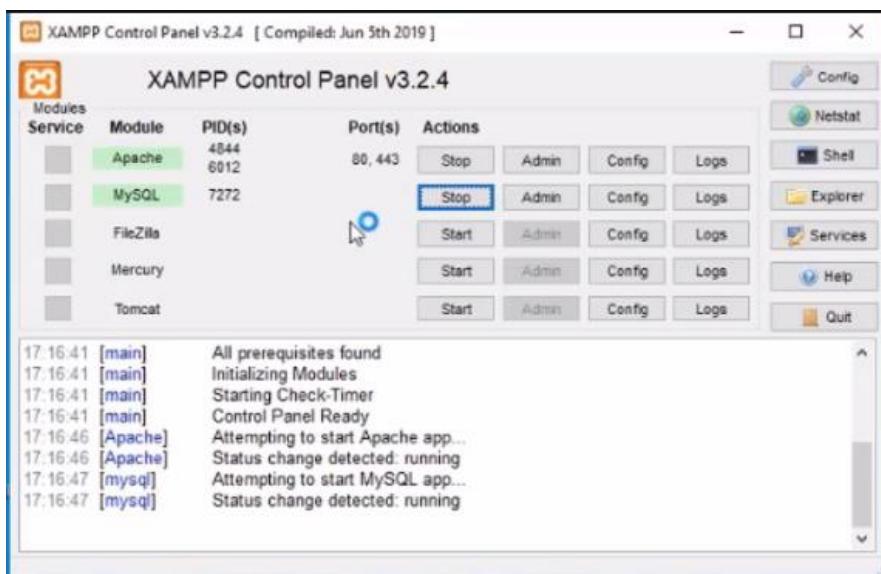
Pada bab ini akan dibahas tentang pembuatan database atau basis data untuk nantinya akan dikoneksikan dengan project web yang akan dibuat. Ada beberapa tabel yang akan dibuat pada studi kasus ini. Diantaranya ada tabel buku dan anggota, serta selebihnya adalah tabel otomatis yang akan dibuat dari library framework codeigniter 4 sendiri.

Tujuan Pembelajaran

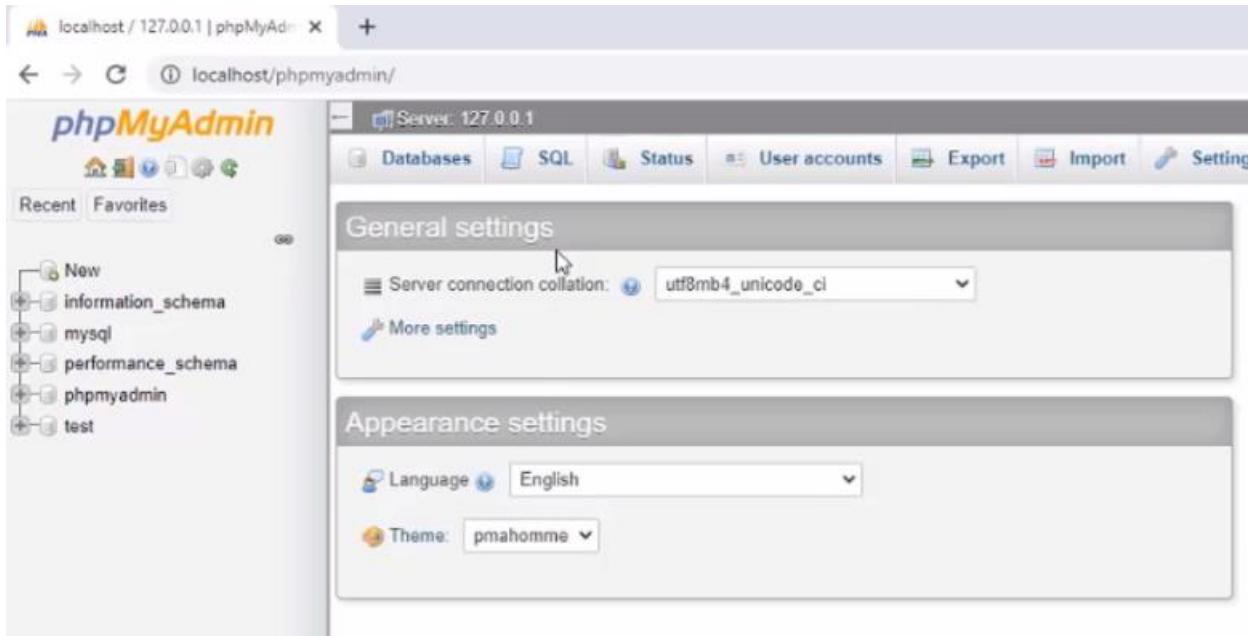
Pada bab ini diharapkan mahasiswa mampu memahami dalam pembuatan database yang sesuai dengan project web yang mereka buat. Dan mampu mengimplementasikannya dengan baik, bagaimana memperhatikan size yang tepat untuk tiap fieldnya dan tabel apa saja yang harus dibuat sesuai kebutuhan program. Sehingga didapatkan manajemen database yang baik dan benar.

3.1. Membuat Database Pustaka

Langkah pertama dalam membuat database adalah mengaktifkan terlebih dahulu Apache dan MySQL yang terdapat pada control panel XAMPP dengan cara klik start pada tombol Apache dan MySQLnya. Maka dari itu silahkan diaktifkan terlebih dahulu seperti gambar di bawah ini:

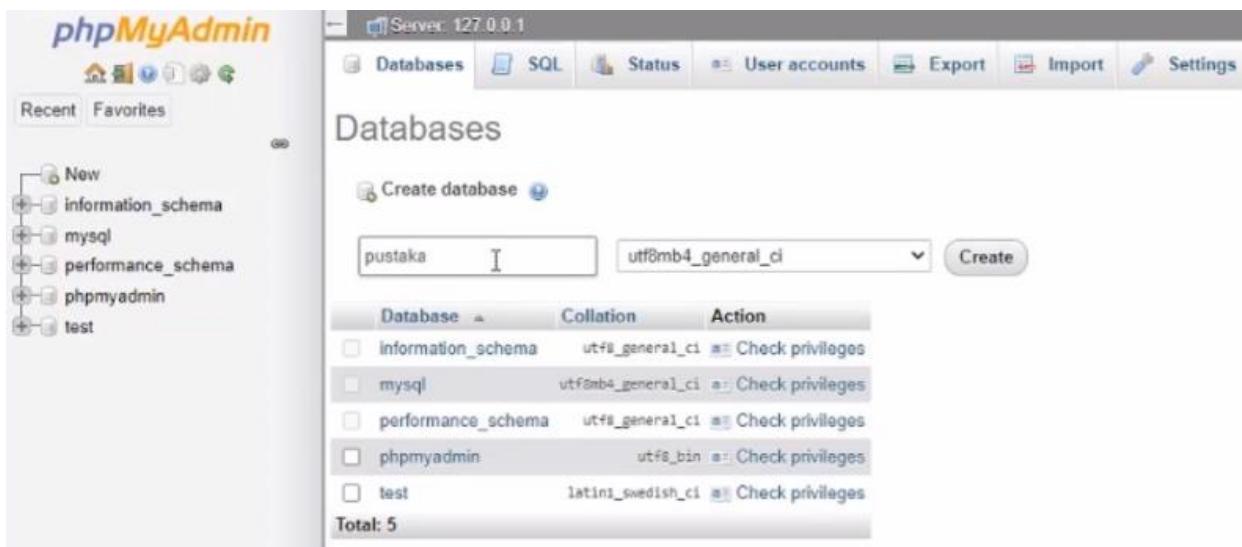


Setelah Apache dan MySQLnya sudah aktif , tahap selanjutnya masuk ke browser lalu ketikkan alamat berikut pada url: localhost/phpmyadmin.



The screenshot shows the 'General settings' section of the phpMyAdmin interface. It includes fields for 'Server connection collation' set to 'utf8mb4_unicode_ci' and a link to 'More settings'. Below this is the 'Appearance settings' section, which shows the 'Language' set to 'English' and the 'Theme' set to 'pmahomme'.

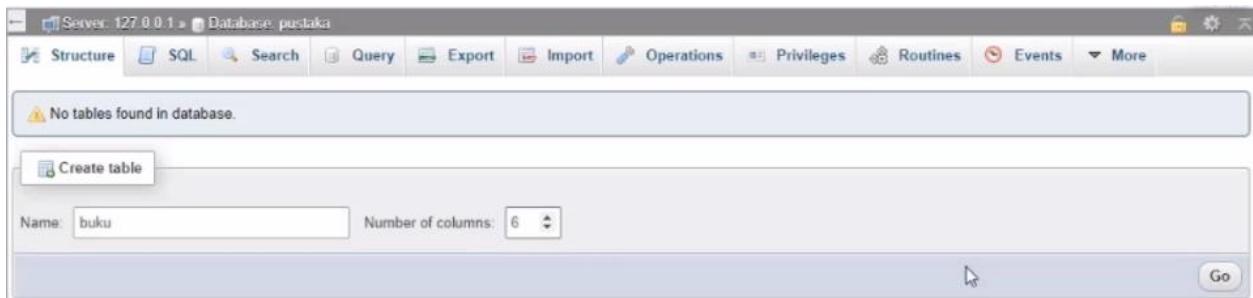
Untuk membuat database klik Menu Bar Databases, lalu masukkan nama databasenya: pustaka. Kemudian klik create.



The screenshot shows the 'Databases' section of the phpMyAdmin interface. A new database named 'pustaka' is being created with a collation of 'utf8mb4_general_ci'. The table lists existing databases: information_schema, mysql, performance_schema, phpmyadmin, and test, each with its respective collation and action links.

| Database | Collation | Action |
|--------------------|--------------------|----------------------------------|
| information_schema | utf8_general_ci | Check privileges |
| mysql | utf8mb4_general_ci | Check privileges |
| performance_schema | utf8_general_ci | Check privileges |
| phpmyadmin | utf8_bin | Check privileges |
| test | latin1_swedish_ci | Check privileges |

Buat tabel yang pertama dengan nama: buku dan jumlah kolom/number of fields: 6.



Lanjutkan dengan mengisi nama kolom, type, dan length-nya seperti gambar di bawah ini:

| Table name: buku | | | | |
|------------------|---------|---------------|---------|-----------|
| Name | Type | Length/Values | Default | Collation |
| id_buku | INT | | None | |
| judul | VARCHAR | 50 | None | |
| pengarang | VARCHAR | 25 | None | |
| penerbit | VARCHAR | 25 | None | |
| tahun_terbit | YEAR | 4 | None | |
| sampul | VARCHAR | 100 | None | |

Jangan lupa untuk menambahkan primary key pada id_buku yakni kolom yang pertama. Untuk lebih detailnya bisa dilihat pada gambar di bawah ini:

Setelah ditambahkan primary key pada id_buku lanjutkan dengan klik save yang terletak di pojok kanan bawah.



Maka hasilnya akan seperti ini:

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|---|---------------------|--------------|--------------------|------------|------|---------|----------|----------------|--------------------|
| 1 | id_buku | int(11) | | | No | None | | AUTO_INCREMENT | Change Drop More |
| 2 | judul | varchar(50) | utf8mb4_general_ci | | No | None | | | Change Drop More |
| 3 | pengarang | varchar(25) | utf8mb4_general_ci | | No | None | | | Change Drop More |
| 4 | penerbit | varchar(25) | utf8mb4_general_ci | | No | None | | | Change Drop More |
| 5 | tahun_terbit | year(4) | | | No | None | | | Change Drop More |
| 6 | sampul | varchar(100) | utf8mb4_general_ci | | No | None | | | Change Drop More |

Lanjutkan dengan membuat tabel yang kedua dengan nama anggota dan jumlah kolom berjumlah 4. Untuk membuat tabel baru bisa dengan cara klik Database:pustaka yang terdapat pada headbar.



Untuk nama kolom,tipe, dan ukuran tabel anggota bisa dilihat pada gambar di bawah ini:

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|---|------------|-------------|--------------------|------------|------|---------|----------|----------------|--------------------|
| 1 | id_anggota | int(11) | utf8mb4_general_ci | | No | None | | AUTO_INCREMENT | Change Drop More |
| 2 | nama | varchar(25) | utf8mb4_general_ci | | No | None | | | Change Drop More |
| 3 | alamat | text | utf8mb4_general_ci | | No | None | | | Change Drop More |
| 4 | telp | varchar(13) | utf8mb4_general_ci | | No | None | | | Change Drop More |

Setelah tabel anggota selesai dibuat tahap selanjutnya isi data pada tabel buku dengan cara klik insert.

| Table | Action | Rows | Type | Collation | Size | Overhead |
|----------|--------|------|--------|--------------------|----------|----------|
| anggota | | 0 | InnoDB | utf8mb4_general_ci | 16.0 KiB | - |
| buku | | 0 | InnoDB | utf8mb4_general_ci | 16.0 KiB | - |
| 2 tables | Sum | 0 | InnoDB | utf8mb4_general_ci | 32.0 KiB | 0 B |

Kosongkan pada bagian id_buku karena id_buku penomorannya dibuat otomatis dan sampul juga dikosongkan terlebih dahulu. Kemudian isikan seperti ini:

Table: buku

Rows: 0 Type: InnoDB Collation: utf8mb4_general_ci Size: 16.0 KiB Overhead: 0 B

Insert

| Column | Type | Function | Null | Value |
|--------------|--------------|----------|------|---------------|
| id_buku | int(11) | | | |
| judul | varchar(50) | | | PHP |
| pengarang | varchar(25) | | | Andre Pratama |
| penerbit | varchar(25) | | | DuniaLkom |
| tahun_terbit | year(4) | | | 2017 |
| sampul | varchar(100) | | | |

Go

Table: buku

Rows: 0 Type: InnoDB Collation: utf8mb4_general_ci Size: 16.0 KiB Overhead: 0 B

Insert

| Column | Type | Function | Null | Value |
|--------------|--------------|----------|------|-----------------|
| id_buku | int(11) | | | |
| judul | varchar(50) | | | Pemrograman Web |
| pengarang | varchar(25) | | | Rohi Abdulloh |
| penerbit | varchar(25) | | | Andi Offset |
| tahun_terbit | year(4) | | | 2019 |
| sampul | varchar(100) | | | |

Console

Cukup isikan 2 data saja lalu klik Go.

| | | |
|--------------|--------------|--|
| judul | varchar(50) | <input type="text" value="Pemrograman Web"/> |
| pengarang | varchar(25) | <input type="text" value="Rohi Abdulloh"/> |
| penerbit | varchar(25) | <input type="text" value="Andi Offset"/> |
| tahun_terbit | year(4) | <input type="text" value="2019"/> |
| sampul | varchar(100) | <input type="text"/> |

and then

Kemudian klik Browse maka akan tampil tampilan sebagai berikut:

✓ Showing rows 0 - 1 (2 total, Query took 0.0009 seconds.)

```
SELECT * FROM `buku`
```

Profiling [Edit inline] [Edit] [Exp]

| | id_buku | judul | pengarang | penerbit | tahun_terbit | sampul |
|--------------------------|---------|-----------------|---------------|-------------|--------------|--------|
| <input type="checkbox"/> | 1 | PHP | Andre Pratama | Duniailkom | 2017 | |
| <input type="checkbox"/> | 2 | Pemrograman Web | Rohi Abdulloh | Andi Offset | 2019 | |

Show all | Number of rows: Filter rows: Sort by key:

Check all With selected:

Show all | Number of rows: Filter rows: Sort by key:

Jika pembuatan database sudah selesai, untuk Apache dan MySQLnya bisa dinonaktifkan kembali.

XAMPP Control Panel v3.2.4 [Compiled: Jun 5th 2019]

XAMPP Control Panel v3.2.4

| Modules | Service | Module | PID(s) | Port(s) | Actions |
|---------|---------|-----------|--------------|---------|---|
| | | Apache | 5208 3408 | 80, 443 | <input type="button" value="Stop"/> <input type="button" value="Admin"/> <input type="button" value="Config"/> <input type="button" value="Logs"/> |
| | | MySQL | 14408 | 3306 | <input type="button" value="Stop"/> <input type="button" value="Admin"/> <input type="button" value="Config"/> <input type="button" value="Logs"/> |
| | | FileZilla | | | <input type="button" value="Start"/> <input type="button" value="Admin"/> <input type="button" value="Config"/> <input type="button" value="Logs"/> |
| | | Mercury | | | <input type="button" value="Start"/> <input type="button" value="Admin"/> <input type="button" value="Config"/> <input type="button" value="Logs"/> |
| | | Tomcat | | | <input type="button" value="Start"/> <input type="button" value="Admin"/> <input type="button" value="Config"/> <input type="button" value="Logs"/> |

PERTEMUAN 4

MVC DAN INSTALASI PLUGINS VISUAL STUDIO CODE

Deskripsi Pembelajaran

Pada bab ini akan dibahas tentang penjelasan mengenai apa itu mvc dan bagaimana cara kerjanya serta menjelaskan plug in apa saja yang harus diinstalasi ke dalam visual studio code yang bertujuan untuk memudahkan dalam hal pengodingan.

Tujuan Pembelajaran

Pada bab ini diharapkan mahasiswa dapat memahami kerangka kerja mvc, yakni model, view dan controller. Selain itu mahasiswa juga diharapkan dapat mengoding dengan mudah dan cepat karena adanya pembelajaran tentang plug in apa yang harus diinstalasi pada visual studio code.

4.1. MVC

MVC merupakan model kerangka kerja yang diterapkan pada codeigniter. MVC sendiri merupakan singkatan dari Model, View, dan Controller. Pada pertemuan kali ini akan dibahas satu persatu mengenai tentang ap itu model, view dan controller.

a. Model

Pada model memiliki tugas yang berkaitan dengan pengolahan database. Model ini lah yang berfungsi sebagai jembatan atau koneksi antara database dengan program web yang dibuat. Dan nantinya sangat berperan penting jika sudah diimplementasikan dengan view. Dimana data yang ada pada database bisa ditampilkan pada fungsi view, dalam hal ini adalah layout.

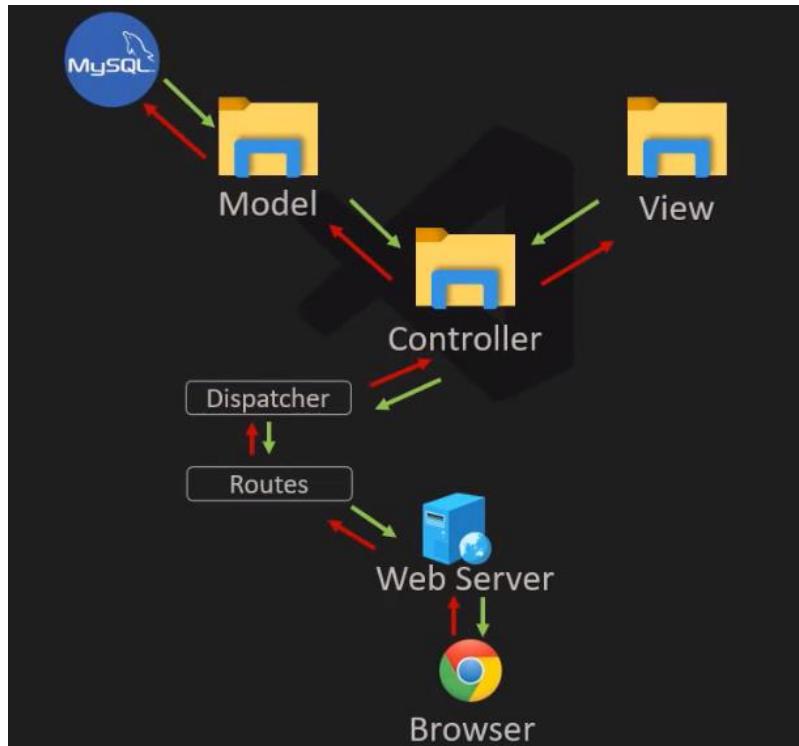
b. View

View memiliki peran dalam hal menampilkan halaman website. Baik back end ataupun front end. Dalam kata lain view ini bertugas sebagai layout.

c. Controller

Untuk controller memiliki fungsi terhadap view dan juga model. Tanpa adanya controller view dan model tidak dapat dijalankan. Jadi controller inilah yang bertugas menjalankan fungsi dari view dan juga model. Dan keduanya pun bisa dijalankan secara langsung oleh control.

Adapun skema atau siklus mvc sendiri dapat dilihat dari gambar di bawah ini:



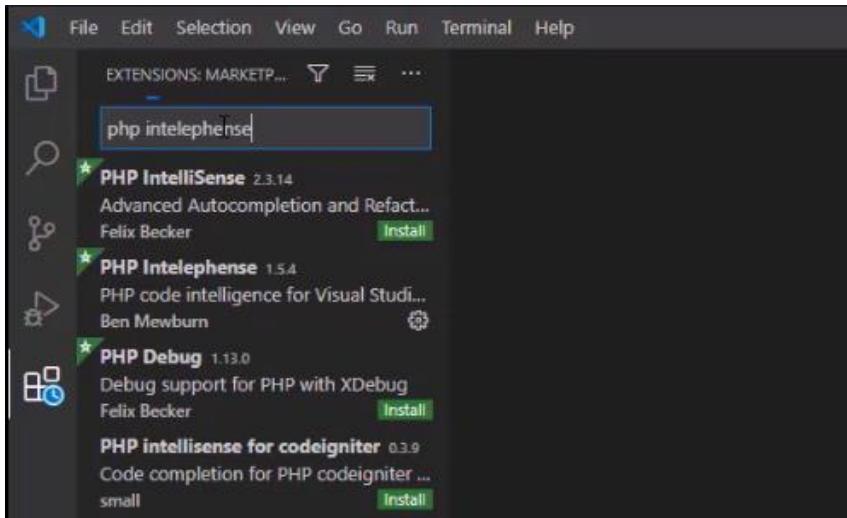
Dari gambar di atas dapat dijelaskan bahwa apa yang ditampilkan dan dijalankan dari browser itu berawal dari kerja web server. Dari web server terdapat routes yang berfungsi sebagai kontrol url yang terdapat pada browser. Kemudian dari routes masuk ke dispatcher. Dispatcher ini lah yang membaca perintah yang dikirimkan oleh controller. Sedangkan controller sendiri membawa fungsi yang dikirimkan oleh model dan view. Dimana model membawa fungsi yang diambil dari database. Database yang dicontohkan disini adalah MySQL.

4.2. Instalasi Plugins Visual Studio Code

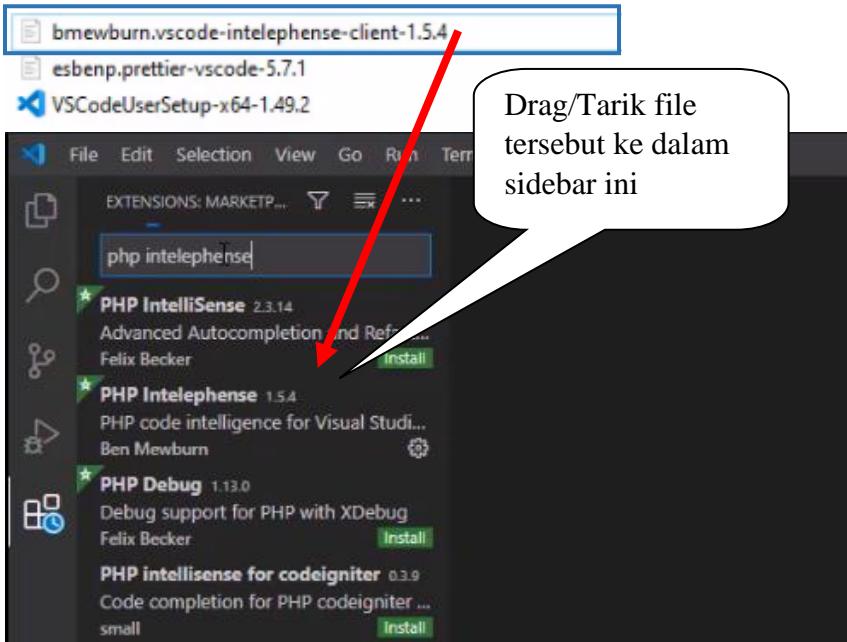
Plugins yang akan diinstalasi pada visual studio code terdapat 3 macam, diantaranya: php intelephense, prettier dan snippet. Adapun tahapan instalasinya sebagai berikut:

a. PHP Intelephense

Cara instalasinya cukup mudah. Buka kembali visual studi code. Lalu ketikkan php intelephense pada search box yang terdapat pada sidebar. Perlu diperhatikan untuk instalasi semua plugins pada visual studio code ini menggunakan koneksi internet. Untuk lebih jelasnya pengetikan di search box pada sidebar visual studio code bisa dilihat pada gambar di bawah ini:



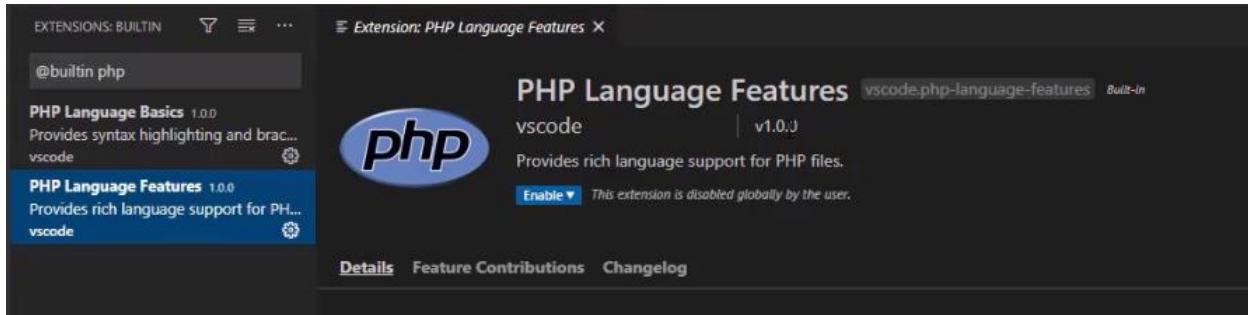
Kemudian klik yang PHP Intelephense. Lalu klik install, maka secara otomatis akan ada attention di pojok kanan bawah untuk install secara manual dan lanjutkan dengan klik saja. Maka akan terunduh file seperti di bawah ini:



Dan tunggu prosesnya hingga selesai.

Jika sudah berhasil diinstalasi, tahap selanjutnya konfigurasi php language featurenya agar php intelephense ini dapat berjalan.

Sama seperti php intelephense ketik di search box: @builtin php . Maka akan tampil seperti gambar di bawah ini:

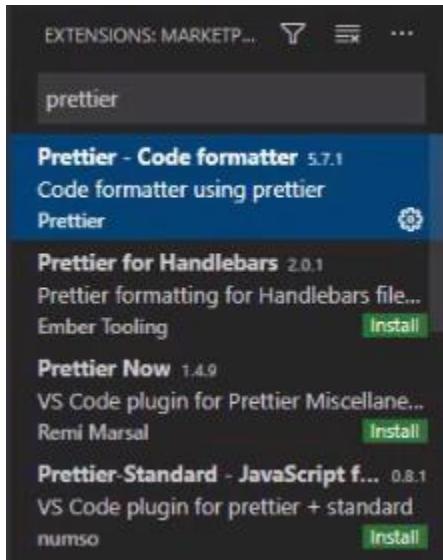


Setelah itu klik disable hingga berubah menjadi enable seperti contoh gambar di atas.

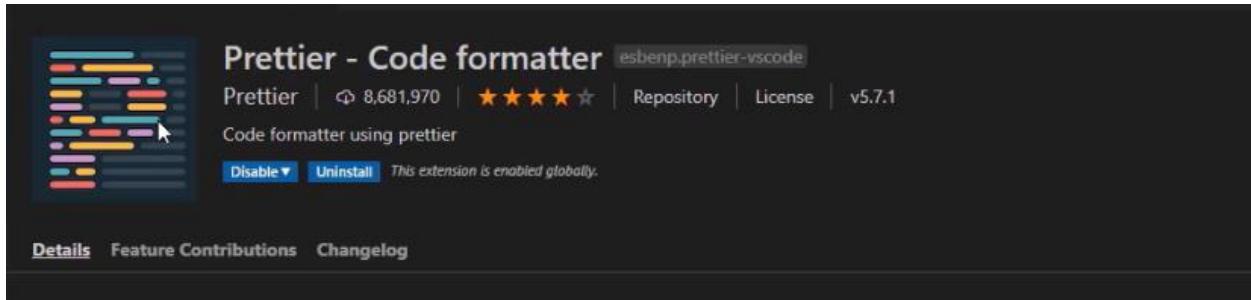
Adapun fungsi dari php intelephense ini sendiri sebagai sense saat hendak mengoding. Misal saja saat ingin mengetikan mysql_connect maka secara otomatis php intelephense ini akan memberikan rekomendasi bar saat kita baru mengetikkan satu atau dua huruf yang kita ketik. Plugins ini mencoba menganalisis apa yang sedang kita fikirkan dan apa yang sedang ingin kita ketik. Jika apa yang kita ingin cocok dengan yang dimaksud otomatis akan tampil di recommendation bar dan tinggal tekan tab saja maka yang ada pada recommendation bar tersebut akan pindah ke dalam screen script kita.

b. Prettier

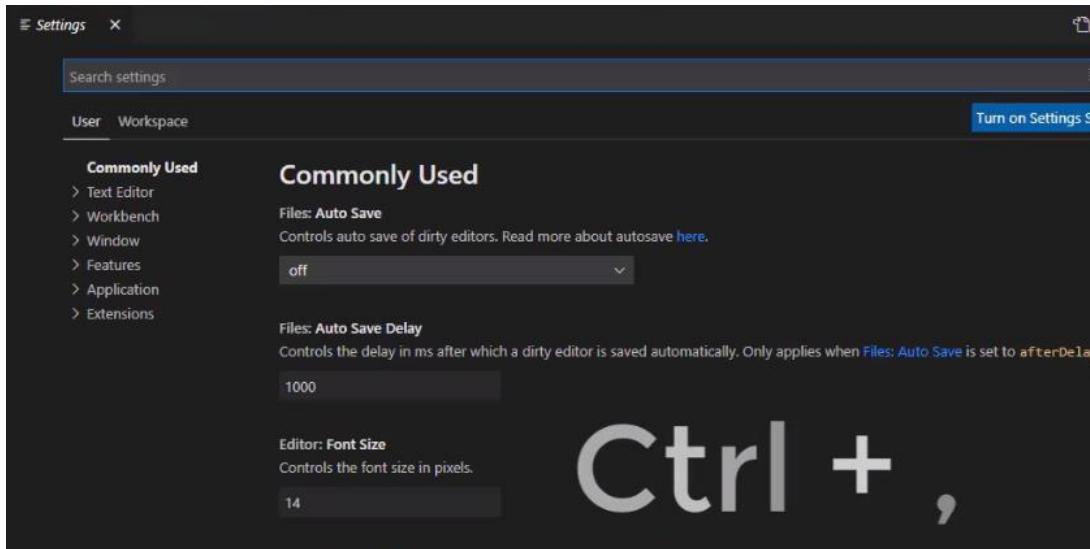
Instalasi prettier sama halnya dengan intelephense. Ketik prettier pada search box. Lalu klik Prettier – Code Formatter.



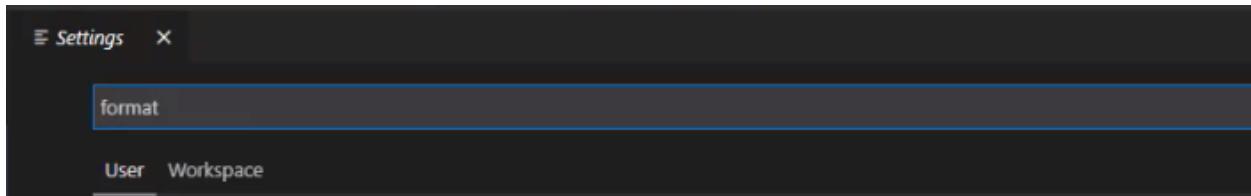
Setelah itu tahpannya sama persis dengan php intelephense. Jika sudah terinstall maka tombol install akan berubah jadi uninstall seperti gambar di bawah ini:



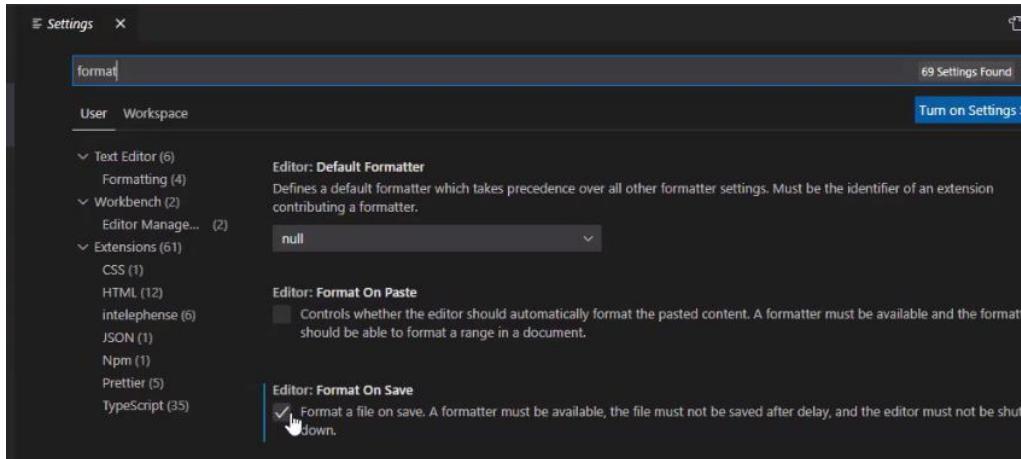
Kemudian tekan “Ctrl + ,“ untuk membuka menu setting.



Ketik format pada bar search settingnya.



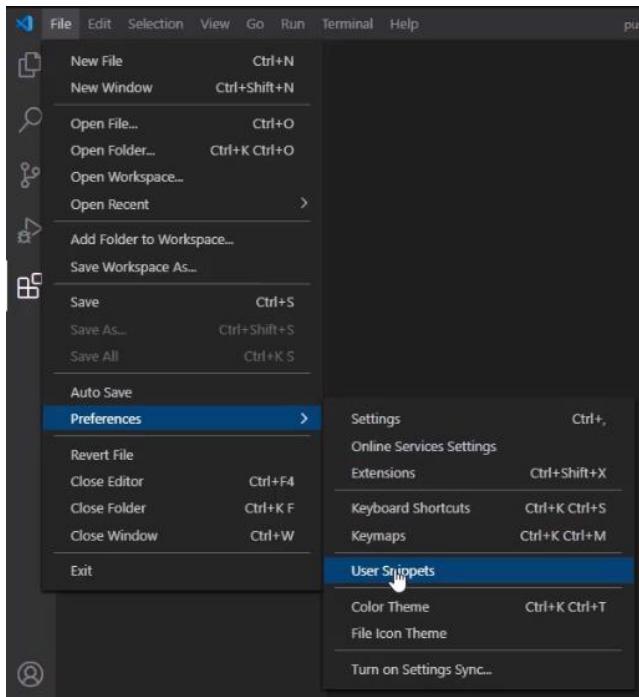
Kemudian checklist pada bagian Editor: Format on Save.



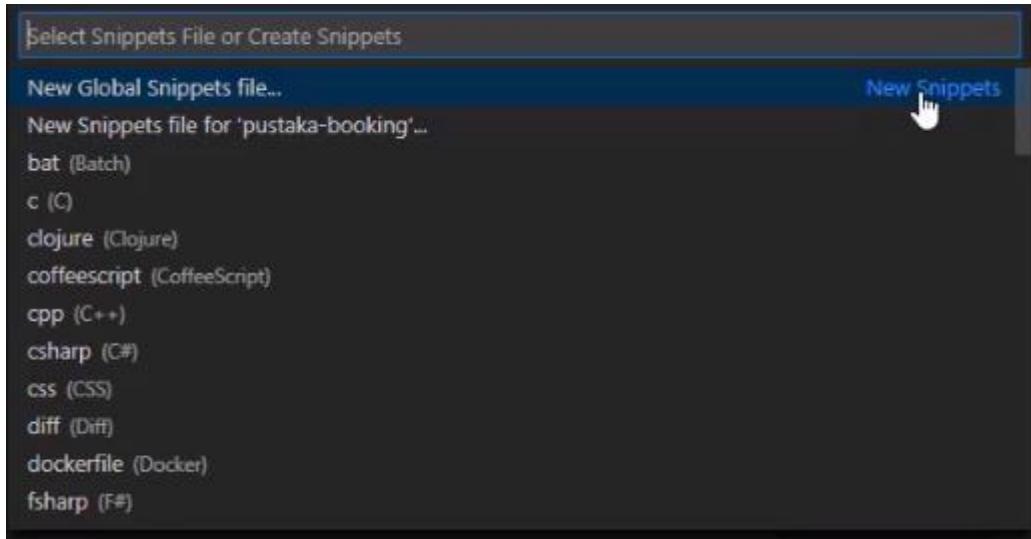
Adapun fungsi dari prettier ini sendiri adalah untuk merapikan script secara otomatis setalah file disimpan atau setelah menekan Ctrl + S.

c. Snippets

Untuk instalasi snippets berbeda dengan php intelephense dan prettier. Caranya klik File -> Preferences -> User Snippets.



Klik new snippets untuk membuat snippet baru:



Buat file baru snippet dengan nama: html.json. Untuk ekstensinya gunakan format json. Penamaan snippet barunya bebas sesuai keinginan. Tapi dalam pembelajaran kali ini disesuaikan seperti gambar di bawah ini:



Lalu ketikkan perintah di bawah ini:

```
Untitled-1  { html.json.code-snippets }
```

C: > User User snippet configuration > Code > User > snippets > html.json.code-snippets > ...

```
1  {
2      "PHP Tag": {
3          "prefix": "php",
4          "body": "<?php $1 ?>"
5      },
6      "Inline Echo": {
7          "prefix": "phpp",
8          "body": "<?= $$1; ?>"
9      }
10 }
11 |
```

A screenshot of the Sublime Text code editor showing a JSON file named 'html.json.code-snippets'. The file contains two snippets: 'PHP Tag' and 'Inline Echo'. The 'PHP Tag' snippet has a prefix of 'php' and a body of '<?php \$1 ?>'. The 'Inline Echo' snippet has a prefix of 'phpp' and a body of '<?= \$\$1; ?>'. The code editor shows line numbers from 1 to 11.

Fungsi dari coding di atas adalah untuk memudahkan dalam menampilkan tag php. Jadi dari sini dapat diketahui fungsi snippet adalah sebagai shortcut dalam membuat fungsi-fungsi php, html dan lain sebagainya yang dengan mudah dapat dipanggil ke dalam script yang sedang dikerjakan. Dengan teknis misal seperti contoh gambar di atas untuk memanggil <?php \$..... ?> hanya dengan mengetikkan php pada script lalu tekan tab maka akan tampil fungsi tag tersebut, yakni: <?php \$..... ?> .

PERTEMUAN 5

PEMBUATAN LAYOUT

Deskripsi Pembelajaran

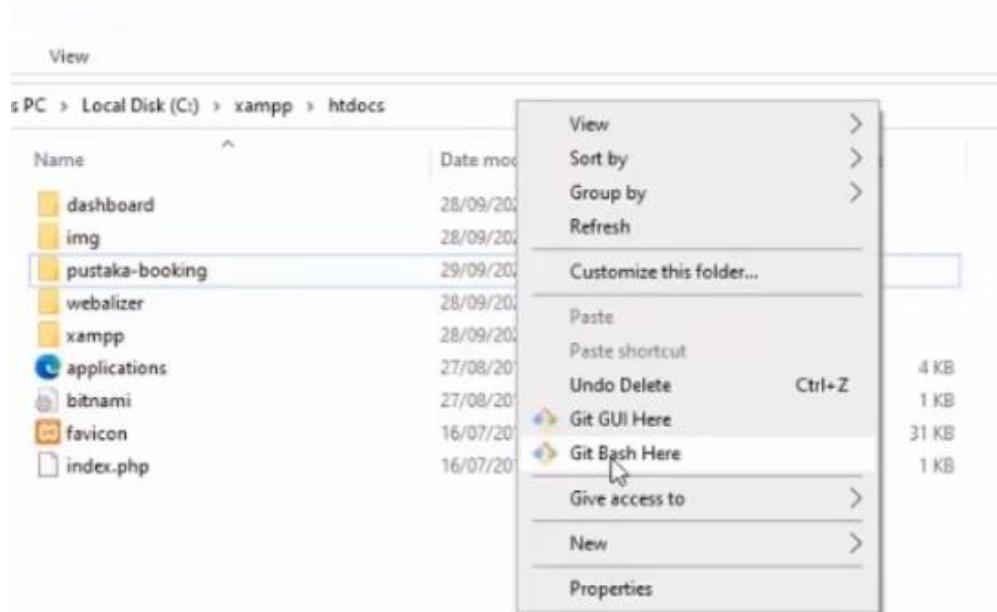
Pada bab ini akan dibahas tentang pembuatan layout menggunakan template bootstrap. Layout yang akan dibuat adalah layout pada bagian back end. Menu yang terdapat pada layout yang akan dibuat nantinya ada menu home, dafta buku, daftar anggota, dan logout.

Tujuan Pembelajaran

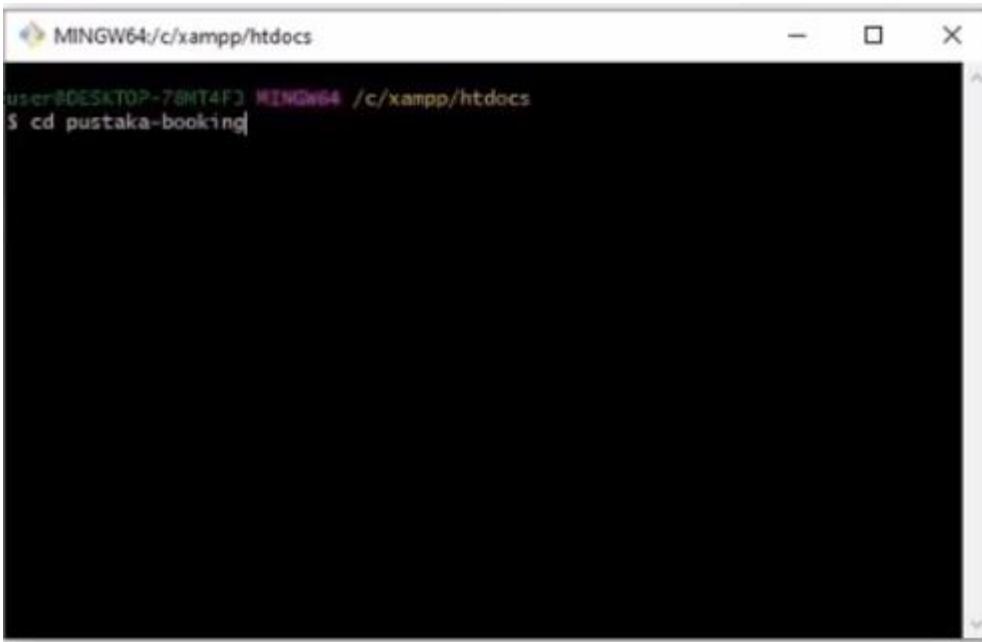
Pada bab ini diharapkan mahasiswa dapat memahami dalam pembuatan layout pada website dengan menggunakan template bootstrap. Dan mahasiswa dapat dengan terampil membuat desain yang menarik dari tampilan websitenya. Karena bootstrap menyediakan banyak sekali template-template yang menarik untuk mempercantik tampilan website.

5.1. Pembuatan Layout

Sebelum masuk ke dalam pembuatan layout, seperti biasa aktifkan terlebih dahulu php server dari codeigniter 4 nya. Caranya masuk ke disk C -> xampp -> htdocs -> klik kanan pilih Git Bash Here.



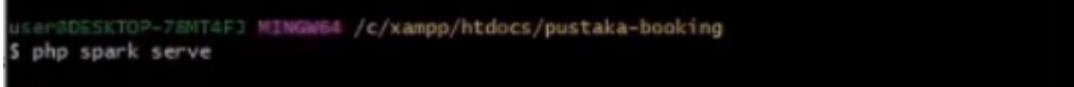
Setelah tampilan git bashnya terbuka dilanjutkan dengan mengetikkan koding di bawah ini untuk masuk ke dalam folder pustaka-booking.



```
MINGW64:/c/xampp/htdocs
user@DESKTOP-78MT4F3 MINGW64 /c/xampp/htdocs
$ cd pustaka-booking
```

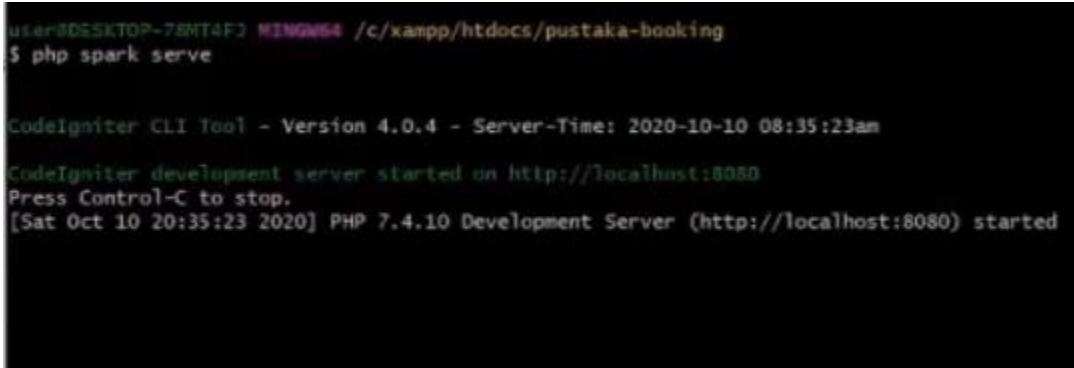
Untuk cd sendiri merupakan singkatan dari change directory. Artinya disini fungsi dialihkan ke dalam folder pustaka-booking.

Tahapan selanjutnya ketikan koding di bawah ini:



```
user@DESKTOP-78MT4F3 MINGW64 /c/xampp/htdocs/pustaka-booking
$ php spark serve
```

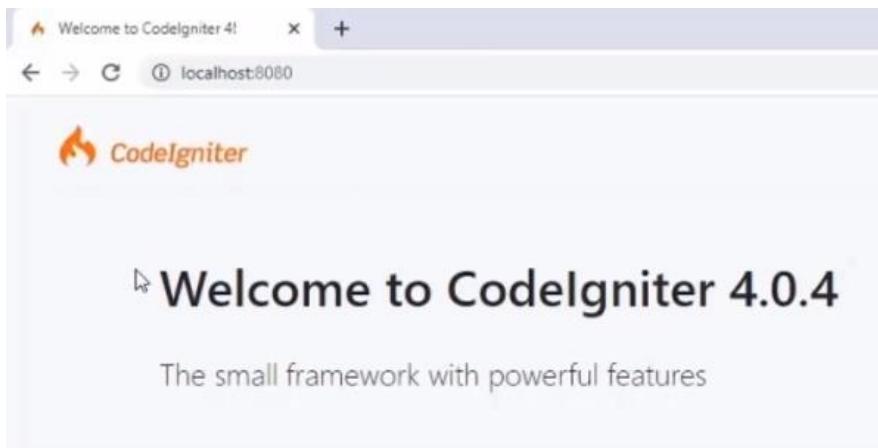
Setelah berhasil diaktifkan tampilannya seperti di bawah ini:



```
user@DESKTOP-78MT4F3 MINGW64 /c/xampp/htdocs/pustaka-booking
$ php spark serve

CodeIgniter CLI Tool - Version 4.0.4 - Server-Time: 2020-10-10 08:35:23am
CodeIgniter development server started on http://localhost:8080
Press Control-C to stop.
[Sat Oct 10 20:35:23 2020] PHP 7.4.10 Development Server (http://localhost:8080) started
```

Saat ingin menghentikan php server dari codeigniter 4 ini sendiri hanya dengan menekan CTRL + C. Kemudian tahap selanjutnya salin tulisan localhost:8080 untuk dipaste di url browser. Atau bisa diketik secara langsung pada browser: localhost:8080 lalu enter. Maka tampilannya akan seperti berikut:



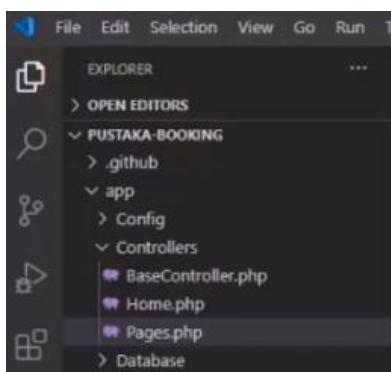
About this page

The page you are looking at is being generated dynamically by CodeIgniter.

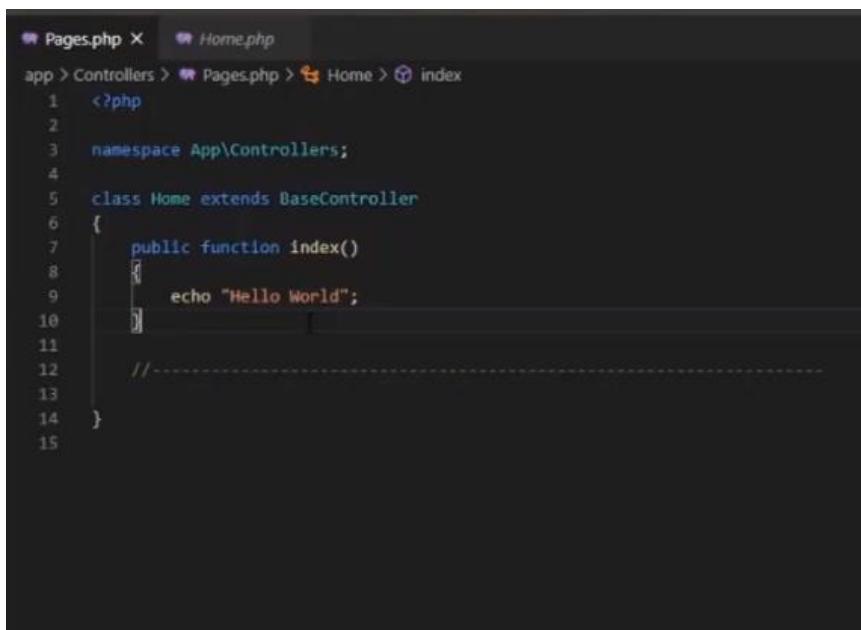
If you would like to edit this page you will find it located at:

app/Views/welcome_message.php

Langkah selanjutnya silahkan buka visual studio code. Kemudian pilih File -> Open -> Select Folder -> Pilih folder project pustaka-booking yang ada di dalam htdocs -> Klik Open. Setelah file terbuka dilanjutkan dengan membuat file baru dengan cara klik folder app yang terdapat di dalam folder pustaka-booking kemudian klik folder app, klik kanan pada folder controllers, pilih new file beri nama filenya: Pages.php. Perlu diperhatikan tiap penulisan nama file controller di awali dengan huruf besar. Untuk detailnya bisa lihat gambar berikut:

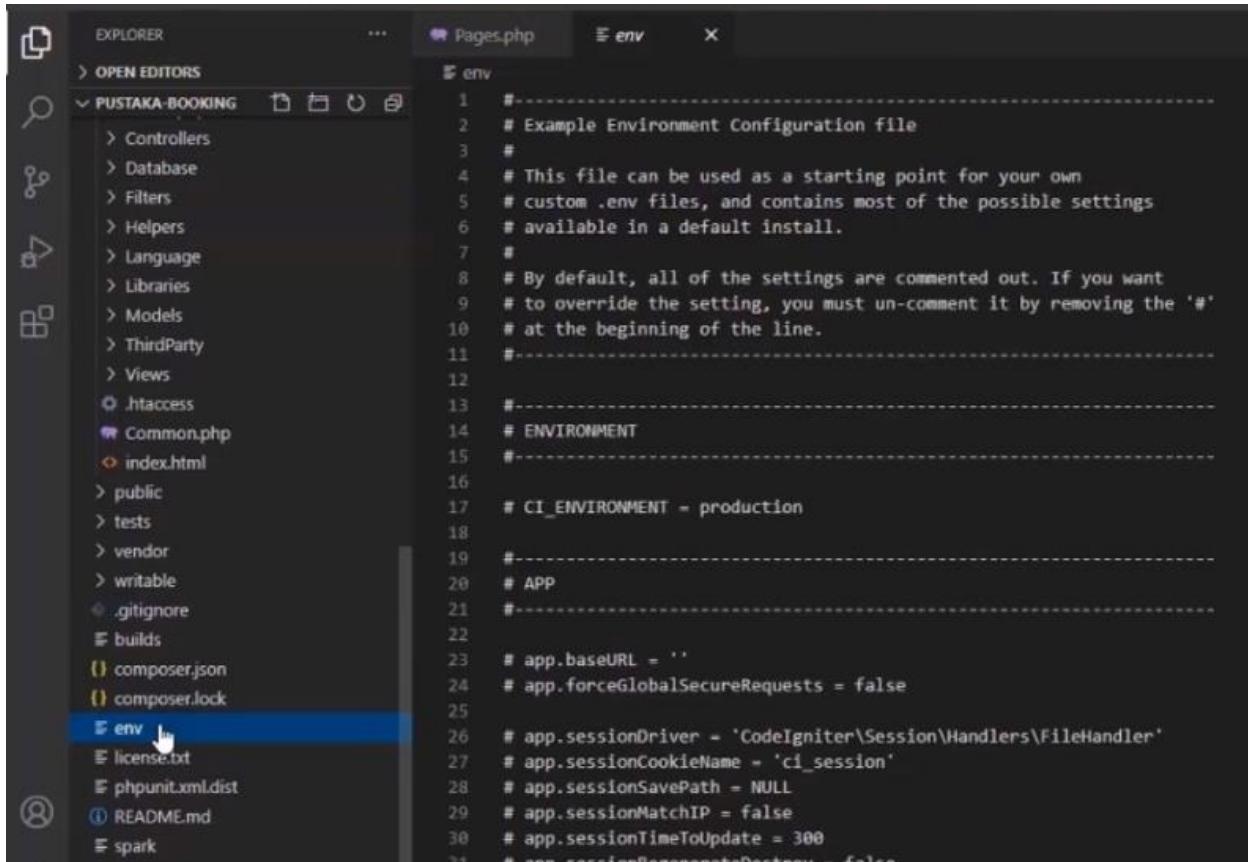


Selanjutnya ketikan perintah echo “Hello World” ; di dalam public function index().



```
Pages.php X Home.php
app > Controllers > Pages.php > Home > index
1  <?php
2
3  namespace App\Controllers;
4
5  class Home extends BaseController
6  {
7      public function index()
8      {
9          echo "Hello World";
10     }
11
12     //
13
14 }
15
```

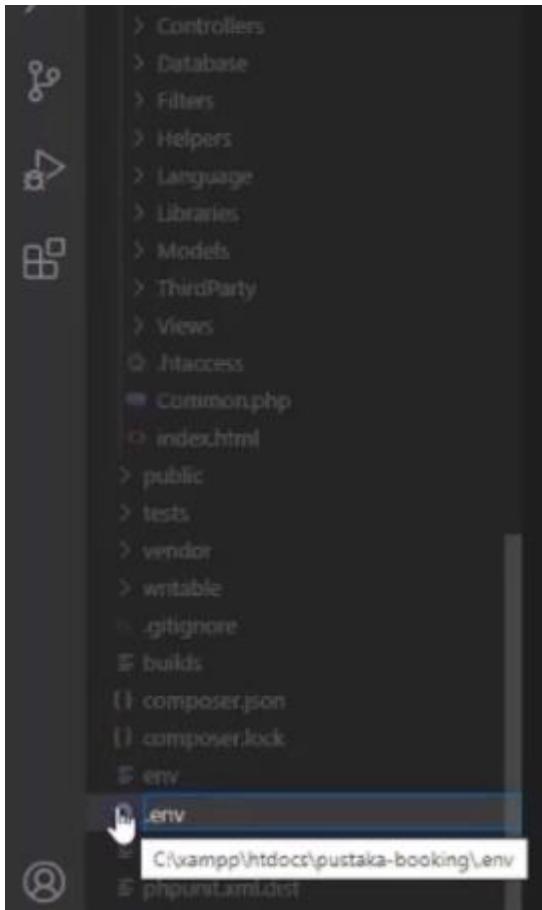
Dilanjutkan dengan klik env.



```
EXPLORER
OPEN EDITORS
PUSTAKA-BOOKING
    > Controllers
    > Database
    > Filters
    > Helpers
    > Language
    > Libraries
    > Models
    > ThirdParty
    > Views
    .htaccess
    Common.php
        index.html
    public
    tests
    vendor
    writable
    .gitignore
    builds
    composer.json
    composer.lock
    env
    license.txt
    phunit.xml.dist
    README.md
    spark
```

```
env
1  #
2  # Example Environment Configuration file
3  #
4  # This file can be used as a starting point for your own
5  # custom .env files, and contains most of the possible settings
6  # available in a default install.
7  #
8  # By default, all of the settings are commented out. If you want
9  # to override the setting, you must un-comment it by removing the '#'
10 # at the beginning of the line.
11 #
12 #
13 #
14 # ENVIRONMENT
15 #
16 #
17 # CI_ENVIRONMENT = production
18 #
19 #
20 # APP
21 #
22 #
23 # app.baseURL = ''
24 # app.forceGlobalSecureRequests = false
25 #
26 # app.sessionDriver = 'CodeIgniter\Session\Handlers\FileHandler'
27 # app.sessionCookieName = 'ci_session'
28 # app.sessionSavePath = NULL
29 # app.sessionMatchIP = false
30 # app.sessionTimeToUpdate = 300
31 # app.sessionRegenerateOnSetCookie = false
```

Env sendiri merupakan singkatan dari environment. Setelah env tersebut diklik. Klik kanan pada env lalu pilih copy. Setalah dicopy/disalin rename dengan cara klik kanan pilih rename. Rename dengan nama: .env. Hasilnya seperti berikut:



Setelah direname maka simbol file env akan berubah menjadi simbol setting atau konfigurasi seperti yang terlihat pada gambar di atas.

Tahap selanjutnya hapus hashtag yang terdapat pada CI_Environment dan ubah jenisnya dari production menjadi developmnet. Hal ini dengan tujuan selama pengodingan ketika terjadi kesalahan dalam script akan diperlihatkan atau ditampilkan. Karena jika jenisnya production ketika program debug maka kesalahan tidak akan diperlihatkan oleh sistem. Adapun detailnya bisa dilihat pada gambar di bawah ini:

```
1 #-----
2 # Example Environment Configuration file
3 #
4 # This file can be used as a starting point for your own
5 # custom .env files, and contains most of the possible settings
6 # available in a default install.
7 #
8 # By default, all of the settings are commented out. If you want
9 # to override the setting, you must un-comment it by removing the '#'
10 # at the beginning of the line.
11 #-----
12 #
13 #
14 # ENVIRONMENT
15 #
16 #
17 # CI_ENVIRONMENT = production
18 |
19 #
20 # APP
21 #
22 #
23 # app.baseURL = ''
24 # app.forceGlobalSecureRequests = false
25 #
26 # app.sessionDriver = 'CodeIgniter\Session\Handlers\FileHandler'
27 # app.sessionCookieName = 'ci_session'
```

Tampilan setelah diubah:

```
1 #-----
2 # Example Environment Configuration file
3 #
4 # This file can be used as a starting point for your own
5 # custom .env files, and contains most of the possible settings
6 # available in a default install.
7 #
8 # By default, all of the settings are commented out. If you want
9 # to override the setting, you must un-comment it by removing the '#'
10 # at the beginning of the line.
11 #
12 #
13 #
14 # ENVIRONMENT
15 #
16 #
17 CI_ENVIRONMENT = development
18 |
19 #
20 # APP
21 #
22 #
23 # app.baseURL = ''
24 # app.forceGlobalSecureRequests = false
```

Jangan lupa untuk selalu CTRL+S setelah melakukan perubahan pada script. Kemudian cek di browser, dan selalu ingat ketika ada yang baru dirubah selalu refresh atau reload laman browsernya agar yang ditampilkan sesuai dengan update-an yang terbaru.

Learn

The User Guide contains an introduction, tutorial, a number of "how to" guides, and then reference documentation for the components that make up the framework. Check the [User Guide](#)!

Discuss

CodeIgniter is a community-developed open source project, with several venues for the community members to gather and exchange ideas. View all the threads on [CodeIgniter's forum](#), or [chat on Slack](#)!

Contribute

CodeIgniter is a community driven project and accepts contributions of code and documentation from the community. Why not [join us](#)?

Page rendered in 0.0251 seconds

Environment: development

© 2020 CodeIgniter Foundation. CodeIgniter is open source project released under the MIT open source licence.

Dan dapat dilihat pada browser di atas pada keterangan Environment berubah menjadi development.

Konfigurasi env selanjutnya yakni pada app.baseUrl=' '. Caranya sama dengan yang sebelumnya hapus hashtag (#) di depan scriptnya.

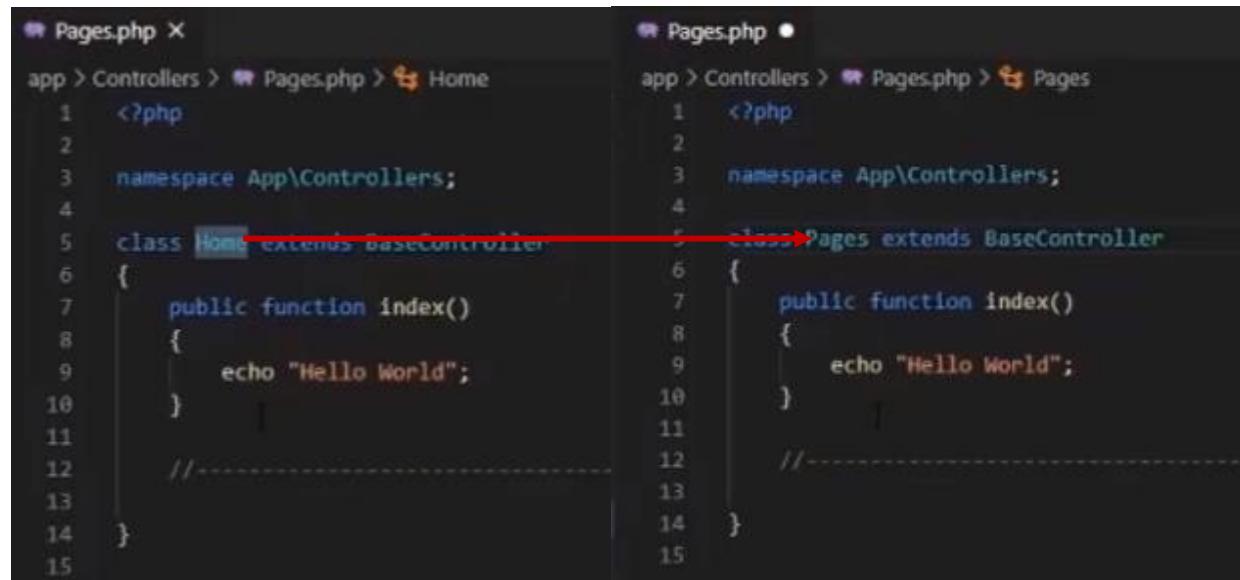
```
① .env
 6 # available in a default install.
 7 #
 8 # By default, all of the settings are commented out. If you want
 9 # to override the setting, you must un-comment it by removing the '#'
10 # at the beginning of the line.
11 #-----
12 #
13 #-----
14 # ENVIRONMENT
15 #-----
16
17 CI_ENVIRONMENT = development
18
19 #-----
20 # APP
21 #-----
22
23 # app.baseUrl = ''
24 # app.forceGlobalSecureRequests = false
25
```

Setelah dihapus hashtag maka script tersebut akan aktif.

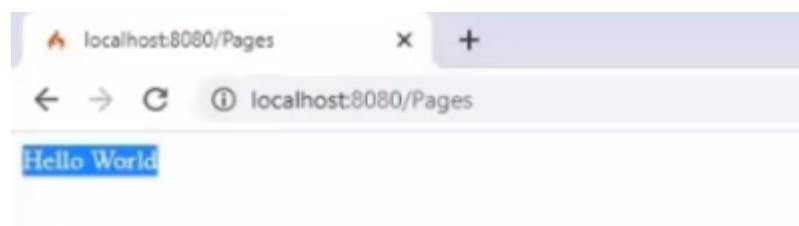
Di dalam kutip baseURL tambahkan hostnamanya.

```
23 app.baseURL = 'http://localhost:8080'  
24 # app.forceGlobalSecureRequests = false  
25  
26 # app.sessionDriver = 'CodeIgniter\Session\Handlers\FileHandler'
```

Ganti script Home pada Controller Pages.php menjadi Pages



Jika sudah, langsung reload di browser. Dan dapat dilihat hasilnya sebagai berikut:



Hal di atas dilakukan untuk testing apakah fungsi controller sudah berfungsi atau belum.

Tahap selanjutnya masuk ke laman bootstrap, pastikan sudah terkoneksi internet. Karena untuk ke laman resmi bootstrap dibutuhkan akses ke internet.



Klik documentation.

The screenshot shows the Bootstrap Documentation homepage. The main content area features a large heading 'Introduction'. Below it, a paragraph explains how to get started with Bootstrap. To the right of the main content, there's a sidebar with a search bar and a list of 'Getting started' links. On the far right, there's a sidebar with links to 'Quick start', 'Starter template', and other documentation sections.

Pilih Starter template.

The screenshot shows the 'Starter template' page. It displays a block of HTML code with syntax highlighting. To the right of the code, there's a 'Copy' button. A sidebar on the right side of the page lists various documentation links, with 'Starter template' being the one currently selected.

Klik copy pilih Starter template.

Starter template

Be sure to have your pages set up with the latest design and development standards. That means using an HTML5 doctype and including a viewport meta tag for proper responsive behaviors. Put it all together and your pages should look like this:

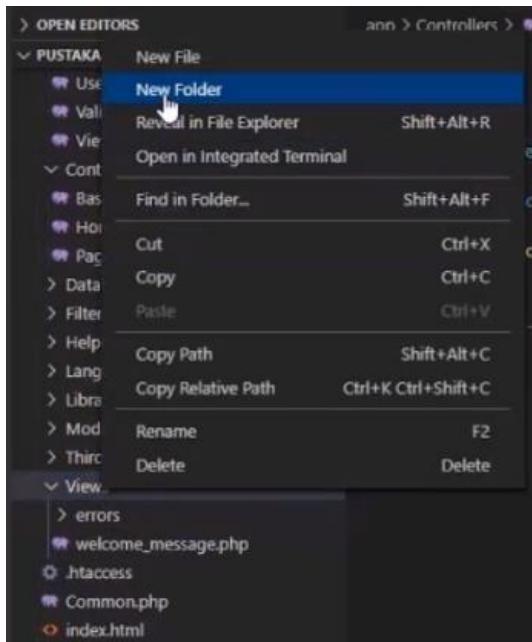
```
<!doctype html>
<html lang="en">
  <head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

    <!-- Bootstrap CSS -->
    <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">

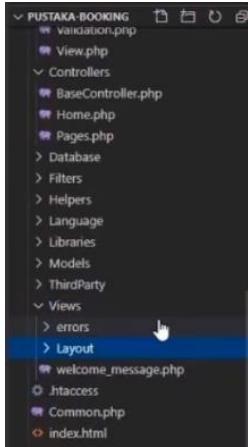
    <title>Hello, world!</title>
  </head>
  <body>
    <h1>Hello, world!</h1>
```

Copy to clipboard
Copy

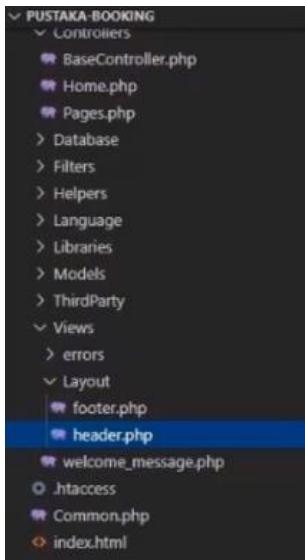
Setelah itu kembali lagi ke visual studio code. Buat folder baru di dalam folder view.



Buat nama foldernya dengan nama Layout.



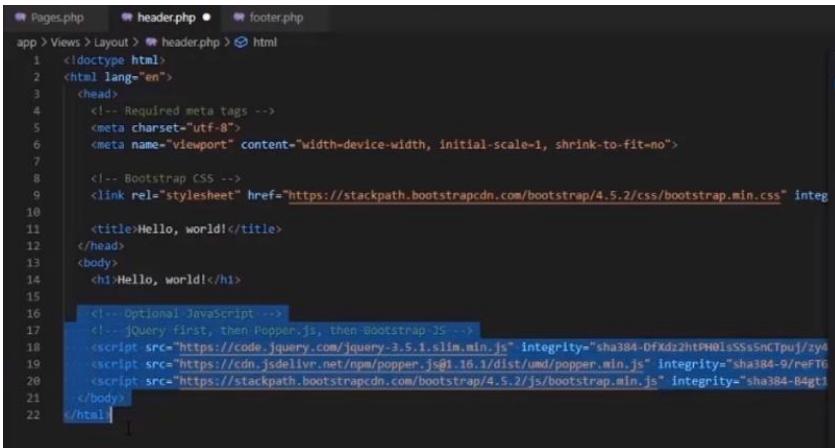
Buat file dengan nama header.php dan footer.php di dalam folder Layout.



Lalu paste hasil salinan sebelumnya di laman file header.php.

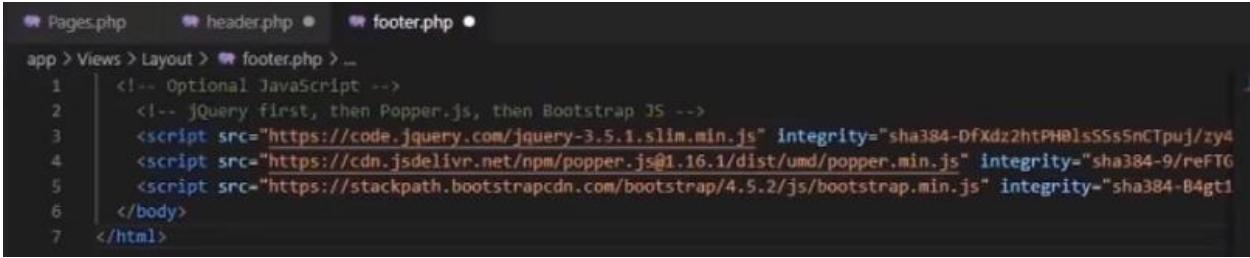
```
<!DOCTYPE html>
<html lang="en">
<head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <!-- Bootstrap CSS -->
    <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" integrity="sha384-BX上前的代码被截断了，显示为省略号。-->
    <title>Hello, world!</title>
</head>
<body>
    <h1>Hello, world!</h1>
    <!-- Optional JavaScript -->
    <!-- jQuery first, then Popper.js, then Bootstrap JS -->
    <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js" integrity="sha384-DfXdz2htPH0lsSSs5nCTpuj/zy4&lt;script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js" integrity="sha384-9/reFTG&lt;script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js" integrity="sha384-B4gT&lt;/body>
```

Pindahkan script yang diblock seperti gambar di bawah ini ke laman file footer.php.



```
Pages.php      header.php • footer.php
app > Views > Layout > header.php > html
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <!-- Required meta tags -->
5     <meta charset="utf-8">
6     <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
7
8     <!-- Bootstrap CSS -->
9     <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" integrity="sha384-Bmorb3zrXQHQaE+oIjCfD9s7SsZ8eHdgD8I7WqDfzDfzJLcA=" crossorigin="an
```

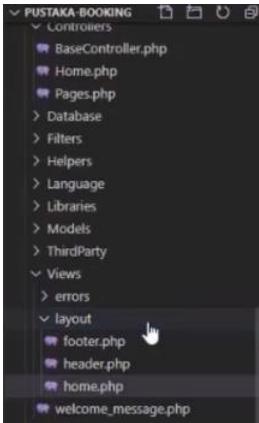
The code editor shows the `header.php` file with a block of JavaScript code highlighted. The code includes comments indicating the order of loading: "jQuery first, then Popper.js, then Bootstrap JS". The highlighted code consists of three `<script>` tags: one from `code.jquery.com`, one from `cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js`, and one from `stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js`.



```
Pages.php      header.php • footer.php •
app > Views > Layout > footer.php > ...
1   <!-- Optional JavaScript -->
2   <!-- jQuery first, then Popper.js, then Bootstrap JS -->
3   <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js" integrity="sha384-DfXdz2htPH0lsSSsnCTpuj/zy4" crossorigin="an
4   <script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js" integrity="sha384-9/reFTG
5   <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js" integrity="sha384-Bmorb3zrXQHQaE+oIjCfD9s7SsZ8eHdgD8I7WqDfzDfzJLcA=" crossorigin="an
```

The code editor shows the `footer.php` file with the same block of JavaScript code now moved here. The code remains identical to the previous screenshot.

Setelah dipindahkan CTRL+S ke kedua halaman tersebut. Tambahkan file home.php di dalam folder layout. Dan rename folder Layout menjadi layout. Dari huruf L besar ke huruf l kecil.



Pindahkan Hello World yang ada di laman file header.php ke dalam laman home.php yang ada di dalam folder layout.

```
Pages.php      home.php     header.php • footer.php
app > Views > layout > header.php > html > body
1   <!doctype html>
2   <html lang="en">
3
4   <head>
5       <!-- Required meta tags -->
6       <meta charset="utf-8">
7       <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
8
9       <!-- Bootstrap CSS -->
10      <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" integrity="sha384-Bm5mORqyjPogA0v3xVxV9cOaxXzT0oZ0nGpHw+XeXh51tQIuN0J3" crossorigin="anonymous">
11
12      <title>Hello, world!</title>
13  </head>
14
15  <body>
16      <h1>Hello, world!</h1>
```

Ubah kodingan yang ada di dalam file Pages.php seperti gambar di bawah ini:

```
Pages.php X home.php header.php footer.php

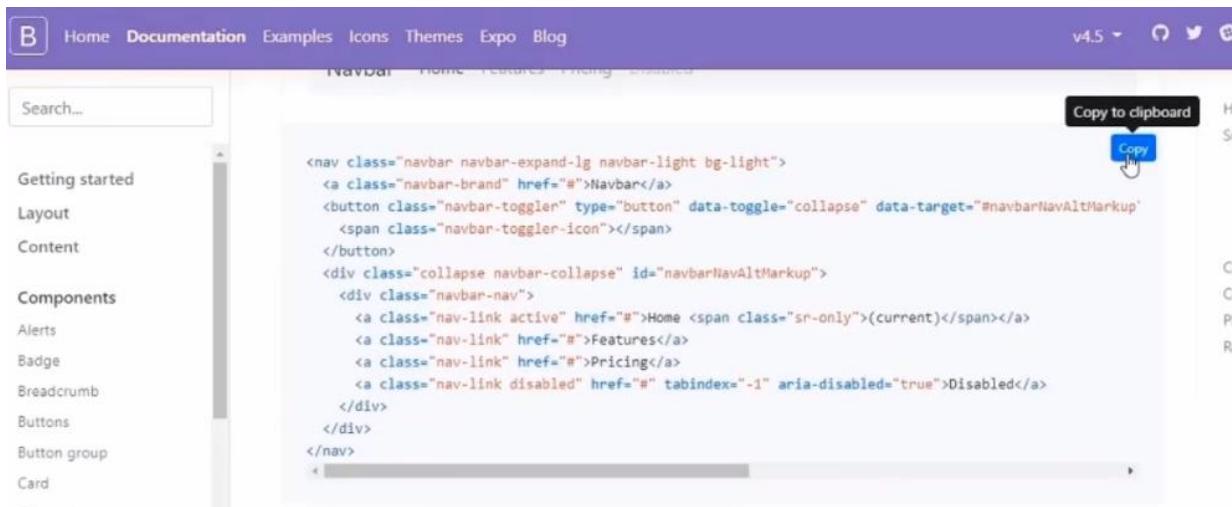
app > Controllers > Pages.php > ...
1 <?php
2
3 namespace App\Controllers;
4
5 class Pages extends BaseController
6 {
7     public function index()
8     {
9         echo view('layout/header');
10        echo view('layout/home');
11        echo view('layout/footer');
12    }
13
14    // ...
15
16 }
17
```

Kembali lagi ke laman bootstrap kemudian ketikkan nav pada search box. Lalu klik yang navbar.

Many of our components require the use of JavaScript to function. Specifically, they require [jQuery](#), [Popper.js](#), and our own JavaScript plugins. Place the following `<script>`s near the end of your page's body, right before the closing `</body>` tag, to enable them. jQuery must come first, followed by Popper.js, and then our JavaScript plugins.

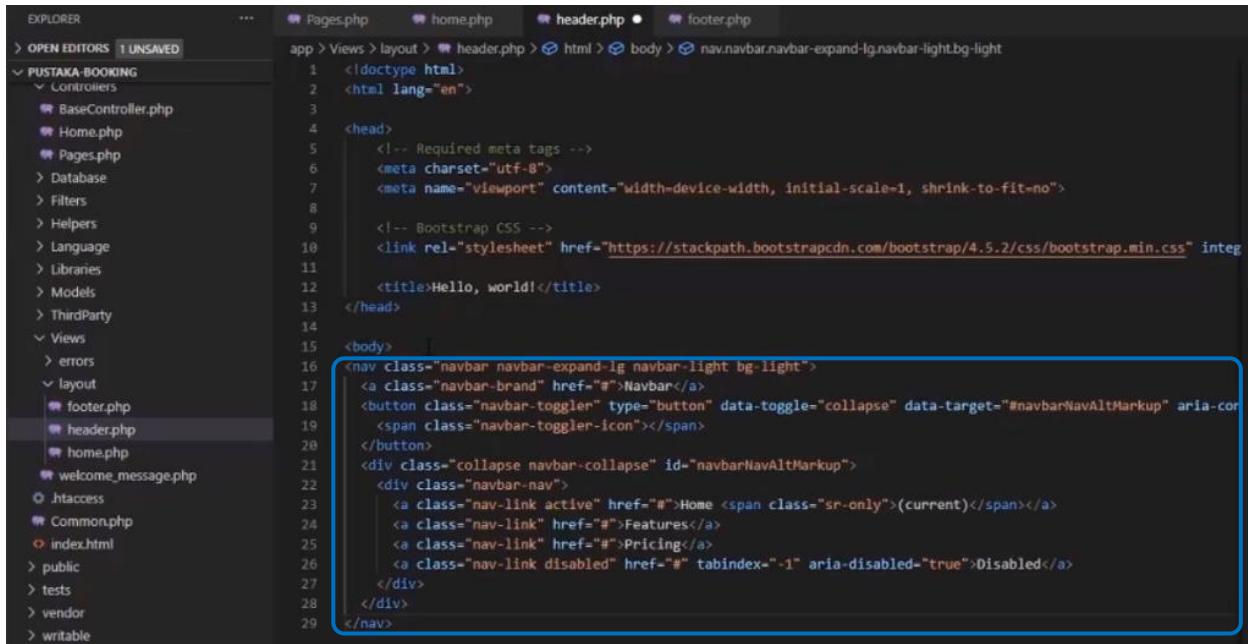
Curious which components explicitly require jQuery, our JS, and Popper.js? Click the show components link below. If you're at all unsure about the general page structure, keep reading for an example page template.

Copy navbar jenis navbar yang terdapat pada baris kedua.



The screenshot shows the Bootstrap documentation website's sidebar with categories like 'Getting started', 'Layout', 'Content', 'Components', 'Alerts', 'Badge', 'Breadcrumb', 'Buttons', 'Button group', and 'Card'. On the right, a code snippet for a 'navbar-expand-lg' navbar is displayed. A 'Copy to clipboard' button with a 'Copy' icon is visible, and a mouse cursor is hovering over it.

Tempel atau paste di laman file header.php di bawah coding <body>



The screenshot shows a code editor with a file structure on the left. The 'header.php' file under the 'Views/layout' directory is selected and highlighted with a blue border. The code in this file is identical to the one shown in the previous screenshot.

```
<nav class="navbar navbar-expand-lg navbar-light bg-light">
  <a class="navbar-brand" href="#">Navbar</a>
  <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNavAltMarkup" aria-controls="navbarNavAltMarkup" aria-expanded="false" aria-label="Toggle navigation">
    <span class="navbar-toggler-icon"></span>
  </button>
  <div class="collapse navbar-collapse" id="navbarNavAltMarkup">
    <div class="navbar-nav">
      <a class="nav-link active" href="#">Home <span class="sr-only">(current)</span></a>
      <a class="nav-link" href="#">Features</a>
      <a class="nav-link" href="#">Pricing</a>
      <a class="nav-link disabled" href="#" tabindex="-1" aria-disabled="true">Disabled</a>
    </div>
  </div>
</nav>
```

Sisipkan koding berikut (pada gambar yang diberikan frame warna biru) dan hilangkan (hapus) koding yang di block sesuai yang terlihat pada gambar di bawah ini.

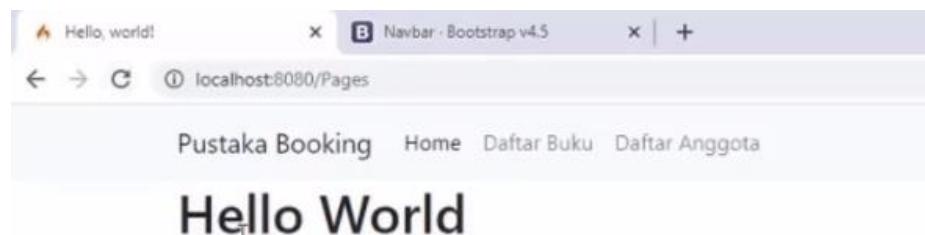
```

<body>
    <nav class="navbar navbar-expand-lg navbar-light bg-light">
        <div class="container">
            <a class="navbar-brand" href="#">Pustaka Booking</a>
            <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNavAltMarkup">
                <span class="navbar-toggler-icon"></span>
            </button>
            <div class="collapse navbar-collapse" id="navbarNavAltMarkup">
                <div class="navbar-nav">
                    <a class="nav-link active" href="#">Home <span class="sr-only">(current)</span></a>
                    <a class="nav-link" href="#">Daftar Buku</a>
                    <a class="nav-link" href="#">Daftar Anggota</a>
                    <a class="nav-link disabled" href="#" tabindex="-1" aria-disabled="true">Disabled</a>
                </div>
            </div>
        </div>
    </nav>

```

Lalu sesuaikan koding yang ada pada laman home.php seperti gambar di bawah ini.

Reload/refresh laman browser maka akan tampil tampilan seperti ini:



Langkah selanjutnya masuk ke folder Controller ->Routes.php. Rubah Home menjadi Pages.

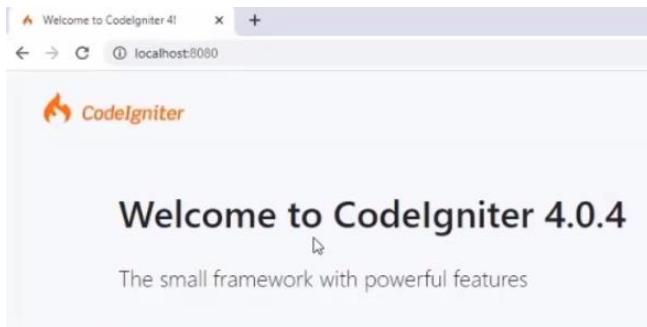
```

    app > Config > Routes.php > ...
    12
    13 /**
    14 *
    15 * Router Setup
    16 *
    17 */
    18 $routes->setDefaultNamespace('App\Controllers');
    19 $routes->setDefaultController('Home');
    20 $routes->setDefaultMethod('index');
    21 $routes->setTranslateURIDashes(false);
    22 $routes->set404Override();
    23 $routes->setAutoRoute(true);
    24 /**
    25 *
    26 * Route Definitions
    27 *
    28 */
    29 */
    30
    31 // We get a performance increase by specifying the default
    32 // route since we don't have to scan directories.
    33 $routes->get('/', 'Home::index');

```

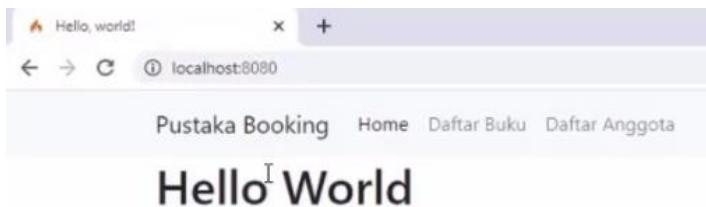
Ketika url localhost:8080/Pages pada kata Pagesnya dihapus akan tetap tampil seperti di bawah ini dan tidak ke halaman welcome lagi.

Halaman Welcome:



About this page

Halaman yang sudah dirubah Routes.phpnya ke Pages:



Artinya disini Indexnya sudah dipindah ke Pages dan bukan lagi ke Home (File Default CI 4)

PERTEMUAN 6

IMPLEMENTASI MODEL DAN CONTROLLER

Deskripsi Pembelajaran

Pada bab ini akan dibahas tentang penerapan model dan controller. Pada model akan dibahas tentang pembuatan fungsi CRUD (Create, Rename, Update dan Delete) dan penerapan DML (Data Manipulating Language). Dari menampilkan data-data buku, menambahkan data buku, mengubah data buku dan menghapus data buku.

Tujuan Pembelajaran

Pada bab ini diharapkan mahasiswa dapat memahami dalam penerapan model dan controller pada framework codeigniter 4. Dari memahami fungsi CRUD sampai dengan DML di dalam framework codeigniter 4.

6.1. Model dan Controller

Melanjutkan dari program selanjutnya. Tambahkan file di dalam folder layout dengan nama template.php. Kemudian pindahkan kodingan yang ada di dalam file header.php dan footer.php ke laman template.php. Lalu hapus file header.php dan footer.php-nya.

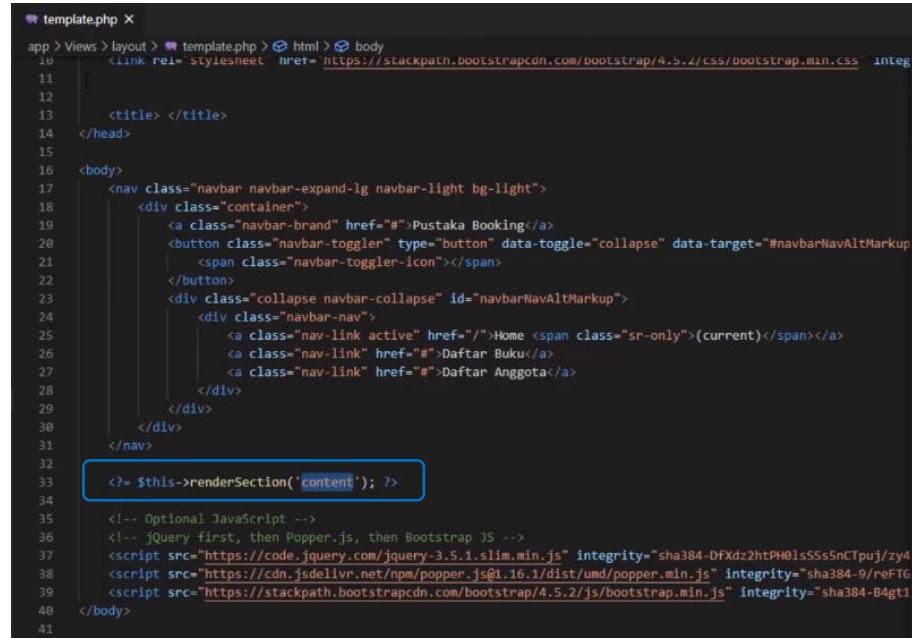
The screenshot shows the Visual Studio Code interface with the following details:

- Explorer View:** On the left, the sidebar lists project files and folders:
 - OPEN EDITORS
 - PUTASKA-BOOKING
 - app
 - Config
 - Controllers
 - Database
 - Filters
 - Helpers
 - Language
 - Libraries
 - Models
 - ThirdParty
 - Views
 - errors
 - layout
 - home.php
 - template.php
 - welcome_message.php
 - .htaccess
 - Common.php
 - index.html
 - public
 - tests
 - vendor
 - writable
 - .env
 - .gitignore
 - builds
- Editor View:** The main area displays the content of the `template.php` file. The code includes HTML structure for a navigation bar and links to Bootstrap CSS and JS files.

```
<head>
    <title> Pustaka Booking </title>
</head>
<body>
    <nav class="navbar navbar-expand-lg navbar-light bg-light">
        <div class="container">
            <a class="navbar-brand" href="#">Pustaka Booking</a>
            <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent">
                <span class="navbar-toggler-icon"></span>
            </button>
            <div class="collapse navbar-collapse" id="navbarSupportedContent">
                <div class="navbar-nav">
                    <a class="nav-link active" href="/">Home <span class="sr-only">(current)</span></a>
                    <a class="nav-link" href="#">Daftar Buku</a>
                    <a class="nav-link" href="#">Daftar Anggota</a>
                </div>
            </div>
        </div>
    </nav>

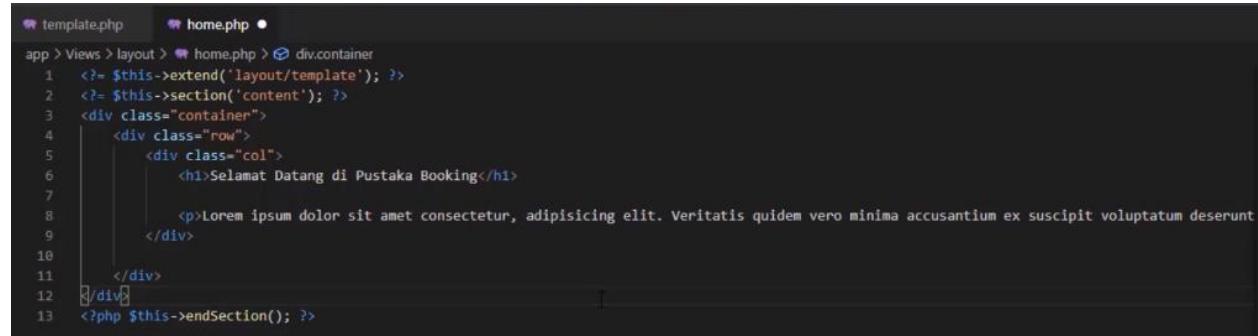
    <!-- Optional JavaScript -->
    <!-- JQuery first, then Popper.js, then Bootstrap JS -->
    <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js" integrity="sha384-DfXdz2htPH0lsS5nCTpuj/zy4&ampgt
    <script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js" integrity="sha384-9/reFTGndij6fWVZNjmTYgvigRo7xK6d4ZVFyjOOtE3x8vS9FeWVgQhD&ampgt
    <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js" integrity="sha384-B&ampgt
```

Tambahkan perintah seperti di bawah ini di dalam script yang ada di laman template.php.



```
template.php
app > Views > layout > template.php > html > body
10   <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" integrity="sha384-BPq5aR2osQ9rH08eDzq49N6IiV6oDyj3wZJnEoWfjKXo7g5Kc0KwvOOGJLJGJUw=">
11
12
13   <title> Pustaka Booking </title>
14 </head>
15
16 <body>
17   <nav class="navbar navbar-expand-lg navbar-light bg-light">
18     <div class="container">
19       <a class="navbar-brand" href="#">Pustaka Booking</a>
20       <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNavAltMarkup">
21         <span class="navbar-toggler-icon"></span>
22       </button>
23       <div class="collapse navbar-collapse" id="navbarNavAltMarkup">
24         <div class="navbar-nav">
25           <a class="nav-link active" href="/">Home <span class="sr-only" style="font-size: small;">(current)</span></a>
26           <a class="nav-link" href="#">Daftar Buku</a>
27           <a class="nav-link" href="#">Daftar Anggota</a>
28         </div>
29       </div>
30     </div>
31   </nav>
32
33   <?= $this->renderSection('content'); ?>
34
35   <!-- Optional JavaScript -->
36   <!-- jQuery first, then Popper.js, then Bootstrap JS -->
37   <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js" integrity="sha384-Dfxz2htPH0lsSS5nCTpuj/zY4js@1.16.1/dist/umd/popper.min.js" integrity="sha384-9/reFTG&lt;script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js" integrity="sha384-9/reFTG&lt;script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js" integrity="sha384-BPq5aR2osQ9rH08eDzq49N6IiV6oDyj3wZJnEoWfjKXo7g5Kc0KwvOOGJLJGJUw=">
```

Beralih dari template.php ke laman home.php yang ada di dalam folder layout. Sesuaikan koding yang ada pada home.php seperti gambar di bawah ini:



```
home.php
app > Views > layout > home.php > div.container
1  <?= $this->extend('layout/template'); ?>
2  <?= $this->section('content'); ?>
3  <div class="container">
4    <div class="row">
5      <div class="col">
6        <h1>Selamat Datang di Pustaka Booking</h1>
7
8        <p>Lorem ipsum dolor sit amet consectetur, adipisicing elit. Veritatis quidem vero minima accusantium ex suscipit voluptatum deserunt.</p>
9      </div>
10    </div>
11  </div>
12 <?= $this->endSection(); ?>
```

Khusus untuk yang script lorem ipsum itu bisa otomatis dengan mengetikkan lorem lalu tab maka kalimat selanjutnya akan muncul otomatis.

Kembali ke template.php, ubah kata Home menjadi Beranda.

```

app > Views > layout > template.php > home.php
app > Views > layout > template.php > html > body > nav.navbar.navbar-expand-lg.navbar-light.bg-light > div.container > div#navbarNavAltMarkup > div.collapse.navbar-collapse id="navbarNavAltMarkup" data-target="#navbarNavAltMarkup" style="height: 0px; overflow: hidden;">
  <title> </title>
  </head>
<body>
  <nav class="navbar navbar-expand-lg navbar-light bg-light">
    <div class="container">
      <a class="navbar-brand" href="#">Pustaka Booking</a>
      <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNavAltMarkup">
        <span class="navbar-toggler-icon"></span>
      </button>
      <div class="collapse navbar-collapse" id="navbarNavAltMarkup">
        <div class="navbar-nav">
          <a class="nav-link active" href="/">Beranda <span class="sr-only" style="font-size: small;">(current)</span></a>
          <a class="nav-link" href="#">Daftar Buku</a>
          <a class="nav-link" href="#">Daftar Anggota</a>
        </div>
      </div>
    </div>
  </nav>

```

Reload/refresh browser untuk lihat hasil perbedaanya dengan yang sebelumnya:



Selanjutnya buat file baru dengan nama Buku.php di dalam folder Controller. Lalu masukkan koding berikut:

```

app > Controllers > Buku.php > ...
1  <?php
2
3  namespace App\Controllers;
4
5  use App\Models\BukuModel;
6
7  class Buku extends BaseController
8  {
9      public function __construct()
10     {
11         $this->BukuModel = new BukuModel();
12     }
13
14     public function index()
15     {
16         $data = [
17             'title' => 'Daftar Buku',
18             'buku' => $this->BukuModel->getBuku()
19         ];
20
21         return view('buku/index', $data);
22     }
23
24     public function detail($idbuku)
25     {
26         $data = [
27             'title' => 'Detail Buku',
28             'buku' => $this->BukuModel->getBuku($idbuku)
29         ];
30
31         return view('buku/detail', $data);
32     }
33 }

```

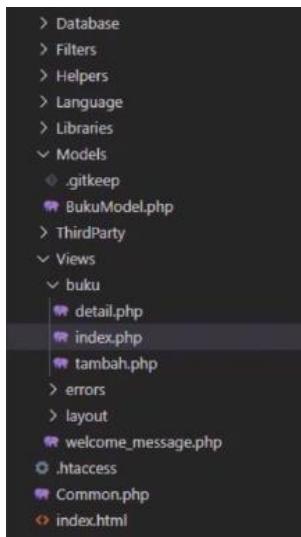
Selanjutnya buat file baru dengan nama BukuModel.php di dalam folder Models dan tuliskan script seperti gambar di bawah ini:

The screenshot shows a code editor with the following details:

- Project Name: PUSTAKA-BOOKING
- File Path: app > Models > BukuModel.php
- File Content (BukuModel.php):

```
1  <?php
2
3  namespace App\Models;
4
5  use CodeIgniter\Model;
6
7  class BukuModel extends Model
8  {
9
10     protected $table = 'buku';
11     protected $primaryKey = 'id_buku';
12
13     public function getBuku($idbuku = false)
14     {
15         if ($idbuku == false) {
16             return $this->findAll();
17         }
18
19         return $this->where(['id_buku' => $idbuku])->first();
20     }
21 }
22 }
```

Tambahkan folder baru di dalam folder View dengan nama buku. Kemudian di dalam folder buku tambahkan 3 file baru dengan nama detail.php, index.php dan tambah.php.



Kembali lagi ke laman bootstrap. Pilih documentation-> ketik di search box table.

The screenshot shows the Bootstrap documentation website at getbootstrap.com/docs/4.5/getting-started/introduction/. A search bar at the top contains the text "table". The sidebar on the left has a "Content" section with a link to "Tables / Tables" which is highlighted in purple. Below it are links for "Tables / Table head options", "Tables / Bordered table", "Tables / Borderless table", and "Tables / Small table". At the bottom of the sidebar is a "Search by Algolia" input field.

Pilih jenis table seperti yang ada pada gambar di bawah ini lalu copy.

The screenshot shows the Bootstrap documentation page for tables at <https://getbootstrap.com/docs/4.5/content/tables/>. The page title is "Tables". It features a table with three rows and four columns. The columns are labeled "#", "First", "Last", and "Handle". The rows contain data for Mark Otto (@mdo), Jacob Thornton (@fat), and Larry Bird (@twitter). Below the table is its source code. A "Copy to clipboard" button is visible on the right side of the code block. The sidebar on the left includes links for "Getting started", "Layout", "Content", "Tables", and "Components".

Paste di halaman index.php.

```
<table class="table">
<thead>
<tr>
<th scope="col">#</th>
<th scope="col">First</th>
<th scope="col">Last</th>
<th scope="col">Handle</th>
</tr>
</thead>
<tbody>
<tr>
<th scope="row">1</th>
<td>Mark</td>
<td>Otto</td>
<td>@mdo</td>
</tr>
<tr>
<th scope="row">2</th>
<td>Jacob</td>
<td>Thornton</td>
<td>@fat</td>
</tr>
<tr>
<th scope="row">3</th>
<td>Larry</td>
<td>the Bird</td>
<td>@twitter</td>
</tr>
</tbody>
```

Hapus bagian yang tidak penting dan sisakan script yang dibutuhkan. Untuk lebih detailnya bisa lihat gambar di bawah ini:

```
<table class="table">
<thead>
<tr>
<th scope="col">#</th>
<th scope="col">First</th>
<th scope="col">Last</th>
<th scope="col">Handle</th>
</tr>
</thead>
<tbody>
<tr>
<th scope="row">1</th>
<td>Mark</td>
<td>Otto</td>
<td>@mdo</td>
</tr>
<tr>
<th scope="row">2</th>
<td>Jacob</td>
<td>Thornton</td>
<td>@fat</td>
</tr>
<tr>
<th scope="row">3</th>
<td>Larry</td>
<td>the Bird</td>
<td>@twitter</td>
</tr>
</tbody>
```

Tambahkan script di index.php seperti di bawah ini:

```
index.php ● home.php  
app > Views > buku > index.php > div.container > div.row > div.col  
1  <?= $this->extend('layout/template'); ?>  
2  <?= $this->section('content'); ?>  
3  <div class="container">  
4      <div class="row">  
5          <div class="col">  
6              <table class="table">  
7                  <thead>  
8                      <tr>  
9                          <th scope="col">#</th>  
10                         <th scope="col">First</th>  
11                         <th scope="col">Last</th>  
12                         <th scope="col">Handle</th>  
13                     </tr>  
14                 </thead>  
15                 <tbody>  
16                     <tr>  
17                         <th scope="row">1</th>  
18                         <td>Mark</td>  
19                         <td>Otto</td>  
20                         <td>@mdo</td>  
21                     </tr>  
22                 </tbody>  
23             </table>  
24     <?= $this->endSection(); ?>
```

Tahap selanjutnya lengkapi scriptnya seperti di bawah ini:

```
OPEN EDITORS  
PUSTAKA-BOOKING  
CONTAINERS  
BaseController.php  
Buku.php  
Home.php  
Pages.php  
Database  
Filters  
Helpers  
Language  
Libraries  
Models  
ThirdParty  
Views  
buku  
detail.php  
index.php  
tambah.php  
errors  
layout  
home.php  
template.php  
welcome_message.php  
.htaccess  
Common.php  
index.html  
public  
app > Views > buku > index.php > div.container > div.row > div.col > table.table > tbody > tr > td > img  
1  <?= $this->extend('layout/template'); ?>  
2  <?= $this->section('content'); ?>  
3  <div class="container">  
4      <div class="row">  
5          <div class="col">  
6              <h1>Daftar Buku</h1>  
7              <table class="table">  
8                  <thead>  
9                      <tr>  
10                         <th scope="col">No.</th>  
11                         <th scope="col">Sampul</th>  
12                         <th scope="col">Judul</th>  
13                         <th scope="col">Aksi</th>  
14                     </tr>  
15                 </thead>  
16                 <tbody>  
17                     <?php  
18                         $i = 1;  
19                         foreach ($buku as $b) :  
20                     ?>  
21                         <tr>  
22                             <th scope="row"><?= $i++; ?></th>  
23                             <td></td>  
24                             <td><?= $b['judul']; ?></td>  
25                             <td><a href="#" class="btn btn-success">Detail</a></td>  
26                         </tr>  
27                     <?php endforeach; ?>  
28                 </tbody>  
29             </table>  
30         <?= $this->endSection(); ?>
```

Halaman index ini nantinya berfungsi sebagai view dari daftar buku. Di dalamnya terdapat fungsi perulangan (foreach) yang berguna untuk memanggil semua data buku yang terdapat pada tabel buku.

Tahapan selanjutnya masukkan beberapa gambar pada disk C -> xampp -> htdocs -> pustaka-booking -> public -> img.

Local Disk (C:) > xampp > htdocs > pustaka-booking > public > img



Setelah itu edit data pada record tabel buku di bagian field sampul.

| + Options | | id_buku | judul | pengarang | penerbit | tahun_terbit | sampul |
|--------------------------|--|---------|-----------------|---------------|-------------|--------------|---------|
| <input type="checkbox"/> | | 11 | PHP | Andre Pratama | DuniaIlkom | 2017 | php.jpg |
| <input type="checkbox"/> | | 12 | Pemrograman Web | Rohi Abdulloh | Andi Offset | 2019 | web.jpg |

Nama sampul harus sesuai dengan nama gambar yang ada pada folder img. Format gambarnya pun harus sama dengan yang ada di dalam folder img.

Ubah beberapa script pada template.php. Yakni pada bagian hrefnya.

```
<body>
    <nav class="navbar navbar-expand-lg navbar-light bg-light">
        <div class="container">
            <a class="navbar-brand" href="#">Pustaka Booking</a>
            <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNavAltMarkup">
                <span class="navbar-toggler-icon"></span>
            </button>
            <div class="collapse navbar-collapse" id="navbarNavAltMarkup">
                <div class="navbar-nav">
                    <a class="nav-link active" href="/pages">Beranda <span class="sr-only">(current)</span></a>
                    <a class="nav-link" href="/buku">Daftar Buku</a>
                    <a class="nav-link" href="#">Daftar Anggota</a>
                </div>
            </div>
        </div>
    </nav>
```

Tambahkan script di Routes.php

```
 29  * -----
 30  */
 31
 32 // We get a performance increase by specifying the default
 33 // route since we don't have to scan directories.
 34 $routes->get('/', 'Pages::index');
 35
 36 $routes->get('/buku/(:any)', 'Buku::detail/$1');
 37
 38 /**
 39
 40
 41
 42
 43
 44
 45
```

Setelah itu jangan lupa setting env untuk koneksi databasenya. Hapus hashtag (#) pada script yang diblock di bawah ini:

```
 46 # app.CSPEnabled = false
 47
 48 #-----
 49 # DATABASE
 50 #
 51
 52 #database.default.hostname = localhost
 53 #database.default.database = ci4
 54 #database.default.username = root
 55 #database.default.password = root
 56 #database.default.DBDriver = MySQL
 57
 58 # database.tests.hostname = localhost
 59 # database.tests.database = ci4
 60 # database.tests.username = root
 61 # database.tests.password = root
 62 # database.tests.DBDriver = MySQL
 63
```

Lalu ubah databasenya menjadi pustaka (sesuai nama database yang sudah di buat di phpmyadmin sebelumnya.

```
 51
 52 database.default.hostname = localhost
 53 database.default.database = pustaka
 54 database.default.username = root
 55 database.default.password = root
 56 database.default.DBDriver = MySQL
 57
 58 # database.tests.hostname = localhost
 59 # database.tests.database = ci4
 60 # database.tests.username = root
 61 # database.tests.password = root
 62 # database.tests.DBDriver = MySQL
 63
```

Dan reload browser untuk melihat hasilnya:

| No. | Sampul | Judul | Aksi |
|-----|--------|-----------------|-------------------------|
| 1 | | PHP | <button>Detail</button> |
| 2 | | Pemrograman Web | <button>Detail</button> |

Selanjutnya untuk script di dalam file detail.php dibutuhkan card file. Untuk card file bisa dengan menyalin di template bootstrap. Search di search box card file. Pilih yang jenis horizontal lalu copy.

Search... gutters and use .col-md-* classes to make the card horizontal at the md breakpoint. Further adjustments may be needed depending on your card content.

Getting started
Layout
Content

Components
Alerts
Badge
Breadcrumb
Buttons
Button group
Card
Carousel
Collapse
Dropdowns
Forms
Input group

Kitchen sink
Header and footer
Sizing
Using grid markup
Using utilities
Using custom CSS
Text alignment
Navigation
Images
Image caps
Image overlays

Horizontal
Card styles
Background and color
Border
Mixins utilities
Card layout
Card groups
Card decks
Grid cards

```
<div class="card mb-3" style="max-width: 540px;">
  <div class="row no-gutters">
    <div class="col-md-4">
      
    </div>
    <div class="col-md-8">
```

Paste di dalam file detail.php.

> OPEN EDITORS

- > PUSTAKA-BOOKING
 - <> Controllers
 - BaseController.php
 - Buku.php
 - Home.php
 - Pages.php
 - > Database
 - > Filters
 - > Helpers
 - > Language
 - > Libraries
 - > Models
 - > ThirdParty
 - > Views
 - <> buku
 - detail.php
 - index.php
 - tambah.php
 - > errors

```
app > Views > buku > detail.php > div.card.mb-3
1  <div class="card mb-3" style="max-width: 540px;">
2    <div class="row no-gutters">
3      <div class="col-md-4">
4        
5      </div>
6      <div class="col-md-8">
7        <div class="card-body">
8          <h5 class="card-title">Card title</h5>
9          <p class="card-text">This is a wider card with supporting text below as a natural lead-in to add
10         <p class="card-text"><small class="text-muted">Last updated 3 mins ago</small></p>
11       </div>
12     </div>
13   </div>
14 </div>
```

Kemudian sempurnakan kodingnya seperti di bawah ini:

> OPEN EDITORS

- > PUSTAKA-BOOKING
 - <> Controllers
 - BaseController.php
 - Buku.php
 - Home.php
 - Pages.php
 - > Database
 - > Filters
 - > Helpers
 - > Language
 - > Libraries
 - > Models
 - > ThirdParty
 - > Views
 - <> buku
 - detail.php
 - index.php
 - tambah.php
 - > errors
 - > layout
 - home.php
 - template.php
 - welcome_message.php
 - > .htaccess
 - > Common.php

```
app > Views > buku > detail.php > div.container > div.row > div.col > div.card.mb-3 > div.row.no-gutters > div.col-md-8 >
1  <?= $this->extend('layout/template'); ?>
2  <?= $this->section('content'); ?>
3  <div class="container">
4    <div class="row">
5      <div class="col">
6        <h3 class="mt-2">Detail Buku</h3>
7        <div class="card mb-3" style="max-width: 540px;">
8          <div class="row no-gutters">
9            <div class="col-md-4">
10           
11         </div>
12         <div class="col-md-8">
13           <div class="card-body">
14             <h5 class="card-title"><?= $buku['judul']; ?></h5>
15             <p class="card-text"><b>Pengarang:</b> <?= $buku['pengarang']; ?></p>
16             <p class="card-text">Penerbit:<?= $buku['penerbit']; ?><small></small></p>
17             <p class="card-text">tahun Terbit:<?= $buku['tahun_terbit']; ?><small></small></p>
18             <a href="#" class="btn btn-warning">Ubah</a>
19             <a href="#" class="btn btn-danger">Hapus</a>
20             <br><br>
21             <a href="/buku">Kembali Ke Daftar Buku</a>
22           </div>
23         </div>
24       </div>
25     </div>
26   <?= $this->endSection(); ?>
27 </div>
```

Pada href di index.php pada tombol edit diubah menjadi sebagai berikut:

```
<tr>
  <th scope="row"><?= $i++; ?></th>
  <td></td>
  <td><?= $b['judul']; ?></td>
  <td><a href="/buku/<?= $b['id_buku']; ?>" class="btn btn-success">Detail</a></td>
</tr>
```

Klik tombol detail pada laman index untuk menampilkan laman detail.

| No. | Sampul | Judul | Aksi |
|-----|--------|-----------------|-------------------------|
| 1 | | PHP | <button>Detail</button> |
| 2 | | Pemrograman Web | <button>Detail</button> |

Setelah klik detail hasil dari laman detailnya seperti gambar di bawah ini:

Detail Buku

Pemrograman Web
Pengarang: Rohi Abdullah
Penerbit: Andi Offset
tahun Terbit: 2019
[Ubah](#) [Hapus](#)
[Kembali Ke Daftar Buku](#)

Jika ingin kembali ke daftar buku bisa dengan klik Kembali Ke Daftar Buku.

Tahapan selanjutnya ke fungsi tambah data buku. Ada yang perlu ditambahkan pada laman ModelBuku.php.

```
1 <?php
2
3 namespace App\Models;
4
5 use CodeIgniter\Model;
6
7 class BukuModel extends Model
8 {
9     protected $table = 'buku';
10    protected $primaryKey = 'id_buku';
11    protected $allowedFields = ['judul', 'pengarang', 'penerbit', 'tahun_terbit', 'sampul'];
12
13    public function getBuku($idbuku = false)
14    {
15        if ($idbuku == false) {
16            return $this->findAll();
17        }
18
19        return $this->where(['id_buku' => $idbuku])->first();
20    }
21
22 }
```

Tambahkan method tambah pada file Buku.php di dalam folder Controllers.

```
app > Controllers > Buku.php > Buku > simpan
```

```
26     $data = [
27         'title' => 'Detail Buku',
28         'buku' => $this->BukuModel->getBuku($idbuku)
29     ];
30
31     return view('buku/detail', $data);
32 }
33
34 public function tambah()
35 {
36     $data = [
37         'title' => 'Form Tambah Data Buku',
38         'validation' => \Config\Services::validation()
39     ];
40
41     return view('buku/tambah', $data);
42 }
43
```

Kurang huruf s,
harusnya Services

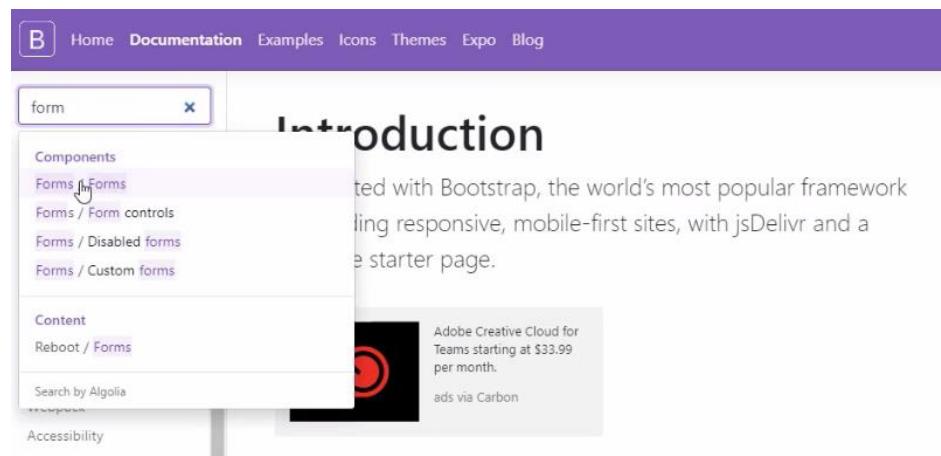
Kemudian tambahkan method simpan.

```
44     public function simpan()
45     {
46         //validasi
47         if (!$this->validate(
48             [
49                 'judul' => [
50                     'rules' => 'required',
51                     'errors' => ['required' => '{field} harus diisi']
52                 ]
53             ]
54         ))
55         {
56             $validation = \Config\Services::validation();
57             return redirect()->to('/buku/tambah')->withInput()->with('validation', $validation);
58         }
59
60         //simpan
61         $this->BukuModel->save(
62             [
63                 'judul' => $this->request->getVar('judul'),
64                 'pengarang' => $this->request->getVar('pengarang'),
65                 'penerbit' => $this->request->getVar('penerbit'),
66                 'tahun_terbit' => $this->request->getVar('tahun'),
67                 'sampul' => $this->request->getVar('sampul')
68             ]
69         );
70
71         session()->setflashdata('pesan', 'Data Berhasil Ditambahkan.');
72         return redirect()->to('/buku');
73     }
```

Tambahkan routes untuk tambah di Routes.php.

```
32     // We get a performance increase by specifying the default
33     // route since we don't have to scan directories.
34     $routes->get('/', 'Pages::index');
35
36     $routes->get('/buku/tambah', 'Buku::tambah');
37     $routes->get('/buku/(:any)', 'Buku::detail/$1');
38
```

Untuk rancangan tambah.php dibutuhkan css dari bootstrap.



Pilih yang jenis form: Horizontal form.

The screenshot shows the official Bootstrap documentation for forms. On the left, there's a sidebar with navigation links for various form-related topics like Layout, Form groups, Form grid, etc. The main content area features a heading 'Forms' and a sub-section titled 'Horizontal form'. Below this, there's a brief description and a small image of a woman's face. A call-to-action banner for Adobe Stock is visible at the bottom of the main content area.

Lalu paste di laman tambah.php

The screenshot shows a code editor with the file 'tambah.php' selected in the sidebar. The code is a PHP script with embedded HTML and CSS. It uses Bootstrap's grid system to create horizontal forms. The code includes labels ('Email', 'Password'), input fields ('text', 'password'), and radio buttons ('radio') for a 'Radios' group. The code is numbered from 1 to 32.

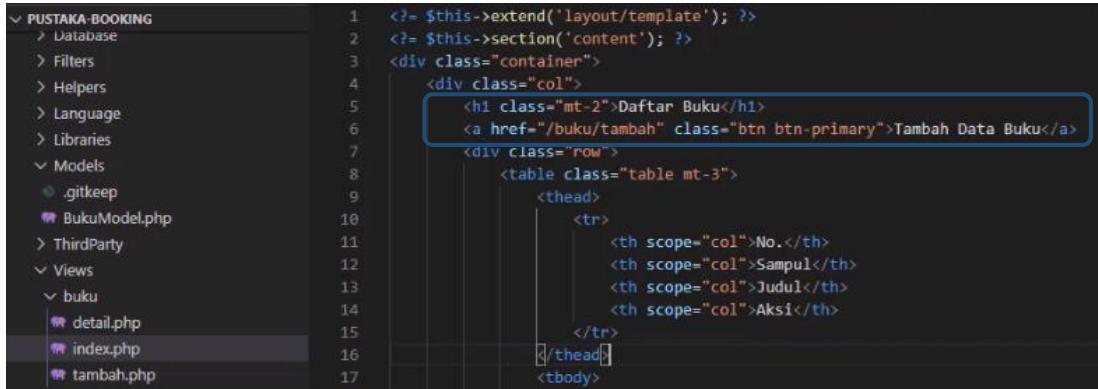
```
<?>= $this->extend('layout/template'); ?>
<?= $this->section('content'); ?>
<div class="container">
  <div class="row">
    <form>
      <div class="form-group row">
        <label for="inputEmail3" class="col-sm-2 col-form-label">Email</label>
        <div class="col-sm-10">
          <input type="email" class="form-control" id="inputEmail3">
        </div>
      </div>
      <div class="form-group row">
        <label for="inputPassword3" class="col-sm-2 col-form-label">Password</label>
        <div class="col-sm-10">
          <input type="password" class="form-control" id="inputPassword3">
        </div>
      </div>
      <fieldset class="form-group">
        <div class="row">
          <legend class="col-form-label col-sm-2 pt-0">Radios</legend>
          <div class="col-sm-10">
            <div class="form-check">
              <input class="form-check-input" type="radio" name="gridRadios" id="gridRadios1" value="option1" checked="checked">
              <label class="form-check-label" for="gridRadios1">
                First radio
              </label>
            </div>
            <div class="form-check">
              <input class="form-check-input" type="radio" name="gridRadios" id="gridRadios2" value="option2">
              <label class="form-check-label" for="gridRadios2">
                Second radio
              </label>
            </div>
          </div>
        </div>
      </fieldset>
    </form>
  </div>
</div>
```

Hapus bagian yang tidak diperlukan dan sesuaikan kodingnya seperti ini:

```
views > buku > tambah.php > div.container > div.col > form.mt-4
<?= $this->extend('layout/template'); ?>
<?= $this->section('content'); ?>
<div class="container">
    <div class="col">

        <h3 class="mt-2">Form Tambah Buku</h3>
        <form action="/buku/simpan" method="post" class="mt-4">
            <?= csrf_field(); ?>
            <div class="form-group row">
                <label for="inputJudul" class="col-sm-2 col-form-label">Judul</label>
                <div class="col-sm-10">
                    <input type="text" class="form-control" <?= ($validation->hasError('judul')) ? 'isinvalid' : ''; ?> name="judul" autofocus>
                    <div class="invalid-feedback">
                        <?= $validation->getError('judul'); ?>
                    </div>
                </div>
            </div>
            <div class="form-group row">
                <label for="inputPengarang" class="col-sm-2 col-form-label">Pengarang</label>
                <div class="col-sm-10">
                    <input type="text" class="form-control" name="pengarang" value="<?= old('pengarang'); ?>">
                </div>
            </div>
            <div class="form-group row">
                <label for="inputPenerbit" class="col-sm-2 col-form-label">Penerbit</label>
                <div class="col-sm-10">
                    <input type="text" class="form-control" name="penerbit" value="<?= old('penerbit'); ?>">
                </div>
            </div>
            <div class="form-group row">
                <label for="inputTahun" class="col-sm-2 col-form-label">Tahun Terbit</label>
                <div class="col-sm-10">
                    <input type="text" class="form-control" name="tahun" value="<?= old('tahun'); ?>">
                </div>
            </div>
            <div class="form-group row">
                <label for="inputSampul" class="col-sm-2 col-form-label">Sampul</label>
                <div class="col-sm-10">
                    <input type="text" class="form-control" name="sampul" value="<?= old('sampul'); ?>">
                </div>
            </div>
            <div class="form-group row">
                <div class="col-sm-10">
                    <button type="submit" class="btn btn-primary">Tambah</button>
                </div>
            </div>
        </form>
    </div>
</div>
```

Tambahkan script berikut di laman file index.php

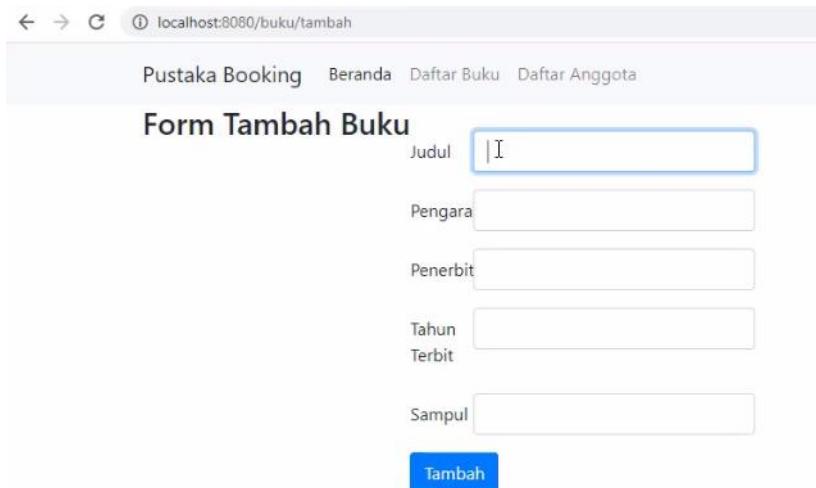


```
<?php $this->extend('layout/template'); ?>
<?= $this->section('content'); ?>
<div class="container">
    <div class="col">
        <h1 class="mt-2">Daftar Buku</h1>
        <a href="/buku/tambah" class="btn btn-primary">Tambah Data Buku</a>
        <div class="row">
            <table class="table mt-3">
                <thead>
                    <tr>
                        <th scope="col">No.</th>
                        <th scope="col">Sampul</th>
                        <th scope="col">Judul</th>
                        <th scope="col">Aksi</th>
                    </tr>
                </thead>
                <tbody>
```

Lalu reload browsernya:



Ketika klik tambah data buku maka ini yang akan tampil:



| | |
|--------------|---|
| Judul | I |
| Pengra | |
| Penerbit | |
| Tahun Terbit | |
| Sampul | |

Tambah

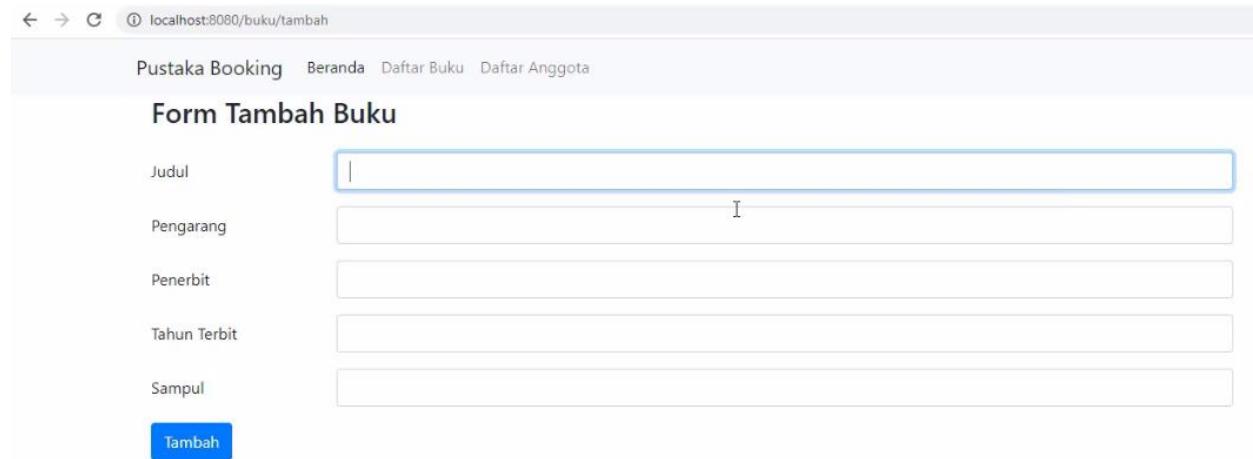
Agar formnya lebih rapi ada script yang harus di hapus pada file tambah.php. Yang dihapus seperti yang diblock biru pada gambar di bawah ini:

```
app > Views > buku >  tambah.php >  div.container >  div.col
1  <?= $this->extend('layout/template'); ?>
2  <?= $this->section('content'); ?>
3  <div class="container">
4      <div class="col">
5          

6              <h3>Tambah Buku</h3>
7              <form action="/buku/tambah" method="post" class="mt-4">
8                  <?= csrf_field(); ?>
9                  <div class="form-group row">
10                     <label for="inputJudul" class="col-sm-2 col-form-label">Judul</label>
11                     <div class="col-sm-10">
12                         <input type="text" class="form-control" <?= ($validation->hasError('judul')) ? 'is-invalid' : ''>
13                         <?= $validation->getErrors('judul'); ?>
14                     </div>
15                 </div>
16
17                 <div class="form-group row">
18                     <label for="inputPenerbit" class="col-sm-2 col-form-label">Penerbit</label>
19                     <div class="col-sm-10">
20                         <input type="text" class="form-control" name="penerbit" value="<?= old('penerbit'); ?>">
21                     </div>
22                 </div>
23                 <div class="form-group row">
24                     <label for="inputTahun" class="col-sm-2 col-form-label">Tahun Terbit</label>
25                     <div class="col-sm-10">
26                         <input type="text" class="form-control" name="tahun" value="<?= old('tahun'); ?>">
27                     </div>
28                 </div>
29                 <div class="form-group row">
30                     <label for="inputSampul" class="col-sm-2 col-form-label">Sampul</label>
31                     <div class="col-sm-10">
32                         <input type="text" class="form-control" name="sampul" value="<?= old('sampul'); ?>">
33                     </div>
34                 </div>
35                 <div class="form-group row">
36                     <div class="col-sm-10">
37                         <button type="submit" class="btn btn-primary">Tambah</button>
38                     </div>
39                 </div>
40             </div>
41         </div>
42     </div>
43     <div class="col">
44         <div class="row">
45             <div class="col">
46                 </div>
47             </div>
48         </div>
49     </div>
50 </div>
51 <?= $this->endSection(); ?>


```

Setelah diperbaiki tampilannya akan lebih rapi:



Pustaka Booking Beranda Daftar Buku Daftar Anggota

Form Tambah Buku

Judul

Pengarang

Penerbit

Tahun Terbit

Sampul

Tambah

Agar text fieldnya tidak terlalu panjang maka panjang kolomnya perlu dirubah. Ubah yang tadinya sizenya 10 menjadi 4.

```
<div class="form-group row">
    <label for="inputJudul" class="col-sm-2 col-form-label">Judul</label>
    <div class="col-sm-10">
        <input type="text" class="form-control" <?= ($validation->hasError('judul')) ? 'is-invalid' : 'is-valid'>
        <div class="invalid-feedback">
            <?= $validation->getError('judul'); ?>
        </div>
    </div>
</div>
<div class="form-group row">
    <label for="inputPengarang" class="col-sm-2 col-form-label">Pengarang</label>
    <div class="col-sm-10">
        <input type="text" class="form-control" name="pengarang" value="<?= old('pengarang'); ?>">
    </div>
</div>
<div class="form-group row">
    <label for="inputPenerbit" class="col-sm-2 col-form-label">Penerbit</label>
    <div class="col-sm-10">
        <input type="text" class="form-control" name="penerbit" value="<?= old('penerbit'); ?>">
    </div>
</div>
<div class="form-group row">
    <label for="inputTahun" class="col-sm-2 col-form-label">Tahun Terbit</label>
    <div class="col-sm-10">
        <input type="text" class="form-control" name="tahun" value="<?= old('tahun'); ?>">
    </div>
</div>
<div class="form-group row">
    <label for="inputSampul" class="col-sm-2 col-form-label">Sampul</label>
    <div class="col-sm-10">
```

Setelah diubah sizenya menjadi 4 maka hasilnya seperti ini:

Pustaka Booking Beranda Daftar Buku Daftar Anggota

Form Tambah Buku

Judul

Pengarang

Penerbit

Tahun Terbit

Sampul

Tambah

Selanjutnya agar data yang ditambahkan terdapat pesan pemeberitahuan bahwa data berhasil ditambahkan pada laman daftar buku maka perlu ditambahkan script sebagai berikut:

```
EXPLORER ... index.php x detail.php tambah.php Buku.php
OPEN EDITORS
> PUSTAKA-BOOKING
  > Config
  > Controllers
    BaseController.php
    Buku.php
    Home.php
    Pages.php
  > Database
  > Filters
  > Helpers
  > Language
  > Libraries
  > Models
    .gitkeep
    BukuModel.php
  > ThirdParty
  > Views
    > buku
      detail.php
      index.php
      tambah.php C:\xampp\htdocs\pustaka-booking\app\Views\buku\index.php + 1;
    > errors
    > layout
    welcome_message.php
  .htaccess
  Common.php
  index.html
  > public
  > PUBLIC
```

```
1  <?= $this->extend('layout/template'); ?>
2  <?= $this->section('content'); ?>
3  <div class="container">
4    <div class="col">
5      <h1 class="mt-2">Daftar Buku</h1>
6      <?php if (session()->getFlashdata('pesan')) : ?>
7        <div class="alert alert-success" role="alert">
8          <?= session()->getFlashdata('pesan') ?>
9        </div>
10       <?php endif; ?>
11      <a href="/buku/tambah" class="btn btn-primary">Tambah Data Buku</a>
12      <div class="row">
13        <table class="table mt-3">
14          <thead>
15            <tr>
16              <th scope="col">No.</th>
17              <th scope="col">Sampul</th>
18              <th scope="col">Judul</th>
19              <th scope="col">Aksi</th>
20            </tr>
21          </thead>
22          <tbody>
23            <?php
24              foreach ($buku as $b) :
25                <tr>
26                  <?php
27                    <th scope="row"><?= $i++ ; ?></th>
28                    <td></td>
29                    <td><?= $b['judul'] ; ?></td>
30                    <td><a href="/buku/<?= $b['id_buku'] ; ?>" class="btn btn-success">Detail</a></td>
31                </tr>
32              endforeach;
33            </tbody>
34          </table>
35      </div>
36    </div>
37  </div>
38</div>
```

Setelah ditambahkan Flashdata maka hasilnya akan seperti ini jika data berhasil ditambahkan:

The screenshot shows a web application interface. At the top, there is a navigation bar with links: Pustaka Booking, Beranda, Daftar Buku, and Daftar Anggota. Below the navigation bar, the title "Daftar Buku" is displayed. A green success message box contains the text "Data Berhasil Ditambahkan." Below this message is a blue button labeled "Tambah Data Buku". The main content area displays a table with three rows of book data. The columns are labeled "No.", "Sampul", "Judul", and "Aksi". Each row contains a small thumbnail image of the book cover under "Sampul", the book title under "Judul", and a green "Detail" button under "Aksi".

| No. | Sampul | Judul | Aksi |
|-----|--------|-----------------|-------------------------|
| 1 | | PHP | <button>Detail</button> |
| 2 | | Pemrograman Web | <button>Detail</button> |
| 3 | | Desain Web | <button>Detail</button> |

Tambahkan method hapus di dalam Controller Buku.

The screenshot shows a code editor with several files open in tabs: index.php, detail.php, tambah.php, and Buku.php. The Buku.php file is the active tab and contains PHP code. A specific section of the code is highlighted with a blue rounded rectangle, indicating the addition of a new method. The highlighted code is:

```
public function hapus($idbuku)
{
    $this->BukuModel->delete($idbuku);

    session()->setflashdata('pesan', 'Data Berhasil Dihapus.');
    return redirect()->to('/buku');
}
```

Tambahkan routes delete di routes.php nya

```
/*
// We get a performance increase by specifying the default
// route since we don't have to scan directories.
$routes->get('/', 'Pages::index');

$routes->get('/buku/tambah', 'Buku::tambah');
$routes->delete('/buku/(:num)', 'Buku::hapus/$1');
$routes->get('/buku/(:any)', 'Buku::detail/$1');
```

Ubah koding href pada aksi hapus di laman detail.phpnya menjadi seperti ini:

```
<div class="col-md-8">
<div class="card-body">
    <h5 class="card-title"><?= $buku['judul']; ?></h5>
    <p class="card-text"><b>Pengarang:</b> <?= $buku['pengarang']; ?></b></p>
    <p class="card-text">Penerbit:<?= $buku['penerbit']; ?><small></p>
    <p class="card-text">tahun Terbit:<?= $buku['tahun_terbit']; ?><small></p>
    <a href="" class="btn btn-warning">Ubah</a>

    <form action="/buku/<?= $buku['id_buku']; ?>" method="post" class="d-inline">
        <?= csrf_field(); ?>
        <input type="hidden" name="_method" value="DELETE">
        <button type="submit" class="btn btn-danger" onclick="return confirm('Yakin Hapus Data ini?')">Hapus</button>
    </form>

    <br><br>
    <a href="/buku">Kembali Ke Daftar Buku</a>
</div>
```

Maka hasilnya akan seperti ini:

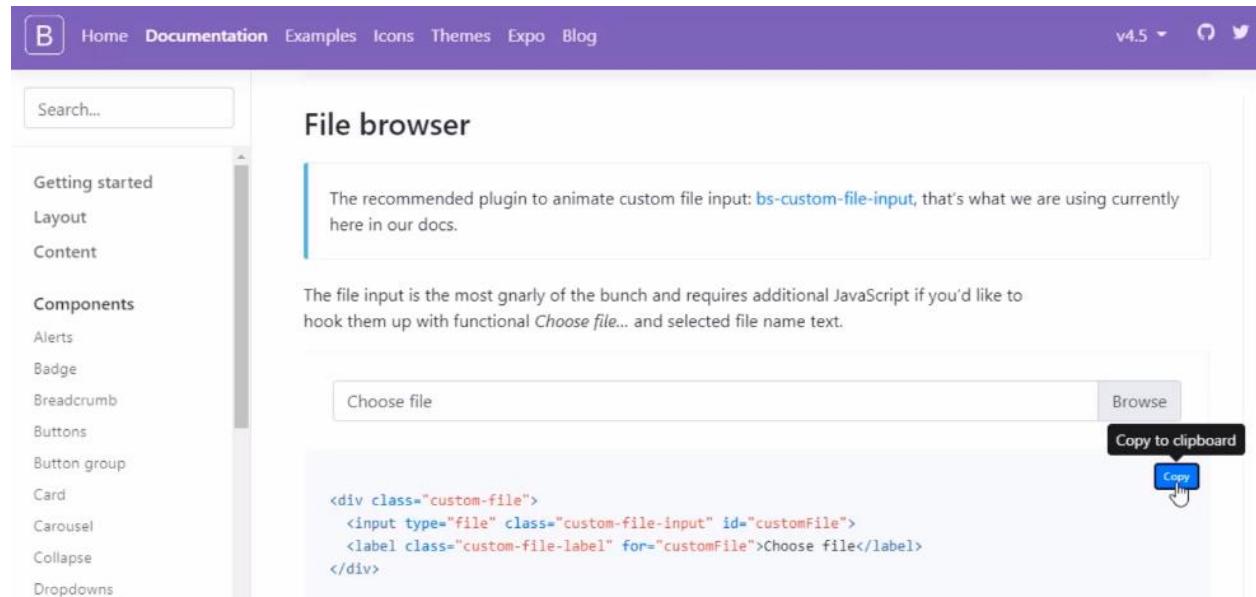


Dan ketika klik OK maka laman buku akan menampilkan pesan data berhasil dihapus.

Daftar Buku

| Data Berhasil Dihapus. | | | |
|------------------------|---|-----------------|-------------------------|
| No. | Sampul | Judul | Aksi |
| 1 |  | PHP | <button>Detail</button> |
| 2 |  | Pemrograman Web | <button>Detail</button> |
| 3 |  | Desain Web | <button>Detail</button> |

Tahapan selanjutnya beralih ke fungsi upload. Untuk fungsi upload coding yang akan dirubah-rubah ada pada laman tambah.php. Dan ada beberapa komponen yang harus ditambahkan yakni file browser. Untuk itu diperlukan css dari template bootstrap kembali. Silakan buka kembali laman dari bootstrap. Kemudian cari file browser.



The screenshot shows the Bootstrap documentation page for the 'File browser' component. The top navigation bar includes links for Home, Documentation, Examples, Icons, Themes, Expo, and Blog, along with a version dropdown set to v4.5 and social media icons for GitHub and Twitter. On the left, a sidebar lists categories like Getting started, Components, and various UI components. The main content area features a heading 'File browser' and a paragraph explaining the recommended plugin for animating custom file inputs. Below this is a code snippet showing the HTML structure for a file input field, which includes a 'Choose file' button, a 'Browse' button, and a 'Copy to clipboard' button with a 'Copy' link underneath it. A cursor arrow points to the 'Copy' link.

Tambahkan script berikut di form action tambah.phpnya, agar form tambah dapat membaca format file yang diupload.

```
<?php
    $validation = Validation::create();
    $validation->rules([
        'judul' => 'required|alpha|min:3|max:100',
        'pengarang' => 'required|alpha|min:3|max:100',
        'penerbit' => 'required|alpha|min:3|max:100',
        'tahun_terbit' => 'required|integer|between:1900,2023',
        'sampul' => 'required|file|mimes:pdf,jpg,png|max:10000'
    ]);
    $validation->messages([
        'required' => 'Field :attribute is required',
        'alpha' => 'Field :attribute must be an alpha value',
        'min' => 'Field :attribute must be at least :min characters long',
        'max' => 'Field :attribute must be no more than :max characters long',
        'integer' => 'Field :attribute must be an integer',
        'between' => 'Field :attribute must be between :min and :max',
        'mimes' => 'Field :attribute must be a file in :mimes format',
        'max' => 'Field :attribute must be no more than :max bytes'
    ]);
    if ($validation->fails()) {
        $errors = $validation->getMessages();
        $error_html = '';
        foreach ($errors as $field => $messages) {
            $error_html .= '

' . $messages[0] . '

' . PHP_EOL;
        }
        echo $error_html;
    } else {
        $data = [
            'judul' => $validation->getInput('judul'),
            'pengarang' => $validation->getInput('pengarang'),
            'penerbit' => $validation->getInput('penerbit'),
            'tahun_terbit' => $validation->getInput('tahun_terbit'),
            'sampul' => $validation->getInput('sampul')
        ];
        $book = Book::create($data);
        echo "Book added successfully!";
    }
}
```

Dan ubah koding pada komponen sampul seperti berikut.

```
36     <div class="form-group row">
37         <label for="inputSampul" class="col-sm-2 col-form-label">Sampul</label>
38         <div class="col-sm-4">
39             <div class="custom-file">
40                 <input type="file" class="custom-file-input" <?= ($validation->hasError('sampul')) ? 'is-invalid' : '' ; ?>
41                     <div class="invalid-feedback">
42                         <?= $validation->getError('sampul'); ?>
43                     </div>
44                 <label class="custom-file-label" for="customFile">Choose file</label>
45             </div>
46         </div>
47     <div class="form-group row">
48         <div class="col-sm-4">
49             <button type="submit" class="btn btn-primary">Tambah</button>
50         </div>
51     </div>
52 </div>
53 >
```

Dan hasilnya bisa dilihat di bawah ini:

The screenshot shows a web page titled "Form Tambah Buku". At the top, there is a navigation bar with links for "Pustaka Booking", "Beranda", "Daftar Buku", and "Daftar Anggota". Below the navigation, the main title "Form Tambah Buku" is displayed. There are five input fields: "Judul", "Pengarang", "Penerbit", "Tahun Terbit", and "Sampul". The "Sampul" field contains a file input field with the placeholder "Choose file" and a "Browse" button. At the bottom of the form is a blue "Tambah" button.

Ubah koding yang ada di dalam method simpan seperti dibawah ini dan hapus yang berkaitan script validation dari services, karena untuk berikutnya akan ditambahkan script session di dalam

BaseController sehingga tidak perlu lagi menambahkan validation dari services pada tiap method.

```
public function simpan()
{
    //validasi
    if (!$this->validate(
        [
            'judul' => [
                'rules' => 'required',
                'errors' => ['required' => '{field} harus diisi']
            ],
            'sampul' => [
                'rules' => 'uploaded[sampul]|max_size[sampul,10000]|is_image[sampul]|mime_in[sampul,image/jpg,image/jpeg,image/png]',
                'errors' => [
                    'uploaded'=>'Gambar Wajib Dipilih',
                    'max_size'=>'Ukuran gambar terlalu besar',
                    'is_image'=>'File Wajib Gambar',
                    'mime_in'=>'File Wajib Gambar'
                ]
            ]
        )
    ) {
        $validation = \Config\Services::validation();
        return redirect()->to('/buku/tambah')->withInput()->with('validation', $validation);
    }
}

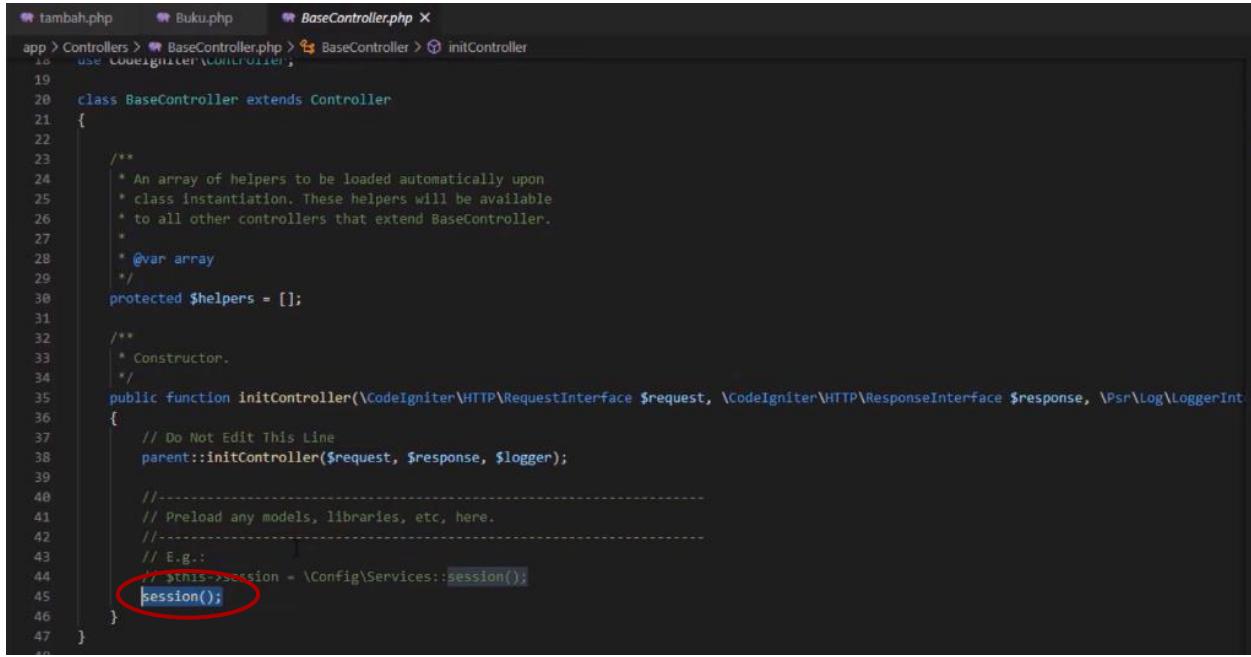
$filesampul = $this->request->getFile('sampul');
$filesampul->move('img');
$nmsampul = $filesampul->getName();

//simpan
$this->BukuModel->save(
    [
        'judul' => $this->request->getVar('judul'),
        'pengarang' => $this->request->getVar('pengarang'),
        'penerbit' => $this->request->getVar('penerbit'),
        'tahun_terbit' => $this->request->getVar('tahun'),
        'sampul' => $nmsampul
    ];
);
```

Pada array mime_in ubah pesannya seperti di bawah ini:

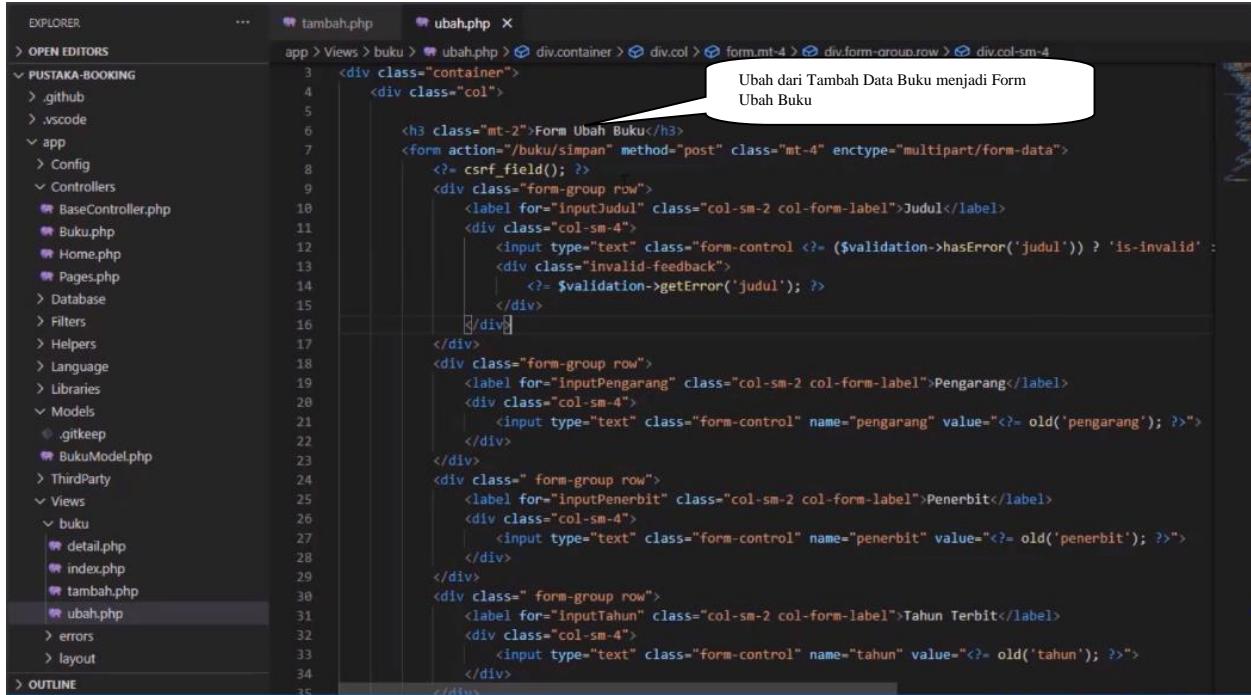
```
public function simpan()
{
    //validasi
    if (!$this->validate(
        [
            'judul' => [
                'rules' => 'required',
                'errors' => ['required' => '{field} harus diisi']
            ],
            'sampul' => [
                'rules' => 'uploaded[sampul]|max_size[sampul,1000]|is_image[sampul]|mime_in[sampul,image/jpg,image/jpeg,image/png]',
                'errors' => [
                    'uploaded'=>'Gambar Wajib Dipilih!',
                    'max_size'=>'Ukuran gambar terlalu besar',
                    'is_image'=>'File Wajib Gambar',
                    'mime_in'=>'Tipe File Gambar Tidak Sesuai'
                ]
            ]
        )
    ) {
}
```

Selanjutnya tambahkan koding session(); di BaseController.php.



```
app > Controllers > BaseController.php > BaseController > initController
16  use CodeIgniter\Controller;
17
18  class BaseController extends Controller
19  {
20
21      /**
22      * An array of helpers to be loaded automatically upon
23      * class instantiation. These helpers will be available
24      * to all other controllers that extend BaseController.
25      */
26
27      var array
28  }
29
30  protected $helpers = [];
31
32  /**
33  * Constructor.
34  */
35
36  public function initController(\CodeIgniter\HTTP\RequestInterface $request, \CodeIgniter\HTTP\ResponseInterface $response, \Psr\Log\LoggerInterface $logger)
37  {
38      // Do Not Edit This Line
39      parent::initController($request, $response, $logger);
40
41      //-----
42      // Preload any models, libraries, etc, here.
43      //-----
44      // E.g.:
45      // $this->session = \Config\Services::session();
46      session();
47  }
48 }
```

Selanjutnya tinggal form ubah. Untuk form ubah kodingannya tinggal copas (copy paste) dari form tambah.php.



EXPLORER

OPEN EDITORS

PUSTAKA-BOOKING

- .github
- .vscode
- app
- Config
- Controllers
- BaseController.php
- Buku.php
- Home.php
- Pages.php
- Database
- Filters
- Helpers
- Language
- Libraries
- Models
- gitkeep
- BukuModel.php
- ThirdParty
- Views
- buku
- detail.php
- index.php
- tambah.php
- ubah.php
- errors
- layout

tambah.php ubah.php

```
3  <div class="container">
4      <div class="col">
5
6          <h3 class="mt-2">Form Ubah Buku</h3>
7          <form action="/buku/simpan" method="post" class="mt-4" enctype="multipart/form-data">
8              <?= csrf_field(); ?>
9              <div class="form-group row">
10                 <label for="inputJudul" class="col-sm-2 col-form-label">Judul</label>
11                 <div class="col-sm-4">
12                     <input type="text" class="form-control" <?= ($validation->hasError('judul')) ? 'is-invalid' : 'is-valid' ?>
13                     <?= $validation->getError('judul'); ?>
14                 </div>
15             </div>
16             <div class="form-group row">
17                 <label for="inputPengarang" class="col-sm-2 col-form-label">Pengarang</label>
18                 <div class="col-sm-4">
19                     <input type="text" class="form-control" name="pengarang" value="<?= old('pengarang'); ?>">
20                 </div>
21             </div>
22             <div class="form-group row">
23                 <label for="inputPenerbit" class="col-sm-2 col-form-label">Penerbit</label>
24                 <div class="col-sm-4">
25                     <input type="text" class="form-control" name="penerbit" value="<?= old('penerbit'); ?>">
26                 </div>
27             </div>
28             <div class="form-group row">
29                 <label for="inputTahun" class="col-sm-2 col-form-label">Tahun Terbit</label>
30                 <div class="col-sm-4">
31                     <input type="text" class="form-control" name="tahun" value="<?= old('tahun'); ?>">
32                 </div>
33             </div>
34         </div>
```

Rubah pada tiap valuenya dari Judul hingga value sampul. Untuk lebih jelasnya bisa lihat gambar berikut:

```
ass="mt-2">Form Ubah Buku</h3>
action="/buku/update" method="post" class="mt-4" enctype="multipart/form-data">
= csrf_field(); ?>
iv class="form-group row">
<label for="inputJudul" class="col-sm-2 col-form-label">Judul</label>
<div class="col-sm-4">
    <input type="text" class="form-control" <?= ($validation->hasError('judul')) ? 'is-invalid' : ''; ?>" name="judul" value=<?= (old('judul')) ?>
    <div class="invalid-feedback">
        <?= $validation->getError('judul'); ?>
    </div>
</div>
iv class="form-group row">
<label for="inputPengarang" class="col-sm-2 col-form-label">Pengarang</label>
<div class="col-sm-4">
    <input type="text" class="form-control" name="pengarang" value=<?= (old('pengarang')) ? old('pengarang') : $buku['pengarang'] ?>">
</div>
iv class="form-group row">
<label for="inputPenerbit" class="col-sm-2 col-form-label">Penerbit</label>
<div class="col-sm-4">
    <input type="text" class="form-control" name="penerbit" value=<?= old('penerbit'); ?>">
</div>
iv class="form-group row">
<label for="inputTahun" class="col-sm-2 col-form-label">Tahun Terbit</label>
<div class="col-sm-4">
```

Tambahkan routes untuk method ubah di Routes.php-nya.

```
// We get a performance increase by specifying the default
// route since we don't have to scan directories.
$routes->get('/', 'Pages::index');

$routes->get('/buku/tambah', 'Buku::tambah');
$routes->get('/buku/ubah', 'Buku::ubah');
$routes->delete('/buku/(:num)', 'Buku::hapus/$1');
$routes->get('/buku/(:any)', 'Buku::detail/$1');
```

Tambahkan method ubah di dalam Controller Buku.

```
public function ubah($idbuku)
{
    $data = [
        'title' => 'Form Ubah Data Buku',
        'validation' => \Config\Services::validation(),
        'buku' => $this->BukuModel->getBuku($idbuku)
    ];

    return view('buku/ubah', $data);
}
```

Rubah form action pada ubah.php-nya.

```
ass="mt-2">Form Ubah Buku</h3>
action="/buku/update/<?= $buku['id_buku']; ?>" method="post" class="mt-4" enctype="multipart/form-data">
= csrf_field(); ?>


<label for="inputJudul" class="col-sm-2 col-form-label">Judul</label>
<div class="col-sm-4">
    <input type="text" class="form-control <?= ($validation->hasError('judul')) ? 'is-invalid' : ''; ?>" name="judul">
    <div class="invalid-feedback">
        <?= $validation->getError('judul'); ?>
    </div>
</div>


```

Dan rubah juga href pada aksi ubah di detail.php

```
<div class="col-md-8">
<div class="card-body">
    <h5 class="card-title"><?= $buku['judul']; ?></h5>
    <p class="card-text"><b>Pengarang: <?= $buku['pengarang']; ?></b></p>
    <p class="card-text">Penerbit:<?= $buku['penerbit']; ?></small></p>
    <p class="card-text">tahun Terbit:<?= $buku['tahun_terbit']; ?></small></p>
    <a href="/buku/ubah/<?= $buku['id_buku']; ?>" class="btn btn-warning">Ubah</a>

    <form action="/buku/<?= $buku['id_buku']; ?>" method="post" class="d-inline">
        <?= csrf_field(); ?>
        <input type="hidden" name="_method" value="DELETE">
        <button type="submit" class="btn btn-danger" onclick="return confirm('Yakin Hapus?')>Hapus</button>
    </form>
</div>
```

Untuk method update perintahnya sama seperti perintah method simpan hanya ada sedikit yang dirubah saja.

```
public function update($idbuku)
{
    //validasi
    if (!$this->validate(
    [
        'judul' => [
            'rules' => 'required',
            'errors' => ['required' => '{field} harus diisi']
        ],
        'sampul' => [
            'rules' => 'uploaded[sampul]|max_size[sampul,1000]|is_image[sampul]|mime_in[sampul,image/jpg,jpeg,png]',
            'errors' => [
                'uploaded' => 'Gambar Wajib Dipilih!',
                'max_size' => 'Ukuran gambar terlalu besar!',
                'is_image' => 'File Wajib Gambar!',
                'mime_in' => 'Tipe File Gambar Tidak Sesuai!'
            ]
        ]
    ])
}
```

```

//simpan
$this->BukuModel->save(
    [
        'id_buku' => $idbuku,
        'judul' => $this->request->getVar('judul'),
        'pengarang' => $this->request->getVar('pengarang'),
        'penerbit' => $this->request->getVar('penerbit'),
        'tahun_terbit' => $this->request->getVar('tahun'),
        'sampul' => $nmsampul
    ]
);

session()->setflashdata('pesan', 'Data Berhasil Diubah.');
return redirect()->to('/buku');
}

```

Tambahkan komponen sampullama dengan type hidden seperti gambar di bawah ini:

```

<h3 class="mt-2">Form Ubah Buku</h3>
<form action="/buku/update/<?= $buku['id_buku']; ?>" method="post" class="mt-4" enctype="multipart/form-data">
    <?= csrf_field(); ?>
    <input type="hidden" name="sampullama" value="<?= $buku['sampul']; ?>">
    <div class="form-group row">
        <label for="inputJudul" class="col-sm-2 col-form-label">Judul</label>
        <div class="col-sm-4">
            <input type="text" class="form-control" <?= ($validation->hasError('judul')) ? 'is-invalid' : 'is-valid' ; ?>
            <div class="invalid-feedback">
                <?= $validation->getError('judul'); ?>
            </div>
        </div>
    </div>

```

Pada bagian method update sesuaikan kodingnya seperti di bawah ini. Untuk letaknya sendiri setelah redirect.

```

    return redirect()->to('/buku/ubah')->withInput();
}

$filesampul = $this->request->getFile('sampul');
//cek gambar kondisi sampul lama
if ($filesampul->getError() == 4) {
    $nmsampul = $this->request->getVar('sampullama');
} else {
    $nmsampul = $filesampul->getName();
    $filesampul->move('img', $nmsampul);
}

```

Setelah itu bisa di uji coba pada form ubah, dirubah nama penerbitnya dari andre pratama menjadi andre.

Sebelum dirubah:

Pustaka Booking Beranda Daftar Buku Daftar Anggota

Form Ubah Buku

| | |
|-------------------------------------|--|
| Judul | <input type="text" value="css"/> |
| Pengarang | <input type="text" value="Andre Pratama"/> |
| Penerbit | <input type="text" value="Andre Pratama"/> |
| Tahun Terbit | <input type="text" value="2019"/> |
| Sampul | <input type="text" value="css.jpg"/> <input type="button" value="Browse"/> |
| <input type="button" value="Ubah"/> | |

Saat dirubah:

Pustaka Booking Beranda Daftar Buku Daftar Anggota

Form Ubah Buku

| | |
|-------------------------------------|--|
| Judul | <input type="text" value="CSS Lanjutan"/> |
| Pengarang | <input type="text" value="Andre Pratama"/> |
| Penerbit | <input type="text" value="Andre"/> |
| Tahun Terbit | <input type="text" value="2019"/> |
| Sampul | <input type="text" value="css.jpg"/> <input type="button" value="Browse"/> |
| <input type="button" value="Ubah"/> | |

Setelah dirubah:

Daftar Buku

Data Berhasil Diubah.

Tambah Data Buku

| No. | Sampul | Judul | Aksi |
|-----|---|-----------------|-------------------------|
| 1 |  | PHP | <button>Detail</button> |
| 2 |  | Pemrograman Web | <button>Detail</button> |
| 3 |  | Desain Web2 | <button>Detail</button> |

PERTEMUAN 7

REVIEW MATERI

Untuk mengetahui sejauh mana mahasiswa memahamai materi yang sudah disampaikan dari pertemuan 1 sampai pertemuan 6, maka dipertemuan 7 ini dibuatkanlah sebuah studi kasus yang harus diselesaikan. Dengan konsep MVC yang sudah dijelaskan pada pertemuan 1 hingga 6 dari instalasi sampai dengan pembuatan program yang lengkap dari insert, save, edit, update dan delete dan juga pembuatan view program. Berikut studi kasus yang harus diselesaikan:

1. Toko XXX Bakery membutuhkan program sederhana untuk menyimpan data bahan baku rotinya. Dengan atribut: id_bahanbaku, nama_bahanbaku, harga_beli, tgl_masuk, stok, dan gambar.
2. Buatlah program web dengan menerapkan konsep MVC dengan kasus pada no.1. Dimana di dalam program tersebut bersifat back end. Buatlah model, view dan controllernya. Untuk konsep tampilan dan pengolahan datanya sama dengan yang sudah dijelaskan di pertemuan 3 sampai dengan pertemuan 6. Dari read data, insert data, edit data, update data, dan delete data.
3. Untuk view dari daftar baku hanya ditampilkan gambar bahan baku dan nama bahan baku saja. Dan ditambahkan tombol detail untuk menampilkan data bahan baku tersebut secara lengkapnya.
4. Dari laman detail ditambahkan hyperlink untuk dapat kembali ke datar bahan baku.

PERTEMUAN 8
UJIAN TENGAH SEMESTER

PERTEMUAN 9

PAGINATION DAN SEARCHING

Deskripsi Pembelajaran

Pertemuan 9 ini membahas tentang cara membuat pencarian dari data yang ditampilkan dan penghalamanan. Jadi untuk pertemuan 9 ini masih melanjutkan sedikit dari pertemuan 6, tapi tidak sebagai inti tapi lebih ke mendukung pengolahan data yang ada. Kegunaan search box ini adalah untuk memudahkan user dalam mencari data yang banyak sehingga user hanya mengetikkan kata kuncinya saja maka data yang dicari dapat langsung ditemukan atau ditampilkan. Dan kegunaan pagination sendiri adalah untuk memudahkan user dalam mengetahui halaman berapa yang saat ini mereka sedang kunjungi dan untuk memudahkan user dalam menemukan data berdasarkan halaman yang mereka cari.

Tujuan Pembelajaran

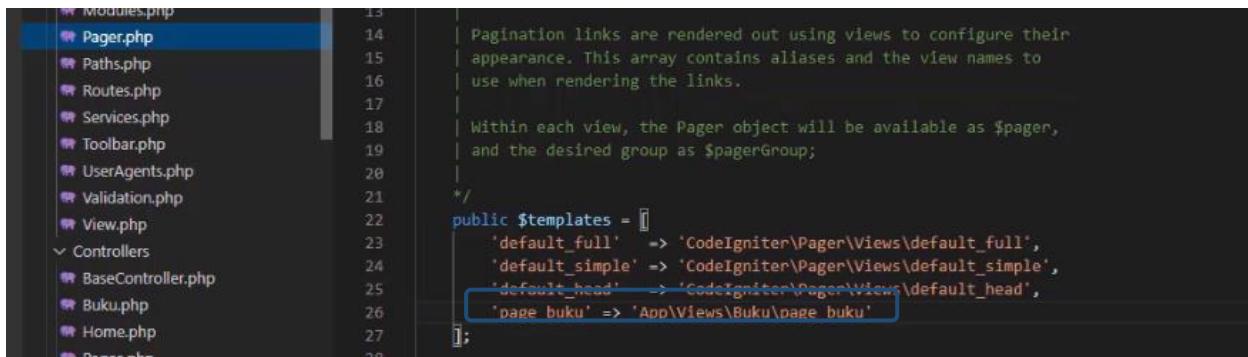
Pada bab ini mahasiswa diharapkan dapat memahami konsep pagination dan pencarian data menggunakan search box serta mampu mengimplementasikannya dengan baik.

9.1. Pagination

Silahkan rubah koding yang di method index pada Controller buku seperti gambar di bawah ini:

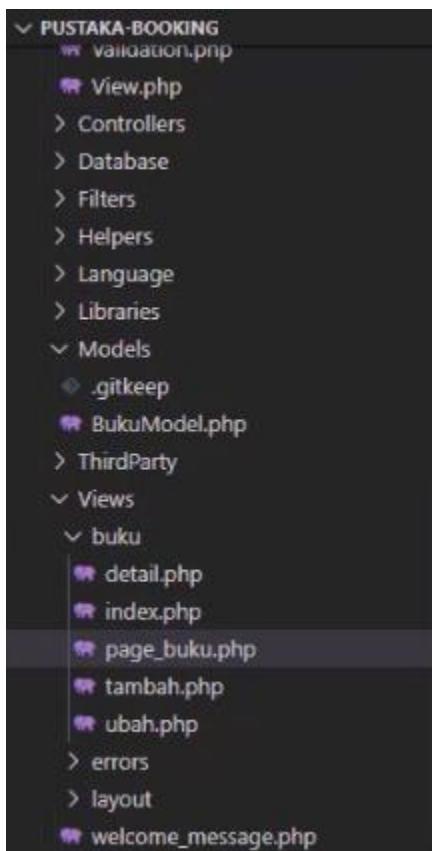
```
public function index()
{
    $current = $this->request->getVar('page_buku') ? $this->request->getVar('page_buku') : 1;
    $data = [
        'title' => 'Daftar Buku',
        'buku' => $this->BukuModel->paginate(2, 'buku'),
        'pager' => $this->BukuModel->pager,
        'current' => $current
    ];
    return view('buku/index', $data);
}
```

Selanjutnya masuk ke dalam folder app -> config -> Pager.php. Kemudian tambahkan scriptnya seperti gambar di bawah ini:



```
13
14 | Pagination links are rendered out using views to configure their
15 | appearance. This array contains aliases and the view names to
16 | use when rendering the links.
17 |
18 | Within each view, the Pager object will be available as $pager,
19 | and the desired group as $pagerGroup;
20 |
21 */
22 public $templates = [
23     'default_full' => 'CodeIgniter\Pager\Views\default_full',
24     'default_simple' => 'CodeIgniter\Pager\Views\default_simple',
25     'default_head' => 'CodeIgniter\Pager\Views\default_head',
26     'page_buku' => 'App\Views\Buku\page_buku'
27 ];
28
```

Buatlah file baru dengan nama page_buku di dalam folder Views -> buku.



Untuk script daripada page_buku dapat disalin atau mencontoh di library web resminya codeigniter sendiri. Untuk itu silahkan buka laman resminya codeigniter lalu pilih read the manual.

Not secure | codeigniter.com

CodeIgniter

Home Download Documentation Community Contribute

CodeIgniter Rocks

CodeIgniter is a powerful PHP framework with a very small footprint, built for developers who need a simple and elegant toolkit to create full-featured web applications.

3,158 stars 1,217 forks 23.3K followers

Download Read the Manual View the Forums On Github

Ketikkan pada kolom pencarian pagination lalu klik pada laman sebelah kanannya pagination.

CodeIgniter

Docs » Search

Search Results

Search finished, found 11 page(s) matching the search query.

Pagination

CodeIgniter4 User Guide

Filesystem Helper

Library Reference

Tahap selanjutnya silahkan cari subject dengan nama creating view lalu salin.

```
public $templates = [
    'default_full' => 'CodeIgniter\Pager\Views\default_full',
    'default_simple' => 'CodeIgniter\Pager\Views\default_simple',
    'front_full' => 'App\Views\Pagers\foundation_full',
];

Once configured, you can specify it as the last parameter in the links(), simpleLinks(), and makeLinks() methods:

<?= $pager->links('group1', 'front_full') ?>
<?= $pager->simpleLinks('group2', 'front_full') ?>
<?= $pager->makeLinks($page, $perPage, $total, 'front_full') ?>
```

Creating the View

When you create a new view, you only need to create the code that is needed for **creating** the **pagination** links themselves. You should not create unnecessary wrapping divs since it might be used in multiple places and you only limit their usefulness. It is easiest to demonstrate **creating** a new view by showing you the existing default_full template:

```
<?php $pager->setSurroundCount(2) ?>

<nav aria-label="Page navigation">
    <ul class="list-group">
        <?php if ($pager->hasPrevious()) : ?>
            <li>
                <a href=<?= $pager->getFirst() ?>" aria-label=<?= lang('Pager.first') ?>">
                    <span aria-hidden="true"><?= lang('Pager.first') ?></span>
                </a>
            </li>
        <?php endif ?>

        <?php foreach ($pager->links() as $link) : ?>
            <li <?= $link['active'] ? 'class="active"' : '' ?>>
                <a href=<?= $link['uri'] ?>">
                    <?= $link['title'] ?>
                </a>
            </li>
        <?php endforeach ?>

        <?php if ($pager->hasNext()) : ?>
            <li>
                <a href=<?= $pager->getNext() ?>" aria-label=<?= lang('Pager.next') ?>">
                    <span aria-hidden="true"><?= lang('Pager.next') ?></span>
                </a>
            </li>
            <li>
                <a href=<?= $pager->getLast() ?>" aria-label=<?= lang('Pager.last') ?>">
                    <span aria-hidden="true"><?= lang('Pager.last') ?></span>
                </a>
            </li>
        <?php endif ?>
    </ul>
</nav>
```

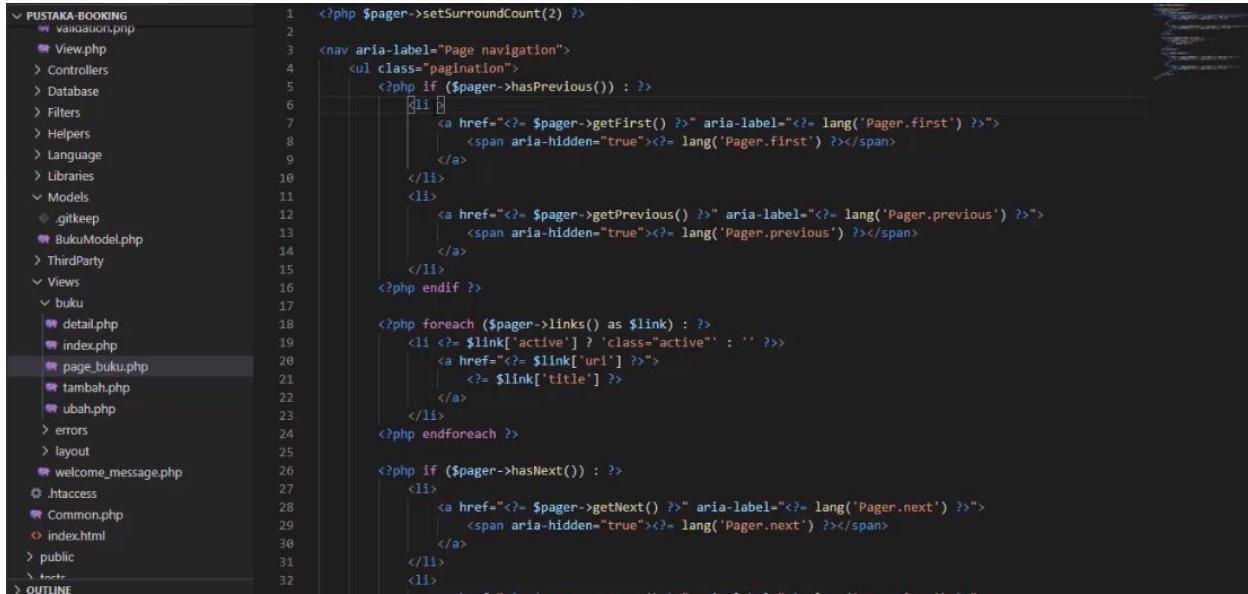
Block semua codingnya lalu salin.

```
</a>
</li>
<li>
    <a href=<?= $pager->getPrevious() ?>" aria-label=<?= lang('Pager.previous') ?>">
        <span aria-hidden="true"><?= lang('Pager.previous') ?></span>
    </a>
</li>
<?php endif ?>

<?php foreach ($pager->links() as $link) : ?>
    <li <?= $link['active'] ? 'class="active"' : '' ?>>
        <a href=<?= $link['uri'] ?>">
            <?= $link['title'] ?>
        </a>
    </li>
<?php endforeach ?>

<?php if ($pager->hasNext()) : ?>
    <li>
        <a href=<?= $pager->getNext() ?>" aria-label=<?= lang('Pager.next') ?>">
            <span aria-hidden="true"><?= lang('Pager.next') ?></span>
        </a>
    </li>
    <li>
        <a href=<?= $pager->getLast() ?>" aria-label=<?= lang('Pager.last') ?>">
            <span aria-hidden="true"><?= lang('Pager.last') ?></span>
        </a>
    </li>
<?php endif ?>
</ul>
</nav>
```

Kemudian pasti di laman page_buku.php.



```
<?php $pager->setSurroundCount(2) ?>

<nav aria-label="Page navigation">
    <ul class="pagination">
        <?php if ($pager->hasPrevious()) : ?>
            <li>
                <a href=<?= $pager->getFirst() ?>" aria-label=<?= lang('Pager.first') ?>">
                    <span aria-hidden="true"><?= lang('Pager.first') ?></span>
                </a>
            </li>
            <li>
                <a href=<?= $pager->getPrevious() ?>" aria-label=<?= lang('Pager.previous') ?>">
                    <span aria-hidden="true"><?= lang('Pager.previous') ?></span>
                </a>
            </li>
        <?php endif ?>
        <?php foreach ($pager->links() as $link) : ?>
            <li ><?= $link['active'] ? 'class="active"' : '' ?>
                <a href=<?= $link['uri'] ?>">
                    <?= $link['title'] ?>
                </a>
            </li>
        <?php endforeach ?>
        <?php if ($pager->hasNext()) : ?>
            <li>
                <a href=<?= $pager->getNext() ?>" aria-label=<?= lang('Pager.next') ?>">
                    <span aria-hidden="true"><?= lang('Pager.next') ?></span>
                </a>
            </li>
        <?php endif ?>
    </ul>
</nav>
```

Edit beberapa koding yang sudah ditandai dengan frame biru.



```
<?php $pager->setSurroundCount(2) ?>

<nav aria-label="Page navigation">
    <ul class="pagination">
        <?php if ($pager->hasPrevious()) : ?>
            <li class="page-item">
                <a class="page-link" href=<?= $pager->getFirst() ?>" aria-label=<?= lang('Pager.first') ?>">
                    <span aria-hidden="true"><?= lang('Pager.first') ?></span>
                </a>
            </li>
            <li class="page-item">
                <a class="page-link" href=<?= $pager->getPrevious() ?>" aria-label=<?= lang('Pager.previous') ?>">
                    <span aria-hidden="true"><?= lang('Pager.previous') ?></span>
                </a>
            </li>
        <?php endif ?>
        <?php foreach ($pager->links() as $link) : ?>
            <li class="page-item" ><?= $link['active'] ? 'class="active"' : '' ?>
                <a class="page-link" href=<?= $link['uri'] ?>">
                    <?= $link['title'] ?>
                </a>
            </li>
        <?php endforeach ?>
    </ul>
</nav>
```

```

<?php if ($pager->hasNext()) : ?>
    <li class="page-item">
        <a class="page-link" href="= $pager-&gt;getNext() ?" aria-label="= lang('Pager.next') ?">
            <span aria-hidden="true"><?= lang('Pager.next') ?></span>
        </a>
    </li>
    <li class="page-item">
        <a class="page-link" href="= $pager-&gt;getLast() ?" aria-label="= lang('Pager.last') ?">
            <span aria-hidden="true"><?= lang('Pager.last') ?></span>
        </a>
    </li>
<?php endif ?>
</ul>
</nav>


```

File structure:

```

└─ PUSTAKA-BOOKING
    └─ view.php
        └─ Controllers
            └─ BaseController.php
            └─ Buku.php
            └─ Home.php
            └─ Pages.php
        └─ Database
        └─ Filters
        └─ Helpers
        └─ Language
        └─ Libraries
        └─ Models
            └─ .gitkeep
            └─ BukuModel.php
        └─ ThirdParty
        └─ Views
            └─ buku
                └─ detail.php
                └─ index.php
                └─ page_buku.php
                └─ tambah.php
                └─ ubah.php
            └─ errors

```

```

<tr>
    <th scope="col">No.</th>
    <th scope="col">Sampul</th>
    <th scope="col">Judul</th>
    <th scope="col">Aksi</th>
</tr>
</thead>
<tbody>
<?php
    $i = 1 + (2 * ($current - 1));
    foreach ($buku as $b) :
    ?>
        <tr>
            <th scope="row"><?= $i++ ; ?></th>
            <td></td>
            <td><?= $b['judul'] ; ?></td>
            <td><a href="/buku/<?= $b['id_buku'] ; ?>" class="btn btn-success">Detail</a></td>
        </tr>
    <?php endforeach; ?>
</tbody>
</table>
<?= $pager->links('buku', 'page_buku'); ?>

```

Hasilnya sebagai berikut:

Pustaka Booking Beranda Daftar Buku Daftar Anggota

Daftar Buku

[Tambah Data Buku](#)

| No. | Sampul | Judul | Aksi |
|-----|--------|-----------------|------------------------|
| 1 | | PHP | Detail |
| 2 | | Pemrograman Web | Detail |

Page navigation: 1 | 2

9.2. Searching

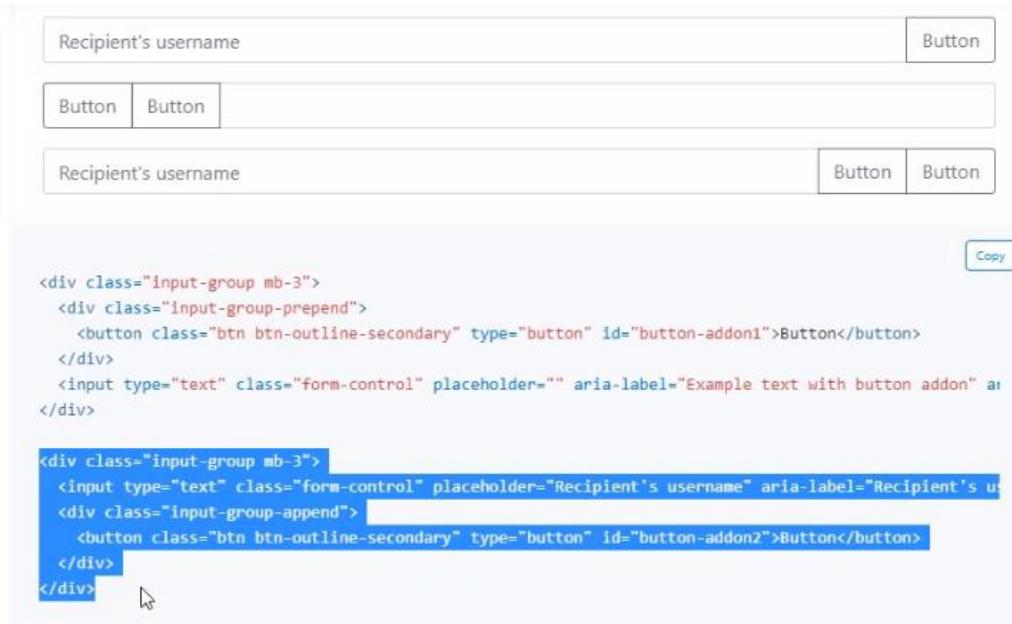
Silahkan masuk ke laman resmi bootstrap, pilih documentation keumdian ketik di pencarian input. Pilih input group.

The screenshot shows the Bootstrap Documentation website's search results for the term "Input". The search bar at the top contains "Input". Below it, a sidebar on the left lists categories: Components (Input group / Input group, Input group / Multiple inputs, Input group / Custom forms > Custom file input, Forms / Range Inputs), Content (Code / User input), and Accessibility. The main content area features a large heading "Introduction" followed by a brief description of Bootstrap's responsive design. A sidebar advertisement for Adobe Creative Cloud is visible. In the center, there is a "Quick start" section with a "View on GitHub" button and a note about using jsDelivr. The overall layout is clean and modern, typical of the Bootstrap design system.

Pilih yang jenisnya button addons

The screenshot shows the Bootstrap Documentation page for "Input group". The main title is "Input group", described as "Easily extend form controls by adding text, buttons, or button groups on either side of textual inputs, custom selects, and custom file inputs.". To the right is a "View on GitHub" button. On the far right, a vertical sidebar lists various documentation topics: Basic example, Wrapping, Sizing, Checkboxes and radios, Multiple inputs, Multiple addons, **Button addons** (with a cursor icon pointing to it), Buttons with dropdowns, Segmented buttons, Custom forms, Custom select, Custom file input, and Accessibility. A sidebar advertisement for Adobe Creative Cloud is also present.

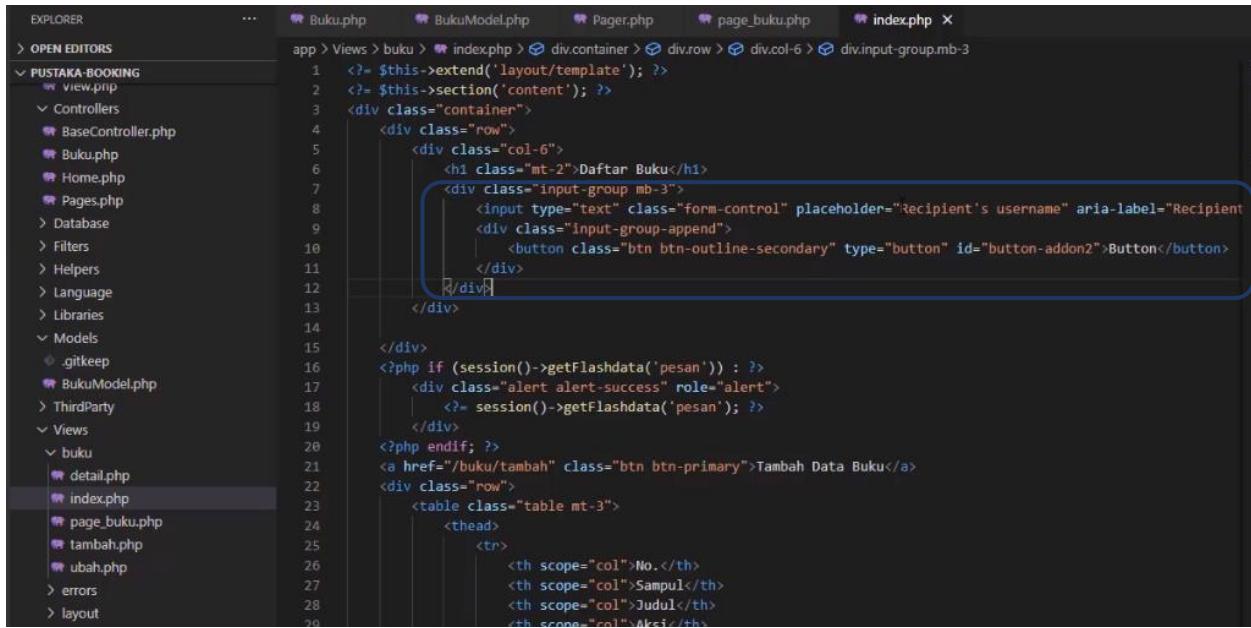
Pilih yang baris kedua, block dan copy.



```
<div class="input-group mb-3">
  <div class="input-group-prepend">
    <button class="btn btn-outline-secondary" type="button" id="button-addon1">Button</button>
  </div>
  <input type="text" class="form-control" placeholder="" aria-label="Example text with button addon" aria-describedby="button-addon1">
</div>

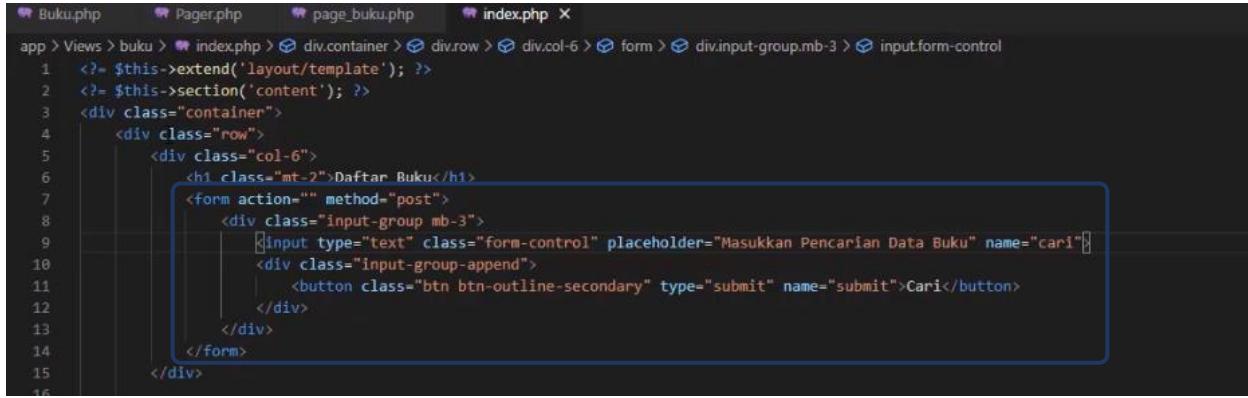
<div class="input-group mb-3">
  <input type="text" class="form-control" placeholder="Recipient's username" aria-label="Recipient's username" aria-describedby="button-addon2">
  <div class="input-group-append">
    <button class="btn btn-outline-secondary" type="button" id="button-addon2">Button</button>
  </div>
</div>
```

Paste di index.php



```
1  <?= $this->extend('layout/template'); ?>
2  <?= $this->section('content'); ?>
3  <div class="container">
4    <div class="row">
5      <div class="col-6">
6        <h1 class="mt-2">Daftar Buku</h1>
7        <div class="input-group mb-3">
8          <input type="text" class="form-control" placeholder="Recipient's username" aria-label="Recipient's username" aria-describedby="button-addon2">
9          <div class="input-group-append">
10            <button class="btn btn-outline-secondary" type="button" id="button-addon2">Button</button>
11          </div>
12        </div>
13      </div>
14    </div>
15    <?php if (session()->getFlashdata('pesan')) : ?>
16      <div class="alert alert-success" role="alert">
17        <?= session()->getFlashdata('pesan'); ?>
18      </div>
19    <?php endif; ?>
20    <a href="/buku/tambah" class="btn btn-primary">Tambah Data Buku</a>
21  <div class="row">
22    <table class="table mt-3">
23      <thead>
24        <tr>
25          <th scope="col">No.</th>
26          <th scope="col">Sampul</th>
27          <th scope="col">Judul</th>
28          <th scope="col">Aksi</th>
29        </tr>
```

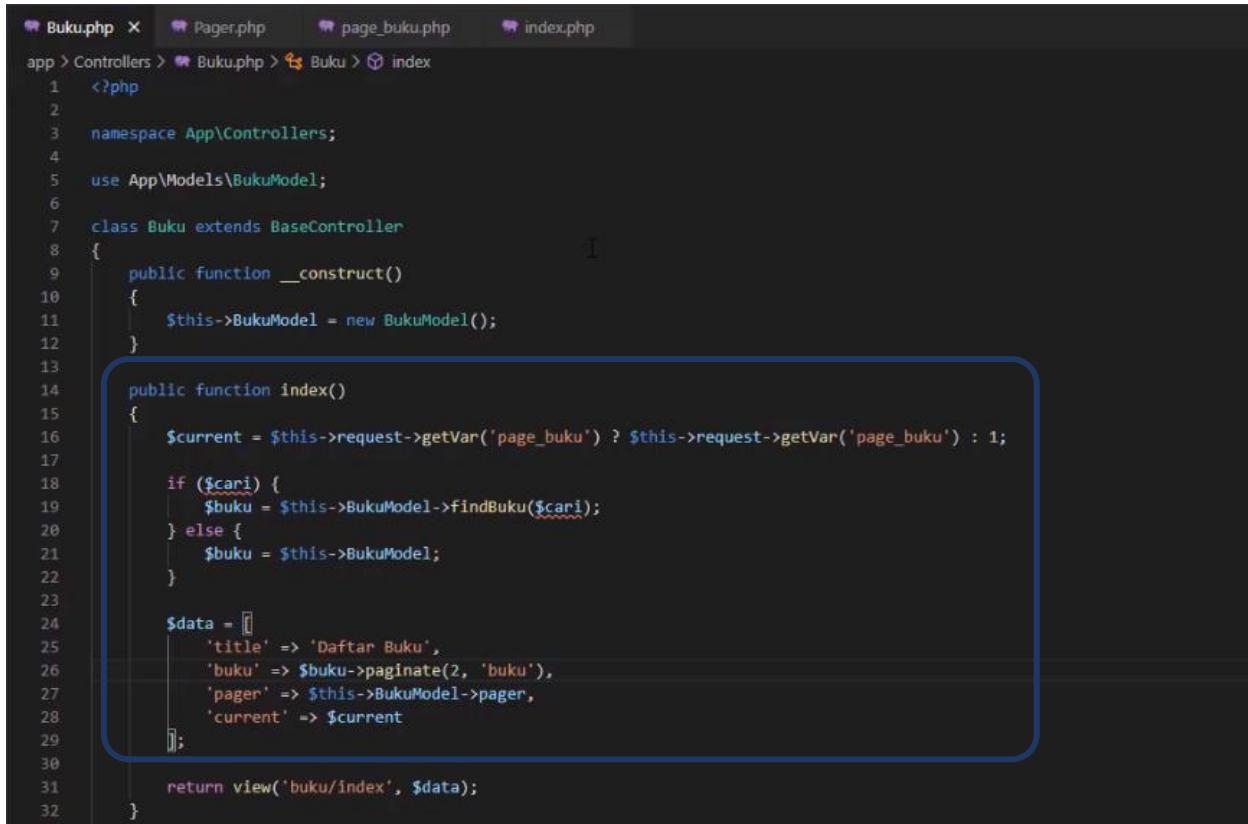
Ubah kodingnya seperti di bawah ini:



The screenshot shows a code editor with several tabs at the top: Buku.php, Pager.php, page_buku.php, and index.php. The index.php tab is active. The code is a PHP template with embedded HTML and Blade syntax. A specific section of the code is highlighted with a blue rounded rectangle:

```
app > Views > buku > index.php > <div>.container > <div>.row > <div>.col-6 > <form> > <div>.input-group.mb-3 > <input>.form-control
1  <?= $this->extend('layout/template'); ?>
2  <?= $this->section('content'); ?>
3  <div class="container">
4      <div class="row">
5          <div class="col-6">
6              <h1 class="mt-2">Daftar Buku</h1>
7              <form action="" method="post">
8                  <div class="input-group mb-3">
9                      <input type="text" class="form-control" placeholder="Masukkan Pencarian Data Buku" name="cari">
10                     <div class="input-group-append">
11                         <button class="btn btn-outline-secondary" type="submit" name="submit">Cari</button>
12                     </div>
13                 </div>
14             </div>
15         </div>
16     </div>
```

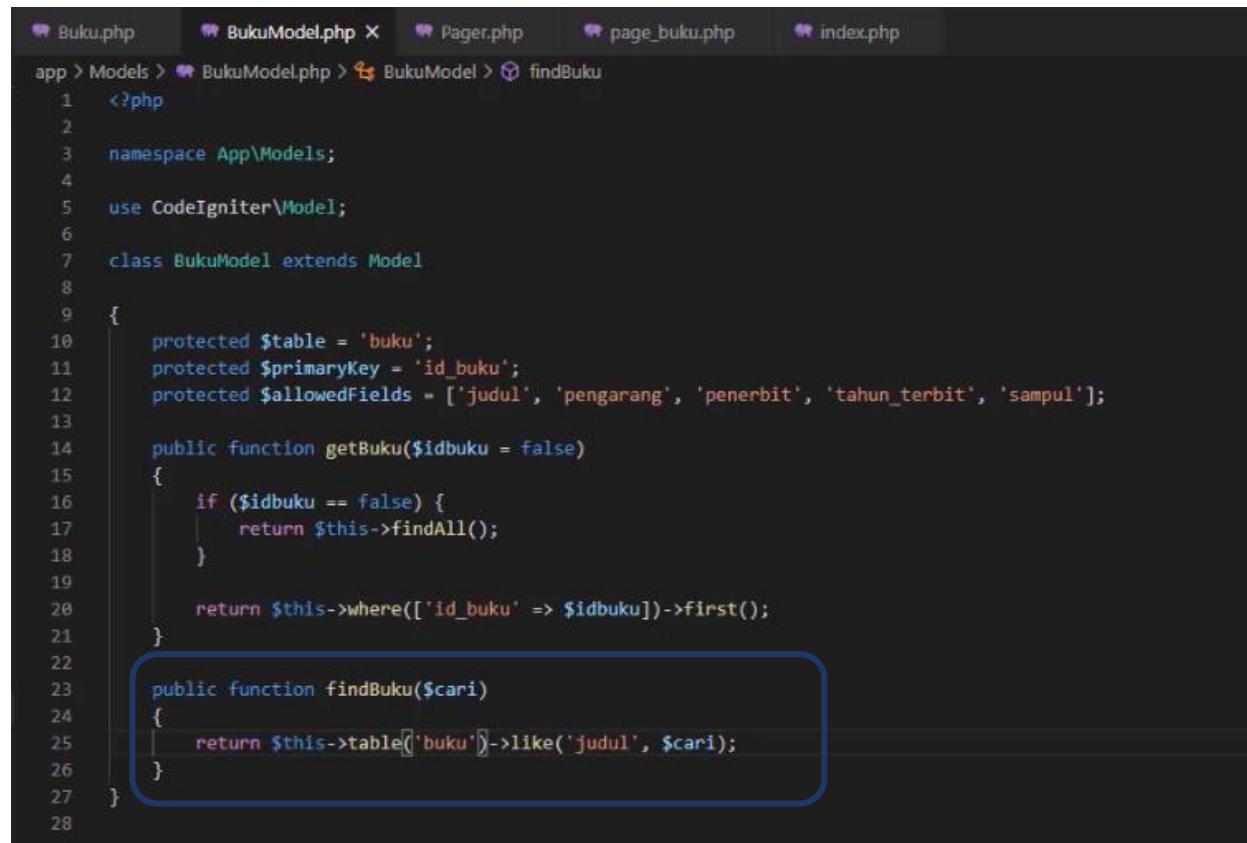
Lanjutkan dengan mengubah koding yang ada di dalam Controller Buku tepatnya di method index.



The screenshot shows a code editor with several tabs at the top: Buku.php, Pager.php, page_buku.php, and index.php. The Buku.php tab is active. The code is a PHP class definition for a controller. A specific section of the code is highlighted with a blue rounded rectangle:

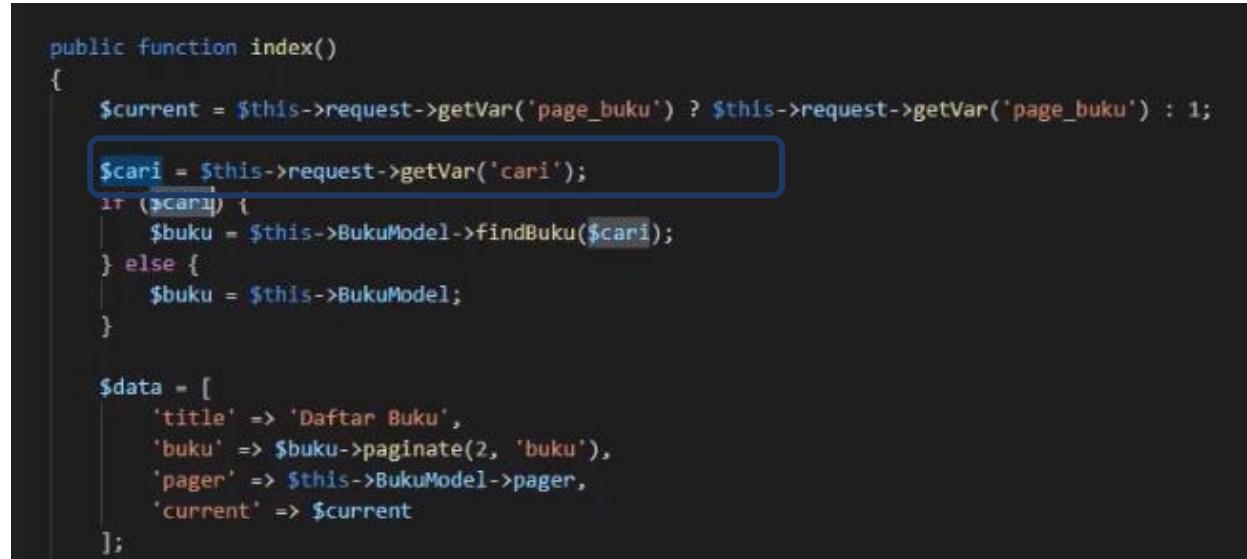
```
app > Controllers > Buku.php > Buku > index
1  <?php
2
3  namespace App\Controllers;
4
5  use App\Models\BukuModel;
6
7  class Buku extends BaseController
8  {
9      public function __construct()
10     {
11         $this->BukuModel = new BukuModel();
12     }
13
14     public function index()
15     {
16         $current = $this->request->getVar('page_buku') ? $this->request->getVar('page_buku') : 1;
17
18         if ($cari) {
19             $buku = $this->BukuModel->findBuku($cari);
20         } else {
21             $buku = $this->BukuModel;
22         }
23
24         $data = [
25             'title' => 'Daftar Buku',
26             'buku' => $buku->paginate(2, 'buku'),
27             'pager' => $this->BukuModel->pager,
28             'current' => $current
29         ];
30
31         return view('buku/index', $data);
32     }
}
```

Tambahkan method findBuku di dalam Model BukuModel.php. Disini dicontohkan pencarian berdasarkan judul.



```
  Buku.php      BukuModel.php X  Pager.php    page_buku.php  index.php
app > Models > BukuModel.php > BukuModel > findBuku
1  <?php
2
3  namespace App\Models;
4
5  use CodeIgniter\Model;
6
7  class BukuModel extends Model
8
9  {
10     protected $table = 'buku';
11     protected $primaryKey = 'id_buku';
12     protected $allowedFields = ['judul', 'pengarang', 'penerbit', 'tahun_terbit', 'sampul'];
13
14     public function getBuku($idbuku = false)
15     {
16         if ($idbuku == false) {
17             return $this->findAll();
18         }
19
20         return $this->where(['id_buku' => $idbuku])->first();
21     }
22
23     public function findBuku($cari)
24     {
25         return $this->table('buku')->like('judul', $cari);
26     }
27 }
28 }
```

Kemudian tambahkan script di Controller Buku tepatnya di method index.



```
public function index()
{
    $current = $this->request->getVar('page_buku') ? $this->request->getVar('page_buku') : 1;

    $cari = $this->request->getVar('cari');
    if ($cari) {
        $buku = $this->BukuModel->findBuku($cari);
    } else {
        $buku = $this->BukuModel;
    }

    $data = [
        'title' => 'Daftar Buku',
        'buku' => $buku->paginate(2, 'buku'),
        'pager' => $this->BukuModel->pager,
        'current' => $current
    ];
}
```

Hasilnya dapat dilihat pada gambar di bawah ini:

Pustaka Booking Beranda Daftar Buku Daftar Anggota

Daftar Buku

Cari

[Tambah Data Buku](#)

| No. | Sampul | Judul | Aksi |
|-----|--------|-----------------|------------------------|
| 1 | | PHP | Detail |
| 2 | | Pemrograman Web | Detail |

1

Misal saja pencarian berdasarkan judul: desain web. Maka setelah diklik cari akan menampilkan hasil seperti di bawah ini.

Saat Pencarian:

Pustaka Booking Beranda Daftar Buku Daftar Anggota

Daftar Buku

Cari

[Tambah Data Buku](#)

| No. | Sampul | Judul | Aksi |
|-----|--------|-----------------|------------------------|
| 1 | | PHP | Detail |
| 2 | | Pemrograman Web | Detail |

1 2

Hasil Pencarian:

Pustaka Booking Beranda Daftar Buku Daftar Anggota

Daftar Buku

Masukkan Pencarian Data Buku

Cari

Tambah Data Buku

| No. | Sampul | Judul | Aksi |
|-----|--------|-------------|-------------------------|
| 1 | | Desain Web2 | <button>Detail</button> |

1

Saat Pencarian:

Pustaka Booking Beranda Daftar Buku Daftar Anggota

Daftar Buku

CSS Lanjutan

Cari

Tambah Data Buku

| No. | Sampul | Judul | Aksi |
|-----|--------|-------------|-------------------------|
| 1 | | Desain Web2 | <button>Detail</button> |

1

Hasil Pencarian:

Pustaka Booking Beranda Daftar Buku Daftar Anggota

Daftar Buku

Masukkan Pencarian Data Buku

Cari

Tambah Data Buku

| No. | Sampul | Judul | Aksi |
|-----|--------|--------------|-------------------------|
| 1 | | CSS Lanjutan | <button>Detail</button> |

1

PERTEMUAN 10

INSTALASI LIBRARY FORM LOGIN CI4

Deskripsi Pembelajaran

Pada bab ini akan dibahas mengenai tentang bagaimana instalasi librabry form login yang dikembangkan oleh developer ci4-nya sendiri. Dan ada beberapa tahapan dalam instalasinya, salah satunya update composer.

Tujuan Pembelajaran

Pada perteuan ini diharapkan mahasiswa mampu memahami instalasi library form login dan mengetahui perbedaan pembuatan form login pada ci3 dan ci4. Jika di ci3 harus membuat layout form login dan registrasinya secara manual tetapi di ci4 layoutnya akan terbuat otomatis setelah instalasi dan konfigurasinya selesai dieksekusi.

10.1. Update Composer

Untuk update composer dilakukan di laman git bash dan membutuhka koneksi internet. Untuk itu silahkan masuk ke git bash. Pastikan git bashnya sudah masuk di disk C-> xampp -> htdocs -> pustaka-booking.

```
MINGW64:/c/xampp/htdocs/pustaka-booking
Thu Dec 3 19:35:10 2020] [::1]:56820 Accepted
Thu Dec 3 19:35:10 2020] [::1]:56821 Accepted
Thu Dec 3 19:35:10 2020] [::1]:56823 Closing
Thu Dec 3 19:35:10 2020] [::1]:56820 [200]: GET /index.php?debugbar
Thu Dec 3 19:35:10 2020] [::1]:56820 Closing
Thu Dec 3 19:35:10 2020] [::1]:56827 Accepted
Thu Dec 3 19:35:10 2020] [::1]:56827 [200]: GET /index.php?debugbar_time=16069
98910
Thu Dec 3 19:35:10 2020] [::1]:56827 Closing
Thu Dec 3 19:35:13 2020] [::1]:56828 Accepted
Thu Dec 3 19:35:13 2020] [::1]:56829 Accepted
Thu Dec 3 19:35:15 2020] [::1]:56828 Closing
Thu Dec 3 19:35:15 2020] [::1]:56829 Closed without sending a request; it was
probably just an unused speculative preconnection
Thu Dec 3 19:35:15 2020] [::1]:56829 Closing
Thu Dec 3 19:35:15 2020] [::1]:56832 Accepted
Thu Dec 3 19:35:15 2020] [::1]:56832 [200]: GET /index.php?debugbar
Thu Dec 3 19:35:15 2020] [::1]:56832 Closing
Thu Dec 3 19:35:15 2020] [::1]:56833 Accepted
Thu Dec 3 19:35:15 2020] [::1]:56833 [200]: GET /img/php.jpg
Thu Dec 3 19:35:15 2020] [::1]:56834 Accepted
Thu Dec 3 19:35:15 2020] [::1]:56834 [200]: GET /img/web.jpg
Thu Dec 3 19:35:15 2020] [::1]:56833 Closing
Thu Dec 3 19:35:15 2020] [::1]:56833 Closing
Thu Dec 3 19:35:15 2020] [::1]:56834 Accepted
Thu Dec 3 19:35:15 2020] [::1]:56834 [200]: GET /index.php?debugbar_time=16069
98915
Thu Dec 3 19:35:15 2020] [::1]:56835 Closing
Thu Dec 3 19:35:25 2020] [::1]:56837 Accepted
Thu Dec 3 19:35:25 2020] [::1]:56838 Accepted
Thu Dec 3 19:35:25 2020] [::1]:56837 Closing
Thu Dec 3 19:35:25 2020] [::1]:56838 [200]: GET /index.php?debugbar
Thu Dec 3 19:35:25 2020] [::1]:56838 Closing
Thu Dec 3 19:35:25 2020] [::1]:56840 Accepted
Thu Dec 3 19:35:25 2020] [::1]:56840 [200]: GET /img/php.jpg
Thu Dec 3 19:35:25 2020] [::1]:56841 Closing
Thu Dec 3 19:35:25 2020] [::1]:56841 Accepted
Thu Dec 3 19:35:25 2020] [::1]:56841 [200]: GET /img/web.jpg
Thu Dec 3 19:35:25 2020] [::1]:56842 Closing
Thu Dec 3 19:35:25 2020] [::1]:56842 Accepted
Thu Dec 3 19:35:25 2020] [::1]:56842 [200]: GET /index.php?debugbar_time=1606998925
Thu Dec 3 19:47:41 2020] [::1]:56948 Closing
Thu Dec 3 19:47:41 2020] [::1]:56948 Accepted
Thu Dec 3 19:47:41 2020] [::1]:56949 Accepted
Thu Dec 3 19:47:49 2020] [::1]:56949 Closing
Thu Dec 3 19:47:49 2020] [::1]:56949 Closed without sending a request; it was probably just an unused speculative preconnection
Thu Dec 3 19:47:49 2020] [::1]:56949 Closing

user@DESKTOP-78MT4F3 MINGW64 /c/xampp/htdocs/pustaka-booking
```

Jika server php codeigniternya masih belum dinonaktifkan seperti contoh gambar di halaman sebelumnya, maka dinonaktifkan terlebih dahulu dengan cara tekan CTRL + C. Selanjutnya ketikan script seperti gambar di bawah ini lalu enter.

```
user@DESKTOP-78MT4FJ MINGW64 /c/xampp/htdocs/pustaka-booking
$ composer update
```

Tunggu hingga prosesnya update-nya selesai. Contoh bisa dilihat seperti:

```
MINGW64:/c/xampp/htdocs/pustaka-booking
Package operations: 31 installs, 1 update, 0 removals
- Updating laminas/laminas-escaper (2.6.1 => 2.7.0): Downloading (100%)
- Installing fzaminotto/faker (dev-master ac73e52): Cloning ac73e52870 from cache
- Installing mikey179/vfsstream (v1.6.8): Downloading (100%)
- Installing sebastian/version (2.0.1): Downloading (100%)
- Installing sebastian/type (1.1.4): Downloading (100%)
- Installing sebastian/resource-operations (2.0.2): Downloading (100%)
- Installing sebastian/recursion-context (3.0.1): Downloading (100%)
- Installing sebastian/object-reflector (1.1.2): Downloading (100%)
- Installing sebastian/object-enumerator (3.0.4): Downloading (100%)
- Installing sebastian/global-state (3.0.1): Downloading (100%)
- Installing sebastian/exporter (3.1.3): Downloading (100%)
- Installing sebastian/environment (4.2.4): Downloading (100%)
- Installing sebastian/diff (3.0.3): Downloading (100%)
- Installing sebastian/comparator (3.0.3): Downloading (100%)
- Installing phpunit/php-timer (2.1.3): Downloading (100%)
- Installing phpunit/php-text-template (1.2.1): Downloading (100%)
- Installing phpunit/php-file-iterator (2.0.3): Downloading (100%)
- Installing theseer/tokenizer (1.2.0): Downloading (100%)
- Installing sebastian/code-unit-reverse-lookup (1.0.2): Downloading (100%)
- Installing phpunit/php-token-stream (4.0.4): Downloading (100%)
- Installing phpunit/php-code-coverage (7.0.14): Downloading (100%)
- Installing doctrine/instantiator (1.4.0): Downloading (100%)
- Installing phpdocumentor/reflection-common (2.2.0): Downloading (100%)
- Installing symfony/polyfill-ctype (v1.20.0): Downloading (100%)
- Installing webmozart/assert (1.9.1): Downloading (100%)
- Installing phpdocumentor/type-resolver (1.4.0): Downloading (100%)
- Installing phpdocumentor/reflection-docblock (5.2.2): Downloading (100%)
- Installing phpspec/prophecy (1.12.1): Downloading (100%)
- Installing phar-io/version (3.0.3): Downloading (100%)
- Installing phar-io/manifest (2.0.1): Downloading (100%)
- Installing myclabs/deep-copy (1.10.2): Downloading (100%)
- Installing phpunit/phpunit (8.5.13): Downloading (100%)
sebastian/global-state suggests installing ext-uopz (*)
sebastian/environment suggests installing ext-posix (*)
phpunit/php-code-coverage suggests installing ext-xdebug (^2.7.2)
phpunit/phpunit suggests installing phpunit/php-invoker (^2.0.0)
phpunit/phpunit suggests installing ext-soap (*)
phpunit/phpunit suggests installing ext-xdebug (*)
Package fzaminotto/faker is abandoned, you should avoid using it. No replacement was suggested.
Package phpunit/php-token-stream is abandoned, you should avoid using it. No replacement was suggested.
writing lock file
Generating autoload files
22 packages you are using are looking for funding.
Use the `composer fund` command to find out more!
Generating autoload files
Generated autoload files

user@DESKTOP-78MT4FJ MINGW64 /c/xampp/htdocs/pustaka-booking
$ |
```

10.2. Instalasi myth/auth

Untuk instalasi myth/auth sendiri masih di dalam git bash meneruskan dari update compser dan instalasi ini dibutuhkan koneksi internet. Untuk kodingnya sebagai berikut:

```
user@DESKTOP-78MT4F3 MINGW64 /c/xampp/htdocs/pustaka-booking
$ composer require myth/auth
```

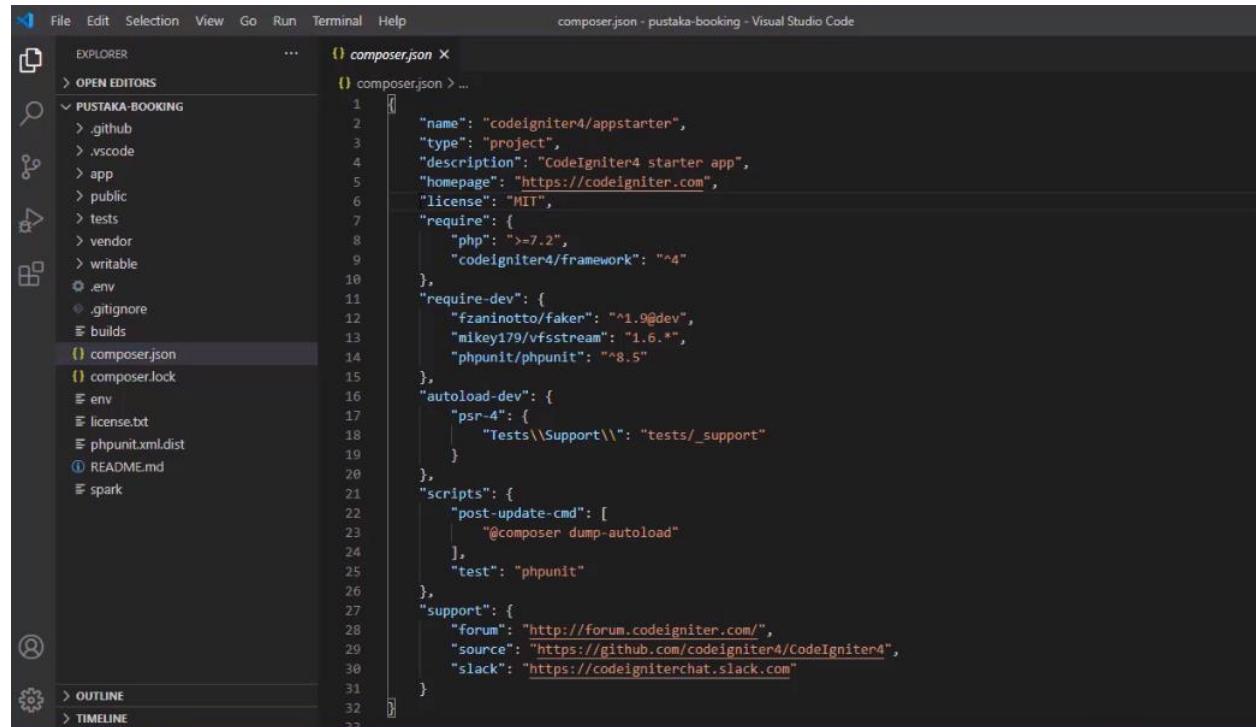
Jika muncul bar warna merah seperti gambar di bawah ini dikarenakan untuk myth/auth sendiri belum ada versi yang stablenya. Jadi kita diminta untuk mengubah konfigurasi myth/auth yang ada di dalam codeigniter 4 nya.

```
user@DESKTOP-78MT4F3 MINGW64 /c/xampp/htdocs/pustaka-booking
$ composer require myth/auth

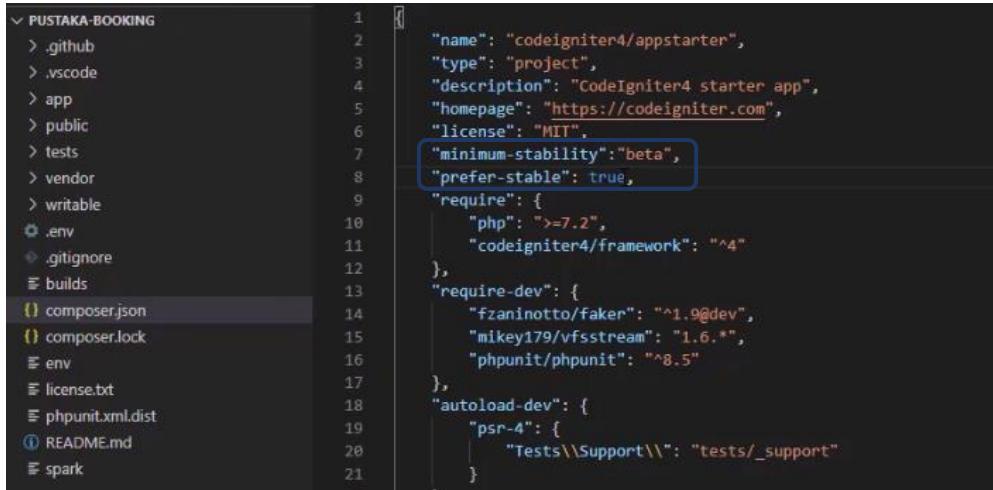
[InvalidArgumentException]
Could not find a version of package myth/auth matching your minimum-stability (stable). Require it with an explicit
version constraint allowing its desired stability.

require [--dev] [--prefer-source] [--prefer-dist] [--fixed] [--no-progress] [--no-suggest] [--no-update] [--no-scripts] [--update-no-dev] [--update-with-dependencies] [--update-with-all-dependencies] [--ignore-platform-reqs] [--prefer-stable] [--prefer-lowest] [--sort-packages] [-o|--optimize-autoloader] [-a|--classmap-authoritative] [--apcu-autoloader] [--] [<packages>]...
```

Untuk konfigurasinya silahkan buka visual studio codenya, kemudian klik pada bagian composer.json.

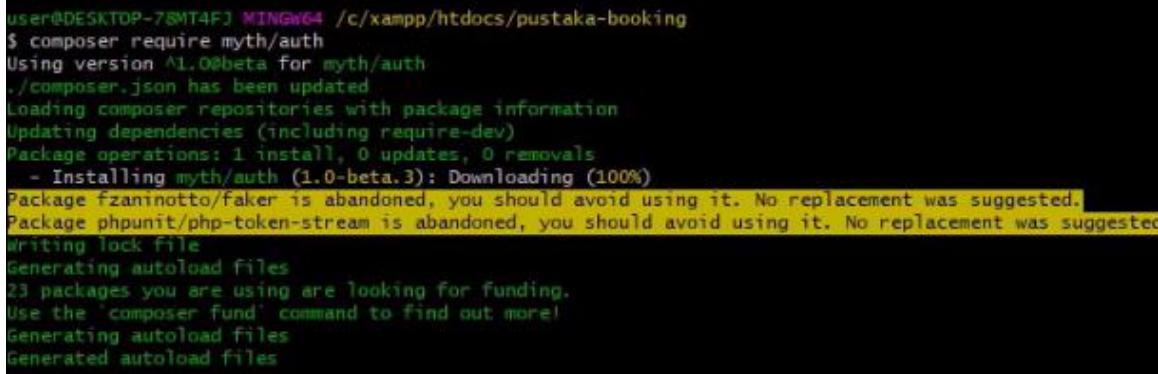


Tambahkan script berikut pada file composer.json-nya.



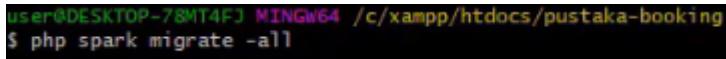
```
1  "name": "codeigniter4/appstarter",
2  "type": "project",
3  "description": "CodeIgniter4 starter app",
4  "homepage": "https://codeigniter.com",
5  "license": "MIT",
6  "minimum-stability": "beta",
7  "prefer-stable": true,
8  "require": {
9      "php": ">=7.2",
10     "codeigniter4/framework": "^4"
11 },
12 "require-dev": {
13     "fzaninotto/faker": "^1.9@dev",
14     "mikey179/vfsstream": "1.6.*",
15     "phpunit/phpunit": "^8.5"
16 },
17 "autoload-dev": {
18     "psr-4": {
19         "Tests\\Support\\": "tests/_support"
20     }
21 }
```

Selanjutnya kembali lagi ke instalasi myth/auth yang sebelumnya ada attention, untuk sekarang sudah tidak akan ada masalah lagi karena sudah dikonfigurasi.



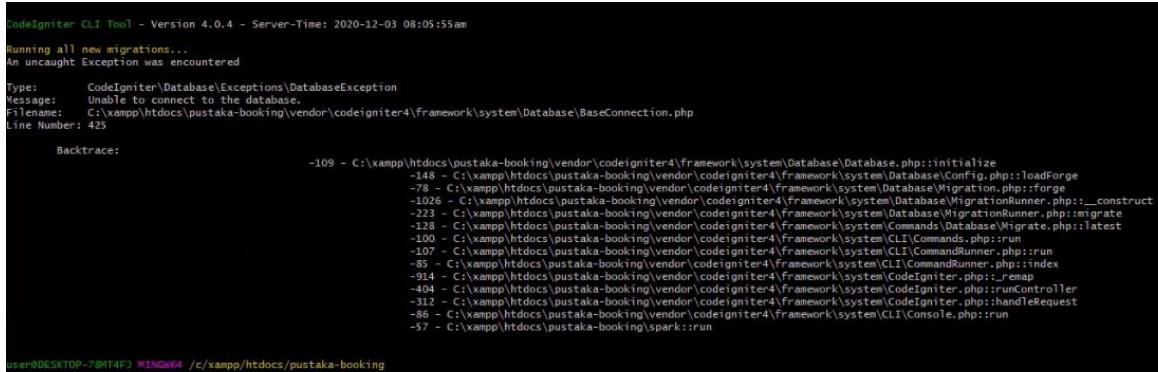
```
user@DESKTOP-78MT4FJ MINGW64 /c/xampp/htdocs/pustaka-booking
$ composer require myth/auth
Using version ^1.0@beta for myth/auth
./composer.json has been updated
Loading composer repositories with package information
Updating dependencies (including require-dev)
Package operations: 1 install, 0 updates, 0 removals
- Installing myth/auth (1.0-beta.3): Downloading (100%)
Package fzaninotto/faker is abandoned, you should avoid using it. No replacement was suggested.
Package phpunit/php-token-stream is abandoned, you should avoid using it. No replacement was suggested.
Writing lock file
Generating autoload files
23 packages you are using are looking for funding.
Use the 'composer fund' command to find out more!
Generating autoload files
Generated autoload files
```

Tahap selanjutnya adalah melakukan migrasi myth/auth. Dengan mengetikkan script di bawah ini lalu enter.



```
user@DESKTOP-78MT4FJ MINGW64 /c/xampp/htdocs/pustaka-booking
$ php spark migrate -all
```

Ketika sudah berhasil dimigrasi hasilnya akan tampak seperti dibawah ini:



```
CodeIgniter CLI Tool - Version 4.0.4 - Server-Time: 2020-12-03 08:05:55am
Running all new migrations...
An uncaught Exception was encountered

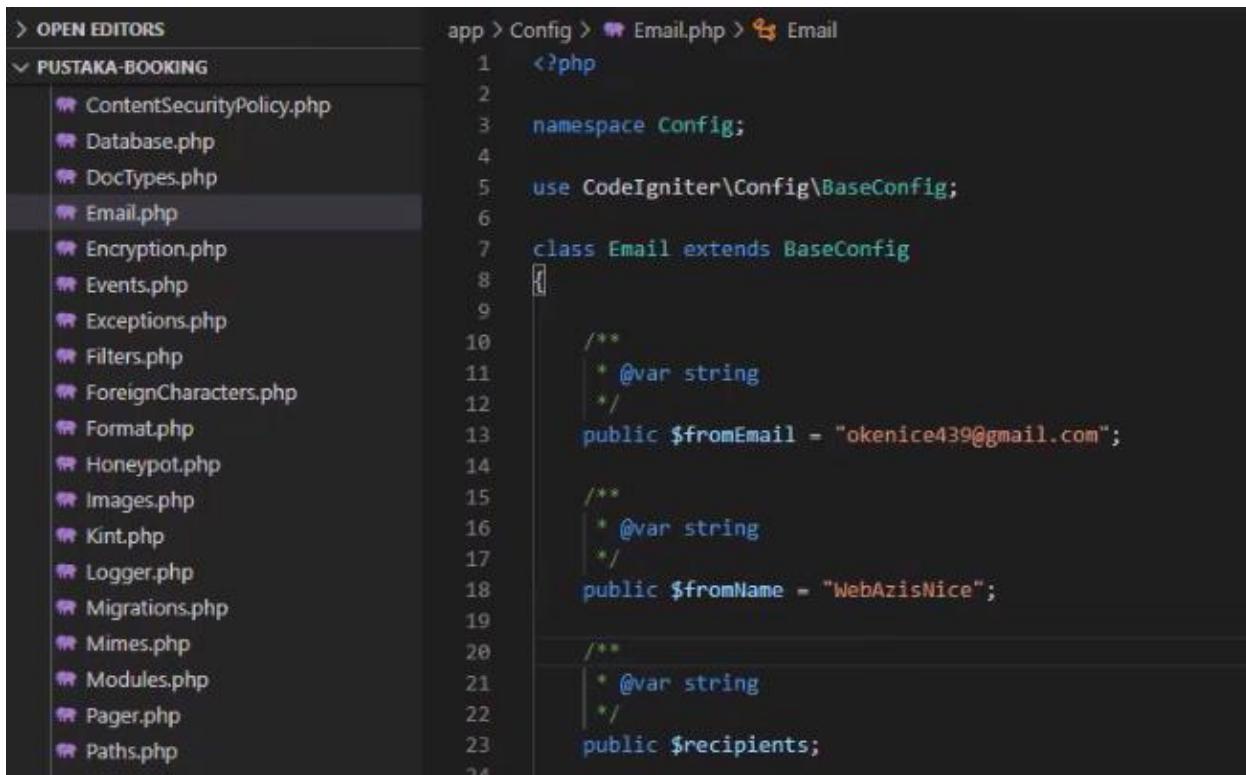
Type:    CodeIgniter\Database\Exceptions\DatabaseException
Message: Unable to connect to the database.
Filename: C:\xampp\htdocs\pustaka-booking\vendor\codeigniter4\framework\system\Database\BaseConnection.php
Line Number: 425

Backtrace:
-109 - C:\xampp\htdocs\pustaka-booking\vendor\codeigniter4\framework\system\Database.php::initialize
-148 - C:\xampp\htdocs\pustaka-booking\vendor\codeigniter4\framework\system\Database\Config.php::loadForge
-78 - C:\xampp\htdocs\pustaka-booking\vendor\codeigniter4\framework\system\Database\Migration.php::forge
-1026 - C:\xampp\htdocs\pustaka-booking\vendor\codeigniter4\framework\system\Database\MigrationRunner.php::__construct
-272 - C:\xampp\htdocs\pustaka-booking\vendor\codeigniter4\framework\system\Database\MigrationRunner.php::migrate
-128 - C:\xampp\htdocs\pustaka-booking\vendor\codeigniter4\framework\system\Database\MigrationRunner.php::latest
-100 - C:\xampp\htdocs\pustaka-booking\vendor\codeigniter4\framework\system\CLI\Commands.php::run
-107 - C:\xampp\htdocs\pustaka-booking\vendor\codeigniter4\framework\system\CLI\CommandRunner.php::run
-85 - C:\xampp\htdocs\pustaka-booking\vendor\codeigniter4\framework\system\CLI\CommandRunner.php::index
-914 - C:\xampp\htdocs\pustaka-booking\vendor\codeigniter4\framework\system\CodeIgniter.php::_remap
-404 - C:\xampp\htdocs\pustaka-booking\vendor\codeigniter4\framework\system\CodeIgniter.php::runController
-312 - C:\xampp\htdocs\pustaka-booking\vendor\codeigniter4\framework\system\CodeIgniter.php::handleRequest
-86 - C:\xampp\htdocs\pustaka-booking\vendor\codeigniter4\framework\system\CLI\Console.php::run
-57 - C:\xampp\htdocs\pustaka-booking\spark::run

user@DESKTOP-78MT4FJ MINGW64 /c/xampp/htdocs/pustaka-booking
$
```

10.3. Konfigurasi Email di Codeigniter 4

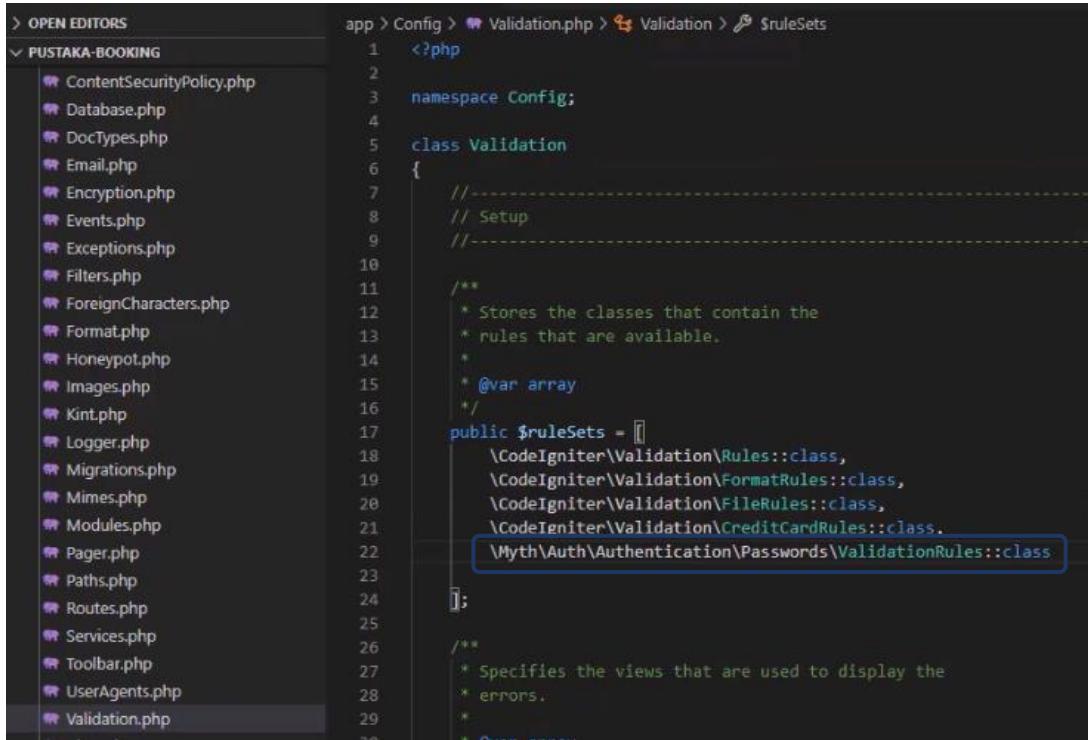
Lanjutkan dengan masuk ke folder app -> config -> Email.php. Kemudian isi yang method fromEmail dan fromName. Untuk email yang digunakan jangan gunakan email utama melainkan email cadangan, sedangkan untuk nama email itu bebas.



The screenshot shows a code editor with a sidebar titled 'OPEN EDITORS' containing a list of PHP files under the 'PUSTAKA-BOOKING' directory. The 'Email.php' file is selected and highlighted in the list. The main pane displays the contents of the Email.php file:

```
<?php  
namespace Config;  
use CodeIgniter\Config\BaseConfig;  
  
class Email extends BaseConfig {  
    /** * @var string */  
    public $fromEmail = "okenice439@gmail.com";  
    /** * @var string */  
    public $fromName = "WebAzisNice";  
    /** * @var string */  
    public $recipients;  
}
```

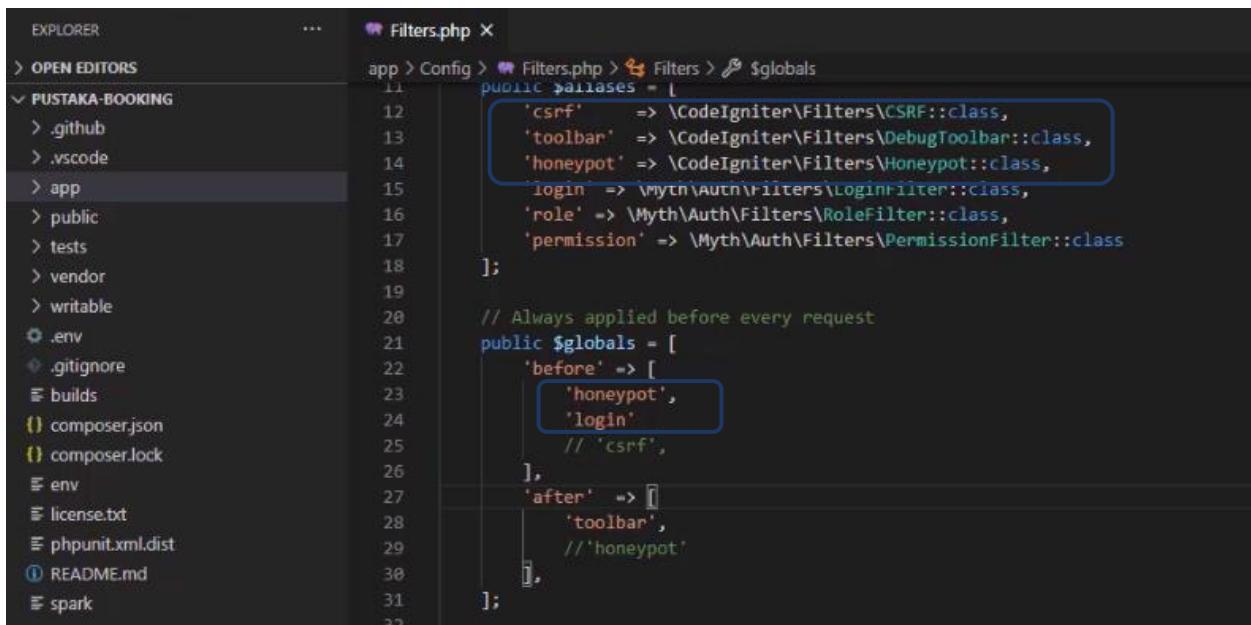
Tahapan berikutnya masuk ke Validation.php yang posisinya masih dalam folder yang sama dengan Email.php.



```
> OPEN EDITORS
> PUSTAKA-BOOKING
  ContentSecurityPolicy.php
  Database.php
  DocTypes.php
  Email.php
  Encryption.php
  Events.php
  Exceptions.php
  Filters.php
  ForeignCharacters.php
  Format.php
  Honeypot.php
  Images.php
  Kint.php
  Logger.php
  Migrations.php
  Mimes.php
  Modules.php
  Pager.php
  Paths.php
  Routes.php
  Services.php
  Toolbar.php
  UserAgents.php
  Validation.php

app > Config > Validation.php > Validation > RuleSets
1  <?php
2
3  namespace Config;
4
5  class Validation
6  {
7      /**
8      * Stores the classes that contain the
9      * rules that are available.
10     *
11     * @var array
12     */
13     public $ruleSets = [
14         \CodeIgniter\Validation\Rules::class,
15         \CodeIgniter\Validation\FormatRules::class,
16         \CodeIgniter\Validation\FileRules::class,
17         \CodeIgniter\Validation\CreditCardRules::class,
18         \Myth\Auth\Authentication\Passwords\ValidationRules::class
19     ];
20
21     /**
22     * Specifies the views that are used to display the
23     * errors.
24     *
25     */
26 }
```

Tetap di folder yang sama, pilih Validation.php. Lalu tambahkan koding berikut:



```
EXPLORER ... Filters.php X
> OPEN EDITORS
> PUSTAKA-BOOKING
  .github
  .vscode
  app
  public
  tests
  vendor
  writable
  .env
  .gitignore
  builds
  composer.json
  composer.lock
  env
  license.txt
  phpunit.xml.dist
  README.md
  spark

app > Config > Filters.php > Filters > Globals
11  public $aliases = [
12      'csrf' => \CodeIgniter\Filters\CSRF::class,
13      'toolbar' => \CodeIgniter\Filters\DebugToolbar::class,
14      'honeypot' => \CodeIgniter\Filters\Honeypot::class,
15      'login' => \Myth\Auth\Filters>LoginFilter::class,
16      'role' => \Myth\Auth\Filters\Rolefilter::class,
17      'permission' => \Myth\Auth\Filters\PermissionFilter::class
18  ];
19
20  // Always applied before every request
21  public $globals = [
22      'before' => [
23          'honeypot',
24          'login'
25          // 'csrf',
26      ],
27      'after' => [
28          'toolbar',
29          // 'honeypot'
30      ],
31  ];
32
```

Dilanjutkan dengan masuk ke LoginFilter.php yang ada di dalam folder vendor -> myth/auth -> src -> Filters. Tambahkan script yang di frame biru.

```
> Authentication      15   * normal execution. However, when an abnormal state
> Authorization       16   * is found, it should return an instance of
> Collectors          17   * CodeIgniter\HTTP\Response. If it does, script
> Commands            18   * execution will end and that Response will be
> Config               19   * sent back to the client, allowing for error pages,
> Controllers          20   * redirects, etc.
> Database             21   *
> Entities             22   *
> Exceptions           23   *
> Filters              24   *
>   LoginFilter.php    25   * @param \CodeIgniter\HTTP\RequestInterface $request
>   PermissionFilter.php 26   *
>   RoleFilter.php      27   *
> Helpers              28   *
> Language             29   *
> Models               30   *
> Views                31   *
> AuthTrait.php        32   *
> AuthTrait.php        33   if (!function_exists('logged_in')) {
> AuthTrait.php        34   helper('auth');
> AuthTrait.php        35   }

public function before(RequestInterface $request, $argument = NULL)
{
    if (!function_exists('logged_in')) {
        helper('auth');

    }

    $current = (string)current_url(true)
        ->setHost('')
        ->setScheme('')
        ->stripQuery('token');
}
```

Satu lagi tambahan pada baris akhir:

```
public function after(RequestInterface $request, ResponseInterface $response, $argument = NULL)
{
}

//-----
```

Selanjutnya masih di dalam folder myth/auth tetapi bukan di folder Filters lagi tapi di folder src/Config/. Hilangkan tag komentar (double slash) pada script yang sudah di frame biru.

```
> OPEN EDITORS
vendor > myth > auth > src > Config > Auth.php > Auth > $validFields
> PUSTAKA-BOOKING
  > mikey179
  > myclabs
  < myth\auth
    > .github
    > bin
    > docs
    < src
      > Authentication
      > Authorization
      > Collectors
      > Commands
      < Config
        < Auth.php
        < Registrar.php
        < Routes.php
  > myclabs
  > mikey179

209 // Password Check Helpers
210 //-----
211 // The PasswordValidator class runs the password through all of these
212 // classes, each getting the opportunity to pass/fail the password.
213 // You can add custom classes as long as they adhere to the
214 // Password\ValidatorInterface.
215 //
216 public $passwordValidators = [
217     'Myth\Auth\Authentication\Passwords\CompositionValidator',
218     'Myth\Auth\Authentication\Passwords\NothingPersonalValidator',
219     'Myth\Auth\Authentication\Passwords\DictionaryValidator',
220     //!'Myth\Auth\Authentication\Passwords\PwnedValidator',
221 ];
222 //-----
223 //-----
224 // Activator classes
225 //-----
226 // Available activators with config settings
227 //
```

Setelah dihilangkan:

```
public $passwordValidators = [
    'Myth\Auth\Authentication\Passwords\CompositionValidator',
    'Myth\Auth\Authentication\Passwords\NothingPersonalValidator',
    'Myth\Auth\Authentication\Passwords\DictionaryValidator',
    'Myth\Auth\Authentication\Passwords\PwnedValidator',
];
```

Kembali ke Email.php. lengkapi script di bawah ini. Untuk SMTPUser samakan dengan fromEmail yang sudah diinput di tahapan sebelumnya.

```
/*
 * The server path to Sendmail.
 *
 * @var string
 */
public $mailPath = '/usr/sbin/sendmail';

/**
 * SMTP Server Address
 *
 * @var string
 */
public $SMTPHost = 'smtp.gmail.com';

/**
 * SMTP Username
 *
 * @var string
 */
public $SMTPUser = 'topenice439@gmail.com';

*/
public $SMTPPass='';
```

Untuk password silahkan diisi sesuai dengan password dari email yang sudah diinput di \$SMTPUser.

```

    < PUSTAKA-BOOKING
      > .github
      > .vscode
      < app
        < Config
          > Boot
          & App.php
          & Autoload.php
          & Cache.php
          & Constants.php
          & ContentSecurityPolicy.php
          & Database.php
          & DocTypes.php
          & Email.php
          & Encryption.php
          & Events.php
          & Exceptions.php
          & Filters.php
          & ForeignCharacters.php
          & Format.php
          & Honeypot.php
          & Images.php
          & Kint.php
          & Logger.php
          & Migrations.php

```

```

66   /**
67   * SMTP Port
68   *
69   * @var integer
70   */
71  public $SMTPPort = 465;

72  /**
73  * SMTP Timeout (in seconds)
74  *
75  * @var integer
76  */
77  public $SMTPTimeout = 60;

78  /**
79  * Enable persistent SMTP connections
80  *
81  * @var boolean
82  */
83  public $SMTPKeepAlive = false;

84  /**
85  * SMTP Encryption. Either tls or ssl
86  *
87  * @var string
88  */
89  public $SMTPCrypto = 'ssl';

90
91
92
93
94

```

```

    & Email.php
    & Encryption.php
    & Events.php
    & Exceptions.php
    & Filters.php
    & ForeignCharacters.php
    & Format.php
    & Honeypot.php
    & Images.php
    & Kint.php
    & Logger.php

```

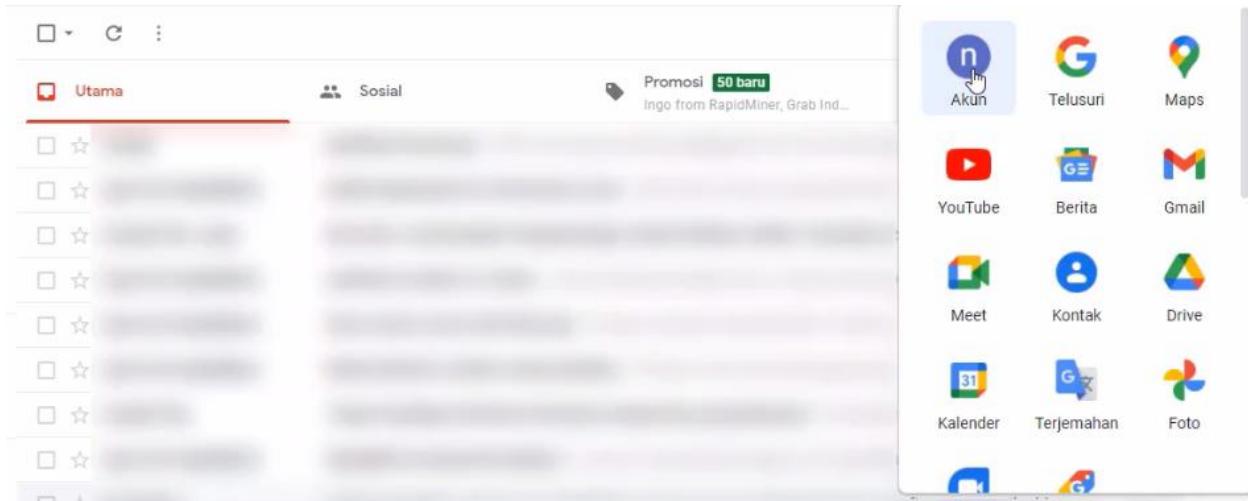
```

104  /**
105  * @var integer
106  */
107  public $wrapChars = 76;

108
109  /**
110  * Type of mail, either 'text' or 'html'
111  *
112  * @var string
113  */
114  public $mailType = 'html';
115

```

Selanjutnya masuk ke gmail dari email yang sudah dikonfigurasi di Email.php. Lalu pilih akun.



Kemudian pilih keamanan.

Selamat datang, nice nice

Keamanan

Privasi & personalisasi

Lihat data di Akun Google Anda dan pilih jenis aktivitas yang disimpan untuk mempersonalisasi pengalaman Anda saat menggunakan Google

[Kelola data & personalisasi Anda](#)

Masalah keamanan ditemukan

Lindungi akun Anda sekarang dengan menyelesaikan masalah ini

[Amankan akun](#)

Lalu aktifkan aksesnya.

Akses aplikasi yang kurang aman

Untuk melindungi akun Anda, aplikasi dan perangkat yang menggunakan teknologi login yang kurang aman diblokir. Untuk menjaga keamanan akun Anda, Google akan otomatis MENONAKTIFKAN setelan ini jika tidak sedang digunakan.

Nonaktif

[Aktifkan akses \(tidak direkomendasikan\)](#)

Geser ke posisi aktif.

← Akses aplikasi yang kurang aman

Beberapa aplikasi dan perangkat menggunakan teknologi login yang kurang aman, sehingga akun Anda menjadi rentan. Anda dapat menonaktifkan akses untuk aplikasi ini, seperti rekomendasi kami, atau mengaktifkan akses jika Anda ingin menggunakan meski berisiko. Google akan otomatis MENONAKTIFKAN setelan ini jika tidak sedang digunakan. [Pelajari lebih lanjut](#)

Izinkan aplikasi yang kurang aman: NONAKTIF



PERTEMUAN 11

IMPLEMENTASI FORM LOGIN DAN FORM REGISTRASI

Deskripsi Pembelajaran

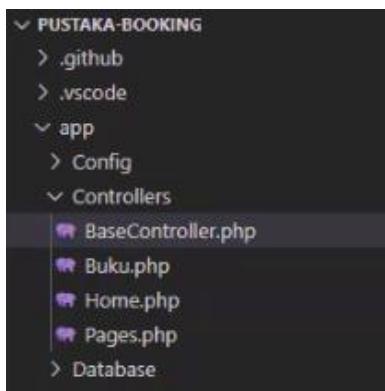
Pada bab kali ini membahas tentang implementasi dari form login dan form registrasi. Dimana kedua form tersebut tidak perlu dibuatkan layoutnya secara manual menggunakan template bootstrap seperti yang dilakukan saat membuat form login dan registrasi di codeigniter 3 karena di codeigniter 4 ini sudah disediakan hanya saja mengkonfigurasinya terlebih dahulu dan form login dan registrasi pun langsung bisa digunakan.

Tujuan Pembelajaran

Pada bab ini diharapkan mahasiswa mampu mengimplementasikan form login dan registrasi dengan baik. Mengetahui bagaimana cara registrasi dan mengetahui cara validasinya.

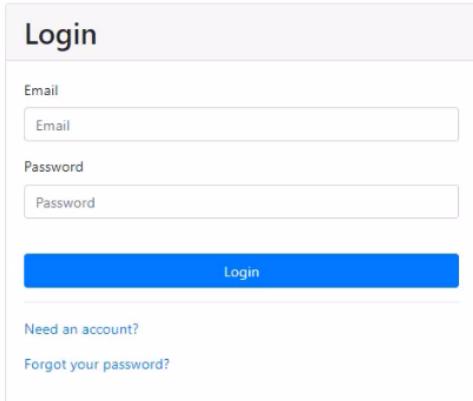
11.1. Implementasi Form Login dan Form Registrasi

Sebelum masuk ke browser seperti biasanya aktifkan terlebih dahulu php server codeigniter 4 dengan menggunakan git bash. Setelah itu masuk ke visual studio code dan lengkapi script yang terdapat di BaseController.php.



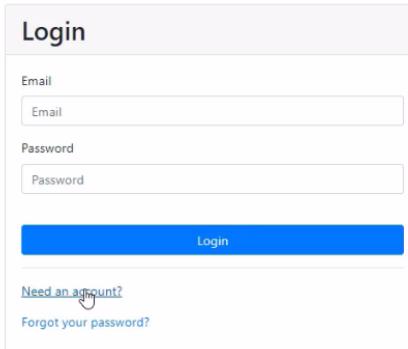
```
18 use CodeIgniter\Controller;
19
20 class BaseController extends Controller
21 {
22
23     /**
24      * An array of helpers to be loaded automatically upon
25      * class instantiation. These helpers will be available
26      * to all other controllers that extend BaseController.
27      *
28      * @var array
29      */
30     protected $helpers = ['auth'];
31 }
```

Cek selanjutnya di browser. Dan hasilnya pun bisa dilihat form login terbuat tanpa perlu membuat layoutnya secara manual.



The screenshot shows a dark header bar with the text "Myth:Auth" and "Home". Below it is a light-colored login form titled "Login". The form contains two input fields: "Email" and "Password", both with placeholder text "Email" and "Password" respectively. Below the fields is a large blue rectangular button labeled "Login". At the bottom of the form, there are two links: "Need an account?" and "Forgot your password?".

Klik need an account untuk menuju ke form registrasinya.



This screenshot is identical to the one above, showing the Myth:Auth login page. However, the "Need an account?" link at the bottom left of the form is now underlined and has a red circle drawn around it, indicating that it has been interacted with.

Kemudian isi semua kolom teksnya sesuai akun dengan email yang ingin didaftarkan. Jika sudah diisi semua, klik register.

Register

Email
azisnice3@gmail.com
We'll never share your email with anyone else.

Username
Azis_Nice3

Password

Repeat Password

Already registered? [Sign In](#)

Jika berhasil akan muncul pesan seperti di bawah ini dan posisi akan kembali lagi ke halaman form login.

MythAuth Home

Login

Please confirm your account by clicking the activation link in the email we have sent.

Email or username

Password

Need an account?
[Forgot your password?](#)

Untuk aktivasi akunnya bisa dengan cek kotak masuk dari email yang baru saja didaftarkan. Lalu klik activate account. Maka secara otomatis akun yang didaftarkan sudah aktif.

Activate your account **Kotak Masuk**

WebAzisNice <okenice439@gmail.com>
kepada saya ▾

This is activation email for your account on <http://localhost:8080>.
To activate your account use this URL.
[Activate account.](#)

If you did not register on this website, you can safely ignore this email.

Dan saat cek database yang sudah dibuat sebelumnya maka akan banyak table yang tercipta otomatis. Hal ini disebabkan karena adanya fitur migrasi myth/auth yang sebelumnya sudah diimport ke dalam codeigniter 4 yang digunakan untuk membuat form log in dan form registrasi ini. Jika flashback dari pertemuan 2, saat itu tabel yang dibuat hanya ada dua. Yakni, tabel anggota dan tabel buku.

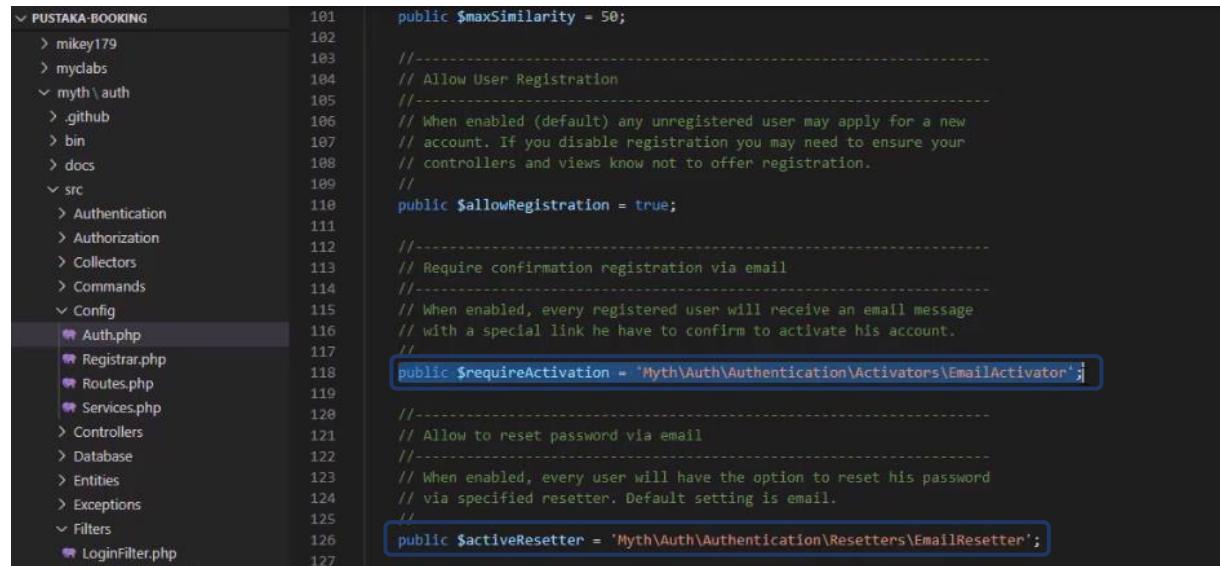
Server: 127.0.0.1 Database: pustaka

| Structure | SQL | Search | Query | Export | Import | Operations | Privileges | Routines | Events | Triggers | ... | |
|---|--------|--------|--------|--------------------|-----------|------------|------------|----------|--------|----------|-----|--|
| Filters | | | | | | | | | | | | |
| Containing the word: <input type="text"/> | | | | | | | | | | | | |
| Table | Action | Rows | Type | Collation | Size | Overhead | | | | | | |
| anggota | | 16 | InnoDB | utf8mb4_general_ci | 16.0 Kib | - | | | | | | |
| auth_activation_attempts | | 16 | InnoDB | utf8_general_ci | 16.0 Kib | - | | | | | | |
| auth_groups | | 16 | InnoDB | utf8_general_ci | 16.0 Kib | - | | | | | | |
| auth_groups_permissions | | 48 | InnoDB | utf8_general_ci | 48.0 Kib | - | | | | | | |
| auth_groups_users | | 48 | InnoDB | utf8_general_ci | 48.0 Kib | - | | | | | | |
| auth_logins | | 4 | InnoDB | utf8_general_ci | 48.0 Kib | - | | | | | | |
| auth_permissions | | 16 | InnoDB | utf8_general_ci | 16.0 Kib | - | | | | | | |
| auth_reset_attempts | | 16 | InnoDB | utf8_general_ci | 16.0 Kib | - | | | | | | |
| auth_tokens | | 48 | InnoDB | utf8_general_ci | 48.0 Kib | - | | | | | | |
| auth_users_permissions | | 48 | InnoDB | utf8_general_ci | 48.0 Kib | - | | | | | | |
| buku | | 4 | InnoDB | utf8mb4_general_ci | 16.0 Kib | - | | | | | | |
| migrations | | 1 | InnoDB | utf8_general_ci | 16.0 Kib | - | | | | | | |
| users | | 1 | InnoDB | utf8_general_ci | 48.0 Kib | - | | | | | | |
| 13 tables | Sum | 18 | InnoDB | utf8mb4_general_ci | 400.0 Kib | 8.8 | | | | | | |

Dan jika dilihat di tabel users, akan terlihat akun yang sudah aktif terdaftar. Dan passwordnya pun otomatis sudah dienkripsi.

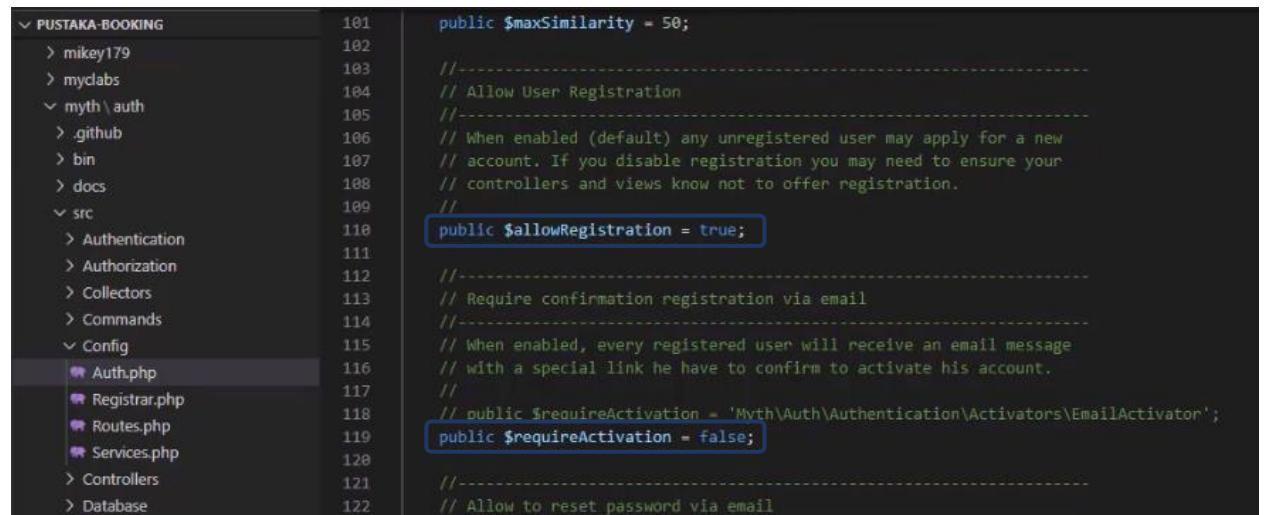
| + Options | | | | | | |
|--------------------------|--|----|---------------------|-------------|--|------------|
| | | id | email | username | password_hash | reset_hash |
| <input type="checkbox"/> | | 1 | azisnice3@gmail.com | azisnice321 | S2y\$10\$0wHaCoH.x3nebcj0RWKp.Xk/g3vV1pLXi6lgFpXNFg... | NULL |

Sedangkan jika tidak ingin menyambungkan form registrasi dan form loginya ke email atau tanpa melalui konfirmasi ke email bisa dilakukan dengan cara menonaktifkannya melalui perubahan script berikut:



```
101 public $maxSimilarity = 50;
102 //-----
103 // Allow User Registration
104 //-----
105 // When enabled (default) any unregistered user may apply for a new
106 // account. If you disable registration you may need to ensure your
107 // controllers and views know not to offer registration.
108 //
109 public $allowRegistration = true;
110 //-----
111 // Require confirmation registration via email
112 //-----
113 // When enabled, every registered user will receive an email message
114 // with a special link he have to confirm to activate his account.
115 //
116 public $requireActivation = 'Myth\Auth\Authentication\Activators\EmailActivator';
117 //-----
118 // Allow to reset password via email
119 //-----
120 // When enabled, every user will have the option to reset his password
121 // via specified resetter. Default setting is email.
122 //
123 public $activeResetter = 'Myth\Auth\Authentication\Resetters\EmailResetter';
124 //-----
125 // -----
126 // -----
127 // -----
```

Rubah menjadi seperti ini:



```
101 public $maxSimilarity = 50;
102 //-----
103 // Allow User Registration
104 //-----
105 // When enabled (default) any unregistered user may apply for a new
106 // account. If you disable registration you may need to ensure your
107 // controllers and views know not to offer registration.
108 //
109 public $allowRegistration = true;
110 //-----
111 // Require confirmation registration via email
112 //-----
113 // When enabled, every registered user will receive an email message
114 // with a special link he have to confirm to activate his account.
115 //
116 // public $requireActivation = 'Myth\Auth\Authentication\Activators\EmailActivator';
117 public $requireActivation = false;
118 //-----
119 // -----
120 // Allow to reset password via email
121 // -----
122 // -----
```

Dan bisa dicoba ulang untuk melakukan registrasi.

Register

Email

We'll never share your email with anyone else.

Username

Password

Repeat Password

[Register](#)

Already registered? [Sign In](#)

Dan saat berhasil tampilannya seperti berikut:

Login

Welcome aboard! Please login with your new credentials.

Email or username

Password

[Login](#)

[Need an account?](#)

PERTEMUAN 12

PEMBUATAN LOGOUT

Deskripsi Pembelajaran

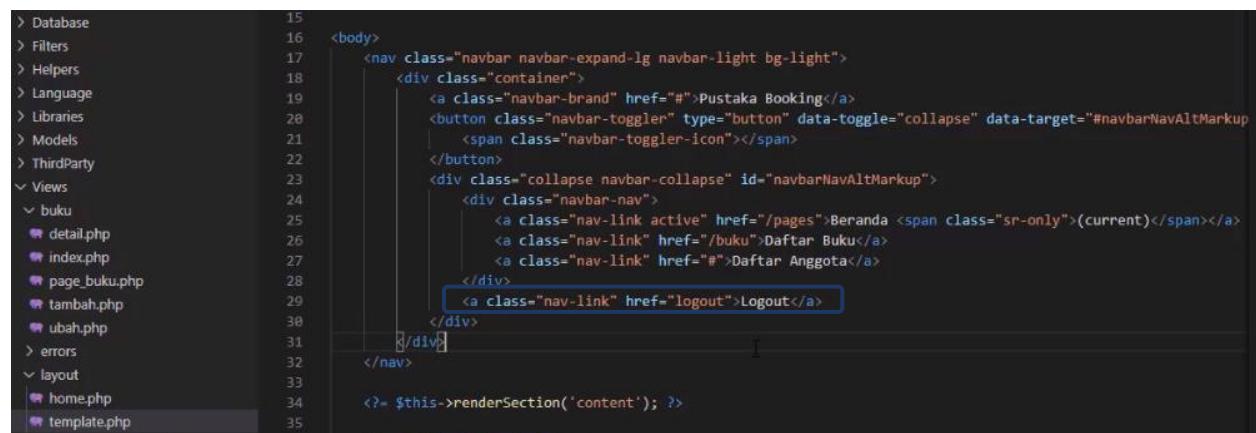
Pada bab ini akan membahas tentang bagaimana keluar dari beranda web setelah melakukan login dengan menggunakan menu yang disebut dengan logout.

Tujuan Pembelajaran

Pada bab ini diharapkan mahasiswa dapat memahami cara breakdown (destroy) dari menu utama di sebuah web untuk kembali lagi ke form login.

12.1. Pembuatan Menu Logout

Logout dibuat pada file template.php. Untuk scriptnya bisa dilihat di bawah ini:



```
> Database          15   <body>
> Filters           16     <nav class="navbar navbar-expand-lg navbar-light bg-light">
> Helpers           17       <div class="container">
> Language          18         <a class="navbar-brand" href="#">Pustaka BookingBeranda <span class="sr-only">(current)</span></a>
<     index.php      25             <a class="nav-link" href="/buku">Daftar Buku</a>
<     page_buku.php  26             <a class="nav-link" href="#">Daftar Anggota</a>
<     tambah.php     27           </div>
<     ubah.php       28         </div>
<   errors          29         </div>
<   layout          30         </div>
<   home.php        31         </div>
<   template.php    32       </nav>
<?> $this->renderSection('content'); ?>
```

Pembuatan logout pada codeigniter 4 tidak membutuhkan coding destroy seperti coding di codeigniter 3 karena codeigniter 4 sudah menyediakan fitur logoutnya sendiri pada librabry-nya. Cukup klik logout maka otomatis halaman akan kembali ke form log in.



PERTEMUAN 13 – 15
PRESENTASI PROJECT

PERTEMUAN 16
UJIAN AKHIR SEMESTER

DAFTAR PUSTAKA

- Foster, Rob. 2015. Codeigniter Web Application Blueprints. Birmingham: Packt Publishing Ltd.
- Harani, Nisa Hanum, Andri Fajar Sunandar. 2020. Aplikasi Prospek Sales Menggunakan Codeigniter. Bekasi: Kreatif.
- Nordeen, Alex. 2020. Learn Codeigniter 24 hours. Ahmedabad: Guru99.
- Subagia, Anton. 2017. Membangun Aplikasi Dengan CodeIgniter Dan Database SQL Server. Jakarta: Elex Media Komputindo.
- Wardana. 2010. Menjadi Master PHP dengan Framework Codeigniter. Jakarta: Elex Media Komputindo.