

Penambahan sintak pada bagian **CONFIG/auth.php** (warna kuning)

```
<?php

return [

    /*
    |-----
    | Authentication Defaults
    |-----
    |
    | This option defines the default authentication "guard" and password
    | reset "broker" for your application. You may change these values
    | as required, but they're a perfect start for most applications.
    |
    */

    'defaults' => [
        'guard' => env('AUTH_GUARD', 'web'),
        'passwords' => env('AUTH_PASSWORD_BROKER', 'users'),
    ],

    /*
    |-----
    | Authentication Guards
    |-----
    |
    | Next, you may define every authentication guard for your application.
    | Of course, a great default configuration has been defined for you
    | which utilizes session storage plus the Eloquent user provider.
    |
    | All authentication guards have a user provider, which defines how the
    | users are actually retrieved out of your database or other storage
    | system used by the application. Typically, Eloquent is utilized.
    |
    | Supported: "session"
    |
    */

    'guards' => [
        'web' => [
            'driver' => 'session',
            'provider' => 'users',
        ],
    ],
];
```

```

    'admin' => [
        'driver' => 'session',
        'provider' => 'admins',
    ],
],

/*
|-----
| User Providers
|-----
|
| All authentication guards have a user provider, which defines how the
| users are actually retrieved out of your database or other storage
| system used by the application. Typically, Eloquent is utilized.
|
| If you have multiple user tables or models you may configure multiple
| providers to represent the model / table. These providers may then
| be assigned to any extra authentication guards you have defined.
|
| Supported: "database", "eloquent"
|
*/

'providers' => [
    'users' => [
        'driver' => 'eloquent',
        'model' => env('AUTH_MODEL', App\Models\User::class),
    ],
    'admins' => [
        'driver' => 'eloquent',
        'model' => App\Models\Admin::class,
    ],

    // 'users' => [
    //     'driver' => 'database',
    //     'table' => 'users',
    // ],
],

/*
|-----
| Resetting Passwords
|-----
|

```

| These configuration options specify the behavior of Laravel's password
| reset functionality, including the table utilized for token storage
| and the user provider that is invoked to actually retrieve users.
|

| The expiry time is the number of minutes that each reset token will be
| considered valid. This security feature keeps tokens short-lived so
| they have less time to be guessed. You may change this as needed.
|

| The throttle setting is the number of seconds a user must wait before
| generating more password reset tokens. This prevents the user from
| quickly generating a very large amount of password reset tokens.
|

*/

```
'passwords' => [  
    'users' => [  
        'provider' => 'users',  
        'table' => env('AUTH_PASSWORD_RESET_TOKEN_TABLE', 'password_reset_tokens'),  
        'expire' => 60,  
        'throttle' => 60,  
    ],  
],
```

/*

|-----
Password Confirmation Timeout

| Here you may define the amount of seconds before a password confirmation
| window expires and users are asked to re-enter their password via the
| confirmation screen. By default, the timeout lasts for three hours.
|

*/

```
'password_timeout' => env('AUTH_PASSWORD_TIMEOUT', 10800),
```

```
];
```

Halaman Registrasi

Pada bagian sebelumnya, Anda berhasil membuat sebuah fungsi login untuk kedua user (user biasa dan admin), membuat middleware, dan controller. Pada bagian ini, Anda akan membuat fungsi untuk melakukan registrasi user baru. Untuk langkah pertama, Anda akan menambahkan metode untuk melakukan registrasi dalam **AuthController.php** yang berlokasi didalam **app/Http/Controllers/Auth/AuthController.php** kemudian tulis kode dibawah ini:

```
1 public function register()
2 {
3     return view('register');
4 }
5
6 public function post_register(Request $request)
7 {
8     $validator = Validator::make($request->all(), [
9         'name' => 'required',
10        'email' => 'required|email:dns',
11        'password' => 'required|min:8|max:8',
12    ]);
13
14    if ($validator->fails()) {
15        Alert::error('Gagal!', 'Pastikan semua terisi dengan benar!');
16        return redirect()->back();
17    }
18
19    $user = User::create([
20        'name' => $request->name,
21        'email' => $request->email,
22        'password' => bcrypt($request->password),
23        'point' => 10000,
24    ]);
25
26    if ($user) {
27        Alert::success('Berhasil!', 'Akun baru berhasil dibuat, silahkan melakukan login!');
28        return redirect('/');
29    } else {
30        Alert::error('Gagal!', 'Akun gagal dibuat, silahkan coba lagi!');
31        return redirect()->back();
32    }
33 }
```

Setelah Anda menambahkan kode diatas, selanjutnya Anda akan membuat sebuah view baru yang berisi halaman registrasi untuk pengguna atau user baru. Tambahkan file baru tersebut dalam lokasi **resources/views** dengan nama filenya adalah **register.blade.php** kemudian tambahkan copy kode dibawah ini dalam file tersebut:

```
<!DOCTYPE html>
<html lang="zxx" class="no-js">

<head>
    <!-- Mobile Specific Meta -->
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <!-- Favicon-->
    <link rel="shortcut icon" href="
        {{ asset('assets/templates/user/img/fav.png') }}">
    <!-- Author Meta -->
    <meta name="author" content="CodePixar">
    <!-- Meta Description -->
    <meta name="description" content="">
    <!-- Meta Keyword -->
    <meta name="keywords" content="">
    <!-- meta character set -->
    <meta charset="UTF-8">
    <!-- Site Title -->
    <title>Merch Store</title>

    <!--
    CSS ===== -->
    <link rel="stylesheet"
        href="{{ asset('assets/templates/user/css/linearicons.css') }}">
    <link rel="stylesheet"
        href="{{ asset('assets/templates/user/css/owl.carousel.css') }}">
    <link rel="stylesheet"
        href="{{ asset('assets/templates/user/css/themify-icons.css') }}">
    <link rel="stylesheet"
        href="{{ asset('assets/templates/user/css/font-awesome.min.css') }}">
    <link rel="stylesheet"
        href="{{ asset('assets/templates/user/css/nice-select.css') }}">
    <link rel="stylesheet"
        href="{{ asset('assets/templates/user/css/nouislider.min.css') }}">
    <link rel="stylesheet"
        href="{{ asset('assets/templates/user/css/bootstrap.css') }}">
    <link rel="stylesheet"
        href="{{ asset('assets/templates/user/css/main.css') }}">
</head>

<body>
    @include('sweetalert::alert')

    <!--=====Login Box Area =====-->
    <section class="login_box_area section_gap">
        <div class="container">
            <div class="row">
                <div class="col-lg-6">
                    <div class="login_box_img">
                        
                    </div>
                </div>
            </div>
        </div>
    </section>
</body>
</html>
```

```

        <div class="hover">
            <p>There are advances being made in science and technology everyday,
            and a good example of this is the</p>
        </div>
    </div>
</div>
<div class="col-lg-6">
    <div class="login_form_inner">
        <h3>Register new account</h3>
        <form class="row login_form" action="{{ route('post.register')
}}}" method="POST" id="contactForm" novalidate="novalidate">
            @csrf
            <div class="col-md-12 form-group">
                <input type="text" class="form-control" id="name"
name="name" placeholder="Name" onfocus="this.placeholder = ''
onblur="this.placeholder = 'Name'">
            </div>
            <div class="col-md-12 form-group">
                <input type="email" class="form-control" id="email"
name="email" placeholder="Email" onfocus="this.placeholder = ''
onblur="this.placeholder = 'Email'">
            </div>
            <div class="col-md-12 form-group">
                <input type="password" class="form-control" id="password"
name="password" placeholder="Password" onfocus="this.placeholder = ''
onblur="this.placeholder = 'Password'">
            </div>
            <div class="col-md-12 form-group">
                <button type="submit" value="submit" class="primary-
btn">Create Account</button>
            </div>
        </form>
    </div>
</div>
</div>
</section>
<!--=====End Login Box Area =====-->

<script src="{{ asset('assets/templates/user/js/vendor/jquery-2.2.4.min.js')
}}"></script>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.11.0/umd/popper.min.js"
integrity="sha384-b/U6ypiBEHp0f/4+1nzFpr53nxSS+GLCKfwbDfNtclqenISfwAzpKaMNFNmj4"
crossorigin="anonymous"></script>
<script
src="{{ asset('assets/templates/user/js/vendor/bootstrap.min.js') }}">
</script>
<script src="{{ asset('assets/templates/user/js/jquery.ajaxchimp.min.js')
}}"></script>
<script src="{{ asset('assets/templates/user/js/jquery.nice-select.min.js')
}}"></script>
<script src="{{ asset('assets/templates/user/js/jquery.sticky.js') }}"></script>
<script src="{{ asset('assets/templates/user/js/nouislider.min.js') }}"></script>
<script src="{{ asset('assets/templates/user/js/jquery.magnific-popup.min.js')
}}"></script>
<script src="{{ asset('assets/templates/user/js/owl.carousel.min.js')
}}"></script>
<!--gmaps Js-->

```

```
<script  
src="https://maps.googleapis.com/maps/api/js?key=AIzaSyCjCGmQ0Uq4exrzdcL6rvxywDD0vfA  
u6eE"></script>  
    <script src="{{ asset('assets/templates/user/js/gmaps.min.js') }}"></script>  
    <script src="{{ asset('assets/templates/user/js/main.js') }}"></script>  
</body>  
</html>
```

Setelah menambahkan fungsi dan halaman untuk pengguna baru melakukan proses registrasi, langkah selanjutnya Anda akan menambahkan route baru. Untuk menambahkan route baru tersebut, tulislah kode dibawah ini dalam file **web.php** yang diberi tanda warna kuning:

```
1 // Guest Route  
2 Route::group(['middleware' => 'guest'], function() {  
3     Route::get('/', function () {  
4         return view('welcome');  
5     });  
6  
7     Route::get('/register', [AuthController::class, 'register'])->name('register');  
8     Route::post('/post-register', [AuthController::class, 'post_register'])->name('post.register');  
9  
10    Route::post('/post-login', [AuthController::class, 'login']);  
11 }->middleware('guest');
```

Selanjutnya jalankan project anda untuk melihat apakah proses registrasi dapat berjalan dengan data dari form Registrasi yang terinput kedalam database.