

LAPORAN TUGAS BESAR

LAPORAN TUGAS BESAR PEMROGRAMAN WEB LANJUT

Disusun sebagai

MATA KULIAH : PEMROGRAMAN WEB LANJUT

oleh :

DIKA AYU WIJAYANTI/1931710061

MI-2C



PROGRAM STUDI MANAJEMEN INFORMATIKA

JURUSAN TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

2020

1. Membuat project baru dengan mengetikkan perintah ““composer create-project --prefer-dist laravel/laravel perpusku_baru” (mohon maaf dalam projek ini saya membuat baru lagi dengan nama perpusku_baru, karena pada perpusku_laravel kemarin terjadi eror saat coba jalankan php artisan db: seed dan saat menjalankan localhost:8000/login yang seharusnya menuju ke halaman login tetapi saat itu langsung menuju ke home dan saya coba benarkan pada source codenya tetep seperti itu sehingga akhirnya saya membuat kembali). Dimana pada projek besar ini saya membuat web E-Perpus yang mana dalam E-Perpus ini sebagai member bisa melakukan peminjaman buku pada perpustakaan ini.

```
Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\xampp\htdocs> composer create-project --prefer-dist laravel/laravel perpusku_baru
Creating a "laravel/laravel" project at "./perpusku_baru"
Installing laravel/laravel (v8.4.2)
  - Installing laravel/laravel (v8.4.2): Downloading (100%)
Created project in C:\xampp\htdocs\perpusku_baru
> @php -r "file_exists('.env') || copy('.env.example', '.env');"
Loading composer repositories with package information
Updating dependencies (including require-dev)
Package operations: 104 installs, 0 updates, 0 removals
  - Installing voku/portable-ascii (1.5.6): Downloading (100%)
  - Installing symfony/polyfill-php80 (v1.20.0): Loading from cache
  - Installing symfony/polyfill-mbstring (v1.20.0): Loading from cache
  - Installing symfony/polyfill ctype (v1.20.0): Loading from cache
  - Installing phppoption/phppoption (1.7.5): Loading from cache
  - Installing graham-campbell/result-type (v1.0.1): Loading from cache
  - Installing vlucas/phpdotenv (v5.2.0): Loading from cache
  - Installing symfony/css-selector (v5.1.8): Loading from cache
  - Installing tijssverkijken/css-to-inline-styles (2.2.3): Loading from cache
  - Installing symfony/var-dumper (v5.1.8): Downloading (100%)
  - Installing symfony/deprecation-contracts (v2.2.0): Loading from cache
Failed to download symfony/deprecation-contracts from dist: Could not delete C:\xampp\htdocs\perpusku_baru\vendor\composer\4c435c41\symfony-deprecation-contracts-5fa56b4:
This can be due to an antivirus or the Windows Search Indexer locking the file while they are analyzed
Now trying to download from source
  - Installing symfony/deprecation-contracts (v2.2.0): Cloning 5fa56b4074 from cache
  - Installing symfony/routing (v5.1.8): Downloading (100%)
  - Installing symfony/process (v5.1.8): Downloading (100%)
  - Installing symfony/polyfill-php72 (v1.20.0): Loading from cache
  - Installing symfony/polyfill-intl-normalizer (v1.20.0): Loading from cache
  - Installing symfony/polyfill-intl-idn (v1.20.0): Loading from cache
  - Installing symfony/polyfill-intl-grapheme (v1.20.0): Downloading (100%)
  - Installing symfony/string (v5.1.8): Downloading (100%)
  - Installing phpdocumentor/reflection-common (2.2.0): Loading from cache
  - Installing webmozart/assert (1.9.1): Loading from cache
  - Installing phpdocumentor/type-resolver (1.4.0): Loading from cache
```

2. Kemudian buatlah folder Anggota, auth,buku,laporan,layouts,transaksi dimana folder – folder tersebut yang nantinya akan diisi berupa file .blade.php dimana cara membuatnya adalah dengan cara “**klik kanan pada folder view kemudian pilih new file.”**

➤ **Folder Auth**

- a. Folder Email(yang didalamnya terdapat **file invite.blade.php** yang digunakan jika anda sudah mendaftarkan e-mail anda ke admin maka akan muncul tampilan “ **Admin telah mendaftarkan e-mail anda ke E- Perpus**

Ceria.Untuk Login silakan anda kunjungi.” Login dengan e-mail Anda dan password”. Jika Anda ingin mengubah password, silahkan kunjungi reset dan masukan e- mail anda.

```
perpusku_baru > resources > views > auth > emails > 📧 invite.blade.php
1   <p>
2   |     Hallo!! Selamat {{ $member->name }}
3   </p>
4   <p>
5   |     Admin kami telah mendaftarkan email Anda {{ $member->email }} ke E- Perpusku Ceria. Untuk login,
6   </p>
7
8   <p>
9   |     Jika Anda ingin mengubah password, silahkan kunjungi <a href="{{ $reset = url('password/reset') }}"
10  </p>
```

b. Verification.blade.php

```
perpusku_baru > resources > views > auth > emails > 📧 verification.blade.php
1 Klik link berikut untuk melakukan aktivasi akun E- Perpusku Ceria:
2 <a href="{{ $link = url('auth/verify', $token). '?email=' . urlencode($user->email) }}"> {{ $link }}[
```

❖ Folder Passwords

Di dalam folder passwords ini terdapat file email.blade.php yang digunakan untuk reset password email dan juga terdapat file reset.blade.php.

a. email.blade.php

```
perpusku_baru > resources > views > auth > passwords > 📧 email.blade.php
1   @extends('layouts.app')
2
3   @section('content')
4   <div class="container">
5       <div class="row">
6           <div class="col-md-8 col-md-offset-2">
7               <div class="panel panel-default">
8                   <div class="panel-heading">
9                       <h2 class="panel-title">Reset Password</h2>
10
11
12               <div class="panel-body">
13                   @if (session('status'))
14                       <div class="alert alert-success">
15                           {{ session('status') }}
16                       </div>
17                   @endif
18
19                   {!! Form::open(['url' => '/password/email', 'class' => 'form-horizontal']) !!}
20                   {{ csrf_field() }}
21
22                   <div class="form-group{{ $errors->has('email') ? ' has-error' : '' }}>
23                       <label for="email" class="control-label">Alamat Email</label>
24
25                       <div class="col-md-6">
26                           {!! Form::email('email', null, ['class' => 'form-control']) !!}
```

```

perpusku_baru > resources > views > auth > passwords > 📧 email.blade.php
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46

```

b. reset.blade.php

```

perpusku_baru > resources > views > auth > passwords > 🌐 reset.blade.php
1   @extends('layouts.app')
2
3   @section('content')
4   <div class="container">
5     <div class="row">
6       <div class="col-md-8 col-md-offset-2">
7         <div class="panel panel-default">
8           <div class="panel-heading">
9             <h2 class="panel-title">Reset Password</h2>
10            </div>
11
12            <div class="panel-body">
13              @if (session('status'))
14                <div class="alert alert-success">
15                  {{ session('status') }}
16                </div>
17              @endif
18
19              {!! Form::open(['url' => '/password/reset', 'class' => 'form-horizontal']) !!}
20              {{ csrf_field() }}
21
22              <input type="hidden" name="token" value="{{ $token }}>
23
24              <div class="form-group{{ $errors->has('email') ? ' has-error' : '' }}>
25                {!! Form::label('email', 'Alamat Email', ['class' => 'col-md-4 control-]
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46

```

```
perpusku_baru > resources > views > auth > passwords > 🗑 reset.blade.php
23
24          <div class="form-group{{ $errors->has('email') ? ' has-error' : '' }}>
25              {!! Form::label('email', 'Alamat Email', ['class' => 'col-md-4 control-l
26
27                  <div class="col-md-6">
28                      {!! Form::email('email', null, ['class' => 'form-control']) !!}
29                      {!! $errors->first('email', '<p class="help-block">:message</p>') !!}
30                  </div>
31          </div>
32
33          <div class="form-group{{ $errors->has('password') ? ' has-error' : '' }}>
34              {!! Form::label('password', 'Password', ['class' => 'col-md-4 control-la
35
36                  <div class="col-md-6">
37                      {!! Form::password('password', ['class' => 'form-control']) !!}
38                      {!! $errors->first('password', '<p class="help-block">:message</p>') !!}
39                  </div>
40          </div>
41
42          <div class="form-group{{ $errors->has('password_confirmation') ? ' has-error
43              {!! Form::label('password_confirmation', 'Konfirmasi Password', ['class'
44                  <div class="col-md-6">
45                      {!! Form::password('password_confirmation', ['class' => 'form-contro
46                      {!! $errors->first('password_confirmation', '<p class="help-block">:
47                  </div>
48          </div>
49
50          <div class="form-group">
51              <div class="col-md-6 col-md-offset-4">
52                  <button type="submit" class="btn btn-primary">
53                      <i class="fa fa-btn fa-refresh"></i> Reset Password
54                  </button>
55              </div>
56          </div>
57          {!! Form::close() !!}
58      </div>
59  </div>
60 </div>
61 </div>
62 </div>
63 @endsection
64
```

a. login.blade.php

```
perpusku_baru > resources > views > auth > 🗑 login.blade.php
 1  @extends('layouts.app')
 2
 3  @section('content')
 4      <div class="container">
 5          <div class="row">
 6              <div class="col-md-8 col-md-offset-2">
 7                  <div class="panel panel-default">
 8                      <div class="panel-heading">
 9                          | <h2 class="panel-title">Login</h2>
10                      </div>
11
12                  <div class="panel-body">
13                      {!! Form::open(['url' => '/login', 'class' => 'form-horizontal']) !!}
14                      {{ csrf_field() }}
15
16                      <div class="form-group{{ $errors->has('email') ? ' has-error' : '' }}>
17                          {!! Form::label('email', 'Alamat Email', ['class' => 'col-md-4 control'])
18
19                          <div class="col-md-6">
20                              {!! Form::email('email', null, ['class' => 'form-control']) !!}
21                              {!! $errors->first('email', '<p class="help-block">:message</p>')
22                          </div>
23                      </div>
24
25                      <div class="form-group{{ $errors->has('password') ? ' has-error' : '' }}>
26                          {!! Form::label('password', 'Password', ['class' => 'col-md-4 control'])
```

```
perpusku_baru > resources > views > auth > 🗑 login.blade.php
28          <div class="col-md-6">
29              {!! Form::password('password', ['class' => 'form-control']) !!}
30              {!! $errors->first('password', '<p class="help-block">:message</p>')
31          </div>
32      </div>
33
34      <div class="form-group">
35          <div class="col-md-6 col-md-offset-4">
36              <div class="checkbox">
37                  <label>
38                      {!! Form::checkbox('remember') !!} Ingat saya
39                  </label>
40              </div>
41          </div>
42      </div>
43
44      <div class="form-group">
45          <div class="col-md-8 col-md-offset-4">
46              <button type="submit" class="btn btn-primary">
47                  | <i class="fa fa-btn fa-sign-in"></i> Login
48              </button>
49
50              <a class="btn btn-link" href="{{ route('password.request') }}">
51                  | Lupa password
52              </a>
53          </div>
54      </div>
55      {!! Form::close() !!}
56  </div>
57      </div>
58  </div>
59      </div>
60  </div>
61  @endsection
62
```

a. register.blade.php

```
perpusku_baru > resources > views > auth > 📄 register.blade.php
1  @extends('layouts.app')
2
3  @section('content')
4  <div class="container">
5      <div class="row">
6          <div class="col-md-8 col-md-offset-2">
7              <div class="panel panel-default">
8                  <div class="panel-heading">
9                      <h2 class="panel-title">Register</h2>
10                 </div>
11
12                 <div class="panel-body">
13                     {!! Form::open(['url' => '/register', 'class' => 'form-horizontal']) !!}
14                     {{ csrf_field() }}
15
16                     <div class="form-group{{ $errors->has('name') ? ' has-error' : '' }}>
17                         {!! Form::label('name', 'Nama', ['class' => 'col-md-4 control-label']) !!}
18
19                         <div class="col-md-6">
20                             {!! Form::text('name', null, ['class' => 'form-control']) !!}
21                             {!! $errors->first('name', '<p class="help-block">:message</p>') !!}
22                         </div>
23                     </div>
24
25                     <div class="form-group{{ $errors->has('email') ? ' has-error' : '' }}>
26                         {!! Form::label('email', 'Alamat Email', ['class' => 'col-md-4 control-label']) !!}
27
28                         <div class="col-md-6">
29                             {!! Form::email('email', null, ['class' => 'form-control']) !!}
30                             {!! $errors->first('email', '<p class="help-block">:message</p>') !!}
31                         </div>
32                     </div>
33
34                     <div class="form-group{{ $errors->has('password') ? ' has-error' : '' }}>
35                         {!! Form::label('password', 'Password', ['class' => 'col-md-4 control-label']) !!}
36
37                         <div class="col-md-6">
38                             {!! Form::password('password', ['class' => 'form-control']) !!}
39                             {!! $errors->first('password', '<p class="help-block">:message</p>') !!}
40                         </div>
41                     </div>
42
43                     <div class="form-group">
44                         {!! Form::label('password_confirmation', 'Konfirmasi Password', ['class' => 'col-md-4 control-label']) !!}
45
46                         <div class="col-md-6">
47                             {!! Form::password('password_confirmation', ['class' => 'form-control']) !!}
48                             {!! $errors->first('password_confirmation', '<p class="help-block">:message</p>') !!}
49                         </div>
50                     </div>
51                 </div>
```

```
perpusku_baru > resources > views > auth > 📄 register.blade.php
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
```

```

perpusku_baru > resources > views > auth > 🐻 register.blade.php

51
52         <!--<div class="form-group{{ $errors->has('g-recaptcha-response') ? ' has
53             <div class="col-md-6 col-md-offset-4">
54                 {!! app('captcha')->display() !!}
55                 {!! $errors->first('g-recaptcha-response', '<p class="help-block"')
56             </div>!-->
57             <div class="form-group">
58                 <div class="col-md-6 col-md-offset-4">
59                     <button type="submit" class="btn btn-primary">
60                         <i class="fa fa-btn fa-user"></i> Daftar
61                     </button>
62                 </div>
63             </div>
64             {!! Form::close() !!}
65         </div>
66     </div>
67 </div>
68 </div>
69 </div>
70 @endsection
71

```

➤ Folder Authors

a. File “_form.blade.php”

⇒ Dimana file ini digunakan untuk form tempat penulis buku, baik itu untuk tambah ,ubah.

```

perpusku_baru > resources > views > authors > 🐻 _form.blade.php
1  <div class="form-group{{ $errors->has('name') ? ' has-error': '' }}>
2      {!! Form::label('name', 'Nama', ['class' => 'col-md-2 control-label']) !!}
3
4      <div class="col-md-4">
5          {!! Form::text('name', null, ['class' => 'form-control']) !!}
6          {!! $errors->first('name', '<p class="help-block">:message</p>') !!}
7      </div>
8  </div>
9
10 <div class="form-group">
11     <div class="col-md-4 col-md-offset-2">
12         {!! Form::submit('Simpan', ['class' => 'btn btn-primary']) !!}
13     </div>
14 </div>
15

```

b. File “create.blade.php”

⇒ Digunakan untuk jika ingin melakukan penambahan penulis buku

```
perpusku_baru > resources > views > authors > 📄 create.blade.php
1  @extends('layouts.app')
2
3  @section('content')
4  <div class="container">
5      <div class="row">
6          <div class="col-md-12">
7              <ul class="breadcrumb">
8                  <li><a style="color:teal;" href="{{ url('/home') }}>Home</a></li>
9                  <li><a style="color:blue;" href="{{ url('/admin/authors') }}>Penulis</a></li>
10                 <li style="color:skyblue;" class="active">Tambah Penulis</li>
11             </ul>
12             <div class="panel panel-default">
13                 <div class="panel-heading">
14                     <h2 style="color:skyblue;" class="panel-title">Tambah Penulis</h2>
15                 </div>
16
17                 <div class="panel-body">
18                     {!! Form::open(['url' => route('authors.store'), 'method' => 'post', 'class' => 'form-horizontal']) !!}
19                     @include('authors._form')
20                     {!! Form::close() !!}
21                 </div>
22             </div>
23         </div>
24     </div>
25 </div>
26 @endsection
```

c. File “edit.bladde.php”

Digunakan untuk tempat melakukan ubah atau edit data penulis,jika ada data penulis yang tidak tepat..

```
perpusku_baru > resources > views > authors > 📄 edit.blade.php
1  @extends('layouts.app')
2
3  @section('content')
4  <div class="container">
5      <div class="row">
6          <div class="col-md-12">
7              <ul class="breadcrumb">
8                  <li><a style="color:teal;" href="{{ url('/home') }}>Home</a></li>
9                  <li><a style="color:green;" href="{{ url('/admin/authors') }}>Penulis</a></li>
10                 <li style="color:blue;" class="active">Ubah Penulis</li>
11             </ul>
12             <div class="panel panel-default">
13                 <div class="panel-heading">
14                     <h2 style="color:blue;" class="panel-title">Ubah Penulis</h2>
15                 </div>
16
17                 <div class="panel-body">
18                     {!! Form::model($author, ['url' => route('authors.update', $author->id), 'method' => 'put', 'class' => 'form-horizontal']) !!}
19                     @include('authors._form')
20                     {!! Form::close() !!}
21                 </div>
22             </div>
23         </div>
24     </div>
25 </div>
26 @endsection
```

d. index.blade.php

⇒ digunakan sebagai tampilan utama pada data penulis

```
perpusku_baru > resources > views > authors > 📄 index.blade.php
1  @extends('layouts.app')
2
3  @section('content')
4  <div class="container">
5    <div class="row">
6      <div class="col-md-12">
7        <ul class="breadcrumb">
8          <li><a style="color:teal;" href="{{ url('/home') }}>Home</a></li>
9          <li style="color:green;" class="active">Penulis</li>
10         </ul>
11         <div class="panel panel-default">
12           <div class="panel-heading">
13             <h2 style="color:green;" class="panel-title">Penulis</h2>
14           </div>
15
16           <div class="panel-body">
17             <p><a class="btn btn-primary" href="{{ route('authors.create') }}>Tambah</a>
18             {!! $html->table(['class' => 'table table-striped']) !!}
19           </div>
20         </div>
21       </div>
22     </div>
23   </div>
24   @endsection
25   @section('scripts')
26     {!! $html->scripts() !!}

```

➤ Folder books

a. _form.blade.php

```
perpusku_baru > resources > views > books > 📄 _form.blade.php
1  <div class="form-group{{ $errors->has('title') ? ' has-error': '' }}>
2    {!! Form::label('title', 'Judul', ['class' => 'col-md-2 control-label']) !!}
3    <div class="col-md-4">
4      {!! Form::text('title', null, ['class' => 'form-control']) !!}
5      {!! $errors->first('title', '<p class="help-block">:message</p>') !!}
6    </div>
7  </div>
8
9  <div class="form-group{{ $errors->has('author_id') ? ' has-error': '' }}>
10   {!! Form::label('author_id', 'Penulis', ['class' => 'col-md-2 control-label']) !!}
11   <div class="col-md-4">
12     {!! Form::select('author_id', [''=>'']+App\Author::pluck('name', 'id')->all(), null, ['class' => 'form-control']) !!}
13     {!! $errors->first('author_id', '<p class="help-block">:message</p>') !!}
14   </div>
15 </div>
16
17 <div class="form-group{{ $errors->has('amount') ? ' has-error': '' }}>
18   {!! Form::label('amount', 'Jumlah', ['class' => 'col-md-2 control-label']) !!}
19   <div class="col-md-4">
20     {!! Form::number('amount', null, ['class' => 'form-control', 'min' => 1]) !!}
21     {!! $errors->first('amount', '<p class="help-block">:message</p>') !!}
22     @if (isset($book))
23       <p class="help-block">{{ $book->borrowed }} buku sedang dipinjam</p>
24     @endif
25   </div>
26 </div>
```

```
perpusku_baru > resources > views > books > _form.blade.php
25     </div>
26 </div>
27
28 <div class="form-group{{ $errors->has('cover') ? ' has-error' : '' }}">
29     {!! Form::label('cover', 'Cover', ['class' => 'col-md-2 control-label']) !!}
30     <div class="col-md-4">
31         {!! Form::file('cover') !!}
32         @if (isset($book) && $book->cover)
33             <p>
34                 {!! Html::image(asset('img/' . $book->cover), null, ['class' => 'img-rounded img-re
35             </p>
36         @endif
37        >{!! $errors->first('cover', '<p class="help-block">:message</p>') !!}
38     </div>
39 </div>
40
41 <div class="form-group">
42     <div class="col-md-4 col-md-offset-2">
43         {!! Form::submit('Simpan', ['class' => 'btn btn-primary']) !!}
44     </div>
45 </div>
```

b. Import.blade.php

⇒ Fitur ini berfungsi sebagai untuk import data buku ke pdf

```
perpusku_baru > resources > views > books > _import.blade.php
1  <div class="form-group{{ $errors->has('title') ? ' has-error' : '' }}">
2      {!! Form::label('template', 'Gunakan template terbaru', ['class' => 'col-md-2 control-label'])
3
4      <div class="col-md-4">
5          <a class="btn btn-success btn-xs" href="{{ route('template.books') }}><i class="fa fa-cl
6      </div>
7 </div>
8
9 <div class="form-group{{ $errors->has('excel') ? ' has-error' : '' }}">
10    {!! Form::label('excel', 'Pilih file', ['class' => 'col-md-2 control-label']) !!}
11
12    <div class="col-md-4">
13        {!! Form::file("excel") !!}
14        {!! $errors->first('excel', '<p class="help-block">:message</p>') !!}
15    </div>
16 </div>
17
18 <div class="form-group">
19     <div class="col-md-4 col-md-offset-2">
20         {!! Form::submit('Simpan', ['class' => 'btn btn-primary']) !!}
21     </div>
22 </div>
```

```

perpusku_baru > resources > views > books > 📚 _form.blade.php
27
28     <div class="form-group{{ $errors->has('cover') ? ' has-error' : '' }}>
29         {!! Form::label('cover', 'Cover', ['class' => 'col-md-2 control-label']) !!}
30         <div class="col-md-4">
31             {!! Form::file('cover') !!}
32             @if (isset($book) && $book->cover)
33                 <p>
34                     {!! Html::image(asset('img/' . $book->cover), null, ['class' => 'img-rounded img-responsive']) !!}
35                 </p>
36             @endif
37             {!! $errors->first('cover', '<p class="help-block">:message</p>') !!}
38         </div>
39     </div>
40
41     <div class="form-group">
42         <div class="col-md-4 col-md-offset-2">
43             {!! Form::submit('Simpan', ['class' => 'btn btn-primary']) !!}
44         </div>
45     </div>

```

c. Fitur create.blade.php yang digunakan untuk membuat buku baru

```

perpusku_baru > resources > views > books > 📚 create.blade.php
1     layouts.app')
2
3     content')
4     = "container"
5     lass="row">
6     iv class="col-md-12">
7         <ul class="breadcrumb">
8             <li><a style="color:teal;" href="{{ url('/home') }}>Home</a></li>
9             <li><a style="color:blue;" href="{{ url('/admin/books') }}>Buku</a></li>
10            <li class="active">Tambah Buku</li>
11        </ul>
12        <div class="panel panel-default">
13            <div class="panel-heading">
14                <h2 class="panel-title">Tambah Buku</h2>
15            </div>
16
17            <div class="panel-body">
18                <ul class="nav nav-tabs" role="tablist">
19                    <li role="presentation" class="active">
20                        <a href="#form" aria-controls="form" role="tab" data-toggle="tab">
21                            <i class="fa fa-pencil-square-o"></i> Isi Form
22                        </a>
23                    </li>
24                    <li role="presentation">
25                        <!--<a href="#upload" aria-controls="upload" role="tab" data-toggle="tab">!-->
26                        <!--<i class="fa fa-cloud-upload"></i> Upload Excel!-->

```

```

perpusku_baru > resources > views > books > 📚 create.blade.php
26                                <!--<i class="fa fa-cloud-upload"></i> Upload Excel!-->
27
28
29    </li>
30
31    <div class="tab-content">
32        <div role="tabpanel" class="tab-pane active" id="form">
33            {!! Form::open(['url' => route('books.store'), 'method' => 'post', 'files' => ''])
34
35            @include('books._form')
36            {!! Form::close() !!}
37        </div>
38
39        <div role="tabpanel" class="tab-pane" id="upload">
40            {!! Form::open(['url' => route('import.books'), 'method' => 'post', 'files' => ''])
41
42            @include('books._import')
43            {!! Form::close() !!}
44        </div>
45
46    </div>
47
48    div>
49

```

d. Fitur edit.blade.php

⇒ Fitur ini digunakan untuk edit data pada buku

```
perpusku_baru > resources > views > books > 📖 edit.blade.php
1  @extends('layouts.app')
2
3  @section('content')
4  <div class="container">
5      <div class="row">
6          <div class="col-md-12">
7              <ul class="breadcrumb">
8                  <li><a style="color:teal;" href="{{ url('/home') }}>Home</a></li>
9                  <li><a style="color:blue;" href="{{ url('/admin/books') }}>Buku</a></li>
10                 <li class="active">Ubah Buku</li>
11             </ul>
12
13             <div class="panel panel-default">
14                 <div class="panel-heading">
15                     <h2 class="panel-title">
16                         <h2 style="color:charcoal;" class="panel-title">Ubah Buku</h2>
17                     </h2>
18
19                     <div class="panel-body">
20                         {!! Form::model($book, ['url' => route('books.update', $book->id), 'method' => 'PUT', 'class' => 'form-horizontal']) !!}
21                         @include('books._form')
22                         {!! Form::close() !!}
23                     </div>
24                 </div>
25             </div>
26         </div>
```

b. export.blade.php

⇒ fitur ini digunakan untuk export data buku seperti judul , jumlah, penulis, stok

```
perpusku_baru > resources > views > books > 📁 export.blade.php
1  @extends('layouts.app')
2
3  @section('content')
4  <div class="container">
5      <div class="row">
6          <div class="col-md-12">
7              <ul class="breadcrumb">
8                  <li><a style="color:teal;" href="{{ url('/home') }}>Home</a></li>
9                  <li><a style="color:blue;" href="{{ url('/admin/books') }}>Buku</a></li>
10                 <li class="active">Export Buku</li>
11             </ul>
12             <div class="panel panel-default">
13                 <div class="panel-heading">
14                     <h2 class="panel-title">Export Buku</h2>
15                 </div>
16
17                 <div class="panel-body">
18                     {!! Form::open(['url' => route('export.books.post'), 'method' => 'post', 'class' => 'form-horizontal']) !!}
19
20                     <div class="form-group {!! $errors->has('author_id') ? 'has-error' : '' !!}">
21                         {!! Form::label('author_id', 'Penulis', ['class' => 'col-md-2 control-label']) !!}
22
23                         <div class="col-md-4">
24                             {!! Form::select('author_id[]', ['' => '']+App\Author::pluck('name', 'id')) !!}
25
26                         {!! $errors->first('author_id', '<p class="help-block">:message</p>') !!}
```

```
perpusku_baru > resources > views > books > 📄 export.blade.php
26
27
28
29
30     <!! $errors->first('author_id', '<p class="help-block">:message</p>') !!
28
29
30     <div class="form-group {!! $errors->has('type') ? 'has-error' : '' !!}">
31         {!! Form::label('type', 'Pilih Output', ['class' => 'col-md-2 control-lab
32
33             <div class="col-md-4 checkbox">
34                 <!--{!! Form::radio('type', 'xls', true) !!} Excel!-->
35                 {!! Form::radio('type', 'pdf') !!} PDF
36                 {!! $errors->first('type', '<p class="help-block">:message</p>') !!}
37             </div>
38         </div>
39
40         <div class="form-group">
41             <div class="col-md-4 col-md-offset-2">
42                 {!! Form::submit('Download', ['class' => 'btn btn-primary']) !!}
43             </div>
44         </div>
45
46         {!! Form::close() !!}
47     </div>
48
49     </div>
50 </div>
```

c. import-review.blade.php

⇒ digunakan untuk jika ingin yakin menghapus data buku

```
perpusku_baru > resources > views > books > import-review.blade.php
25 |     <tr>
26 |         <th>Judul</th>
27 |         <th>Penulis</th>
28 |         <th>Jumlah</th>
29 |         <th></th>
30 |     </tr>
31 | </thead>
32 | <tbody>
33 |     @foreach ($books as $book)
34 |     <tr>
35 |         <td>{{ $book->title }}</td>
36 |         <td>{{ $book->author->name }}</td>
37 |         <td>{{ $book->amount }}</td>
38 |         <td>
39 |             {!! Form::open(['url' => route('books.destroy', $book->id
40 |             'id'           => 'form-' . $book->id, 'method' => 'd
41 |             'data-confirm' => 'Yakin menghapus ' . $book->title
42 |             'class'        => 'form-inline js-review-delete'])}
43 |
44 |             {!! Form::submit('Hapus', ['class' => 'btn btn-xs btn
45 |             {!! Form::close() !!}]
46 |         </td>
47 |     </tr>
48 |     @endforeach
49 | </tbody>
50 | </table>
```

```
perpusku_baru > resources > views > books > import-review.blade.php
45 |         <td>
46 |             <{!! Form::close() !!}>
47 |         </td>
48 |     @endforeach
49 | </tbody>
50 | </table>
51 | <p>
52 |     <a class="btn btn-success" href="{{ url('/admin/books') }}>Selesai</a>
53 | </p>
54 | </div>
55 | </div>
56 | </div>
57 | </div>
58 | </div>
59 | @endsection
60 |
```

d. index.blade.php

⇒ Digunakan untuk tampilan awal jika ingin mengimport data buku ke pdf

```
perpusku_baru > resources > views > books > index.blade.php
 1  @extends('layouts.app')
 2
 3  @section('content')
 4  <div class="container">
 5    <div class="row">
 6      <div class="col-md-12">
 7        <ul class="breadcrumb">
 8          <li><a style="color:teal;" href="{{ url('/home') }}>Home</a></li>
 9          <li style="color:blue;" class="active">Buku</li>
10        </ul>
11        <div class="panel panel-default">
12          <div class="panel-heading">
13            <h2 style="color:blue;" class="panel-title">Buku</h2>
14          </div>
15
16          <div class="panel-body">
17            <p>
18              <a class="btn btn-primary" href="{{ url('/admin/books/create') }}>Tambah
19              <a style="color:charcoal;" class="btn btn-primary" href="{{ url('/admin/e
20              </p>
21
22              {!! $html->table(['class' => 'table table-striped']) !!}
23            </div>
24          </div>
25        </div>
26      </div>
27    </div>
28  @endsection
29
30  @section('scripts')
31    {!! $html->scripts() !!}
32  @endsection
33
```

➤ Folder dashboard

Didalam folder ini terdapat 2 file yaitu admin.blade.php dan member.blade.php dimana jika admin yang login, maka akan menuju ke admin.blade.php yang sebagai tampilan utama atau home khusus admin dan jika member login maka akan menuju ke member.blade.php sebagai tampilan utama atau home khusus member.

a. Admin.blade.php

```
perpusku_baru > resources > views > dashboard > admin.blade.php
1  @extends('layouts.app')
2
3  @section('content')
4  <div class="container">
5      <div class="row">
6          <div class="col-md-12">
7              <div class="panel panel-default">
8                  <div class="panel-heading">
9                      <h2 style="color:teal;" class="panel-title"><b>Home</b></h2>
10                     </div>
11
12                     <div class="panel-body"><div class="jumbotron jumbotron-fluid" style="background-color: #f8d7da; color: black; text-align: center;">
13                         <div style="color:charcoal;">
14                             <div style="text-align: center;">Selamat Datang di Menu Administrasi
15                                 |   |   |   |   E- Perpus Ceria.</h1>
16                                 |   |   |   |
17                                 |   |   |   <br>
18                                 |   |   |   <hr>
19                                 |   |   |   <br>
20                                 |   |   |   <br>
21                                 |   |   |   <br>
22                                 |   |   |   <br>
23                                 |   |   |   <h4 style="color:gray;">Statistik Penulis</h4>
24                                 |   |   |   <canvas id="chartPenulis" width="450" height="150"> </canvas>
25
26             </div>
27         </div>
28     </div>
```

```
perpusku_baru > resources > views > dashboard > admin.blade.php

24 |         |     </div>
25 |         |     </div>
26 |     </div>
27 |     </div>
28 </div>
29 @endsection
30
31 @section('scripts')
32     <script src="js/Chart.min.js"></script>
33     <script>
34         var data = {
35             labels: {!! json_encode($authors) !!},
36             datasets: [
37                 label: 'Jumlah buku',
38                 data: {!! json_encode($books) !!},
39                 backgroundColor: "rgba(151,187,205,0.5)",
40                 borderColor: "rgba(151,187,205,0.8)",
41             ]
42         };
43
44         var options = {
45             scales: {
46                 yAxes: [
47                     ticks: {
48                         beginAtZero: true,
49                         stepSize: 1
50                     }
51                 ]
52             }
53         };
54
55         var chart = new Chart(document.getElementById("chart"), {
56             type: "bar",
57             data: data,
58             options: options
59         });
60     </script>
61 
```

```
perpusku_baru > resources > views > dashboard > admin.blade.php
 49           stepSize: 1
 50       }
 51   }]
 52 }
 53 };
 54
 55 var ctx = document.getElementById("chartPenulis").getContext("2d");
 56
 57 var authorChart = new Chart(ctx, {
 58     type: 'bar',
 59     data: data,
 60     options: options
 61 });
 62 </script>
 63 @endsection
```

b. member.blade.php

```
perpusku_baru > resources > views > dashboard > 📚 member.blade.php
26 @foreach ($borrowLogs as $borrowLog)
27     <li>
28         {!! Form::open(['url' => route('member.books.return', $borrowLog->id)])
29             !!>
30             {{ $borrowLog->book->title }}
31             {!! Form::submit('Kembalikan', ['class' => 'button']) !!}
32         <!! Form::close() !!}
33     </li>
34 @endforeach
35 </ul>
36 </td>
37 </tr>
38 </tbody>
39 </table>
40 </div>
41 </div>
42 </div>
43 </div>
44 </div>
45 </div>
46 @endsection
47
```

➤ Folder datatable

Dimana folder ini terdapat file tipe “**.blade.php**” yaitu _action.blade.php yang berfungsi sebagai action untuk hapus atau delete maupun ubah atau edit.

```
perpusku_baru > resources > datatable > 🗑_action.blade.php
1  {!! Form::model($model, ['url' => $form_url, 'method' => 'delete', 'class' => 'form-inline js-form-delete'])
2      <a href="{{ $edit_url }}>Ubah</a>
3      {!! Form::submit('Hapus', ['class' => 'btn btn-xs btn-danger']) !!}
4  <!! Form::close() !!}
5
```

➤ Folder Errors

Dimana folder ini terdapat file tipe “**.blade.php**” yaitu 430.blade.php yang berfungsi sebagai action jika bagi yang tidak memiliki akses ke halaman tertentu.

```
perpusku_baru > resources > views > errors > 🛡 403.blade.php
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>No access</title>
5
6      <link href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300" rel="stylesheet">
7
8      <style>
9          html, body {
10              height: 100%;
11          }
12
13          body {
14              margin: 0;
15              padding: 0;
16              width: 100%;
17              color: #B0BEC5;
18              display: table;
19              font-weight: 300;
20              font-family: 'Source Sans Pro';
21          }
22
23          .container {
24              text-align: center;
25              display: table-cell;
26              vertical-align: middle;
```

```
perpusku_baru > resources > views > errors > 403.blade.php
23     .container {
24         text-align: center;
25         display: table-cell;
26         vertical-align: middle;
27     }
28
29     .content {
30         text-align: center;
31         display: inline-block;
32     }
33
34     .title {
35         font-size: 72px;
36         margin-bottom: 40px;
37     }
38 
```

</style>

```
39 </head>
40
41 <body>
42     <div class="container">
43         <div class="content">
44             <div class="title">:(</div>
45             <p>Maaf, Anda tidak memiliki akses untuk fitur ini. </p>
46             <p><a href="{{ url('/') }}>Kembali ke halaman awal</a></p>
47         </div>
48     </div>
```

➤ Folder layouts

Dimana pada folder layouts terdapat 2 file yaitu _flash.blade.php dan app.blade.php dimana yang berfungsi sebagai tampilan navbar

a. _flash.blade.php

```
perpusku_baru > resources > views > layouts > _flash.blade.php
1  @if (session()->has('flash_notification.message'))
2      <div class="container">
3          <div class="alert alert-{{ session()->get('flash_notification.level') }}">
4              <button type="button" class="close" data-dismiss="alert" aria-hidden="true">&times;</
5
6                  {!! session()->get('flash_notification.message') !!}
7              </div>
8          </div>
9      @endif
10
```

b. app.blade.php

```
perpusku_baru > resources > views > layouts > app.blade.php
 1  <!DOCTYPE html>
 2  <html lang="{{ config('app.locale') }}">
 3  <head>
 4      <meta charset="utf-8">
 5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
 6      <meta name="viewport" content="width=device-width, initial-scale=1">
 7      <title> E- Perpus Ceria</title>
 8      <!-- CSRF Token -->
 9      <meta name="csrf-token" content="{{ csrf_token() }}">
10
11      <title>{{ config('app.name', 'Laravel') }}</title>
12
13      <!-- Bootstrap core CSS, Bootswatch: Lumen -->
14      <link href="{{ asset('css/bootstrap.min.css') }}" rel="stylesheet">
15
16      <!-- Font-Awesome -->
17      <link href="{{ asset('css/font-awesome.min.css') }}" rel="stylesheet">
18
19      <!-- DataTable CSS -->
20      <link href="{{ asset('css/dataTables.bootstrap.min.css') }}" rel="stylesheet">
21
22      <!-- Selectize -->
23      <link href="{{ asset('css/selectize.css') }}" rel="stylesheet">
24      <link href="{{ asset('css/selectize.bootstrap3.css') }}" rel="stylesheet">
25
26      <!-- Styles -->
```

```
perpusku_baru > resources > views > layouts > app.blade.php
26      <!-- Styles -->
27      <link href="{{ asset('css/app.css') }}" rel="stylesheet">
28
29      <!-- Scripts -->
30      <script>
31          window.Laravel = {!! json_encode([
32              'csrfToken' => csrf_token(),
33          ]) !!};
34      </script>
35  </head>
36  <body>
37      <div id="app">
38          <nav class="navbar navbar-default navbar-static-top">
39              <div class="container">
40                  <div class="navbar-header">
41
42                      <!-- Collapsed Hamburger -->
43                      <button type="button" class="navbar-toggle collapsed" data-toggle="collapse">
44                          <span class="sr-only">Toggle Navigation</span>
45                          <span class="icon-bar"></span>
46                          <span class="icon-bar"></span>
47                          <span class="icon-bar"></span>
48                      </button>
49
50                      <!-- Branding Image -->
51                      <a style="color:blue;" class="navbar-brand" href="{{ url('/') }}>
```

```
perpusku_baru > resources > views > layouts > app.blade.php
50      <!-- Branding Image -->
51      <a style="color:blue;" class="navbar-brand" href="{{ url('/') }}"
52          | {{ config('app.name', 'Laravel') }}>
53      </a>
54  </div>
55  <font color="Navy">
56  <div class="collapse navbar-collapse" id="app-navbar-collapse">
57      <!-- Left Side Of Navbar -->
58      <ul class="nav navbar-nav">
59          @if (Auth::check())
60              {!! Html::smartNav(url('/home'), 'Home') !!}
61          @endif
62
63          @role('admin')
64              {!! Html::smartNav(route('authors.index'), 'Penulis') !!}
65              {!! Html::smartNav(route('books.index'), 'Buku') !!}
66              {!! Html::smartNav(route('members.index'), 'Members') !!}
67              {!! Html::smartNav(route('statistics.index'), 'Peminjaman') !!}
68          @endrole
69
70          @if (auth()->check())
71              {!! Html::smartNav(url('/settings/profile'), 'Profile') !!}
72          @endif
73      </ul>
74
```

```
perpusku_baru > resources > views > layouts > app.blade.php
75      <!-- Right Side Of Navbar -->
76      <ul class="nav navbar-nav navbar-right">
77          <!-- Authentication Links -->
78          @if (Auth::guest())
79              <li><a style="color:skyblue;" href="{{ route('login') }}">Login</a></li>
80              <li><a style="color:skyblue;" href="{{ route('register') }}">Daftar</a>
81          @else
82              <li class="dropdown">
83                  <a href="#" class="dropdown-toggle" data-toggle="dropdown" role="button"
84                      | {{ Auth::user()->name }} <span class="caret"></span>
85                  </a>
86
87                  <ul class="dropdown-menu" role="menu">
88                      <li><a style="color:skyblue;" href="{{ url('settings/password') }}>Change Password</a>
89                      <li>
90                          <a style="color:skyblue;" href="{{ route('logout') }}"
91                              | onclick="event.preventDefault();
92                              |         document.getElementById('logout-form').submit();>
93                          Logout
94
95                          <form id="logout-form" action="{{ route('logout') }}" method="POST">
96                              | {{ csrf_field() }}
97                          </form>
98                      </li>
99                  </ul>
100
```

```
perpusku_baru > resources > views > layouts > 📄 app.blade.php
100
101
102
103
104
105
106
107
108     @include('layouts._flash')
109     @yield('content')
110 
```

```
111
112     <!-- Bootstrap core JavaScript
113     ===== -->
114     <!-- Placed at the end of the document so the pages load faster -->
115     <script src="{{ asset('js/jquery.min.js') }}"></script>
116     <script src="{{ asset('js/bootstrap.min.js') }}"></script>
117
118     <!-- DataTable JS -->
119     <script src="{{ asset('js/jquery.dataTables.min.js') }}"></script>
120     <script src="{{ asset('js/dataTables.bootstrap.min.js') }}"></script>
121
122     <!-- Selectize -->
123     <script src="{{ asset('js/selectize.min.js') }}"></script>
124
```

➤ Folder members

Dimana dalam folder ini terdapat ada file _form.blade.php, create.blade.php, edit.blade.php, index.blade.php, show.blade.php yang mana pada form.blade.php adalah untuk membuat form memasukkan data member, sedangkan create.blade.php adalah untuk membuat atau menambahkan data member baru, serta edit.blade.php adalah untuk edit data member yang telah dimasukkan atau didaftarkan , dan show.blade.php untuk dashboard tampilan status buku yang dipinjam, tanggal pinjam, dan tanggal kembali beserta judul.

a. _form.blade.php

```
perpusku_baru > resources > views > members > 🐘 _form.blade.php
 1  <div class="form-group{{ $errors->has('name') ? ' has-error' : '' }}>
 2  |   {!! Form::label('name', 'Nama', ['class' => 'col-md-2 control-label']) !!}
 3
 4  |   <div class="col-md-4">
 5  |   |   {!! Form::text('name', null, ['class' => 'form-control']) !!}
 6  |   |   {!! $errors->first('name', '<p class="help-block">:message</p>') !!}
 7  |   </div>
 8  </div>
 9
10 <div class="form-group{{ $errors->has('email') ? ' has-error' : '' }}>
11 |   {!! Form::label('email', 'Email', ['class' => 'col-md-2 control-label']) !!}
12
13 |   <div class="col-md-4">
14 |   |   {!! Form::email('email', null, ['class' => 'form-control']) !!}
15 |   |   {!! $errors->first('email', '<p class="help-block">:message</p>') !!}
16 |   </div>
17 </div>
18
19 <div class="form-group">
20 |   <div class="col-md-4 col-md-offset-2">
21 |   |   {!! Form::submit('Simpan', ['class' => 'btn btn-primary']) !!}
22 |   </div>
23 </div>
24
```

b. Create.blade.php

```
perpusku_baru > resources > views > members > 🗄 create.blade.php
 1  @extends('layouts.app')
 2
 3  @section('content')
 4  <div class="container">
 5    <div class="row">
 6      <div class="col-md-12">
 7        <ul class="breadcrumb">
 8          <li><a style="color:teal;" href="{{ url('/home') }}>Home</a></li>
 9          <li><a style="color:coral;" href="{{ url('/admin/members') }}>Member</a></li>
10          <li class="active">Tambah Member</li>
11        </ul>
12        <div class="panel panel-default">
13          <div class="panel-heading">
14            <h2 style="color:coral;" class="panel-title">Tambah Member</h2>
15          </div>
16
17          <div class="panel-body">
18            {!! Form::open(['url' => route('members.store'), 'method' => 'post', 'files' => true]) !!}
19
20            @include('members._form')
21            {!! Form::close() !!}
22          </div>
23        </div>
24      </div>
25    </div>
26  </div>
```

c. edit.blade.php

```
perpusku_baru > resources > views > members > 🗄 edit.blade.php
 1  @extends('layouts.app')
 2
 3  @section('content')
 4  <div class="container">
 5    <div class="row">
 6      <div class="col-md-12">
 7        <ul class="breadcrumb">
 8          <li><a style="color:teal;" href="{{ url('/home') }}>Home</a></li>
 9          <li><a style="color:coral;" href="{{ url('/admin/members') }}>Member</a></li>
10          <li style="color:indigo;" class="active">Ubah Member</li>
11        </ul>
12        <div class="panel panel-default">
13          <div class="panel-heading">
14            <h2 style="color:indigo;" class="panel-title">Ubah Member</h2>
15          </div>
16
17          <div class="panel-body">
18            {!! Form::model($member, ['url' => route('members.update', $member->id), 'method' => 'put']) !!}
19
20            @include('members._form')
21            {!! Form::close() !!}
22          </div>
23        </div>
24      </div>
25    </div>
26  </div>
```

d. index.blade.php

```
perpusku_baru > resources > views > members > index.blade.php
 1  @extends('layouts.app')
 2
 3  @section('content')
 4  <div class="container">
 5    <div class="row">
 6      <div class="col-md-12">
 7        <ul class="breadcrumb">
 8          <li><a style="color:teal;" href="{{ url('/home') }}>Dashboard</a></li>
 9          <li style="color:coral;" class="active">Member</li>
10        </ul>
11        <div class="panel panel-default">
12          <div class="panel-heading">
13            <h2 style="color:coral;" class="panel-title">Member</h2>
14          </div>
15
16          <div class="panel-body">
17            <p><a style="color:brown;" class="btn btn-primary" href="{{ url('/admin/member') }}>Detail</a></p>
18
19            {!! $html->table(['class' => 'table table-striped']) !!}
20
21          </div>
22        </div>
23      </div>
24    </div>
25  @endsection
26
```

e. show.blade.php

```
perpusku_baru > resources > views > members > show.blade.php
 1  @extends('layouts.app')
 2
 3  @section('content')
 4  <div class="container">
 5    <div class="row">
 6      <div class="col-md-12">
 7        <ul class="breadcrumb">
 8          <li><a style="color:teal;" href="{{ url('/home') }}>Home</a></li>
 9          <li><a style="color:coral;" href="{{ url('/admin/members') }}>Member</a></li>
10          <li style="color:coral;" class="active">Detail {{ $member->name }}</li>
11        </ul>
12        <div class="panel panel-default">
13          <div class="panel-heading">
14            <h2 style="color:coral;" class="panel-title">Detail {{ $member->name }}</h2>
15          </div>
16
17          <div class="panel-body">
18            <p style="color:green;">Buku yang sedang dipinjam:</p>
19
20            <table class="table table-condensed table-striped">
21              <thead>
22                <tr>
23                  <td style="color:olive;">Judul</td>
24                  <td style="color:olive;">Tanggal Peminjaman</td>
25                </tr>
26              </thead>
```

```
perpusku_baru > resources > views > members > 📄 show.blade.php
25 |           </tr>
26 |       </thead>
27 |       <tbody>
28 |           @forelse ($member->borrowLogs()->borrowed()->get() as $log)
29 |               <tr>
30 |                   <td>{{ $log->book->title }}</td>
31 |                   <td>{{ $log->created_at }}</td>
32 |               </tr>
33 |           @empty
34 |               <tr>
35 |                   <td style="color:olive;" colspan="2">Tidak ada data</td>
36 |               </tr>
37 |           @endforelse
38 |       </tbody>
39 |   </table>
40 |
41 |   <p style="color:green;">Buku yang telah dikembalikan:</p>
42 |
43 |   <table class="table table-condensed table-striped">
44 |       <thead>
45 |           <tr>
46 |               <td style="color:olive;">Judul</td>
47 |               <td style="color:olive;">Tanggal Kembali</td>
48 |           </tr>
49 |       </thead>
50 |       <tbody>
```

```
perpusku_baru > resources > views > members > 📄 show.blade.php
51 |           @forelse ($member->borrowLogs()->returned()->get() as $log)
52 |               <tr>
53 |                   <td>{{ $log->book->title }}</td>
54 |                   <td>{{ $log->updated_at }}</td>
55 |               </tr>
56 |           @empty
57 |               <tr>
58 |                   <td style="color:olive;" colspan="2">Tidak ada data</td>
59 |               </tr>
60 |           @endforelse
61 |       </tbody>
62 |   </table>
63 |   </div>
64 | </div>
65 | </div>
66 | </div>
67 | </div>
68 | @endsection
69 | 
```

➤ Folder pdf

Dimana pada folder pdf ini terdapat file books.blade.php yang digunakan untuk action download jika setelah klik export ke pdf .

a. books.blade.php

```
perpusku_baru > resources > views > pdf > 📄 books.blade.php
 1  <!DOCTYPE html>
 2  <html>
 3  <head>
 4      <title>Data Buku Larapus</title>
 5      <style>
 6
 7  /* -----
 8
 9   Hartija Css Print Framework
10   * Version: 1.0
11
12  ----- */
13
14 body {
15 width:100% !important;
16 margin:0 !important;
17 padding:0 !important;
18 line-height: 1.45;
19 font-family: Garamond,"Times New Roman", serif;
20 color: #000;
21 background: none;
22 font-size: 14pt; }
23
24 /* Headings */
25 h1,h2,h3,h4,h5,h6 { page-break-after:avoid; }
26 h1{font-size:19pt;}
```

```
perpusku_baru > resources > views > pdf > 📄 books.blade.php
28 h3{font-size:15pt;}
29 h4,h5,h6{font-size:14pt;}
30
31
32 p, h2, h3 { orphans: 3; widows: 3; }
33
34 code { font: 12pt Courier, monospace; }
35 blockquote { margin: 1.2em; padding: 1em; font-size: 12pt; }
36 hr { background-color: #ccc; }
37
38 /* Images */
39 img { float: left; margin: 1em 1.5em 1.5em 0; max-width: 100% !important; }
40 a img { border: none; }
41
42 /* Links */
43 a:link, a:visited { background: transparent; font-weight: 700; text-decoration: underline; color:#000; }
44 a:link[href^="http://"]::after, a[href^="http://"]::visited::after { content: "(" attr(href) ")"; }
45
46 abbr[title]:after { content: "(" attr(title) ")"; }
47
48 /* Don't show linked images */
49 a[href^="http://"] {color:#000; }
50 a[href$=".jpg"]::after, a[href$=".jpeg"]::after, a[href$=".gif"]::after, a[href$=".png"]::after { con
```

```
perpusku_baru > resources > views > pdf > 📖 books.blade.php
51
52     /* Don't show links that are fragment identifiers, or use the `javascript:` pseudo protocol .. ta
53     a[href^="#"]::after, a[href^="javascript:"]::after {content: "";}
54
55     /* Table */
56     table { margin: 1px; text-align:left; }
57     th { border-bottom: 1px solid #333; font-weight: bold; }
58     td { border-bottom: 1px solid #333; }
59     th,td { padding: 4px 10px 4px 0; }
60     tfoot { font-style: italic; }
61     caption { background: #fff; margin-bottom:2em; text-align:left; }
62     thead {display: table-header-group;}
63     img, tr {page-break-inside: avoid;}
64
65     /* Hide various parts from the site
66     #header, #footer, #navigation, #rightSideBar, #leftSideBar
67     {display:none;}
68     */
69     </style>
70 </head>
71
72 <body>
73     <h1>Data Buku E-Perpus Ceria</h1>
74     <hr>
75     <table>
```

```
perpusku_baru > resources > views > pdf > 📖 books.blade.php
76         <thead>
77             <tr>
78                 <td background-color: #0000FF;>Judul</td>
79                 <td background-color: #0000FF;>Jumlah</td>
80                 <td background-color: #0000FF;>Stok</td>
81                 <td background-color: #0000FF;>Penulis</td>
82             </tr>
83         </thead>
84         <tbody>
85             @foreach ($books as $book)
86                 <tr>
87                     <td> {{ $book->title }} </td>
88                     <td> {{ $book->amount }} </td>
89                     <td> {{ $book->stock }} </td>
90                     <td> {{ $book->author->name }} </td>
91                 </tr>
92             @endforeach
93         </tbody>
94     </table>
95 </body>
96
97 </html>
98
```

➤ Folder Setting

Dimana pada folder ini terdapat 3 file yang digunakan untuk setting atau ubah password yaitu edit-password.blade.php, edit-profil.blade.php, profi.blade.php

a. edit-password.blade.php

```
1 @extends('layouts.app')
2
3 @section('content')
4 <div class="container">
5     <div class="row">
6         <div class="col-md-12">
7             <ul class="breadcrumb">
8                 <li><a style="color:teal;" href="{{ url('/home') }}><h3>Home</a></h3></li>
9                 <li style="color:fuchsia;" class="active"><b>Ubah Password</b></li>
10            </ul>
11            <div class="panel panel-default">
12                <div class="panel-heading">
13                    <h2 style="color:fuchsia;" class="panel-title">Ubah Password</h2>
14                </div>
15
16                <div class="panel-body">
17                    {!! Form::open(['url' => url('/settings/password'), 'method' => 'post', 'clas
18
19                        <div class="form-group{{ $errors->has('password') ? ' has-error': '' }}">
20                            {!! Form::label('password', 'Password lama', ['class' => 'col-md-4 contro
21
22                                <div class="col-md-6">
23                                    {!! Form::password('password', ['class' => 'form-control']) !!}
24                                    {!! $errors->first('password', '<p class="help-block">:message</p>')
25                                </div>
```

perpusku_baru > resources > views > settings > edit-password.blade.php

```
26          </div>
27
28          <div class="form-group{{ $errors->has('new_password') ? ' has-error' : '' }}>
29              {!! Form::label('new_password', 'Password baru', ['class' => 'col-md-4 co
30
31                  <div class="col-md-6">
32                      {!! Form::password('new_password', ['class' => 'form-control']) !!}
33                      {!! $errors->first('new_password', '<p class="help-block">:message</p
34                  </div>
35          </div>
36
37          <div class="form-group{{ $errors->has('new_password_confirmation') ? ' has-er
38              {!! Form::label('new_password_confirmation', 'Konfirmasi password baru',
39
40                  <div class="col-md-6">
41                      {!! Form::password('new_password_confirmation', ['class' => 'form-con
42                      {!! $errors->first('new_password_confirmation', '<p class="help-block
43                  </div>
44          </div>
45
46          <div class="form-group">
47              <div class="col-md-6 col-md-offset-4">
48                  {!! Form::submit('Simpan', ['class' => 'btn btn-primary']) !!}
49              </div>
50          </div>
51          {!! Form::close() !!}
52      
```

```

perpusku_baru > resources > views > settings > edit-password.blade.php
26     </div>
27
28     <div class="form-group{{ $errors->has('new_password') ? ' has-error' : '' }}>
29         {!! Form::label('new_password', 'Password baru', ['class' => 'col-md-4 co
30
31             <div class="col-md-6">
32                 {!! Form::password('new_password', ['class' => 'form-control']) !!}
33                 {!! $errors->first('new_password', '<p class="help-block">:message</p
34             </div>
35         </div>
36
37         <div class="form-group{{ $errors->has('new_password_confirmation') ? ' has-er
38             !! Form::label('new_password_confirmation', 'Konfirmasi password baru',
39
40                 <div class="col-md-6">
41                     {!! Form::password('new_password_confirmation', ['class' => 'form-con
42                     {!! $errors->first('new_password_confirmation', '<p class="help-block
43                 </div>
44         </div>
45
46         <div class="form-group">
47             <div class="col-md-6 col-md-offset-4">
48                 {!! Form::submit('Simpan', ['class' => 'btn btn-primary']) !!}
49             </div>
50         </div>

```

```

perpusku_baru > resources > views > settings > edit-password.blade.php
50     </div>
51     {!! Form::close() !!}
52 </div>
53 </div>
54 </div>
55 </div>
56 </div>
57 @endsection

```

b. edit-profil.blade.php

```

perpusku_baru > resources > views > settings > edit-profile.blade.php
1     @extends('layouts.app')
2
3     @section('content')
4     <div class="container">
5         <div class="row">
6             <div class="col-md-12">
7                 <ul class="breadcrumb">
8                     <li><a href="{{ url('/home') }}>Home</a></li>
9                     <li style="color:turquoise;" class="active"><b>Ubah Profil</b></li></ul>
10
11             <div class="panel panel-default">
12                 <div class="panel-heading">
13                     <h2 style="color:turquoise;" class="panel-title"><b>Ubah Profil</b></h2></div>
14
15             <div class="panel-body">
16                 {!! Form::model(auth()->user(), ['url' => url('/settings/profile'), 'method'
17
18
19                     <div class="form-group{{ $errors->has('name') ? ' has-error' : '' }}>
20                         {!! Form::label('name', 'Nama', ['class' => 'col-md-4 control-label']) !!}
21
22                         <div class="col-md-6">
23                             {!! Form::text('name', null, ['class' => 'form-control']) !!}
24                             {!! $errors->first('name', '<p class="help-block">:message</p>') !!}
25                         </div>
26                     </div>

```

```

perpusku_baru > resources > views > settings > edit-profile.blade.php
26     </div>
27
28     <div class="form-group{{ $errors->has('email') ? ' has-error' : '' }}>
29         {!! Form::label('email', 'Email', ['class' => 'col-md-4 control-label'])}
30
31         <div class="col-md-6">
32             {!! Form::email('email', null, ['class' => 'form-control']) !!}
33             {!! $errors->first('email', '<p class="help-block">:message</p>') !!}
34         </div>
35     </div>
36
37     <div class="form-group">
38         <div class="col-md-6 col-md-offset-4">
39             {!! Form::submit('Simpan', ['class' => 'btn btn-primary']) !!}
40         </div>
41     </div>
42     {!! Form::close() !!}
43 </div>
44 </div>
45 </div>
46 </div>
47 @endsection

```

c. Profil.blade.php

```

perpusku_baru > resources > views > settings > profile.blade.php
1     @extends('layouts.app')
2
3     @section('content')
4     <div class="container">
5         <div class="row">
6             <div class="col-md-12">
7                 <ul class="breadcrumb">
8                     <li><a style="color:teal;" href="{{ url('/home') }}><h3>Home</h3></a></li>
9                     <li style="color:violet;" class="active"><b>Profil</b></li>
10                </ul>
11                <div class="panel panel-default">
12                    <div class="panel-heading">
13                        <h1 style="color:violet;" class="panel-title"><b>Profil</b></h1>
14                    </div>
15
16                    <div class="panel-body">
17                        <table class="table">
18                            <tbody>
19                                <tr>
20                                    <td style="color:DeepSkyBlue;" class="text-muted">Nama</td>
21                                    <td >{{ auth()->user()->name }}</td>
22                                </tr>
23                                <tr>
24                                    <td style="color:DeepSkyBlue;" class="text-muted">Email</td>
25                                    <td >{{ auth()->user()->email }}</td>
26                                </tr>

```

```
perpusku_baru > resources > views > settings > profile.blade.php

26
27
28    </tr>
29    <tr>
30        <td style="color:DeepSkyBlue;" class="text-muted">Login terakhir</td>
31        <td>{{ auth()->user()->last_login }}</td>
32    </tr>
33 </tbody>
34 </table>
35 <a class="btn btn-primary" href="{{ url('/settings/profile/edit') }}>Ubah</a>
36
37 </div>
38 </div>
39 @endsection
```

➤ Folder statistic

```
perpusku_baru > resources > views > statistics > 📈 index.blade.php
1  @extends('layouts.app')
2
3  @section('content')
4      <div class="container">
5          <div class="row">
6              <div class="col-md-12">
7                  <ul class="breadcrumb">
8                      <li><a style="color:teal;" href="{{ url('/home') }}>Home</a></li>
9                      <li class="active">Data Peminjaman</li>
10                 </ul>
11                 <div class="panel panel-default">
12                     <div class="panel-heading">
13                         <h2 style="color:blue;" class="panel-title">Data Peminjaman</h2>
14                     </div>
15
16                     <div class="panel-body">
17                         {!! $html->table(['class'=>'table table-striped']) !!}
18                     </div>
19                 </div>
20             </div>
21         </div>
22     </div>
23     @endsection
24
25     @section('scripts')
26         {!! $html->scripts() !!}
27     </section>
```

```
perpusku_baru > resources > views > statistics > 📈 index.blade.php
27  <script>
28      $(function() {
29          $('\
30              <div id="filter_status" class="dataTables_length">\
31                  <label>Status \
32                      <select name="filter_status" aria-controls="filter_status" \
33                          class="form-control input-sm" style="width: auto;">\
34                          <option value="all" selected="selected">Semua</option>\
35                          <option value="returned">Sudah Dikembalikan</option>\
36                          <option value="notReturned">Belum Dikembalikan</option>\
37                      </select>\
38                  </label>\
39              </div>\
40          ).insertAfter('.dataTables_length');
41
42          $('#dataTableBuilder').on('preXhr.dt', function(e, settings, data) {
43              data.status = $('select[name="filter_status"]').val();
44          });
45
46          $('select[name="filter_status"]').change(function() {
47              window.LaravelDataTables['dataTableBuilder'].ajax.reload();
48          });
49      });
50  </script>
51 @endsection
```

❖ home.blade.php

```
perpusku_baru > resources > views > 🏠 home.blade.php
1  @extends('layouts.app')
2
3  @section('content')
4  <div class="container">
5      <div class="row">
6          <div class="col-md-8 col-md-offset-2">
7              <div class="panel panel-default">
8                  <div style="color:teal;" class="panel-heading">Home</div>
9
10                 <div style="color:grey;" class="panel-body">
11                     Anda berhasil login!
12                 </div>
13             </div>
14         </div>
15     </div>
16  @endsection
17
18
```

❖ Welcome.blade.php

```
perpusku_baru > resources > views > 📄 welcome.blade.php
 1  @extends('layouts.app')
 2
 3  @section("content")
 4  <div class="container">
 5    <div class="row">
 6      <div class="col-md-8 col-md-offset-2">
 7        <div class="panel panel-default">
 8          <div class="panel-heading">Hallo!</div>
 9          <div style="text-align: center;">SELAMAT DATANG di E- PERPU CERIA UNIVERSITAS MA
10          </div>
11        </div>
12      </div>
13    </div>
14  </div>
15  @endsection
16
```

3. Selanjutnya Membuat Controller

a. BooksController.php

```
perpus_Dika > app > Http > Controllers > 📄 BooksController.php
 1  <?php
 2
 3  namespace App\Http\Controllers;
 4
 5  use App\Book;
 6  use Illuminate\Http\Request;
 7
 8  class BooksController extends Controller
 9  {
10
11    public function __construct()
12    {
13      $this->middleware('auth');
14    }
15
16    /**
17     * Display a listing of the resource.
18     *
19     * @return \Illuminate\Http\Response
20     */
21    public function index()
22    {
23      $books = Book::all();
24      return view('books.index', compact('books'));
25    }
26
```

```
perpus_Dika > app > Http > Controllers > BooksController.php
27  /**
28   * Show the form for creating a new resource.
29   *
30   * @return \Illuminate\Http\Response
31   */
32  public function create()
33  {
34      return view('books.create');
35  }
36
37 /**
38  * Store a newly created resource in storage.
39  *
40  * @param \Illuminate\Http\Request $request
41  * @return \Illuminate\Http\Response
42  */
43  public function store(Request $request)
44  {
45      $this->validate($request, [
46          'name' => 'required|unique:books|max:30',
47          'description' => 'required|max:100',
48          'penerbit' => 'required|max:30',
49          'tanggal_terbit' => 'required|date',
50          'stock' => 'required|integer'
51      ]);
52
53      $book = new Book;
54      $book->name = $request->name;
55      $book->description = $request->description;
56      $book->penerbit = $request->penerbit;
57      $book->tanggal_terbit = $request->tanggal_terbit;
58      $book->stock = $request->stock;
59      $book->save();
60
61      return redirect()->route('books.index');
62  }
63
64 /**
65  * Show the form for editing the specified resource.
66  *
67  * @param \App\Book $book
68  * @return \Illuminate\Http\Response
69  */
70  public function edit(Book $book)
71  {
72      return view('books.edit', compact('book'));
73  }
74
75
```

```

76     /**
77      * Update the specified resource in storage.
78      *
79      * @param \Illuminate\Http\Request $request
80      * @param \App\Book $book
81      * @return \Illuminate\Http\Response
82      */
83     public function update(Request $request, Book $book)
84     {
85         $this->validate($request, [
86             'name' => 'required|max:30',
87             'description' => 'required|max:100',
88             'penerbit' => 'required|max:30',
89             'tanggal_terbit' => 'required|date',
90             'stock' => 'required|integer'
91         ]);
92
93         $book->update($request->all());
94         return redirect()->route('books.index');
95     }
96
97     /**
98      * Remove the specified resource from storage.
99      *
100     * @param \App\Book $book
101     * @return \Illuminate\Http\Response
102     */
103    public function destroy(Book $book)
104    {
105        $book->delete();
106        return redirect()->route('books.index');
107    }
108 }
109

```

b. HomeController.php

```

perpusku_baru > app > Http > Controllers > BooksController.php
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6  use Illuminate\Support\Facades\Session;
7  use Illuminate\Support\Facades\File;
8  use Illuminate\Database\Eloquent\ModelNotFoundException;
9  use Illuminate\Support\Facades\Auth;
10 use App\BorrowLog;
11 use App\Exceptions\BookException;
12 use App\Book;
13 use App\Http\Requests\StoreBookRequest;
14 use App\Http\Requests\UpdateBookRequest;
15 use Yajra\Datatables\Html\Builder;
16 use Yajra\Datatables\Datatables;
17 use Excel;
18 use PDF;
19 use App\Author;
20 use Validator;
21
22 class BooksController extends Controller
23 {
24     /**
25      * Display a listing of the resource.
26      *

```

```
25     * Display a listing of the resource.
26     *
27     * @return \Illuminate\Http\Response
28     */
29    public function index(Request $request, Builder $htmlBuilder)
30    {
31        if ($request->ajax()) {
32            $books = Book::with('author');
33
34            return Datatables::of($books)
35                ->addColumn('action', function($book) {
36                    return view('datatable._action', [
37                        'model'          => $book,
38                        'form_url'       => route('books.destroy', $book->id),
39                        'edit_url'        => route('books.edit', $book->id),
40                        'confirm_message' => 'Yakin mau menghapus ' . $book->title . '?'
41                    ]);
42                })->make(true);
43        }
44
45        $html = $htmlBuilder
46            ->addColumn(['data' => 'title', 'name' => 'title', 'title' => 'Judul'])
47            ->addColumn(['data' => 'amount', 'name' => 'amount', 'title' => 'Jumlah'])
48            ->addColumn(['data' => 'author.name', 'name' => 'author.name', 'title' => 'Penulis'])
49            ->addColumn(['data' => 'action', 'name' => 'action', 'title' => '', 'orderable' => fa
50
50
51        return view('books.index')->with(compact('html'));
52    }
53
54    /**
55     * Show the form for creating a new resource.
56     *
57     * @return \Illuminate\Http\Response
58     */
59    public function create()
60    {
61        return view('books.create');
62    }
63
64    /**
65     * Store a newly created resource in storage.
66     *
67     * @param \Illuminate\Http\Request $request
68     * @return \Illuminate\Http\Response
69     */
70    public function store(StoreBookRequest $request)
71    {
72
73        $book = Book::create($request->except('cover'));
74    }
```

```
75     // Isi field cover jika ada cover yang diupload
76     if ($request->hasFile('cover')) {
77         // Mengambil file yang diupload
78         $uploaded_cover = $request->file('cover');
79
80         // Mengambil extension file
81         $extension = $uploaded_cover->getClientOriginalExtension();
82
83         // Membuat nama file random berikut extension
84         $filename = md5(time()) . '.' . $extension;
85
86         // Menyimpan cover ke folder public/img
87         $destinationPath = public_path() . DIRECTORY_SEPARATOR . 'img';
88         $uploaded_cover->move($destinationPath, $filename);
89
90         // Mengisi field cover di book dengan filename yang harus dibuat
91         $book->cover = $filename;
92         $book->save();
93     }
94
95     Session::flash("flash_notification", [
96         "level"    => "success",
97         "message"  => "Berhasil menyimpan $book->title"
98     ]);

```

```
100    return redirect()->route('books.index');
101 }
102
103 /**
104 * Display the specified resource.
105 *
106 * @param int $id
107 * @return \Illuminate\Http\Response
108 */
109 public function show($id)
110 {
111     //
112 }
113
114 /**
115 * Show the form for editing the specified resource.
116 *
117 * @param int $id
118 * @return \Illuminate\Http\Response
119 */
120 public function edit($id)
121 {
122     $book = Book::find($id);
123
124     return view('books.edit')->with(compact('book'));
125 }
```

```
125     }
126
127     /**
128      * Update the specified resource in storage.
129      *
130      * @param \Illuminate\Http\Request $request
131      * @param int $id
132      * @return \Illuminate\Http\Response
133     */
134     public function update(UpdateBookRequest $request, $id)
135     {
136
137         $book = Book::find($id);
138         if (!$book->update($request->all())) return redirect()->back();
139
140         // Isi field cover jika ada cover yang diupload
141         if ($request->hasFile('cover')) {
142             // Mengambil file yang diupload berikut eksistensinya
143             $filename = null;
144             $uploaded_cover = $request->file('cover');
145
146             // Mengambil extension file
147             $extension = $uploaded_cover->getClientOriginalExtension();
148
149             // Membuat nama file random berikut extension
150             $filename = md5(time()) . '.' . $extension;
```

```
152         // Menyimpan cover ke folder public/img
153         $destinationPath = public_path() . DIRECTORY_SEPARATOR . 'img';
154         $uploaded_cover->move($destinationPath, $filename);
155
156         // Hapus cover lama, jika ada
157         if ($book->cover) {
158             $old_cover = $book->cover;
159             $filepath = public_path() . DIRECTORY_SEPARATOR . 'img' . DIRECTORY_SEPARATOR . $old_cover;
160
161             try {
162                 File::delete($filepath);
163             } catch (FileNotFoundException $e) {
164                 // File sudah dihapus / tidak ada
165             }
166         }
167
168         // Ganti field cover dengan cover yang baru
169         $book->cover = $filename;
170         $book->save();
171     }
172
173     Session::flash("flash_notification", [
174         "level" => "success",
175         "message" => "Berhasil menyimpan $book->title"
176     ]);
177
```

```
177     return redirect()->route('books.index');
178 }
179 
180 /**
181 * Remove the specified resource from storage.
182 *
183 * @param int $id
184 * @return \Illuminate\Http\Response
185 */
186 public function destroy(Request $request, $id)
187 {
188     $book = Book::find($id);
189     $cover = $book->cover;
190 
191     if (!$book->delete()) return redirect()->back();
192 
193     // Handle hapus buku via Ajax
194     if ($request->ajax()) return response()->json(['id' => $id]);
195 
196     // Hapus cover lama jika ada
197     if ($cover) {
198         $old_cover = $book->cover;
199         $filepath = public_path() . DIRECTORY_SEPARATOR . 'img' . DIRECTORY_SEPARATOR . $book
200 
201     try {
```

```
202         try {
203             File::delete($filepath);
204         } catch (FileNotFoundException $e) {
205             // File sudah dihapus / tidak ada
206         }
207     }
208 
209     Session::flash("flash_notification", [
210         "level"      => "success",
211         "message"   => "Buku berhasil dihapus"
212     ]);
213 
214     return redirect()->route('books.index');
215 }
216 
217 public function borrow($id)
218 {
219     try {
220         $book = Book::findOrFail($id);
221         Auth::user()->borrow($book);
222 
223         Session::flash("flash_notification", [
224             "level"      => "success",
225             "message"   => "Berhasil meminjam $book->title"
226         ]);
227     }
```

```
228     } catch (BookException $e) {
229         Session::flash("flash_notification", [
230             "level"      => "danger",
231             "message"   => $e->getMessage()
232         ]);
233     } catch (ModelNotFoundException $e) {
234         Session::flash("flash_notification", [
235             "level"      => "danger",
236             "message"   => "Buku tidak ditemukan."
237         ]);
238     }
239
240     return redirect('/');
241 }
242
243 public function returnBack($book_id)
244 {
245     $borrowLog = BorrowLog::where('user_id', Auth::user()->id)
246         ->where('book_id', $book_id)
247         ->where('is_returned', 0)
248         ->first();
249
250     if ($borrowLog) {
251         $borrowLog->is_returned = true;
252         $borrowLog->save();
253     }

```

```
255     Session::flash("flash_notification", [
256         "level"      => "success",
257         "message"   => "Berhasil mengembalikan " . $borrowLog->book->title
258     ]);
259
260     return redirect('/home');
261 }
262
263 public function export() {
264     return view('books.export');
265 }
266
267
268 public function exportPost(Request $request) {
269     // Validasi
270     $this->validate($request, [
271         'author_id' => 'required',
272         'type' => 'required|in:pdf,xls'
273     ], [
274         'author_id.required' => 'Anda belum memilih penulis. Pilih minimal 1 penulis.'
275     ]);
276
277     $books = Book::whereIn('id', $request->get('author_id'))->get();
278
279     $handler = 'export' . ucfirst($request->get('type'));
280

```

```
280
281     return $this->$handler($books);
282 }
283
284 private function exportXls($books)
285 {
286     Excel::create('Data Buku E-Perpus', function($excel) use ($books) {
287         // Set Property
288         $excel->setTitle('Data Buku E-Perpus')->setCreator('Dika Ayu Wijayanti');
289
290         $excel->sheet('Data Buku', function($sheet) use ($books) {
291             $row = 1;
292             $sheet->row($row, [
293                 'Judul',
294                 'Jumlah',
295                 'Stok',
296                 'Penulis'
297             ]);
298             foreach ($books as $book) {
299                 $sheet->row(++$row, [
300                     $book->title,
301                     $book->amount,
302                     $book->stock,
303                     $book->author->name
304                 ]);
305             }
306         });
307     }
308 }
309
310 public function exportPdf($books) {
311     $pdf = PDF::loadview('pdf.books', compact('books'));
312
313     return $pdf->download('books.pdf');
314 }
315
316 public function generateExcelTemplate() {
317     Excel::create('Template Import Buku', function($excel) {
318         // Set the properties
319         $excel->setTitle('Template Import Buku')
320             ->setCreator('E-Perpus')
321             ->setCompany('E-Perpus')
322             ->setDescription('Template import buku untuk E-Perpus');
323
324         $excel->sheet('Data Buku', function($sheet) {
325             $row = 1;
326             $sheet->row($row, [
327                 'judul',
328                 'penulis',
329                 'jumlah'
330             ]);
331         });
332     })->export('xlsx');
333 }
```

```
335     public function importExcel(Request $request) {
336         // Validasi untuk memastikan file yang diupload adalah excel
337         $this->validate($request, ['excel' => 'required|mimes:xls,xlsx']);
338
339         // Ambil file yang baru diupload
340         $excel = $request->file('excel');
341
342         // Baca sheet pertama
343         $excels = Excel::selectSheetsByIndex(0)->load($excel, function($reader) {
344             // Options, jika ada
345         })->get();
346
347         // Rule untuk validasi setiap row pada file excel
348         $rowRules = [
349             'judul'      => 'required',
350             'penulis'    => 'required',
351             'jumlah'    => 'required'
352         ];
353
354         // Catat semua id buku baru
355         // ID ini kita butuhkan untuk menghitung total buku yang berhasil diimport
356         $books_id = [];
357
358         // Looping setiap baris, mulai dari baris ke 2 (karena baris ke 1 adalah nama kolom)
359         foreach ($excels as $row) {
360
361             // Membuat validasi untuk row di excel
362             // Disini kita ubah baris yang sedang diproses menjadi array
363             $validator = Validator::make($row->toArray(), $rowRules);
364
365             // Skip baris ini jika tidak valid, langsung ke baris selanjutnya
366             if ($validator->fails()) continue;
367
368             // Syntax dibawah dieksekusi jika baris excel ini valid
369
370             // Cek apakah Penulis sudah terdaftar di database
371             $author = Author::where('name', $row['penulis'])->first();
372
373             // Buat penulis jika belum ada
374             if (!$author) {
375                 $author = Author::create(['name' => $row['penulis']]);
376             }
377
378             // Buat buku baru
379             $book = Book::create([
380                 'title'      => $row['judul'],
381                 'author_id'  => $author->id,
382                 'amount'     => $row['jumlah']
383             ]);
384
385             // Tambahkan id buku baru ke array
386             $books_id[] = $book->id;
387
388         }
389
390         // Setelah selesai looping, hitung total buku yang berhasil diimport
391         $totalBooks = count($books_id);
392
393         // Simpan data ke database
394         $books = Book::insert($books_id);
395
396         // Jika berhasil, kirim pesan sukses
397         if ($books) {
398             return response()->json(['status' => 'success', 'message' => 'Buku berhasil diimport!']);
399         } else {
400             return response()->json(['status' => 'error', 'message' => 'Terjadi kesalahan saat mencoba menyimpan data ke database']);
401         }
402     }
403 }
```

```

383
384     // Catat id dari buku yang baru dibuat
385     array_push($books_id, $book->id);
386 }
387
388 // Ambil semua buku yang baru dibuat
389 $books = Book::whereIn('id', $books_id)->get();
390
391 // Redirect ke form jika tidak ada buku yang berhasil diimport
392 if ($books->count() == 0) {
393     Session::flash("flash_notification", [
394         "level" => "danger",
395         "message" => "Tidak ada buku yang berhasil diimport."
396     ]);
397     return redirect()->back();
398 }
399
400 // Set feedback
401 Session::flash("flash_notification", [
402     "level" => "success",
403     "message" => "Berhasil mengimport " . $books->count() . " buku."
404 ]);
405
406 // Tampilkan index buku
407 return view('books.import-review')->with(compact('books'));
408 }

```

c. GuestController.php

```

perpusku_baru > app > Http > Controllers > 🏠 GuestController.php
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6  use App\Http\Requests;
7  use Yajra\Datatables\Html\Builder;
8  use Yajra\Datatables\Datatables;
9  use App\Book;
10 use Laratrust\LaratrustFacade as Laratrust;
11
12 class GuestController extends Controller
13 {
14
15     public function index(Request $request, Builder $htmlBuilder)
16     {
17         if ($request->ajax()) {
18             $books = Book::with('author');
19
20             return Datatables::of($books)
21                 ->addColumn('stock', function($book) {
22                     return $book->stock;
23                 })
24                 ->addColumn('action', function($book) {
25                     if ($Laratrust::hasRole('admin')) return '';
26                     return '<a class="btn btn-xs btn-primary" href="'.route('guest.books.borrow',

```

```
27     }->make(true);
28 }
29
30     $html = $htmlBuilder
31         ->addColumn(['data' => 'title', 'name' => 'title', 'title' => 'Judul'])
32         ->addColumn(['data' => 'stock', 'name' => 'stock', 'title' => 'Stock', 'orderable'
33         ->addColumn(['data' => 'author.name', 'name' => 'author.name', 'title' => 'Penulis',
34         ->addColumn(['data' => 'action', 'name' => 'action', 'title' => '', 'orderable' =
35
36     return view('guest.index')->with(compact('html'));
37 }
38 }
39 }
```

d. MemberController.php

```
perpusku_baru > app > Http > Controllers > MembersController.php
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6  use App\Role;
7  use App\User;
8  use Yajra\Datatables\Html\Builder;
9  use Yajra\Datatables\Facades\Datatables;
10 use App\Http\Requests\StoreMemberRequest;
11 use Illuminate\Support\Facades\Session;
12 use Illuminate\Support\Facades\Mail;
13 use App\Http\Requests\UpdateMemberRequest;
14
15 class MembersController extends Controller
16 {
17     /**
18      * Display a listing of the resource.
19      *
20      * @return \Illuminate\Http\Response
21      */
22     public function index(Request $request, Builder $htmlBuilder)
23     {
24         if ($request->ajax()) {
25             $members = Role::where('name', 'member')->first()->users;
```

```
26
27     return Datatables::of($members)
28         ->addColumn('name', function($member) {
29             return '<a href="'.route('members.show', $member->id).'">'.$member->name.'</a>';
30         })
31         ->addColumn('action', function($member) {
32             return view('datatable._action', [
33                 'model'          => $member,
34                 'form_url'       => route('members.destroy', $member->id),
35                 'edit_url'        => route('members.edit', $member->id),
36                 'confirm_message' => 'Yakin mau menghapus ' . $member->name . '?'
37             ]);
38         })
39         ->rawColumns(['name', 'action'])->make(true);
40     }
41
42     $html = $htmlBuilder
43         ->addColumn(['data' => 'name', 'name' => 'name', 'title' => 'Nama'])
44         ->addColumn(['data' => 'email', 'name' => 'email', 'title' => 'Email'])
45         ->addColumn(['data' => 'action', 'name' => 'action', 'title' => '', 'orderable' => false]
46
47     return view('members.index', compact('html'));
48 }
49
```

```
50 /**
51  * Show the form for creating a new resource.
52  *
53  * @return \Illuminate\Http\Response
54  */
55 public function create()
56 {
57     return view('members.create');
58 }
59
60 /**
61  * Store a newly created resource in storage.
62  *
63  * @param \Illuminate\Http\Request $request
64  * @return \Illuminate\Http\Response
65  */
66 public function store(StoreMemberRequest $request)
67 {
68     $password = str_random(6);
69     $data = $request->all();
70     $data['password'] = bcrypt($password);
71     // bypass verifikasi
72     $data['is_verified'] = 1;
73
74     $member = User::create($data);
```

```
76     // Set role
77     $memberRole = Role::where('name', 'member')->first();
78     $member->attachRole($memberRole);
79
80     // Kirim email
81     Mail::send('auth.emails.invite', compact('member', 'password'), function($m) use ($member)
82         $m->to($member->email, $member->name)->subject('Anda telah didaftarkan di Larapus!');
83     });
84
85     Session::flash("flash_notification", [
86         "level"      => "success",
87         "message"    => "Berhasil menyimpan member dengan email " . "<strong>" . $data['email']
88     ]);
89
90     return redirect()->route('members.index');
91 }
92
93 /**
94 * Display the specified resource.
95 *
96 * @param int $id
97 * @return \Illuminate\Http\Response
98 */
99 public function show($id)
100 {
101     $member = User::find($id);
102
103     return view('members.show', compact('member'));
104 }
105
106 /**
107 * Show the form for editing the specified resource.
108 *
109 * @param int $id
110 * @return \Illuminate\Http\Response
111 */
112 public function edit($id)
113 {
114     $member = User::find($id);
115     return view('members.edit')->with(compact('member'));
116 }
117
118 /**
119 * Update the specified resource in storage.
120 *
121 * @param \Illuminate\Http\Request $request
122 * @param int $id
123 * @return \Illuminate\Http\Response
124 */
125 public function update(UpdateMemberRequest $request, $id)
126 {
127     $member = User::find($id);
128     $member->update($request->only(['name', 'email']));

```

```

128     $member->update($request->only('name', 'email'));
129
130     Session::flash("flash_notification", [
131         "level"      => "success",
132         "message"   => "Berhasil menyimpan $member->name"
133     ]);
134
135     return redirect()->route('members.index');
136 }
137
138 /**
139 * Remove the specified resource from storage.
140 *
141 * @param int $id
142 * @return \Illuminate\Http\Response
143 */
144 public function destroy($id)
145 {
146     $member = User::find($id);
147
148     if ($member->hasRole('member')) {
149         $member->delete();
150
151         Session::flash("flash_notification", [
152             "level"      => "success",
153             "message"   => "Member berhasil dihapus"
154         ]);
155     }
156
157     return redirect()->route('members.index');
158 }
159 }
160

```

e. settingController.php

```

perpusku_baru > app > Http > Controllers > 🏷 SettingsController.php
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6  use Illuminate\Support\Facades\Auth;
7  use Illuminate\Support\Facades\Session;
8
9  class SettingsController extends Controller
10 {
11
12     public function __construct()
13     {
14         $this->middleware('auth');
15     }
16
17     public function profile()
18     {
19         return view('settings.profile');
20     }
21
22     public function editProfile()
23     {
24         return view('settings.edit-profile');
25     }
26

```

```
25     }
26
27     public function updateProfile(Request $request)
28     {
29         $user = Auth::user();
30
31         $this->validate($request, [
32             'name' => 'required',
33             'email' => 'required|unique:users,email,' . $user->id
34         ]);
35
36         $user->name = $request->get('name');
37         $user->email = $request->get('email');
38         $user->save();
39
40         Session::flash("flash_notification", [
41             "level" => "success",
42             "message" => "Profile berhasil diubah"
43         ]);
44
45         return redirect('settings/profile');
46     }
47
48     public function editPassword()
49     {
50         return view('settings.edit-password');
51     }
52
53     public function updatePassword(Request $request)
54     {
55         $user = Auth::user();
56
57         $this->validate($request, [
58             'password' => 'required|passcheck:' . $user->password,
59             'new_password' => 'required|confirmed|min:6',
60         ], [
61             'password.passcheck' => 'Password lama tidak sesuai'
62         ]);
63
64         $user->password = bcrypt($request->get('new_password'));
65         $user->save();
66
67         Session::flash("flash_notification", [
68             "level" => "success",
69             "message" => "Password berhasil diubah"
70         ]);
71
72         return redirect('settings/password');
73     }
74 }
```

f. StatisticController.php

```
perpusku_baru > app > Http > Controllers > StatisticsController.php
 1  <?php
 2
 3  namespace App\Http\Controllers;
 4
 5  use Illuminate\Http\Request;
 6  use App\Http\Requests;
 7  use App\BorrowLog;
 8  use Yajra\Datatables\Html\Builder;
 9  use Yajra\Datatables\Facades\Datatables;
10
11 class StatisticsController extends Controller
12 {
13     public function index(Request $request, Builder $htmlBuilder)
14     {
15         if ($request->ajax()) {
16             $stats = BorrowLog::with('book', 'user');
17
18             if ($request->get('status') == 'returned') $stats->returned();
19             if ($request->get('status') == 'notReturned') $stats->borrowed();
20
21             return Datatables::of($stats)
22                 ->addColumn('returned_at', function($stat){
23                     if ($stat->is_returned) {
24                         return $stat->updated_at;
25                     }
26                     return "Masih dipinjam";
27
28     })
29
30     $html = $htmlBuilder
31         ->addColumn(['data' => 'book.title', 'name' => 'book.title', 'title' => 'Judul'])
32         ->addColumn(['data' => 'user.name', 'name' => 'user.name', 'title' => 'Peminjam'])
33         ->addColumn(['data' => 'created_at', 'name' => 'created_at', 'title' => 'Tanggal Pinj')
34         ->addColumn(['data' => 'returned_at', 'name' => 'returned_at', 'title' => 'Tanggal Kembal
35
36     return view('statistics.index')->with(compact('html'));
37
38 }
39 }
```

➤ Folder Auth

a. ForgotPasswordController.php

```
perpusku_baru > app > Http > Controllers > Auth > 📄 ForgotPasswordController.php
 1  <?php
 2
 3  namespace App\Http\Controllers\Auth;
 4
 5  use App\Http\Controllers\Controller;
 6  use Illuminate\Foundation\Auth\SendsPasswordResetEmails;
 7
 8  class ForgotPasswordController extends Controller
 9  {
10      /*
11      |--------------------------------------------------------------------------
12      | Password Reset Controller
13      |--------------------------------------------------------------------------
14
15      |
16      | This controller is responsible for handling password reset emails and
17      | includes a trait which assists in sending these notifications from
18      | your application to your users. Feel free to explore this trait.
19      |
20      */
21
22      use SendsPasswordResetEmails;
23
24      /**
25       * Create a new controller instance.
26       *
27       * @return void
28
29       */
30
31       public function __construct()
32     {
33         $this->middleware('guest');
34     }
35
36 }
```

b. LoginController.php

```
perpusku_baru > app > Http > Controllers > Auth > 📄 LoginController.php
 1  <?php
 2
 3  namespace App\Http\Controllers\Auth;
 4
 5  use App\Http\Controllers\Controller;
 6  use Illuminate\Foundation\Auth\AuthenticatesUsers;
 7
 8  class LoginController extends Controller
 9  {
10      /*
11      |--------------------------------------------------------------------------
12      | Login Controller
13      |--------------------------------------------------------------------------
14
15      |
16      | This controller handles authenticating users for the application and
17      | redirecting them to your home screen. The controller uses a trait
18      | to conveniently provide its functionality to your applications.
19      |
20      */
21
22      use AuthenticatesUsers;
23
24      /**
25       * Where to redirect users after login.
26       *
27       * @var string
28   }
```

```

27     */
28     protected $redirectTo = '/home';
29
30     /**
31      * Create a new controller instance.
32      *
33      * @return void
34      */
35     public function __construct()
36     {
37         $this->middleware('guest', ['except' => 'logout']);
38         $this->middleware('user-should-verified');
39     }
40 }
41

```

c. RegisterController.php

```

perpusku_baru > app > Http > Controllers > Auth > RegisterController.php
1  <?php
2
3  namespace App\Http\Controllers\Auth;
4
5  use App\User;
6  use App\Role;
7  use App\Http\Controllers\Controller;
8  use Illuminate\Support\Facades\Validator;
9  use Illuminate\Foundation\Auth\RegistersUsers;
10 use Illuminate\Http\Request;
11 use Illuminate\Support\Facades\Auth;
12 use Illuminate\Support\Facades\Session;
13
14 class RegisterController extends Controller
15 {
16     /*
17     |-----
18     | Register Controller
19     |-----
20     |
21     | This controller handles the registration of new users as well as their
22     | validation and creation. By default this controller uses a trait to
23     | provide this functionality without requiring any additional code.
24     |
25     */
26
27     use RegistersUsers;
28
29     /**
30      * Where to redirect users after registration.
31      *
32      * @var string
33      */
34     protected $redirectTo = '/home';
35
36     /**
37      * Create a new controller instance.
38      *
39      * @return void
40      */
41     public function __construct()
42     {
43         $this->middleware('guest');
44         $this->middleware('user-should-verified');
45     }
46
47     /**
48      * Get a validator for an incoming registration request.
49      *
50      * @param array $data
51      * @return \Illuminate\Contracts\Validation\Validator
52      */

```

```
53     protected function validator(array $data)
54     {
55         return Validator::make($data, [
56             'name' => 'required|max:255',
57             'email' => 'required|email|max:255|unique:users',
58             'password' => 'required|min:6|confirmed',
59         ]);
60     }
61
62     /**
63      * Create a new user instance after a valid registration.
64      *
65      * @param  array  $data
66      * @return User
67      */
68     protected function create(array $data)
69     {
70         $user = User::create([
71             'name' => $data['name'],
72             'email' => $data['email'],
73             'password' => bcrypt($data['password']),
74         ]);
75         $memberRole = Role::where('name', 'member')->first();
76         $user->attachRole($memberRole);
77         $user->sendVerification();
78         return $user;

```

```
79     }
80
81     public function verify(Request $request, $token) {
82         $email = $request->get('email');
83         $user = User::where('verification_token', $token)->where('email', $email)->first();
84
85         if ($user) {
86             $user->verify();
87
88             Session::flash("flash_notification", [
89                 "level"    => "success",
90                 "message"  => "Berhasil melakukan verifikasi."
91             ]);
92             Auth::login($user);
93         }
94         return redirect('/');
95     }
96
97     public function sendVerification(Request $request)
98     {
99         $user = User::where('email', $request->get('email'))->first();
100
101        if ($user && !$user->is_verified) {
102            $user->sendVerification();
103
104        }
105    }

```

```
103     |         Session::flash("flash_notification", [
104     |             "level"      => "success",
105     |             "message"   => "Silahkan klik pada link aktivasi yang telah kami kirim."
106     |         ]);
107     |
108     }
109
110     return redirect('/login');
111 }
112
113 }
```

d. ResetPasswordController.php

```
perpusku_baru > app > Http > Controllers > Auth > 🗑 ResetPasswordController.php
1  <?php
2
3  namespace App\Http\Controllers\Auth;
4
5  use App\Http\Controllers\Controller;
6  use Illuminate\Foundation\Auth\ResetsPasswords;
7
8  class ResetPasswordController extends Controller
9  {
10     /*
11     |--------------------------------------------------------------------------
12     | Password Reset Controller
13     |--------------------------------------------------------------------------
14     |
15     | This controller is responsible for handling password reset requests
16     | and uses a simple trait to include this behavior. You're free to
17     | explore this trait and override any methods you wish to tweak.
18     |
19     */
20
21     use ResetsPasswords;
22
23     /**
24     * Where to redirect users after resetting their password.
25     *
26     * @var string
27
28     protected $redirectTo = '/home';
29
30     /**
31     * Create a new controller instance.
32     *
33     * @return void
34     */
35     public function __construct()
36     {
37         $this->middleware('guest');
38     }
39 }
```

4. Langkah selanjutnya yaitu membuat Model dengan cara “ **php artisan make:model Author**”, dimana pada projek ini ada 6 model sebagai berikut :

a. Authors

```
C:\xampp\htdocs\perpusku_baru>php artisan make:model Author
Model created successfully.
```

```
perpusku_baru > app > Author.php
1  ?php
2
3  namespace App;
4
5  use Illuminate\Database\Eloquent\Model;
6  use Illuminate\Support\Facades\Session;
7
8  class Author extends Model
9  {
10     protected $fillable = ['name'];
11
12     public static function boot()
13     {
14         parent::boot();
15
16         self::deleting(function($author) {
17             // Untuk Mengecek apakah penulis masih punya buku
18             if ($author->books->count() > 0) {
19                 // Menyiapkan pesan error
20                 $html = 'Penulis tidak bisa dihapus karena masih memiliki buku : ';
21                 $html .= '<ul>';
22                 foreach ($author->books as $book) {
23                     $html .= "<li>$book->title</li>";
24                 }
25                 $html .= '</ul>';
26
27             Session::flash("flash_notification", [
28                 "level" => "danger",
29                 "message" => $html
30             ]);
31             // Untuk Membatalkan proses penghapusan
32             return false;
33         });
34     });
35
36     public function books()
37     {
38         return $this->hasMany('App\Book');
39     }
40 }
41
42 }
```

b. Book.php

```
C:\xampp\htdocs\perpusku_baru>php artisan make:model Book
Model created successfully.
```

```
perpusku_baru > app > Book.php
1  <?php
2
3  namespace App;
4
5  use Illuminate\Database\Eloquent\Model;
6  use Illuminate\Support\Facades\Session;
7
8  class Book extends Model
9  {
10     protected $fillable = ['title', 'author_id', 'amount'];
11
12     public static function boot()
13     {
14         parent::boot();
15
16         self::updating(function ($book)
17         {
18             if ($book->amount < $book->borrowed) {
19                 Session::flash("flash_notification", [
20                     "level"      => "danger",
21                     "message"   => "Jumlah buku $book->title harus >= " . $book->borrowed
22                 ]);
23
24             return false;
25         }
26     });
27
28     self::deleting(function ($book)
29     {
30         if ($book->borrowLogs()->count() > 0) {
31             Session::flash("flash_notification", [
32                 "level"      => "danger",
33                 "message"   => "Buku $book->title sudah pernah dipinjam."
34             ]);
35             return false;
36         }
37     });
38 }
39
40 public function getBorrowedAttribute()
41 {
42     return $this->borrowLogs()->borrowed()->count();
43 }
44
45 public function author()
46 {
47     return $this->belongsTo('App\Author');
48 }
49
50 public function borrowLogs()
51 {
```

```
55
56     public function getStockAttribute()
57     {
58         $borrowed = $this->borrowLogs()->borrowed()->count();
59         $stock = $this->amount - $borrowed;
60
61         return $stock;
62     }
63 }
64
```

c. BorrowLog.php

```
perpusku_baru > app > BorrowLog.php
1  <?php
2
3  namespace App;
4
5  use Illuminate\Database\Eloquent\Model;
6
7  class BorrowLog extends Model
8  {
9      protected $casts = ['is_returned' => 'boolean'];
10
11     protected $fillable = ['book_id', 'user_id', 'is_returned'];
12
13     public function book()
14     {
15         return $this->belongsTo('App\Book');
16     }
17
18     public function user()
19     {
20         return $this->belongsTo('App\User');
21     }
22
23     public function scopeReturned($query)
24     {
25         return $query->where('is_returned', 1);
26     }
27
28     public function scopeBorrowed($query)
29     {
30         return $query->where('is_returned', 0);
31     }
32 }
33
```

d. User.php

```
C:\xampp\htdocs\perpusku_baru>php artisan make:model User
Model created successfully.
```

```
perpusku_baru > app > User.php
1  <?php
2
3  namespace App;
4
5  use Illuminate\Notifications\Notifiable;
6  use Illuminate\Foundation\Auth\User as Authenticatable;
7  use Illuminate\Support\Facades\Mail;
8  use Laratrust\Traits\LaratrustUserTrait;
9  use App\Book;
10 use App\BorrowLog;
11 use App\Exceptions\BookException;
12
13 class User extends Authenticatable
14 {
15     use LaratrustUserTrait;
16     use Notifiable;
17
18     protected $casts = [
19         'is_verified' => 'boolean',
20     ];
21
22     /**
23      * The attributes that are mass assignable.
24      *
25      * @var array
26     */
```

```
perpusku_baru > app > User.php
26     */
27     protected $fillable = [
28         'name', 'email', 'password',
29     ];
30
31     /**
32      * The attributes that should be hidden for arrays.
33      *
34      * @var array
35     */
36     protected $hidden = [
37         'password', 'remember_token',
38     ];
39
40     public function borrow(Book $book)
41     {
42         // Cek apakah masih ada stock buku
43         if ($book->stock < 1) {
44             throw new BookException("Buku $book->title sedang tidak tersedia.");
45         }
46
47         // Cek apakah buku ini sedang dipinjam oleh user
48         if ($this->borrowLogs()->where('book_id', $book->id)->where('is_returned', 0)->count() >
49             throw new BookException("Buku $book->title sedang Anda pinjam.");
50     }
```

```

52     $borrowLog = BorrowLog::create(['user_id' => $this->id, 'book_id' => $book->id]);
53     return $borrowLog;
54 }
55
56 public function borrowLogs()
57 {
58     return $this->hasMany('App\BorrowLog');
59 }
60
61 public function verify()
62 {
63     $this->is_verified = 1;
64     $this->verification_token = null;
65     $this->save();
66 }
67
68 public function generateVerificationToken()
69 {
70     $token = $this->verification_token;
71
72     if (!$token) {
73         $token = str_random(40);
74         $this->verification_token = $token;
75         $this->save();
76     }
77
78     return $token;
79 }
80
81 public function sendVerification()
82 {
83     $token = $this->generateVerificationToken();
84     $user = $this;
85
86     Mail::send('auth.emails.verification', compact('user', 'token'), function ($m) use ($user)
87     {
88         $m->to($user->email, $user->name)->subject('Verifikasi Akun E-Perpus');
89     });
90 }
91

```

5. Selanjutnya,membuat migrate , dimana **Migration** adalah sebuah fasilitas dilaravel digunakan untuk mempermudah kita ketika ada perubahan dalam database. **Schema Builder** digunakan untuk membuat sebuah skema database. Dengan menggunakan migrations dan schema builder kita tidak perlu repot – repot membuka phpmyadmin , ataupun aplikasi Sql lainnya untuk membuat database. Dengan migrations dan schema builder juga akan lebih mudah ketika kita membuat project besar dan dengan developer yang berbeda-beda jadinya si developer ini tidak perlu import/export database tapi langsung saja menggunakan migration.

Setiap file Migration terdapat *prefix timestamp* yang berfungsi untuk mengurutkan Migration yang telah dibuat. Sedangkan code yang terdapat didalam file Migration baru tersebut secara otomatis akan membuat dua buah *method* yang masih kosong yakni **up** dan **down** , dimana bahwa **Method up** : akan berisi perintah untuk memanipulasi *table*, meliputi: membuat mengatur

atau mengubah *field* yang terdapat di dalam *table*, sedangkan **Method down** bertugas sebaliknya.

- Untuk membuat file migrasi, jalankan perintah berikut pada cmd. Jangan lupa untuk masuk ke folder project aplikasi laravel berada dan lakukan pada membuat migration lainnya.

```
C:\xampp\htdocs\perpusku_baru>php artisan make:migration create_books_table
```

Berikut hasil file migration tersebut(Namun untuk nama- nama migration ini seharusnya 2020-12-20_05241_create_books_table.php tetapi karena saat saya jalankan terus eror akhirnya saya samakan dengan pada create user dan password reset dan setelah itu baru berhasil saya migrate:

- a. 2014_10_12_000000_create_users_table.php

```
1  <?php
2
3  use Illuminate\Support\Facades\Schema;
4  use Illuminate\Database\Schema\Blueprint;
5  use Illuminate\Database\Migrations\Migration;
6
7  class CreateUsersTable extends Migration
8  {
9      /**
10      * Run the migrations.
11      *
12      * @return void
13      */
14      public function up()
15      {
16          Schema::create('users', function (Blueprint $table) {
17              $table->increments('id');
18              $table->string('name');
19              $table->string('email')->unique();
20              $table->string('password');
21              $table->rememberToken();
22              $table->timestamps();
23          });
24      }
25
26      /**
27      * Reverse the migrations.
28      *
29      * @return void
30      */
31      public function down()
32      {
33          Schema::dropIfExists('users');
34      }
35  }
```

b. 2014_10_12_100000_create_password_resets_table.php

```
1  <?php
2
3  use Illuminate\Support\Facades\Schema;
4  use Illuminate\Database\Schema\Blueprint;
5  use Illuminate\Database\Migrations\Migration;
6
7  class CreatePasswordResetsTable extends Migration
8  {
9      /**
10      * Run the migrations.
11      *
12      * @return void
13      */
14      public function up()
15      {
16          Schema::create('password_resets', function (Blueprint $table) {
17              $table->string('email')->index();
18              $table->string('token')->index();
19              $table->timestamp('created_at')->nullable();
20          });
21      }
22
23      /**
24      * Reverse the migrations.
25      *
26      * @return void
27      */
28      public function down()
29      {
30          Schema::dropIfExists('password_resets');
31      }
32  }
```

c. 2017_03_05_133706_laratrust_setup_tables.php

```
1  <?php
2  use Illuminate\Database\Migrations\Migration;
3  use Illuminate\Database\Schema\Blueprint;
4
5  class LaratrustSetupTables extends Migration
6  {
7      /**
8      * Run the migrations.
9      *
10     * @return void
11     */
12     public function up()
13     {
14         // Create table for storing roles
15         Schema::create('roles', function (Blueprint $table) {
16             $table->increments('id');
17             $table->string('name')->unique();
18             $table->string('display_name')->nullable();
19             $table->string('description')->nullable();
20             $table->timestamps();
21         });
22
23         // Create table for associating roles to users (Many-to-Many)
24         Schema::create('role_user', function (Blueprint $table) {
25             $table->integer('user_id')->unsigned();
26             $table->integer('role_id')->unsigned();
27         });
28     }
29
30     /**
31     * Reverse the migrations.
32     *
33     * @return void
34     */
35     public function down()
36     {
37         Schema::dropIfExists('roles');
38         Schema::dropIfExists('role_user');
39     }
40 }
```

```
28 |     $table->foreign('user_id')->references('id')->on('users')
29 |         ->onUpdate('cascade')->onDelete('cascade');
30 |     $table->foreign('role_id')->references('id')->on('roles')
31 |         ->onUpdate('cascade')->onDelete('cascade');
32 |
33 |     $table->primary(['user_id', 'role_id']);
34 | });
35 |
36 | // Create table for storing permissions
37 | Schema::create('permissions', function (Blueprint $table) {
38 |     $table->increments('id');
39 |     $table->string('name')->unique();
40 |     $table->string('display_name')->nullable();
41 |     $table->string('description')->nullable();
42 |     $table->timestamps();
43 | });
44 |
45 | // Create table for associating permissions to roles (Many-to-Many)
46 | Schema::create('permission_role', function (Blueprint $table) {
47 |     $table->integer('permission_id')->unsigned();
48 |     $table->integer('role_id')->unsigned();
49 |
50 |     $table->foreign('permission_id')->references('id')->on('permissions')
51 |         ->onUpdate('cascade')->onDelete('cascade');
52 |     $table->foreign('role_id')->references('id')->on('roles')
53 |         ->onUpdate('cascade')->onDelete('cascade');
54 |
55 |     $table->primary(['permission_id', 'role_id']);
56 | });
57 |
58 |
59 | /**
60 | * Reverse the migrations.
61 | *
62 | * @return void
63 | */
64 | public function down()
65 | {
66 |     Schema::dropIfExists('permission_role');
67 |     Schema::dropIfExists('permissions');
68 |     Schema::dropIfExists('role_user');
69 |     Schema::dropIfExists('roles');
70 | }
71 |
72 | }
```

d. 2017_03_05_152747_create_authors_table.php

```
1 <?php
2
3 use Illuminate\Support\Facades\Schema;
4 use Illuminate\Database\Schema\Blueprint;
5 use Illuminate\Database\Migrations\Migration;
6
7 class CreateAuthorsTable extends Migration
8 {
9     /**
10      * Run the migrations.
11      *
12      * @return void
13      */
14     public function up()
15     {
16         Schema::create('authors', function (Blueprint $table) {
17             $table->increments('id');
18             $table->string('name');
19             $table->timestamps();
20         });
21     }
22
23     /**
24      * Reverse the migrations.
25      *
26      * @return void
27      */
28     public function down()
29     {
30         Schema::dropIfExists('authors');
31     }
32 }
33
```

e. 2017_03_05_152755_create_books_table.php

```
1 <?php
2
3 use Illuminate\Support\Facades\Schema;
4 use Illuminate\Database\Schema\Blueprint;
5 use Illuminate\Database\Migrations\Migration;
6
7 class CreateBooksTable extends Migration
8 {
9     /**
10      * Run the migrations.
11      *
12      * @return void
13      */
14     public function up()
15     {
16         Schema::create('books', function (Blueprint $table) {
17             $table->increments('id');
18             $table->string('title');
19             $table->integer('author_id')->unsigned();
20             $table->integer('amount')->unsigned();
21             $table->string('cover')->nullable();
22             $table->timestamps();
23
24             $table->foreign('author_id')->references('id')->on('authors')->onDelete('cascade');
25         });
26     }
27 }
```

```
28     /**
29      * Reverse the migrations.
30      *
31      * @return void
32      */
33     public function down()
34     {
35         Schema::dropIfExists('books');
36     }
37 }
38
```

e. 2017_03_07_091524_create_borrow_logs_table.php

```
1  <?php
2
3  use Illuminate\Support\Facades\Schema;
4  use Illuminate\Database\Schema\Blueprint;
5  use Illuminate\Database\Migrations\Migration;
6
7  class CreateBorrowLogsTable extends Migration
8  {
9      /**
10      * Run the migrations.
11      *
12      * @return void
13      */
14     public function up()
15     {
16         Schema::create('borrow_logs', function (Blueprint $table) [
17             $table->increments('id');
18             $table->integer('book_id')->unsigned()->index();
19             $table->foreign('book_id')->references('id')->on('books')->onDelete('cascade')->onUpd
20             $table->integer('user_id')->unsigned()->index();
21             $table->foreign('user_id')->references('id')->on('users')->onDelete('cascade')->onUpd
22             $table->boolean('is_returned')->default(false);
23             $table->timestamps();
24         ]);
25     }
26
27     /**
28      * Reverse the migrations.
29      *
30      * @return void
31      */
32     public function down()
33     {
34         Schema::dropIfExists('borrow_logs');
35     }
36 }
37
```

f. 2017_03_09_032851_add_verification_to_users.php

```
1  <?php
2
3  use Illuminate\Support\Facades\Schema;
4  use Illuminate\Database\Schema\Blueprint;
5  use Illuminate\Database\Migrations\Migration;
6
7  class AddVerificationToUsers extends Migration
8  {
9      /**
10      * Run the migrations.
11      *
12      * @return void
13      */
14      public function up()
15      {
16          Schema::table('users', function (Blueprint $table) {
17              $table->string('verification_token')->nullable();
18              $table->boolean('is_verified')->default(0);
19          });
20      }
21
22      /**
23      * Reverse the migrations.
24      *
25      * @return void
26      */
27
28      public function down()
29      {
30          Schema::table('users', function (Blueprint $table) {
31              $table->dropColumn('verification_token');
32              $table->dropColumn('is_verified');
33          });
34      }
35  }
```

g. 2017_03_13_145340_add_last_login_to_users_table.php

```
1  <?php
2
3  use Illuminate\Support\Facades\Schema;
4  use Illuminate\Database\Schema\Blueprint;
5  use Illuminate\Database\Migrations\Migration;
6
7  class AddLastLoginToUsersTable extends Migration
8  {
9      /**
10      * Run the migrations.
11      *
12      * @return void
13      */
14      public function up()
15      {
16          Schema::table('users', function (Blueprint $table) {
17              $table->timestamp('last_login')->after('remember_token')->nullable();
18          });
19      }
20
21      /**
22      * Reverse the migrations.
23      *
24      * @return void
25      */
26      public function down()
```

```
21  /**
22  * Reverse the migrations.
23  *
24  * @return void
25  */
26  public function down()
27  {
28      Schema::table('users', function (Blueprint $table) {
29          $table->dropColumn('last_login');
30      });
31  }
32 }
33
```

- Untuk dapat bekerja dengan database, sebelumnya kita harus melakukan konfigurasi koneksi database. Untuk melakukannya kita bisa mengisi variabel yang tersedia pada file konfigurasi database pada **config/database.php**.

```
perpusku_baru > config > database.php
41
42     'mysql' => [
43         'driver' => 'mysql',
44         'host' => env('DB_HOST', '127.0.0.1'),
45         'port' => env('DB_PORT', '3306'),
46         'database' => env('DB_DATABASE', 'forge'),
47         'username' => env('DB_USERNAME', 'forge'),
48         'password' => env('DB_PASSWORD', ''),
49         'charset' => 'utf8mb4',
50         'collation' => 'utf8mb4_unicode_ci',
51         'prefix' => '',
52         'strict' => true,
53         'engine' => null,
54     ],
55 ].
```

- Misal kita akan menghubungkan aplikasi Laravel dengan mysql, maka kita langsung menuju konfigurasi mysql seperti berikut pada file tersebut. Jika diperhatikan, file tersebut menggunakan fungsi `env("DB_HOST", "127.0.0.1")` dimana file tersebut menggunakan nilai yang ada pada file `.env`. Maka, kita isikan konfigurasi koneksi kita pada file `.env` di root direktori aplikasi.

```
perpusku_baru > .env
1 APP_ENV=laravel
2 APP_KEY=base64:K0cz7b7+M9QlRji0w1+f5UTUpK1q0g5CqcIwrqwTugY=
3 APP_DEBUG=true
4 APP_LOG_LEVEL=debug
5 APP_URL=http://localhost
6
7 DB_CONNECTION=mysql
8 DB_HOST=127.0.0.1
9 DB_PORT=3306
10 DB_DATABASE=perpusku_baru
11 DB_USERNAME=root
12 DB_PASSWORD=
13
```

Sehingga kita pengaturan nama host, port, nama database, username, dan password dari database yang kita miliki dilakukan pada file.env bukan pada config/database.php.

- Sebelum menjalankan perintah migration pastikan , untuk membuat koneksi pada database maka buat database baru di phpmyadmin untuk menampung data – data yang telah di inputkan di laravel dengan database yang telah diberi nama “**perpusku_baru**”.



- Selanjutnya lakukan eksekusi database dengan perintah “**php artisan migrate**”

```
C:\xampp\htdocs\perpusku_baru>php artisan migrate
Migration table created successfully.
Migrating: 2014_10_12_00000_create_users_table
Migrated: 2014_10_12_00000_create_users_table
Migrating: 2014_10_12_100000_create_password_resets_table
Migrated: 2014_10_12_100000_create_password_resets_table
Migrating: 2017_03_05_133706_laratrust_setup_tables
Migrated: 2017_03_05_133706_laratrust_setup_tables
Migrating: 2017_03_05_152747_create_authors_table
Migrated: 2017_03_05_152747_create_authors_table
Migrating: 2017_03_05_152755_create_books_table
Migrated: 2017_03_05_152755_create_books_table
Migrating: 2017_03_07_091524_create_borrow_logs_table
Migrated: 2017_03_07_091524_create_borrow_logs_table
Migrating: 2017_03_09_032851_add_verification_to_users
Migrated: 2017_03_09_032851_add_verification_to_users
Migrating: 2017_03_13_145340_add_last_login_to_users_table
Migrated: 2017_03_13_145340_add_last_login_to_users_table
```

6. Kemudian membuat seeding , dimana *Seeding* merupakan fungsi yang disediakan oleh Laravel untuk memberikan data *dummy* pada database menggunakan Class seed. Class seed disimpan di direktori ***database/seeding***. Penulisan nama Class seeding tidak dibatasi, kita dapat memberikan nama apapun. Tetapi, sebaiknya menggunakan format yang seragam untuk memudahkan seperti ***UserSeeder***, dan lain sebagainya.

a. BookSeeder.php

```
C:\xampp\htdocs\perpusku_baru>php artisan make:seeder BooksSeeder  
Seeder created successfully.
```

```
perpusku_baru > database > seeds > BooksSeeder.php  
1  <?php  
2  
3  use Illuminate\Database\Seeder;  
4  use App\Book;  
5  use App\Author;  
6  use App\User;  
7  use App\BorrowLog;  
8  
9  class BooksSeeder extends Seeder  
10 {  
11     /**  
12      * Run the database seeds.  
13      *  
14      * @return void  
15      */  
16     public function run()  
17     {  
18         // Sample penulis  
19         $author1 = Author::create(['name' => 'Mohammad Abdullah Rismanto']);  
20         $author2 = Author::create(['name' => 'Sulaiman Riskyatnar']);  
21         $author3 = Author::create(['name' => 'Amin Maarufdai']);  
22  
23         // Sample buku  
24         $book1 = Book::create(['title' => 'Pemrograman Dasar Java', 'amount' => 3, 'cover' => 'buku_dasar.java');  
25         $book2 = Book::create(['title' => 'Cerita Sejarah Para Pejuang Dahulu', 'amount' => 2, 'cover' => 'cerita_sejarah_pejuang');  
26         $book3 = Book::create(['title' => 'Kesejahteraan Sosial', 'amount' => 4, 'cover' => 'buku_kesejahteraan_sosial');  
27         $book4 = Book::create(['title' => 'Surga Yang Diridhoi', 'amount' => 1, 'cover' => 'buku_surga_yang_diridhoi');  
28  
29  
30         // Sample peminjaman buku  
31         $member = User::where('email', 'member@gmail.com')->first();  
32         //BorrowLog::create(['user_id' => $member->id, 'book_id' => $book1->id, 'is_returned' => 0]  
33         BorrowLog::create(['user_id' => $member->id, 'book_id' => $book1->id, 'is_returned' => 1]  
34         BorrowLog::create(['user_id' => $member->id, 'book_id' => $book2->id, 'is_returned' => 2]  
35         BorrowLog::create(['user_id' => $member->id, 'book_id' => $book3->id, 'is_returned' => 3]  
36         BorrowLog::create(['user_id' => $member->id, 'book_id' => $book4->id, 'is_returned' => 4]  
37     }  
38 }  
39 }
```

b. DatabaseSeeder.php

```
1 <?php
2
3 use Illuminate\Database\Seeder;
4
5 class DatabaseSeeder extends Seeder
6 {
7     /**
8      * Run the database seeds.
9      *
10     * @return void
11     */
12    public function run()
13    {
14        $this->call(UsersSeeder::class);
15        $this->call(BooksSeeder::class);
16    }
17 }
18
```

c. UsersSeeder.php

```
1 <?php
2
3 use Illuminate\Database\Seeder;
4 use App\Role;
5 use App\User;
6
7 class UsersSeeder extends Seeder
8 {
9     /**
10      * Run the database seeds.
11      *
12      * @return void
13      */
14     public function run()
15     {
16         // Membuat role admin
17         $adminRole = new Role();
18         $adminRole->name = "admin";
19         $adminRole->display_name = "Admin";
20         $adminRole->save();
21
22         // Membuat role member
23         $memberRole = new Role();
24         $memberRole->name = "member";
25         $memberRole->display_name = "Member";
26         $memberRole->save();
27     }
28 }
```

```
28     // Membuat sample Admin
29     $admin = new User();
30     $admin->name = "Admin Perpus Ceria";
31     $admin->email = "admin@gmail.com";
32     $admin->password = bcrypt("123");
33     $admin->is_verified = 1;
34     $admin->save();
35     $admin->attachRole($adminRole);
36
37     // Membuat sample member
38     $member = new User();
39     $member->name = "Fatimah";
40     $member->email = "member1@gmail.com";
41     $member->password = bcrypt("121");
42     $member->is_verified = 1;
43     $member->save();
44     $member->attachRole($memberRole);
45
46     $member = new User();
47     $member->name = "Yusuf Mansyur";
48     $member->email = "member2@gmail.com";
49     $member->password = bcrypt("122");
50     $member->is_verified = 1;
51     $member->save();
52     $member->attachRole($memberRole);
53
```

```
54     $member = new User();
55     $member->name = "Ruqayah Syahrani";
56     $member->email = "member3@gmail.com";
57     $member->password = bcrypt("124");
58     $member->is_verified = 1;
59     $member->save();
60     $member->attachRole($memberRole);
61
62     $member = new User();
63     $member->name = "Chandra Risdianto";
64     $member->email = "member4@gmail.com";
65     $member->password = bcrypt("125");
66     $member->is_verified = 1;
67     $member->save();
68     $member->attachRole($memberRole);
69
70     $member = new User();
71     $member->name = "Qirani Aisyah";
72     $member->email = "member5@gmail.com";
73     $member->password = bcrypt("126");
74     $member->is_verified = 1;
75     $member->save();
76     $member->attachRole($memberRole);
77 }
78 }
79 }
```

- Setelah itu jalankan php artisan db:seed

```
C:\xampp\htdocs\perpusku_baru>php artisan db:seed
Seeding: UsersSeeder
Seeding: BooksSeeder
```

- Mengecek apakah data yang di insert masuk pada tabel dengan cara cek di phpmyadmin

Tabel	Tindakan	Baris	Jenis	Penyortiran	Ukuran	Beban
authors	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	3	InnoDB	utf8mb4_unicode_ci	16.0 KB	-
books	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	6	InnoDB	utf8mb4_unicode_ci	32.0 KB	-
borrow_logs	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	2	InnoDB	utf8mb4_unicode_ci	80.0 KB	-
migrations	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	8	InnoDB	utf8mb4_unicode_ci	16.0 KB	-
password_resets	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	0	InnoDB	utf8mb4_unicode_ci	48.0 KB	-
permissions	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	0	InnoDB	utf8mb4_unicode_ci	32.0 KB	-
permission_role	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	0	InnoDB	utf8mb4_unicode_ci	32.0 KB	-
roles	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	2	InnoDB	utf8mb4_unicode_ci	32.0 KB	-
role_user	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	9	InnoDB	utf8mb4_unicode_ci	32.0 KB	-
users	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	9	InnoDB	utf8mb4_unicode_ci	48.0 KB	-
10 tabel	Jumlah	39	InnoDB	utf8mb4_general_ci	368.0 KB	0 B

7. Selanjutnya buat route dimana untuk mengambil data dari database

```
perpusku_baru > routes > web.php
 1  <?php
 2
 3  /*
 4  | -----
 5  | Web Routes
 6  | -----
 7  |
 8  | Here is where you can register web routes for your application. These
 9  | routes are loaded by the RouteServiceProvider within a group which
10  | contains the "web" middleware group. Now create something great!
11  |
12  */
13
14 Auth::routes();
15
16 Route::get('/', 'GuestController@index');
17
18 Route::get('/home', 'HomeController@index');
19
20 Route::group(['prefix' => 'admin', 'middleware' => ['auth', 'role:admin']], function() {
21     Route::resource('authors', 'AuthorsController');
22     Route::resource('books', 'BooksController');
23     Route::resource('members', 'MembersController');
24     Route::get('statistics', [
25         'as'      => 'statistics.index',
26         'uses'   => 'StatisticsController@index'
```

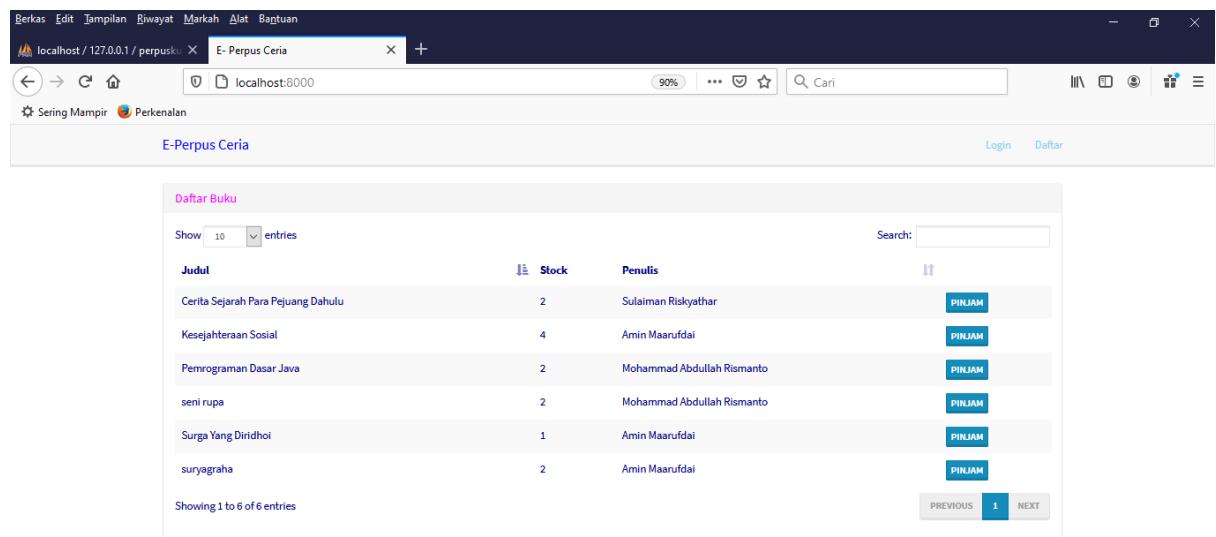
```
27     ]);  
28  
29     Route::get('export/books', [  
30         'as' => 'export.books',  
31         'uses' => 'BooksController@export'  
32     ]);  
33     Route::post('export/books', [  
34         'as' => 'export.books.post',  
35         'uses' => 'BooksController@exportPost'  
36     ]);  
37  
38     Route::get('template/books', [  
39         'as' => 'template.books',  
40         'uses' => 'BooksController@generateExcelTemplate'  
41     ]);  
42     Route::post('import/books', [  
43         'as' => 'import.books',  
44         'uses' => 'BooksController@importExcel'  
45     ]);  
46 };  
47  
48 Route::get('books/{book}/borrow', [  
49     'middleware' => ['auth', 'role:member'],  
50     'as' => 'guest.books.borrow',  
51     'uses' => 'BooksController@borrow'  
52 ]);
```

```
perpusku_baru > routes > 🌐 web.php  
54     Route::put('books/{book}/return', [  
55         'middleware' => ['auth', 'role:member'],  
56         'as' => 'member.books.return',  
57         'uses' => 'BooksController@returnBack'  
58 ]);  
59  
60     Route::get('auth/verify/{token}', 'Auth\RegisterController@verify');  
61     Route::get('auth/send-verification', 'Auth\RegisterController@sendVerification');  
62  
63     Route::get('settings/profile', 'SettingsController@profile');  
64     Route::get('settings/profile/edit', 'SettingsController@editProfile');  
65     Route::post('settings/profile', 'SettingsController@updateProfile');  
66     Route::get('settings/password', 'SettingsController@editPassword');  
67     Route::post('settings/password', 'SettingsController@updatePassword');
```

8. Lalu jalankan program dengan alamat sebagai berikut:

➤ **Jika Sebagai member**

1. Berikut adalah tampilan awal jika jalankan program dengan alamat sebagai berikut, namun dalam tampilan ini jika klik pinjam maka akan diarahkan ke halaman form login, dimana jika kita ingin pinjam buku diharuskan login terlebih dahulu.



The screenshot shows a web browser window titled "E-Perpus Ceria". The address bar displays "localhost:8000". The main content area is a table titled "Daftar Buku" (Book Catalog) with columns: Judul (Title), Stock (Stock), Penulis (Author), and PINJAM (Loan). The table lists six books. At the bottom of the table, it says "Showing 1 to 6 of 6 entries". Below the table are "PREVIOUS" and "NEXT" buttons. The browser's status bar at the bottom shows the date and time: "19/12/2020 22:37".

Judul	Stock	Penulis	PINJAM
Cerita Sejarah Para Pejuang Dahulu	2	Sulaiman Riskyathar	PINJAM
Kesejahteraan Sosial	4	Amin Maarufdai	PINJAM
Pemrograman Dasar Java	2	Mohammad Abdullah Rismanto	PINJAM
seni rupa	2	Mohammad Abdullah Rismanto	PINJAM
Surga Yang Diridhoi	1	Amin Maarufdai	PINJAM
suryagraha	2	Amin Maarufdai	PINJAM

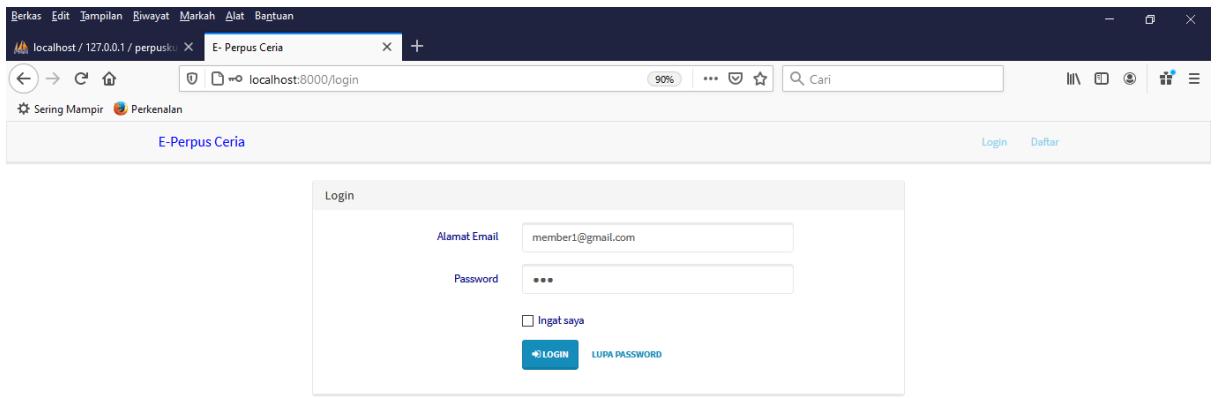
2. Jika kita klik “ Login” maka akan muncul tampilan sebagai berikut dimana dalam form login tersebut kita diharapkan memasukkan email dan password yang telah didaftarkan , tetapi jika kita belum memiliki akun, maka anda harus klik “ Daftar ” terlebih dahulu.

a. Tampilan Register / daftar

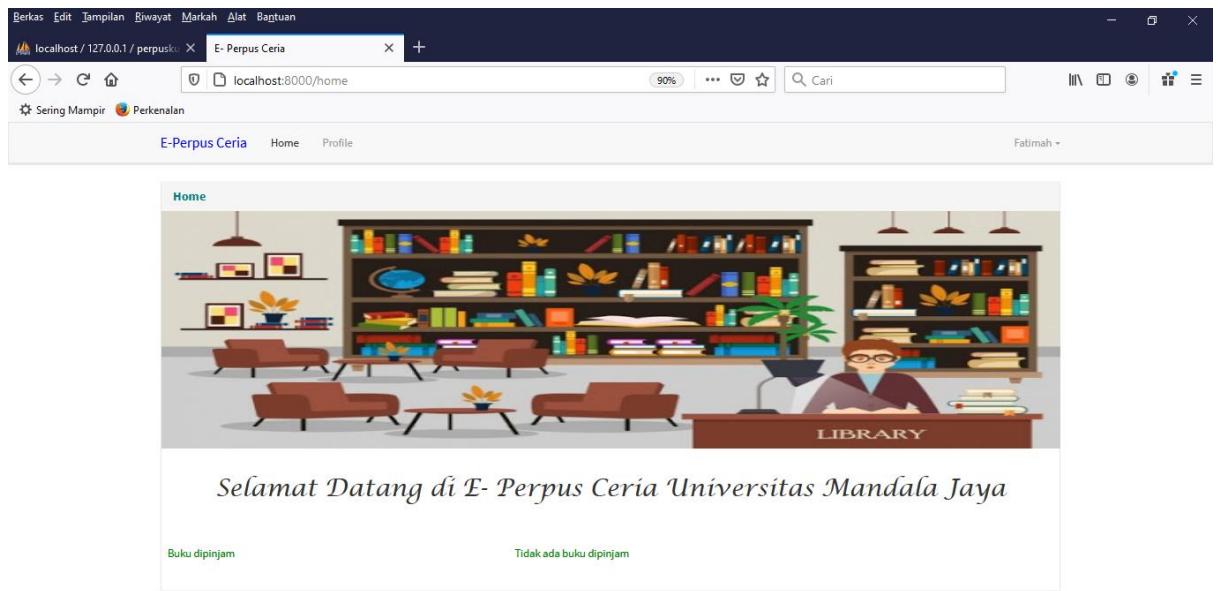
The screenshot shows a Windows desktop environment with a browser window titled "E-Perpus Ceria". The URL in the address bar is "localhost:8000/register". The page content is a registration form titled "Register" with four input fields: "Nama", "Alamat Email", "Password", and "Konfirmasi Password". Below the fields is a blue "DAFTAR" button.

The screenshot shows a Windows desktop environment with a browser window titled "E-Perpus Ceria". The URL in the address bar is "localhost:8000/register". The page content is a registration form titled "Register" with the following data entered: "Nama" is "Fatimah", "Alamat Email" is "member1@gmail.com", "Password" and "Konfirmasi Password" both show three masked asterisks. Below the fields is a blue "DAFTAR" button.





3. Setelah login maka akan diarahkan ke halaman home , dimana jika kita sebagai member maka akan diarahakan ke halaman home untuk member



Dalam halaman utama ini ,untuk member hanya mempunyai hak akses melihat profil, home dan juga peminjaman buku yang mana jika ingin meminjam buku maka klik “**E-Perpus Ceria**”

- Jika kita klik “ E-Perpus ” maka akan muncul tampilan berikut

The screenshot shows a Windows desktop environment with a browser window titled "E-Perpus Ceria". The URL is "localhost:8000". The page displays a table of books with columns: Judul, Stock, Penulis, and PINJAM. The books listed are:

Judul	Stock	Penulis	PINJAM
Cerita Sejarah Para Pejuang Dahulu	2	Sulaiman Riskyathar	PINJAM
Kesejahteraan Sosial	4	Amin Maarufdai	PINJAM
Pemrograman Dasar Java	3	Mohammad Abdullah Rismanto	PINJAM
seni rupa	2	Mohammad Abdullah Rismanto	PINJAM
Surga Yang Dindhi	1	Amin Maarufdai	PINJAM
suryagraha	2	Amin Maarufdai	PINJAM

At the bottom, there are navigation buttons: PREVIOUS, 1, and NEXT.

- Jika klik “ Profil ” maka akan muncul tampilan sebagai berikut

The screenshot shows a Windows desktop environment with a browser window titled "E-Perpus Ceria". The URL is "localhost:8000/settings/profile". The page displays a "Profil" section with the following information:

Nama	Fatimah
Email	member1@gmail.com
Login terakhir	2020-12-19 14:52:33

At the bottom, there is a blue "UBAH" button.

Dimana fungsi fitur ini adalah untuk melihat data anda beserta login terakhir member tersebut.

- Dan jika klik button pinjam pada data buku di E-Perpus maka akan muncul tampilan sebagai berikut

Berhasil meminjam Pemrograman Dasar Java

Judul	Stock	Penulis	Action
Cerita Sejarah Para Pejuang Dahulu	2	Sulaiman Riskyathar	PINJAM
Kesejahteraan Sosial	4	Amin Maarufdai	PINJAM
Pemrograman Dasar Java	2	Mohammad Abdullah Rismanto	PINJAM
seni rupa	2	Mohammad Abdullah Rismanto	PINJAM
Surga Yang Diridhoi	1	Amin Maarufdai	PINJAM
suryagraha	2	Amin Maarufdai	PINJAM

Showing 1 to 6 of 6 entries

- Maka setelah melakukan peminjaman , akan muncul pada dashboard / home tampilan dimana bahwa tercatat buku “ Pemrograman Dasar Java” dipinjam.

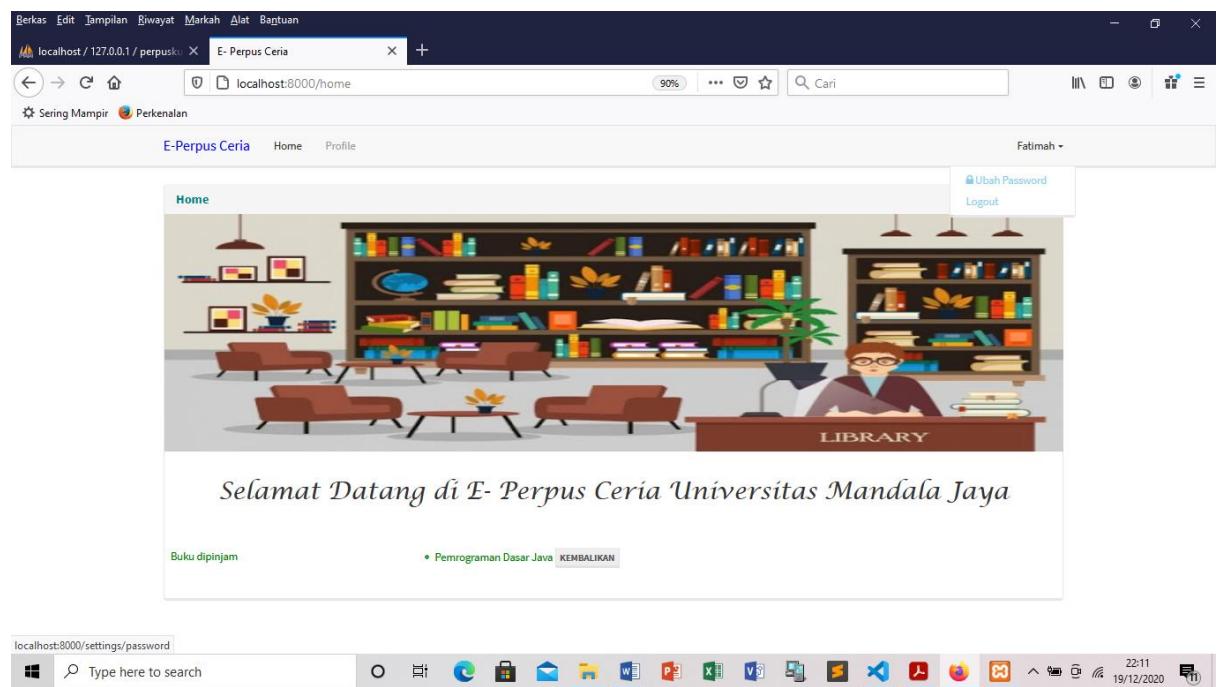
Home

Selamat Datang di E-Perpus Ceria Universitas Mandala Jaya

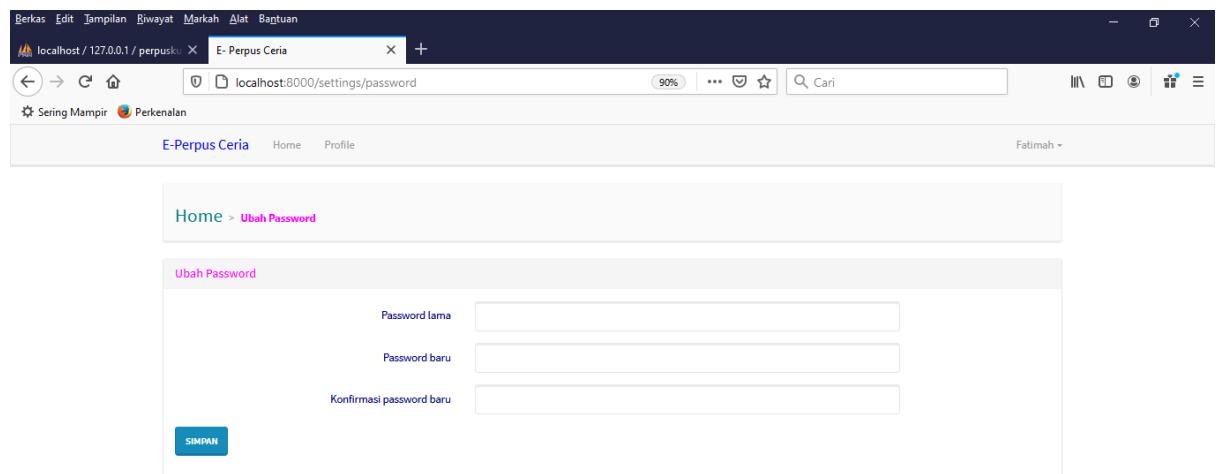
Buku dipinjam • Pemrograman Dasar Java KEMBALIKAN

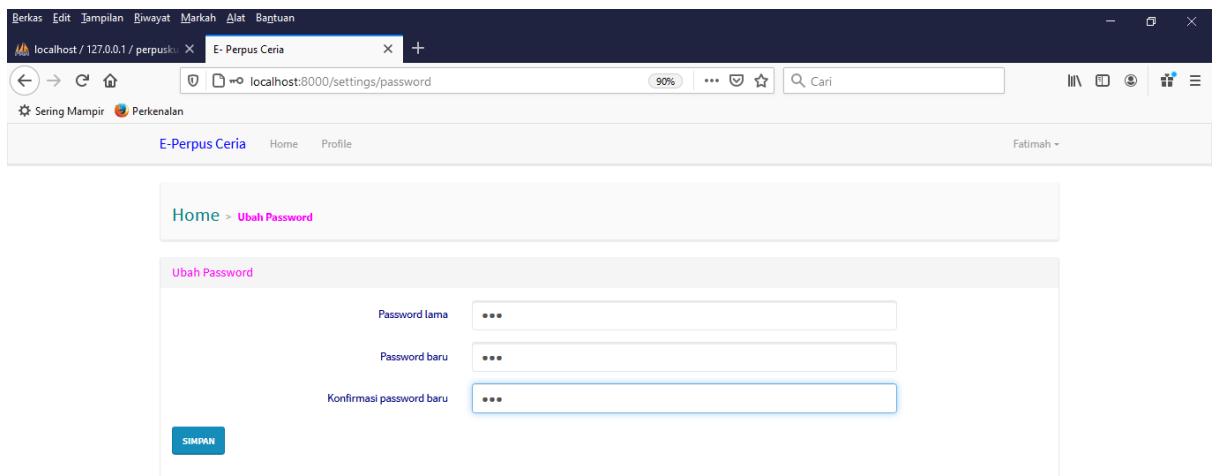


- Jika ingin ubah password maka klik pilihan ubah password pada button login

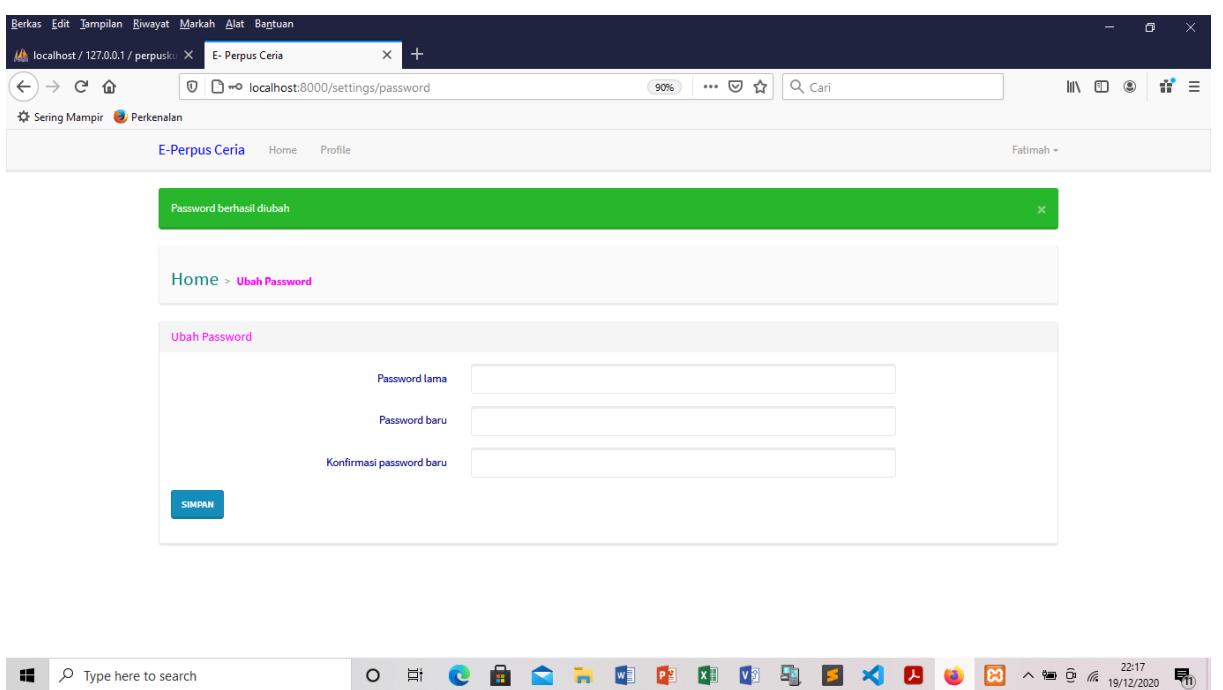


Maka akan muncul tampilan sebagai berikut dan di tampilan ini diarahkan bahwa kita harus mengisi dengan password lama, password baru dan konfirmasi password baru.





Dan jika klik simpan maka akan muncul tampilan sebagai berikut



4. Setelah itu klik logout, maka secara otomatis akan keluar dari sistem itu dan kembali ke tampilan awal.

➤ **Jika sebagai admin**

1. Berikut adalah tampilan awal jika jalankan program dengan alamat sebagai berikut, namun dalam tampilan ini jika klik pinjam maka akan diarahkan ke halaman form login, dimana jika kita ingin pinjam buku diharuskan login terlebih dahulu.

The screenshot shows a web application titled "E-Perpus Ceria". The main content is a table titled "Daftar Buku" displaying six entries. The columns are "Judul", "Stock", and "Penulis". Each row contains a "PINJAM" button. The table includes a "Search:" input field, a "Show: 10" dropdown, and navigation buttons for "PREVIOUS" and "NEXT". The browser's address bar shows "localhost:8000". The taskbar at the bottom of the screen shows various pinned applications like File Explorer, Microsoft Word, and Microsoft Excel.

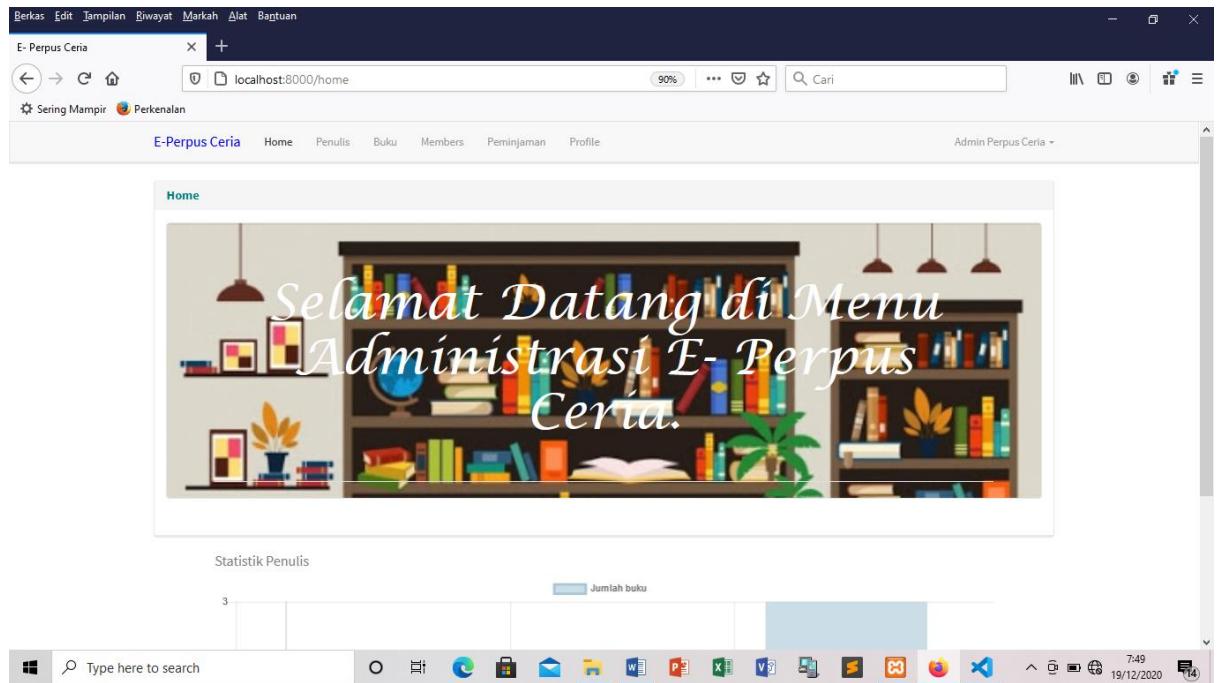
Judul	Stock	Penulis	
Cerita Sejarah Para Pejuang Dahulu	2	Sulaiman Riskyathar	<button>PINJAM</button>
Kesejahteraan Sosial	4	Amin Maarufdai	<button>PINJAM</button>
Pemrograman Dasar Java	2	Mohammad Abdullah Rismanto	<button>PINJAM</button>
seni rupa	2	Mohammad Abdullah Rismanto	<button>PINJAM</button>
Surga Yang Diridhoi	1	Amin Maarufdai	<button>PINJAM</button>
suryagraha	2	Amin Maarufdai	<button>PINJAM</button>

2. Jika kita klik “ Login” maka akan muncul tampilan sebagai berikut dimana dalam form login tersebut kita diharapkan memasukkan email dan password yang telah didaftarkan .

The screenshot shows a "Login" page from the "E-Perpus Ceria" application. The page features a form with "Alamat Email" and "Password" fields, both pre-filled with "admin@gmail.com" and "password" respectively. There is a "Ingat saya" checkbox, a "LOGIN" button, and a "LUPA PASSWORD" link. The browser's address bar shows "localhost:8000/login". The taskbar at the bottom of the screen shows various pinned applications like File Explorer, Microsoft Word, and Microsoft Excel.



3. Setelah login maka akan diarahkan ke halaman home , dimana jika kita sebagai admin maka akan diarahakan ke halaman home untuk admin



Dalam halaman utama ini ,untuk admin hanya mempunyai hak akses melihat profil, home, Penulis, Buku, Members, Peminjaman buku yang mana dalam peminjaman ini hanya berupa data buku yang dipinjam oleh member beserta tanggal pinjam dan tanggal kembali.

- Jika klik Penulis maka akan muncul tampilan berikut, dimana dalam penulis ini terdapat fitur tambah, ubah, hapus

Nama	
Amin Maarufdal	Ubah HAPUS
Mohammad Abdullah Rismanto	Ubah HAPUS
Sulaiman Riskyethar	Ubah HAPUS



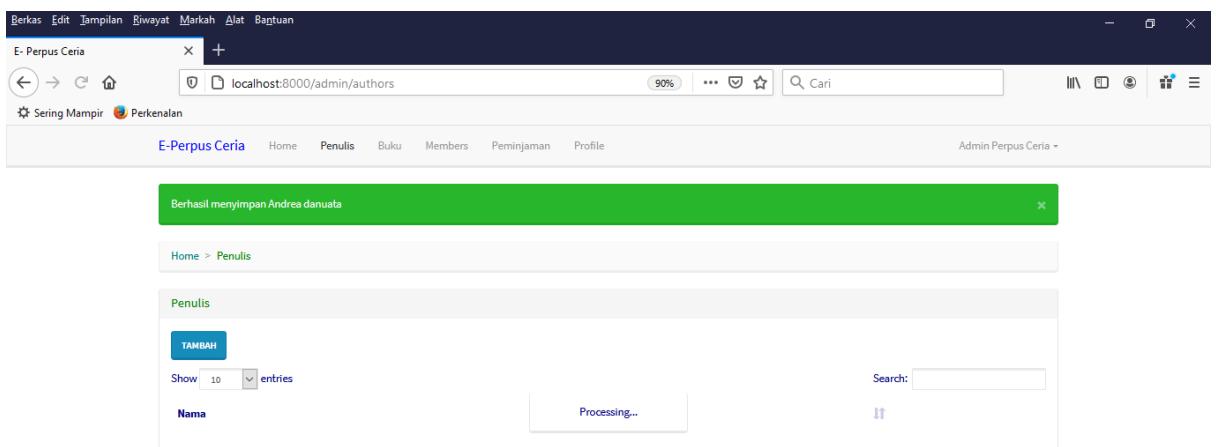
- Jika klik button tambah maka akan muncul tampilan sebagai berikut, dimana fitur ini berfungsi untuk membuat penulis baru/ menambahkan penulis baru

The screenshot shows a web browser window with the URL `localhost:8000/admin/authors/create`. The page title is "Tambah Penulis". There is a single input field labeled "Nama" with the placeholder "Masukkan Nama". Below the input field is a blue "SIMPAN" button.

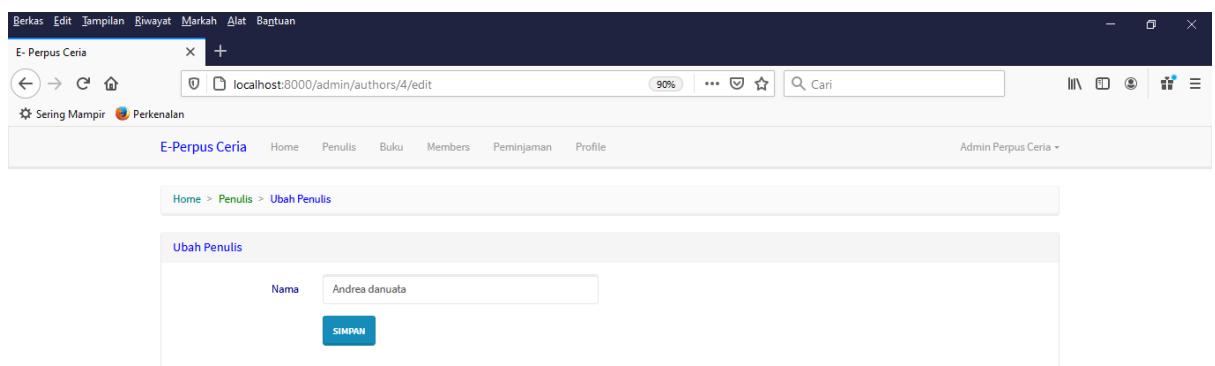
The screenshot shows the same browser window after the name "Andrea danuta" has been entered into the "Nama" field. The "SIMPAN" button remains visible below the input field.

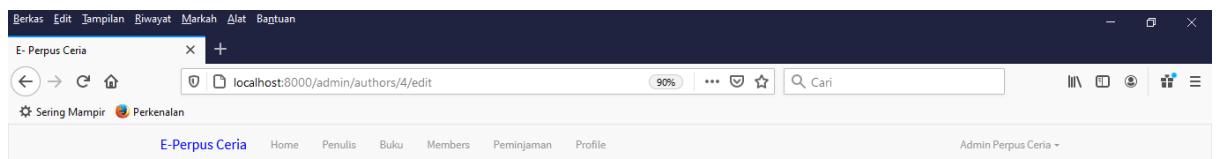


Setelah klik button simpan maka akan muncul tampilan sebagai berikut,dimana pada tampilan ini menunjukan bahwa penulis yang kita tambahkan telah berhasil masuk ke database.



- Dan jika klik button ubah maka akan muncul tampilan sebagai berikut , dimana dalam fitur ubah ini hanya mengubah nama saja





Berkas Edit Tampilan Riwayat Markah Alat Bantuan

E-Perpus Ceria

Sering Mampir Perkenalan

localhost:8000/admin/authors

90% Cari

E-Perpus Ceria Home Penulis Buku Members Peminjaman Profile Admin Perpus Ceria

Berhasil menyimpan Andrea danuata, S.T.

Home > Penulis

Penulis

TAMBAH

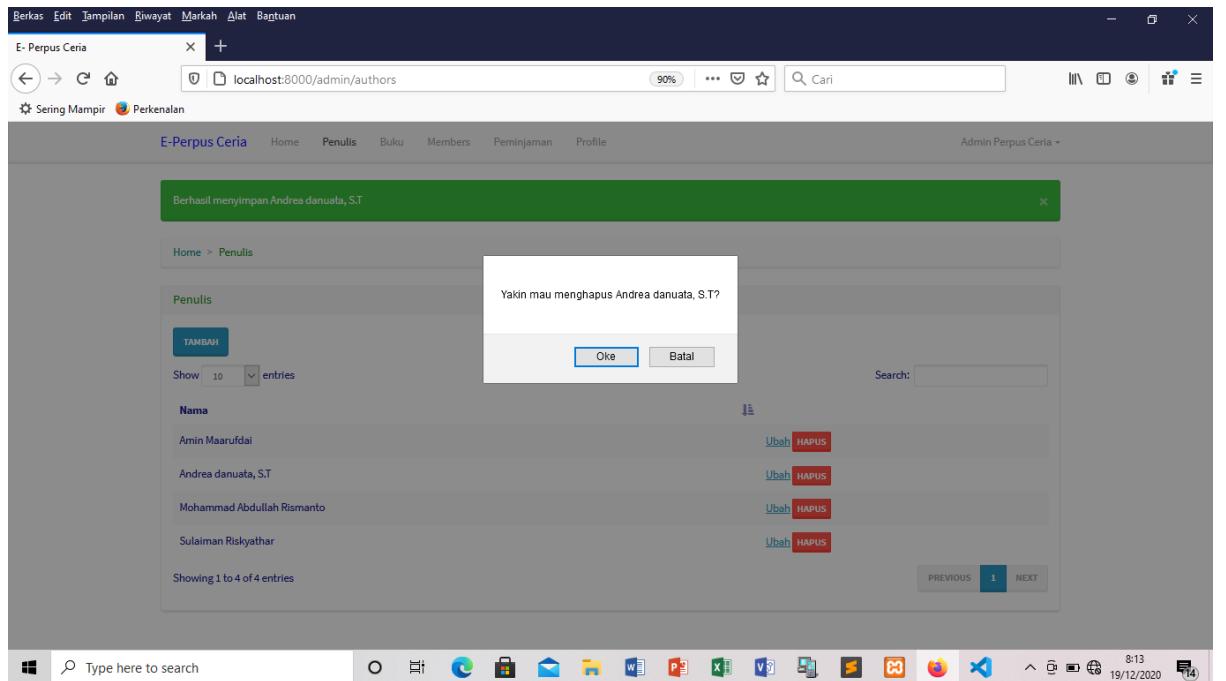
Show: 10 entries

Search:

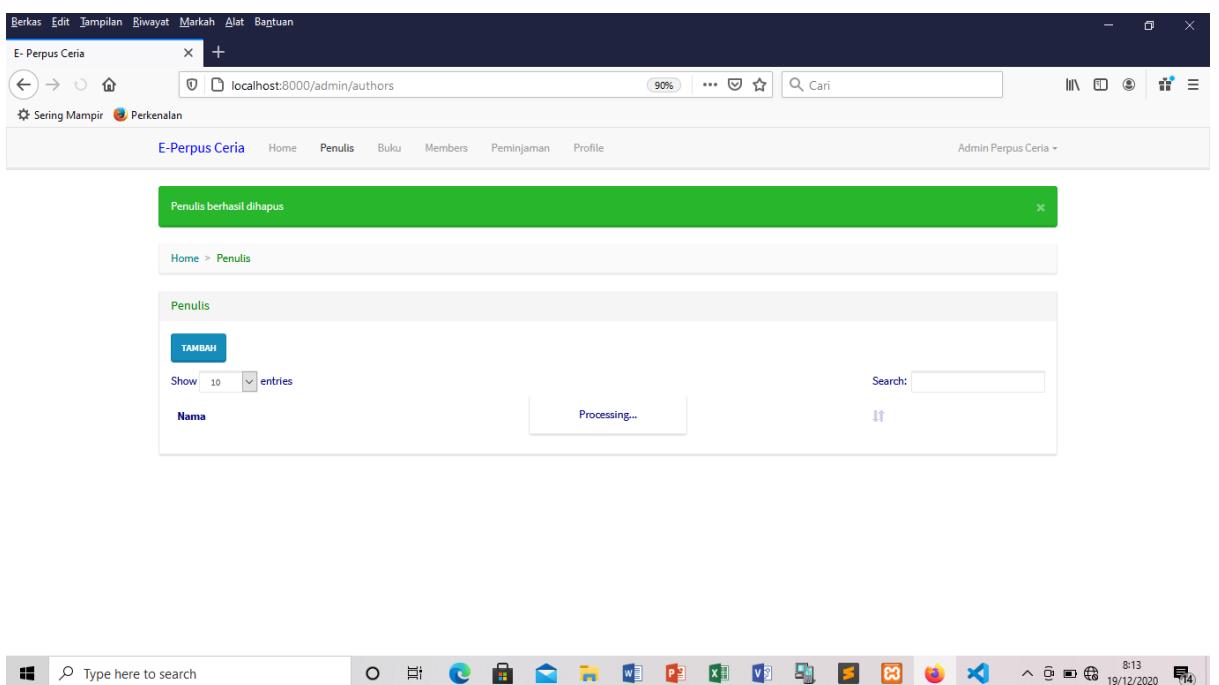
Nama	Action
Andrea danuata, S.T.	Processing...



- Dan pada fitur hapus hanya bisa menghapus jika pada penulis tidak memiliki buku, tetapi jika penulis masih memiliki buku maka akan muncul peringatan “**bawa penulis tidak bisa dihapus karena masih memiliki buku**”.



Serta jika berhasil dihapus maka akan muncul tampilan berikut



- Selanjutnya jika admin klik fitur buku maka akan muncul tampilan sebagai berikut, yang mana dalam fitur ini terdapat fitur – fitur lainnya seperti tambah, ubah,delet, export

Judul	Jumlah	Penulis	
Cerita Sejarah Para Pejuang Dahulu	2	Sulaiman Riskyathar	Ubah HAPUS
Kesejahteraan Sosial	4	Amin Maarufdai	Ubah HAPUS
Pemrograman Dasar Java	3	Mohammad Abdullah Rismanto	Ubah HAPUS
seni rupa	2	Mohammad Abdullah Rismanto	Ubah HAPUS
Surga Yang Diridhoi	1	Amin Maarufdai	Ubah HAPUS
suryagraha	2	Amin Maarufdai	Ubah HAPUS

Dan jika klik button Tambah maka akan muncul tampilan berikut, dimana pada fitur ini berfungsi untuk menambahkan buku baru.



Berkas Edit Tampilan Riwayat Markah Alat Bantuan

E-Perpus Ceria

Sering Mampir Perkenalan

E-Perpus Ceria Home Penulis Buku Members Peminjaman Profile Admin Perpus Ceria

Home > Buku > Tambah Buku

Tambah Buku

Isi Form

Judul: Pemrograman Desain Interface

Penulis: Sulaiman Riskyather

Jumlah: 1

Cover: Telusuri... desainInterface.jpg

SIMPAN



Jika berhasil disimpan maka akan muncul tampilan berikut

Berkas Edit Tampilan Riwayat Markah Alat Bantuan

E-Perpus Ceria

Sering Mampir Perkenalan

E-Perpus Ceria Home Penulis Buku Members Peminjaman Profile Admin Perpus Ceria

Berhasil menyimpan Pemrograman Desain Interface

Home > Buku

Buku

TAMBAH **EXPORT**

Show 10 entries

Search:

Judul	Jumlah	Processing...	Penulis
Pemrograman Desain Interface	1	Processing...	Sulaiman Riskyather



- Lalu jika klik fitur ubah maka akan muncul sebagai berikut , dimana pada fitur ubah ini sebagai admin bisa ubah judul buku, penulis, jumlah buku, cover , namun cover ini hanya muncul pada jika admin ubah atau menambahkan saja, sedangkan pada fitur E-Perpus(jika ingin pinjam) yang bisa di akses oleh member, fitur ini hanya menyajikan atau menampilkan judul buku, nama penulis, jumlah, penulis ke berapa.

The screenshot shows the 'Tambah Buku' (Add Book) form. It has a title 'Tambah Buku' and a sub-section 'Isi Form'. The form fields are:

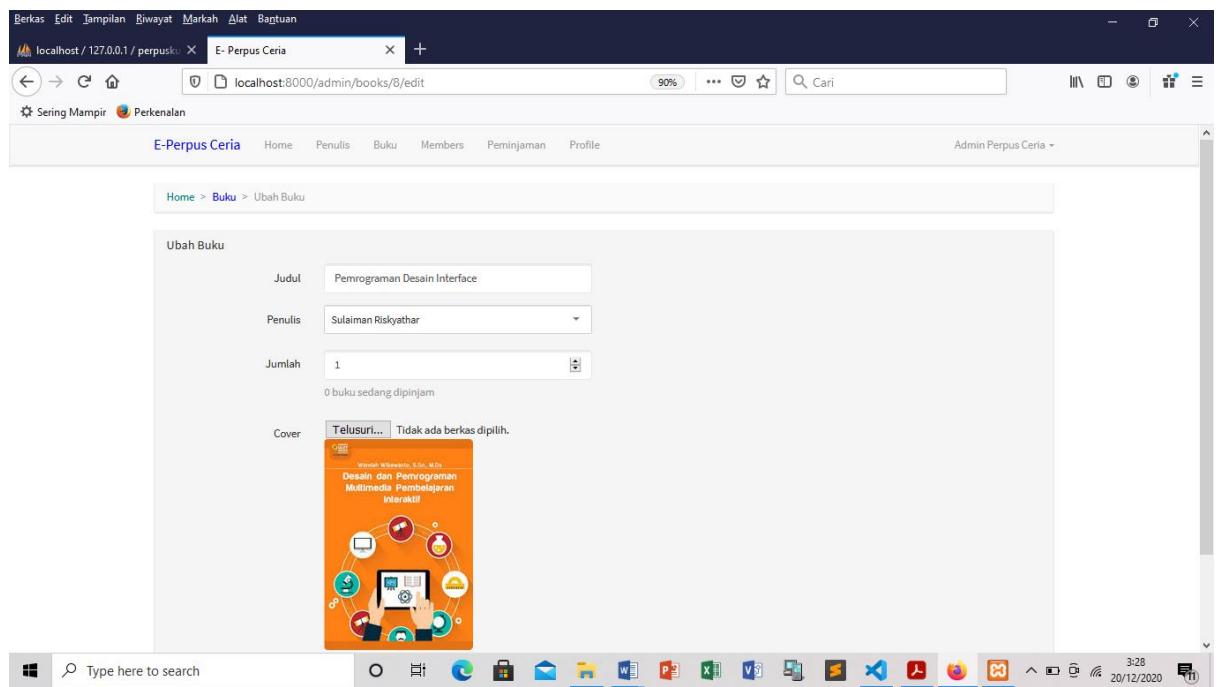
- Judul:** Pemrograman Desain Interface
- Penulis:** Sulaiman Riskyathar
- Jumlah:** 1
- Cover:** Telusuri... desainInterface.jpg

At the bottom is a blue 'SIMPAN' (Save) button.

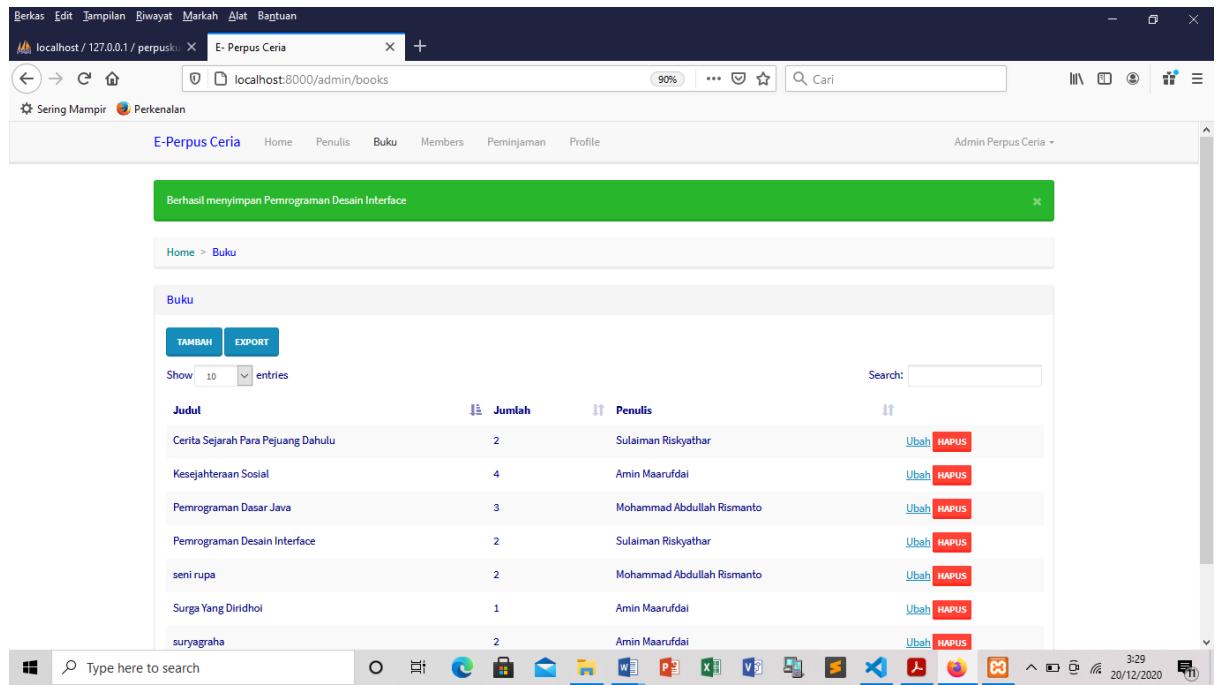
The screenshot shows the 'Buku' (Books) list page. At the top, there is a green success message: 'Berhasil menyimpan Pemrograman Desain Interface'. The page has a header with 'TAMBAH' and 'EXPORT' buttons, and a search bar. The main table lists books with columns: Judul, Jumlah, Penulis, and actions (Ubah, HAPUS).

Judul	Jumlah	Penulis	
Cerita Sejarah Para Pejuang Dahulu	2	Sulaiman Riskyathar	Ubah HAPUS
Kesejahteraan Sosial	4	Amin Maarufdai	Ubah HAPUS
Pemrograman Dasar Java	3	Mohammad Abdullah Rismanto	Ubah HAPUS
Pemrograman Desain Interface	1	Sulaiman Riskyathar	Ubah HAPUS
seni rupa	2	Mohammad Abdullah Rismanto	Ubah HAPUS
Surga Yang Diridhoi	1	Amin Maarufdai	Ubah HAPUS
suryagraha	2	Amin Maarufdai	Ubah HAPUS

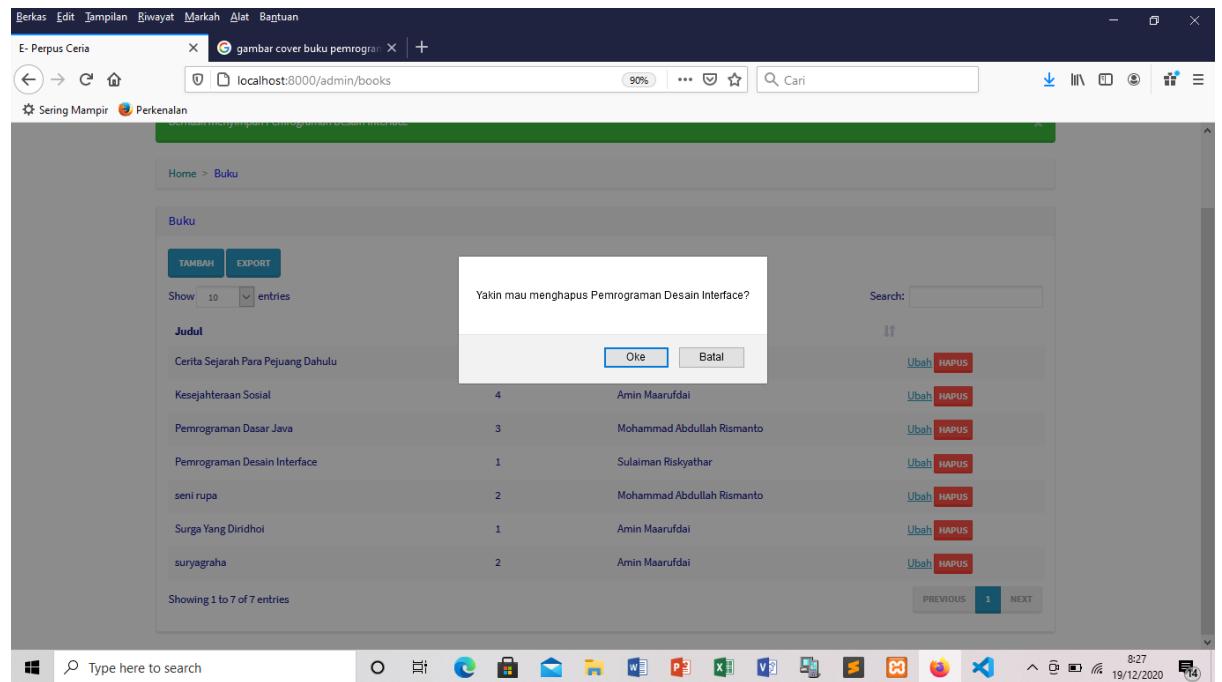
- Jika klik fitur ubah maka akan muncul tampilan berikut



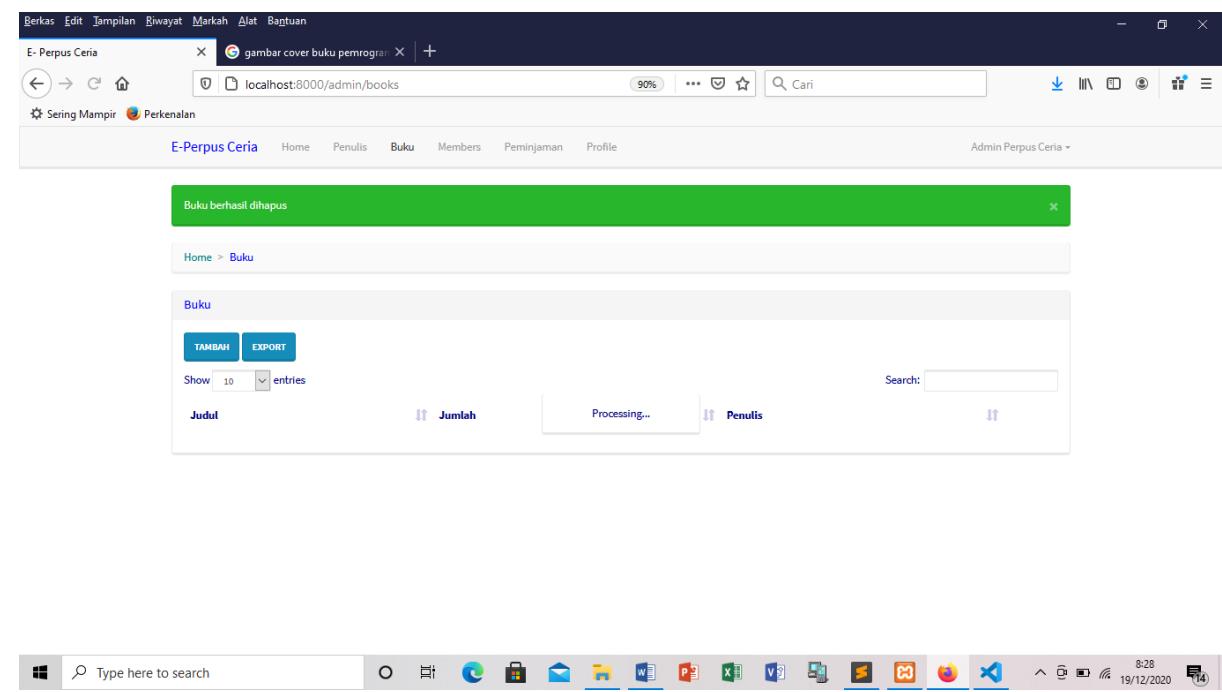
Setelah klik button simpan maka akan muncul tampilan peringatan bahwa data buku yang admin ubah telah berhasil disimpan.



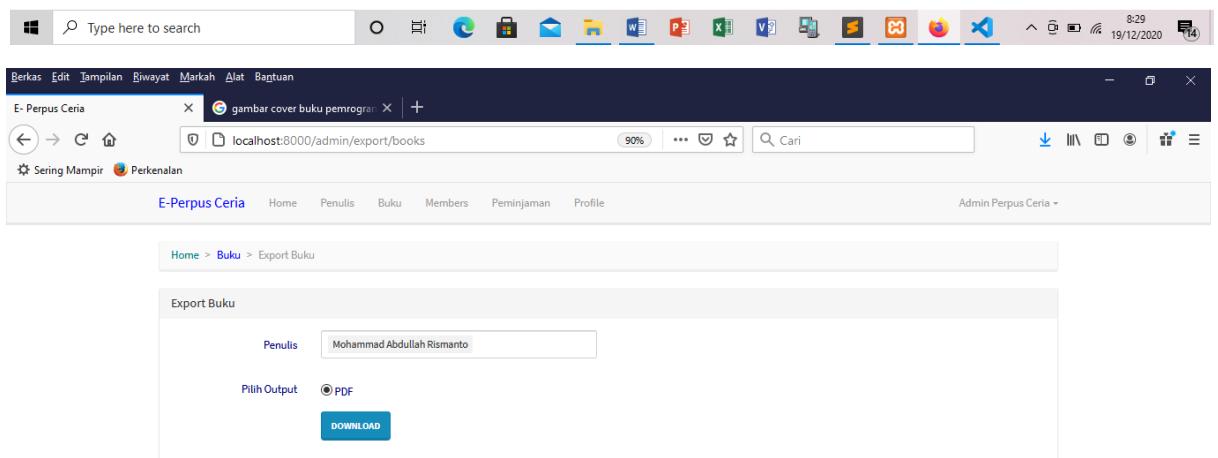
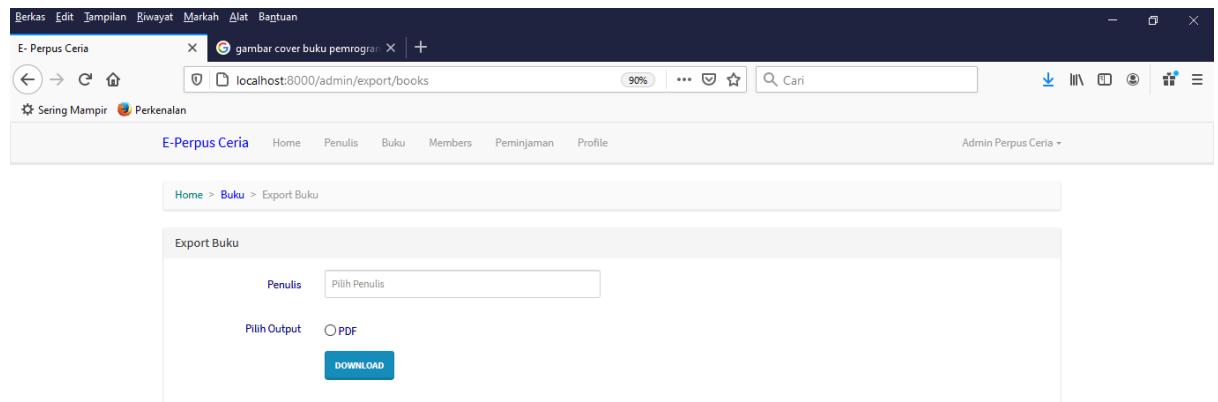
- Jika admin klik button hapus salah satu data buku maka akan muncul tampilan sebagai berikut.



Setelah klik “ok” maka akan muncul tampilan sebagai berikut



- Jika klik button export, maka akan diarahkan ke tampilan silahkan pilih penulis dan klik radio button pdf



Setelah klik button download maka akan muncul tampilan yang mengarahkan untuk download data buku yang admin export ke pdf

The screenshot shows a Firefox browser window with the URL `localhost:8000/admin/export/books`. The page title is "Export Buku". A search bar contains the name "Mohammad Abdullah Rismanto". Below it, a radio button group labeled "Pilih Output" has "PDF" selected. A blue "DOWNLOAD" button is visible. A modal dialog box titled "Membuka books.pdf" is displayed, asking if the user wants to open or save the file. It specifies the file is a PDF document from the URL. The "Buka dengan Firefox" option is selected. At the bottom of the dialog are "Oke" and "Batal" buttons. The taskbar at the bottom of the screen shows several pinned icons and the date/time as 8:30 19/12/2020.

Data Buku E-Perpus Ceria

Judul	Jumlah	Stok	Penulis
Pemrograman Dasar Java	3	2	Mohammad Abdullah Rismanto

- Setelah itu , jika admin klik fitur navbar Member maka akan muncul tampilan data member dan dimana pada fitur member ini terdapat action yaitu tambah member, ubah, dan hapus.

Nama	Email	
ayuningtyas	ayuning@gmail.com	Ubah HAPUS
Chandra Risdianto	member4@gmail.com	Ubah HAPUS
Fatimah	member1@gmail.com	Ubah HAPUS
Qirani Aleyah	member5@gmail.com	Ubah HAPUS
Rugayah Syahrani	member3@gmail.com	Ubah HAPUS
Yusuf Mansyur	member2@gmail.com	Ubah HAPUS

Jika klik button tambah maka akan muncul tampilan seperti berikut



Berkas Edit Tampilan Bantuan Markah Alat Bantuan

localhost / 127.0.0.1 / perpusku E-Perpus Ceria +

Sering Mampir Perkenalan

E-Perpus Ceria Home Penulis Buku Members Peminjaman Profile Admin Perpus Ceria

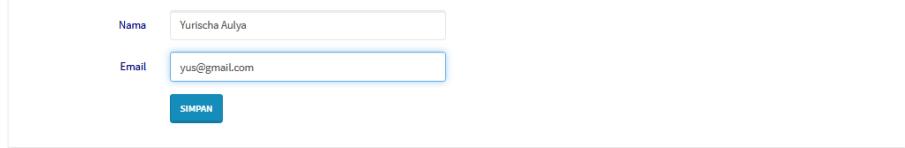
Home > Member > Tambah Member

Tambah Member

Nama Yurischa Aulya

Email yus@gmail.com

SIMPAN



- Jika klik ubah maka, akan muncul tampilan sebagai berikut

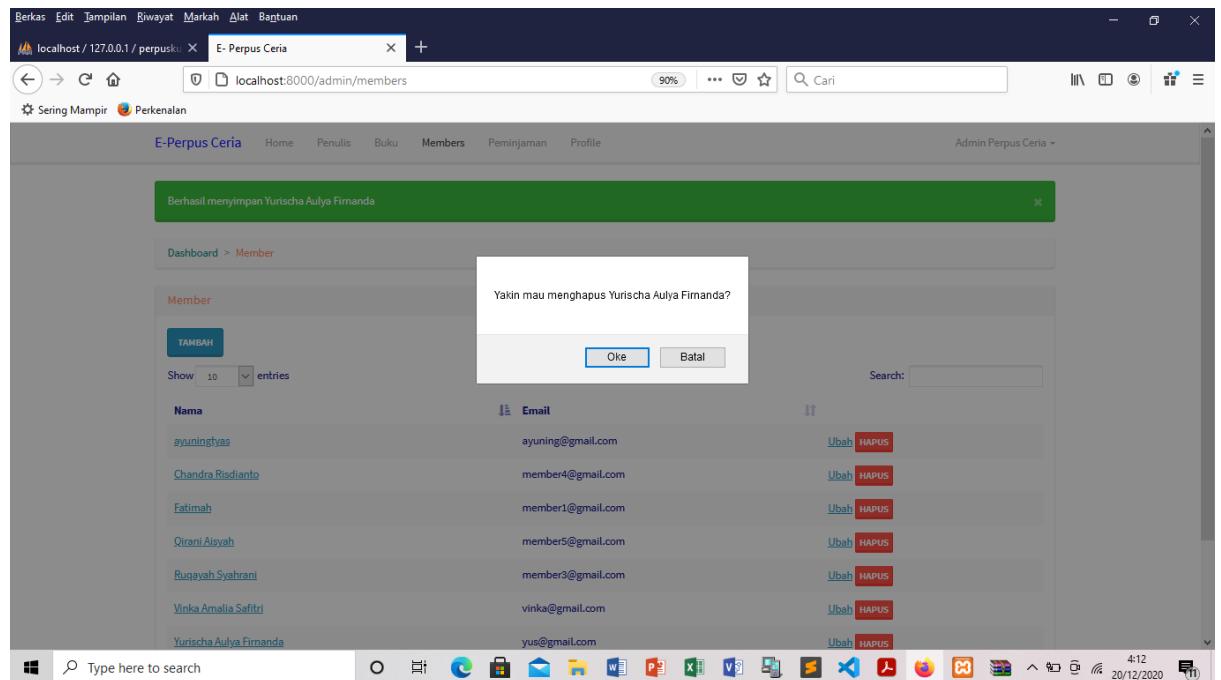
Ubah Member

Nama	<input type="text" value="Yurischa Aulya Firmanda"/>
Email	<input type="text" value="yus@gmail.com"/>
SIMPAN	

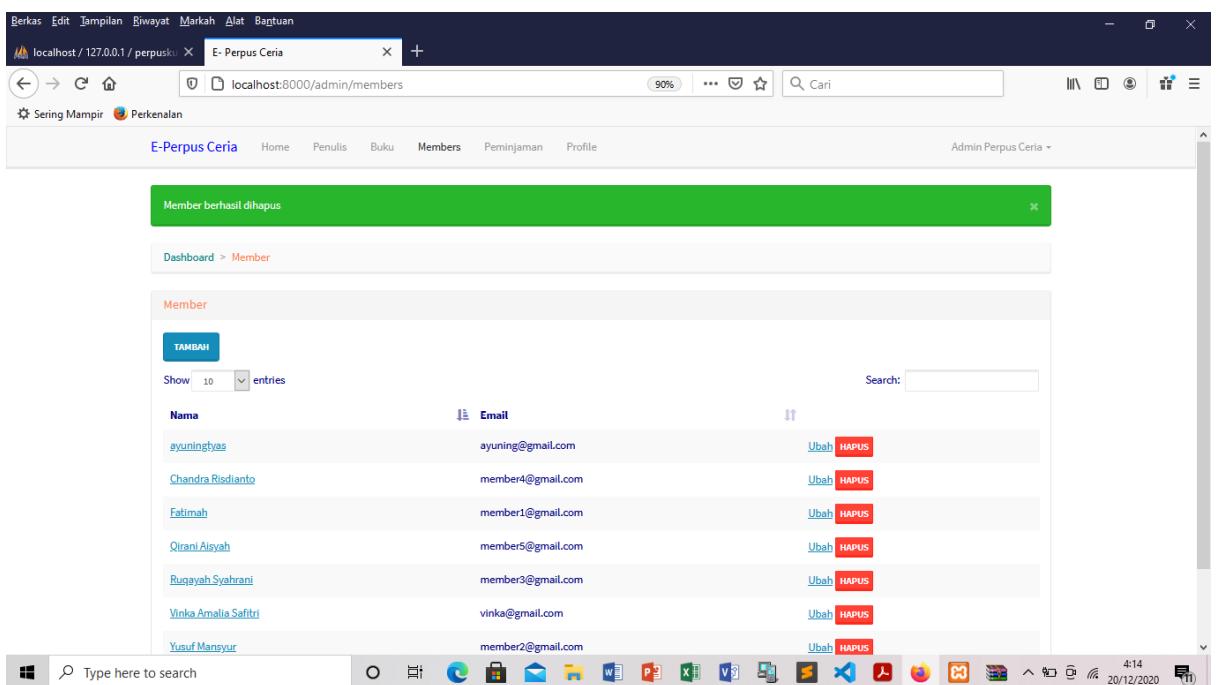
Berhasil menyimpan Yurischa Aulya Firmanda

Member		
TAMBAH	Show 10 entries	Search:
ayuningtyas	ayuning@gmail.com	Ubah HAPUS
Chandra Risdianto	member4@gmail.com	Ubah HAPUS
Fatimah	member1@gmail.com	Ubah HAPUS
Qirani Alisyah	member5@gmail.com	Ubah HAPUS
Ruqayah Syahrani	member3@gmail.com	Ubah HAPUS
Vinka Amalia Safitri	vinka@gmail.com	Ubah HAPUS
Yurischa Aulya Firmanda	yus@gmail.com	Ubah HAPUS

- Jika klik hapus maka akan muncul tampilan peringatan sebagai berikut



Dan jika klik “ ok ” maka akan muncul tampilan berikut



- Untuk fitur peminjaman, dalam fitur ini akan menampilkan data buku yang dipinjam, tanggal pinjam, tanggal kembali.

Judul	Peminjam	Tanggal Pinjam	Tanggal Kembali
Pemrograman Dasar Java	Fatimah	2020-12-18 21:53:13	Masih dipinjam

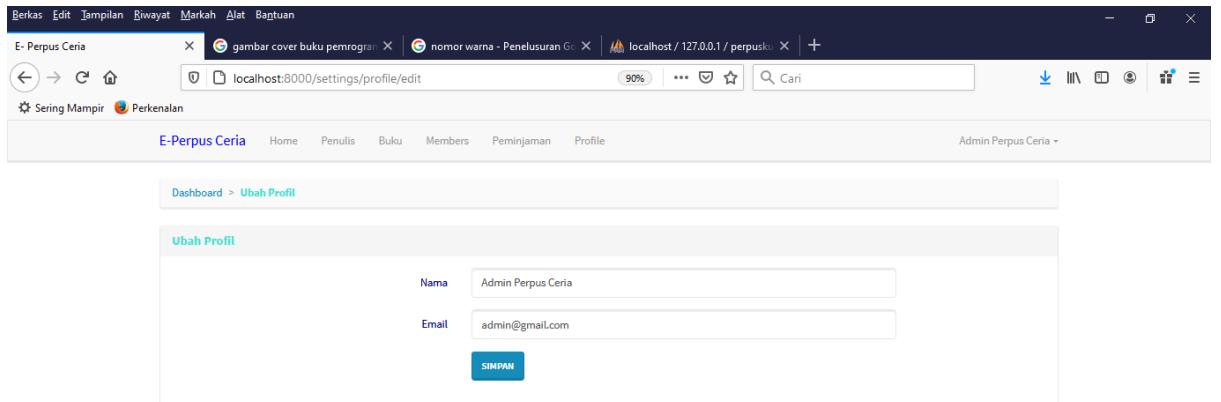


- Jika klik navbar Profil maka akan muncul tampilan yang menjelaskan nama,email, login terakhir.

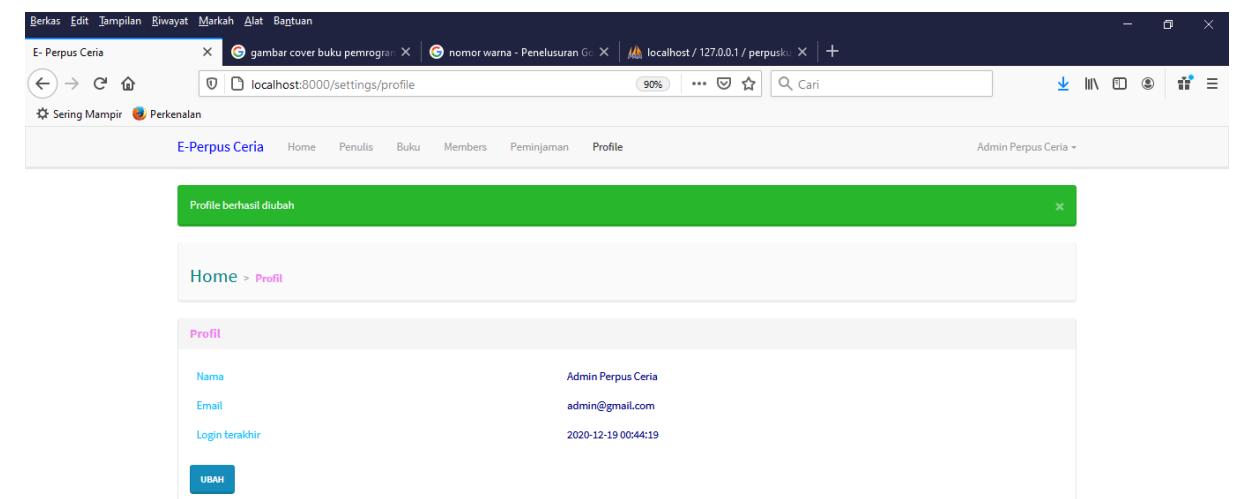
Nama	Admin Perpus Ceria
Email	admin@gmail.com
Login terakhir	2020-12-19 00:44:19



- Sedangkan , jika klik button ubah maka akan tampil form yang mana disitu dapat ubah nama, email lalu klik simpan



- Setelah klik simpan, maka akan ada nada peringatan seperti berikut



- Dan jika ingin ubah password maka klik ubah password , setelah itu akan muncul tampilan sebagai berikut.

Berkas Edit Tampilan Biwayat Markah Alat Bantuan

E-Perpus Ceria gambar cover buku pemrograman nomor warna - Penelusuran Google localhost / 127.0.0.1 / perpusceria

localhost:8000/settings/password 90% Cari

Sering Mampir Perkenalan E-Perpus Ceria Home Penulis Buku Members Peminjaman Profile Admin Perpus Ceria

Home > Ubah Password

Ubah Password

Password lama

Password baru

Konfirmasi password baru

SIMPAN



- Dan jika klik logout maka akan secara otomatis langsung keluar dari sistem ini

Berkas Edit Tampilan Biwayat Markah Alat Bantuan

localhost / 127.0.0.1 / perpusceria E-Perpus Ceria

localhost:8000/home 90% Cari

Sering Mampir Perkenalan E-Perpus Ceria Home Penulis Buku Members Peminjaman Profile Admin Perpus Ceria

Home

Ubah Password Logout

Selamat Datang di Menu Administrasi E-Perpus Ceria.

Statistik Penulis

Jumlah buku

localhost:8000/logout

Type here to search

4.Slesai.