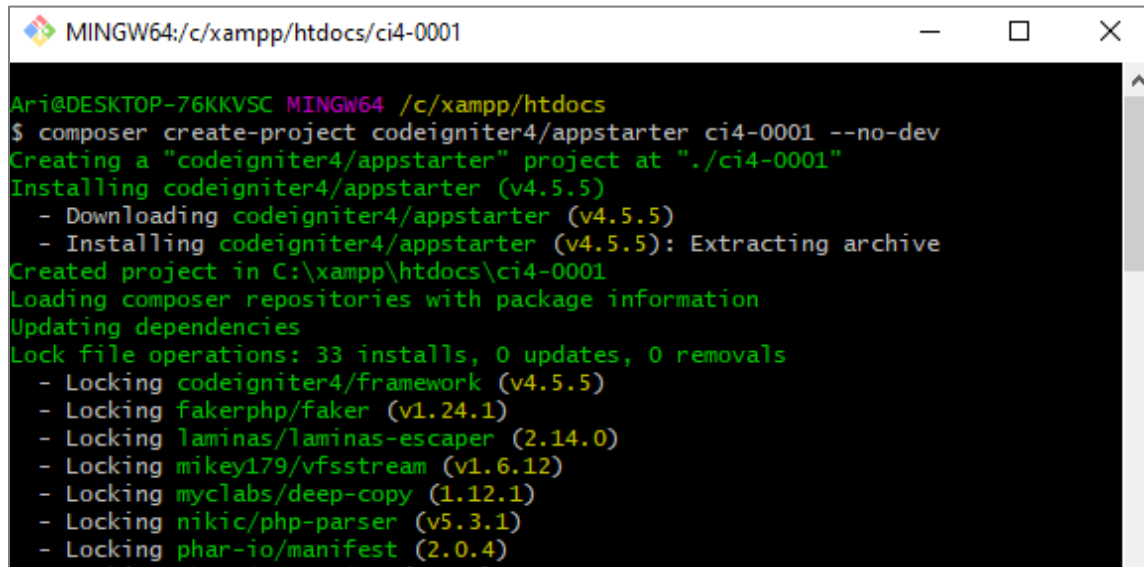


Migration dan Seeder Pada CodeIgniter 4

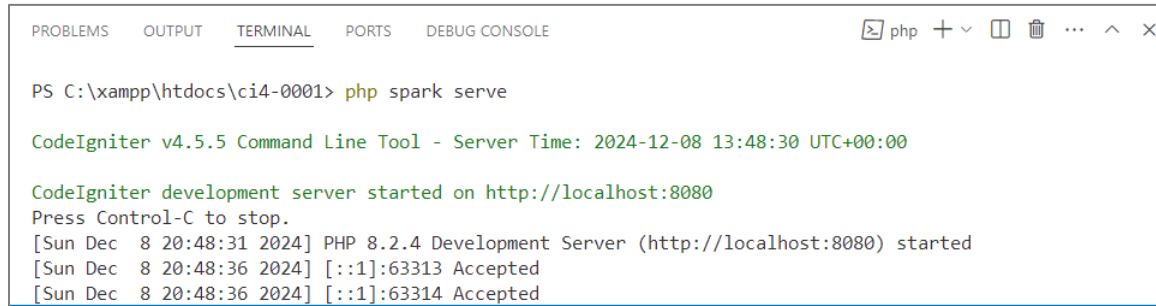
1. Buat proyek baru dengan nama ci4-4digitnim (contoh: ci4-0001) melalui composer, **composer create-project codeigniter4/appstarter ci4-0001 --no-dev**



```

MINGW64:/c:/xampp/htdocs/ci4-0001
Ari@DESKTOP-76KKVSC MINGW64 /c:/xampp/htdocs
$ composer create-project codeigniter4/appstarter ci4-0001 --no-dev
Creating a "codeigniter4/appstarter" project at "./ci4-0001"
Installing codeigniter4/appstarter (v4.5.5)
- Downloading codeigniter4/appstarter (v4.5.5)
- Installing codeigniter4/appstarter (v4.5.5): Extracting archive
Created project in C:\xampp\htdocs\ci4-0001
Loading composer repositories with package information
Updating dependencies
Lock file operations: 33 installs, 0 updates, 0 removals
- Locking codeigniter4/framework (v4.5.5)
- Locking fakerphp/faker (v1.24.1)
- Locking laminas/laminas-escaper (2.14.0)
- Locking mikey179/vfsstream (v1.6.12)
- Locking myclabs/deep-copy (1.12.1)
- Locking nikic/php-parser (v5.3.1)
- Locking phar-io/manifest (2.0.4)
  
```

Pastikan proses instalasi berjalan dengan benar, setelah proses instalasi selesai buka proyek pada VScode lalu jalankan **php spark serve** pada terminal

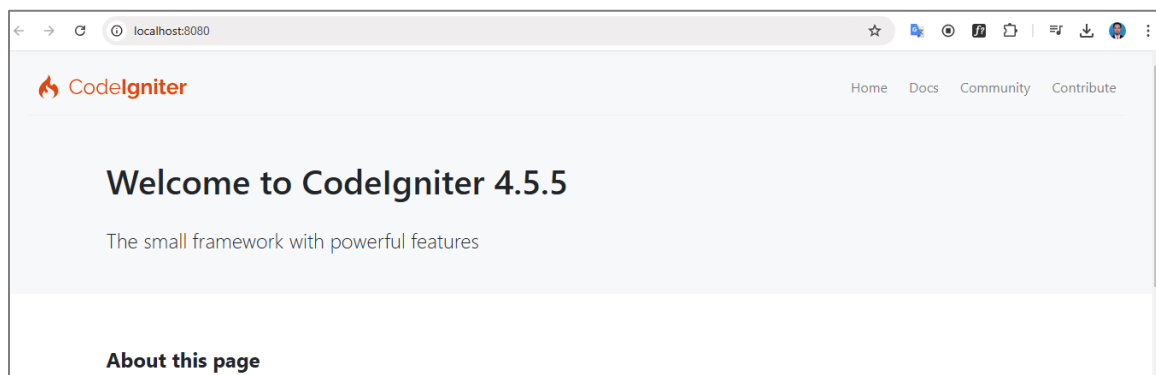


```

PROBLEMS OUTPUT TERMINAL PORTS DEBUG CONSOLE
PS C:\xampp\htdocs\ci4-0001> php spark serve

CodeIgniter v4.5.5 Command Line Tool - Server Time: 2024-12-08 13:48:30 UTC+00:00

CodeIgniter development server started on http://localhost:8080
Press Control-C to stop.
[Sun Dec  8 20:48:31 2024] PHP 8.2.4 Development Server (http://localhost:8080) started
[Sun Dec  8 20:48:36 2024] [::1]:63313 Accepted
[Sun Dec  8 20:48:36 2024] [::1]:63314 Accepted
  
```

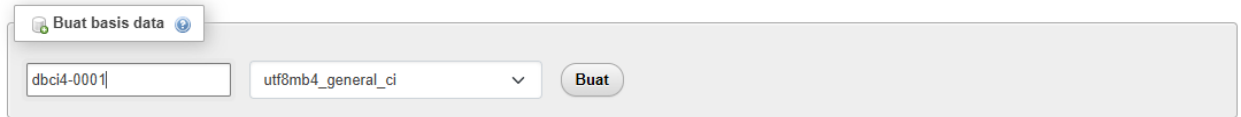


Edit file env menjadai .env dan ubah baris code

17	# CI_ENVIRONMENT = production	➡	17	CI_ENVIRONMENT = development
----	-------------------------------	---	----	------------------------------

2. Membuat dan Konfigurasi Database

a. Buat database dengan nama dbci4-0001



b. Konfigurasi database

Pada file `.env` isikan data sesuai dengan konfigurasi database yang dibuat

```

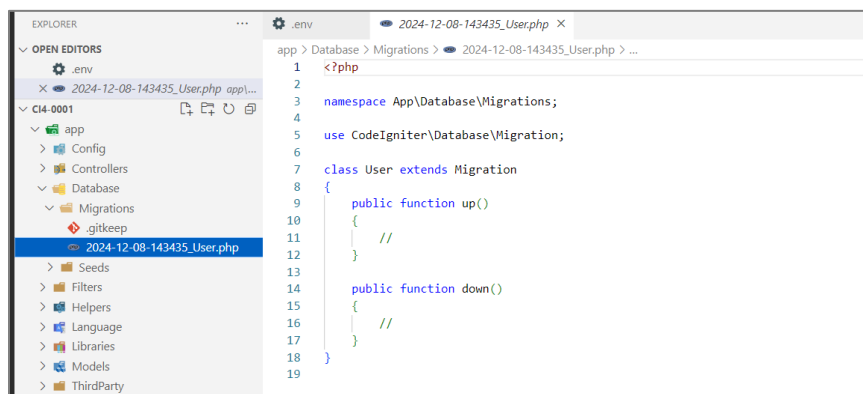
29  #-----
30  # DATABASE
31  #-----
32
33  database.default.hostname = localhost
34  database.default.database = dbci4-0001
35  database.default.username = root
36  database.default.password =
37  database.default.DBDriver = MySQLi
38  database.default.DBPrefix =
39  database.default.port = 3306
    
```

3. Membuat Table

Membuat table dengan nama **user** melalui codeigniter, buka terminal baru lalu ketikkan perintah **php spark make:migration user**

```
PS C:\xampp\htdocs\ci4-0001> php spark make:migration user
```

Akan akan dibuat file baru pada direktori `app/database/migration` dengan nama file keterangan tanggal lalu nama file (`_User`)



Isikan field yang akan dibuat pada tabel user, pada function up()

```

7  class User extends Migration
8  {
9      public function up()
10     {
11         $this->forge->addField([
12             'id' => [
13                 'type'          => 'INT',
14                 'constraint'    => 5,
15                 'unsigned'      => true,
16                 'auto_increment' => true,
17             ],
18             'username' => [
19                 'type'          => 'VARCHAR',
20                 'constraint'    => 255,
21                 'unique'        => true,
22             ],
23             'password' => [
24                 'type'          => 'VARCHAR',
25                 'constraint'    => 255
26             ],
27             'nama_lengkap' => [
28                 'type'          => 'VARCHAR',
29                 'constraint'    => 255
30             ],
31             'email' => [
32                 'type'          => 'VARCHAR',
33                 'constraint'    => 255,
34                 'unique'        => true
35             ],
36             'token' => [
37                 'type'          => 'VARCHAR',
38                 'constraint'    => 255
39             ],
40             'last_login timestamp default now()'
41         ]);
42         $this->forge->addKey('id', TRUE);
43         $this->forge->createTable('user');
44     }

```

Jika sudah, pada function down tambahkan perintah berikut

```

46     public function down()
47     {
48         $this->forge->dropTable('user');
49     }
50 }

```

Untuk mengaplikasikan field ke dalam database phpmyadmin ketikan perintah **php spark migrate**

```
PS C:\xampp\htdocs\ci4-0001> php spark migrate
```

Pastikan tidak ada kesalahan saat melakukan migrate. Selanjutnya bisa dilihat pada database dbci4-0001 akan muncul table user dengan struktur tabel yang sesuai yang dibuat pada codeigniter

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
<input type="checkbox"/> 1	id	int(5)		UNSIGNED	Tidak	Tidak ada		AUTO_INCREMENT	Ubah Hapus Lainnya
<input type="checkbox"/> 2	username	varchar(25)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/> 3	password	varchar(255)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/> 4	nama_lengkap	varchar(255)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/> 5	email	varchar(25)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/> 6	token	varchar(255)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/> 7	last_login	timestamp			Tidak	current_timestamp()			Ubah Hapus Lainnya

4. Membuat Input Data

Input data user dilakukan melalui seeder pada codeigniter, ketikkan perintah **php spark make:seeder user** pada terminal

```
PS C:\xampp\htdocs\ci4-0001> php spark make:seeder user
```

Jika sudah, maka akan dibuat file baru dengan nama User.php pada direktor app/database/migrations

```
app > Database > Seeds > User.php > ...
3 namespace App\Database\Seeds;
4
5 use CodeIgniter\Database\Seeder;
6
7 class User extends Seeder
8 {
9     public function run()
10     {
11         //
12     }
13 }
```

Pada function run(), masukkan data yang akan dibuat, misalnya seperti berikut

```
9     public function run()
10     {
11         //
12         $data = [
13             'username' => 'admin',
14             'password' => password_hash('admin', PASSWORD_DEFAULT),
15             'nama_lengkap' => 'Ari Putra',
16             'email' => 'ariputra.stmikwp@gmail.com'
17         ];
18         $this->db->table('user')->insert($data);
19     }
```

Jika sudah, ketikkan perintah **php spark db:seed user** untuk memasukkan data ke tabel user. Pastikan tidak ada kesalahan maka bisa dicek pada table user pada php myadmid

	id	username	password	nama_lengkap	email	token
<input type="checkbox"/>	1	admin	\$2y\$10\$OB9hAxHWBnF3zny/BcFITup0zUGPNU8cDd52NpNcCE8...	Ari PUkra	ariputra.stmikwp@gmail.co	

5. Membuat Model

Berikutnya buat model dengan nama User.php, pada terminal ketikkan perintah **php spark make:model user**, otomatis akan dibuatkan file baru dengan nama User.php pada direktori app/models. Selanjutnya sesuaikan isian pada variable yang ada dan buat fungsi getData() dan updateData()

```

7  class User extends Model
8  {
9      protected $table = "user";
10     protected $primaryKey = "email";
11     protected $allowedFields = ['username', 'password', 'nama_lengkap', 'token'];
12
13     /**
14      * untuk ambil data
15      */
16     public function getData($parameter)
17     {
18         $builder = $this->table($this->table); //this->table yang pertama adalah f
19         $builder->where('username=', $parameter); // where dan orWhere merupakan f
20         $builder->orWhere('email=', $parameter);
21         $query = $builder->get();
22         return $query->getRowArray();
23     }
24
25     /** untuk update / simpan data */
26     public function updateData($data)
27     {
28         $builder = $this->table($this->table);
29         if ($builder->save($data)) {
30             return true;
31         } else {
32             return false;
33         }
34     }
35 }

```

6. Membuat Controller

Buat controller dengan nama User.php, ketikkan perintah **php spark make:controller user** pada terminal

```
PS C:\xampp\htdocs\ci4-0001> php spark make:controller user
```

```

app > Controllers > User.php > ...
1  <?php
2
3  namespace App\Controllers;
4
5  use App\Controllers\BaseController;
6  use CodeIgniter\HTTP\ResponseInterface;
7
8  class User extends BaseController
9  {
10     public function index()
11     {
12         //
13     }
14 }
15

```

Tambahkan method return view dan arahkan ke tampilan login

```

8  class User extends BaseController
9  {
10     public function index()
11     {
12         return view('login/v_login');
13     }
14 }

```

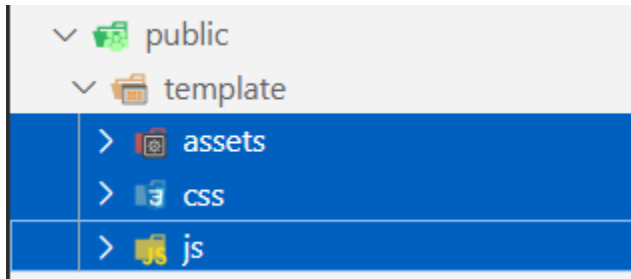
7. Membuat Form Login

Tampilan yang akan digunakan adalah sb-admin yang bisa didownload pada alamat <https://startbootstrap.com/template/sb-admin>

The screenshot shows the Start Bootstrap website's 'sb-admin' template page. The header includes the Start Bootstrap logo and navigation links: Themes, Templates, Bundles, Forms, and Resources. A 'Log In' button is visible in the top right. The main content area features a sidebar with links to Layouts, Pages, Charts, and Tables. The central dashboard preview includes four colored cards (Primary, Warning, Success, Danger), two charts (Area and Bar), and a data table. The data table has columns for Name, Position, Office, Age, Start date, and Salary, with three rows of sample data. A large Adobe Creative Cloud advertisement is on the right, and a 'Free Download' button is at the bottom.

Name	Position	Office	Age	Start date	Salary
Ari Setou	Accountant	Tokyo	33	2008/11/28	\$162,700
Angelica Ramos	Chief Executive Officer (CEO)	London	47	2009/10/09	\$1,200,000
Ashley Cox	Junior Technical Author	San Francisco	66	2009/01/12	\$86,000

Setelah didownload lakukan ekstrak dan copy folder assets, css dan js. Buat folder dengan nama template pada direktori public dari proyek. Lalu paste-kan file yang tadi sudah dicopy.



Berikutnya pada direktori **views**, buat folder dengan nama **login**, lalu buat file baru dengan nama **v_login.php**. untuk v_login.php diisi kode dari login.html dari file sb-admin yang sudah didownload sebelumnya.

```

1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5     <meta charset="utf-8" />
6     <meta http-equiv="X-UA-Compatible" content="IE=edge" />
7     <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=
8     <meta name="description" content="" />
9     <meta name="author" content="" />
10    <title>Login - SB Admin</title>
11    <link href="css/styles.css" rel="stylesheet" />
12    <script src="https://use.fontawesome.com/releases/v6.3.0/js/all.js" crossorigin="a
13 </head>
14
15 <body class="bg-primary">
16     <div id="layoutAuthentication">
17         <div id="layoutAuthentication_content">
18             <main>
19                 <div class="container">
20                     <div class="row justify-content-center">
21                         <div class="col-lg-5">
22                             <div class="card shadow-lg border-0 rounded-lg mt-5">
23                                 <div class="card-header">
24                                     <h3 class="text-center font-weight-light my-4">Log
25                                 </div>

```

Sesuaikan direktori pada href css dan script

11	<link href="css/styles.css" rel="stylesheet" />
11	<link href="<?php echo base_url('template') ?>/css/styles.css" rel="stylesheet" />
71	<script src="js/scripts.js"></script>
71	<script src="<?php echo base_url('template') ?>/js/scripts.js"></script>

Ubah routes

```
8 $routes->get('/', 'User::index');
```

Hasil

