

BAB IV

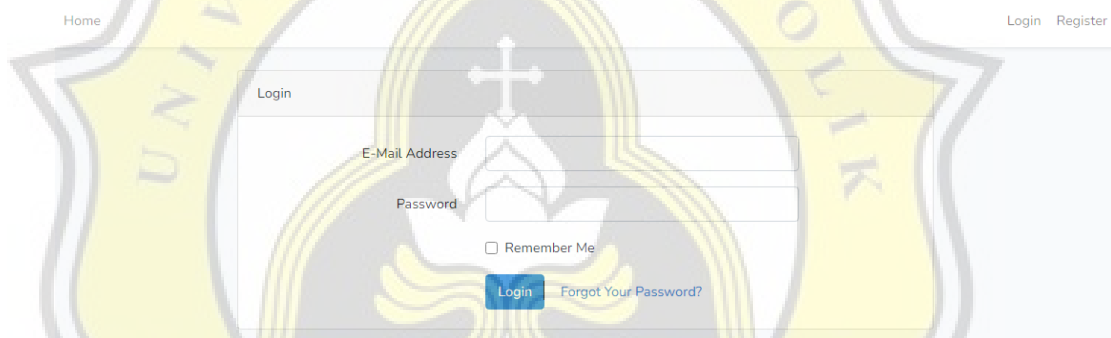
PERANCANGAN APLIKASI DAN PENGUJIAN

4.1 Perancangan Website

Website bertujuan untuk menampilkan berita game kepada pembaca. Dibuat menggunakan PHP dan API berbasis Laravel untuk mendapatkan data dan mengirimkan data ke database.

4.1.1 Halaman Login

Gambar 4.1 menunjukkan tampilan login. Login pada web ini menggunakan E-Mail dan password.



4.1 Halaman login

Kode 4.1 merupakan kode halaman login bagian validate form login yang digunakan untuk validasi email dan password sebelum login. Untuk kode lebih lengkap bisa dilihat dilampiran.

```
$this->validateLogin($request);
```

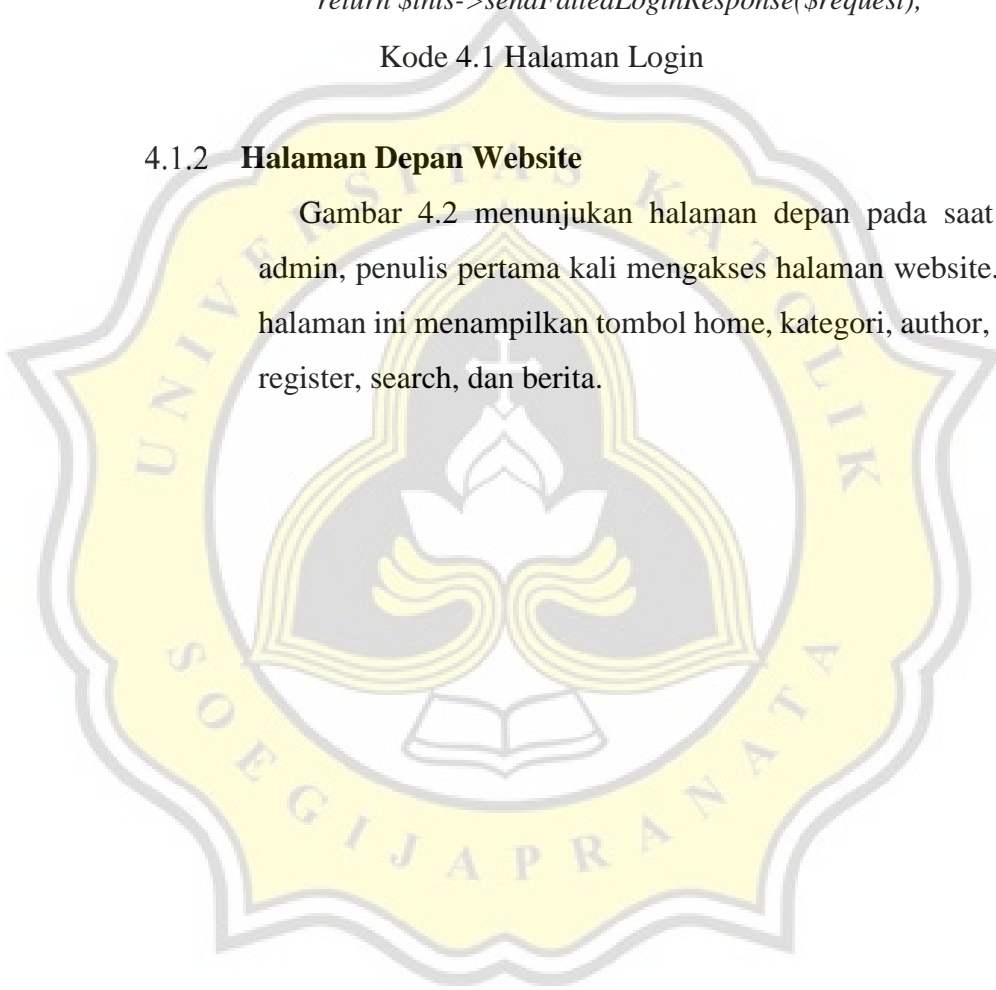
```
if (method_exists($this, 'hasTooManyLoginAttempts') &&  
    $this->hasTooManyLoginAttempts($request)) {  
    $this->fireLockoutEvent($request);  
  
return $this->sendLockoutResponse($request);
```

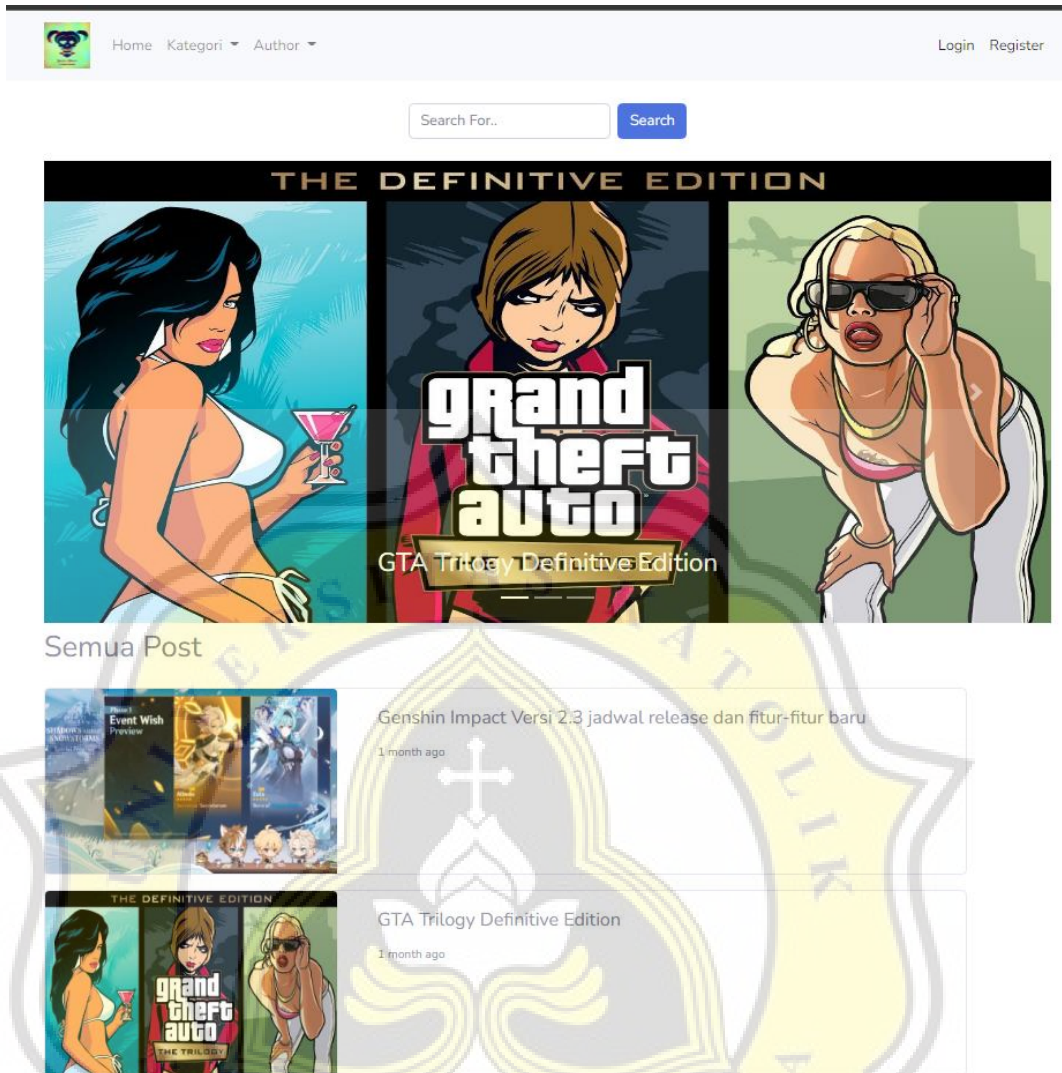
```
}  
  
if ($this->attemptLogin($request)) {  
    return $this->sendLoginResponse($request);  
}  
  
$this->incrementLoginAttempts($request);  
  
return $this->sendFailedLoginResponse($request);
```

Kode 4.1 Halaman Login

4.1.2 Halaman Depan Website

Gambar 4.2 menunjukkan halaman depan pada saat user, admin, penulis pertama kali mengakses halaman website. Pada halaman ini menampilkan tombol home, kategori, author, login, register, search, dan berita.





Gambar 4.2 Halaman depan website

Kode 4.2 merupakan kode untuk menampilkan berita pada halaman depan website. Untuk kode lebih lengkapnya bisa dilihat di lampiran.

```
public function index()
{
    $footer = Footer::select('konten')->first();
    $logo = logo::select('gambar')->first();
    $banner = banner::select('slug','gambar','judul')->latest()->get();
}
```

```

request()->session()->forget('search');

if(request()->search)
{
    $artikel =
post::select('gambar','judul','slug','created_at','status','konten')-
>where('status',1)->where('judul','LIKE','%'.request()->search.'%')-
>latest()->paginate(10);

    if(count($artikel) == 0)
    {
        request()->session()->flash('search','Post yang anda cari
tidak ada');
    }
    $search = request()->search;
}else{
    $artikel =
post::select('gambar','judul','slug','created_at','status','konten')-
>where('status',1)->latest()->paginate(10);
    $search="";
}

$kategori = Kategori::select('slug','nama')-
>orderBy('nama','asc')->get();

$home= true;

$author =User::getAdminPenulis();

return
view('artikel/index',compact('artikel','kategori','banner','logo','footer'
,'home','author','search'));
}

```

Kode 4.2 Halaman depan website

4.1.3 Halaman Artikel Berita

Gambar 4.3 menunjukkan tampilan artikel berita ketika memilih artikel yang ingin dibaca. Pada gambar 4.3 menampilkan cover artikel berita, kategori artikel, waktu upload artikel, tag artikel, pembuat artikel, dan isi artikel.



Gambar 4.3 Halama artikel berita

Kode 4.3 adalah kode yang bertujuan untuk menampilkan cover artikel berita, kategori artikel, waktu upload artikel, tag artikel, penulis artikel, dan isi artikel.

```

public function artikel($slug)
{
    $footer = Footer::select('konten')->first();
    $logo = logo::select('gambar')->first();
    $artikel =
post::select('id','judul','konten','id_kategori','created_at','gambar','id
_user',)->where('slug', $slug)->firstOrFail();

    $kategori = Kategori::select('slug','nama')-
>orderBy('nama','asc')->get();

    $author =User::getAdminPenulis();
    $like = like::where('id_post', $artikel->id)->count();

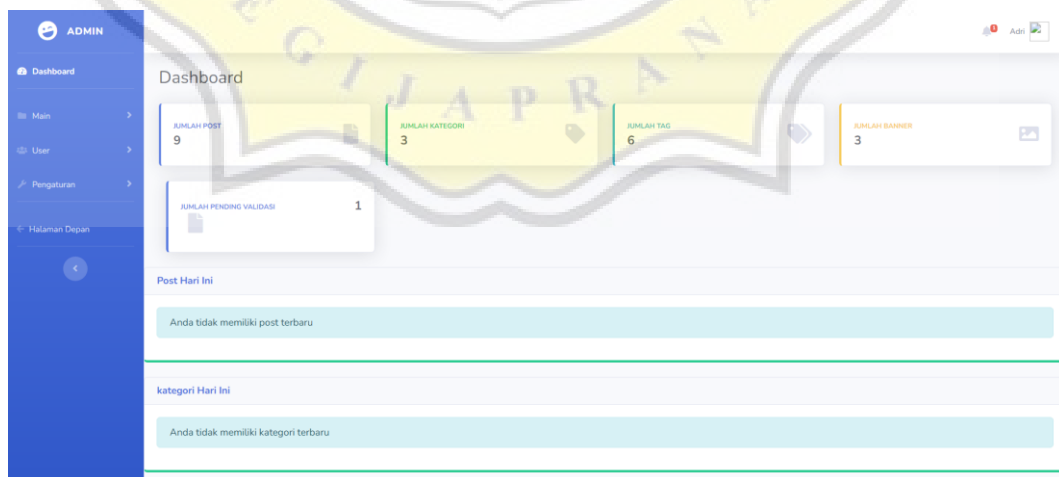
    return
view('artikel/artikel',compact('artikel','kategori','logo','footer','autho
r','like'));
}

```

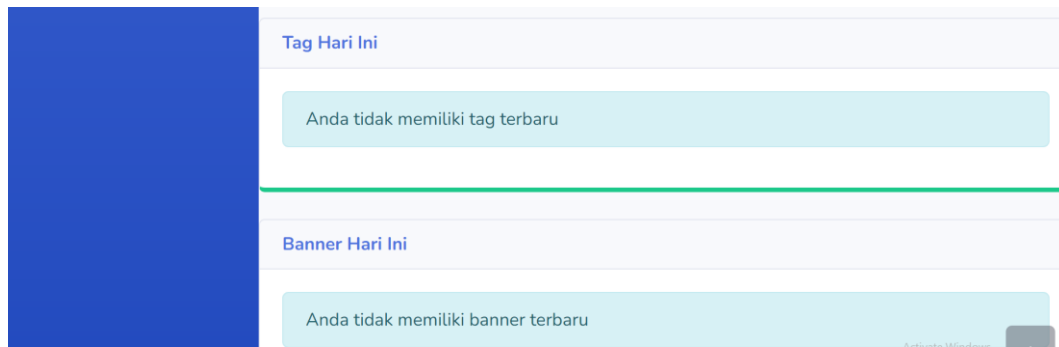
Kode 4.3 Halaman artikel berita

4.1.4 Halaman Dashboard

Gambar 4.4 dan gambar 4.5 menunjukkan tampilan halaman dashboard. Halaman dashboard dapat digunakan untuk melihat jumlah post, jumlah kategori, jumlah tag, jumlah banner, dan jumlah berita pending validasi, post hari ini, kategori hari ini, tag hari ini, dan banner hari ini.



Gambar 4.4 Halaman dashboard A



Gambar 4.5 Halaman dashboard B

Kode 4.4 bertujuan untuk menampilkan jumlah post artikel yang telah di publish di website. Untuk kode lengkap dashboard dapat dilihat di lampiran.

```

public function index()
{
    $jumlah_post = Post::count();
    $jumlah_kategori = Kategori::count();
    $jumlah_tag = Tag::count();
    $jumlah_banner = Banner::count();
    $jumlah_pending = post::where('status',0)->count();

    $hari_ini = Carbon::today();

    $post = Post::select('id','judul','id_kategori','gambar')-
    >whereDate('created_at',$hari_ini)->get();

    $kategori = Kategori::select('nama','slug')-
    >whereDate('created_at',$hari_ini)->get();

    $tag1 = Tag::select('nama','slug')-
    >whereDate('created_at',$hari_ini)->get();

    $banner = Banner::select('gambar','judul','slug')-
    >whereDate('created_at',$hari_ini)->get();
  
```

```

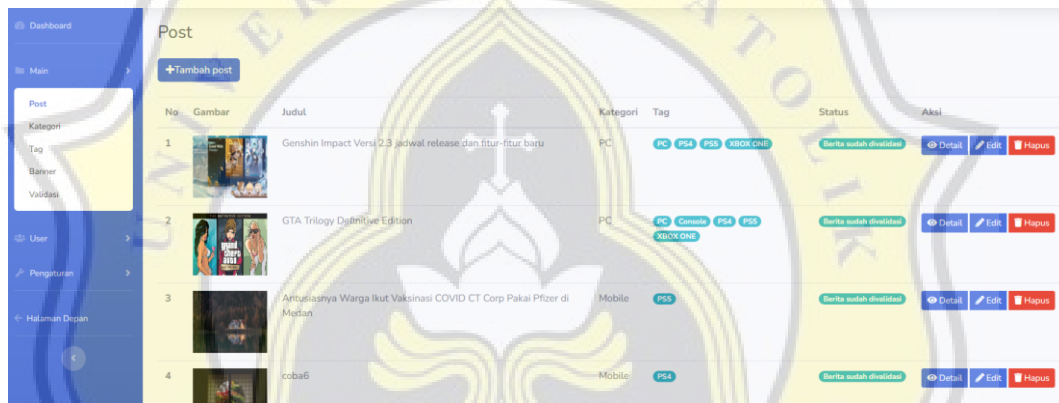
return view('admin/dashboard',
compact('jumlah_post','jumlah_kategori','jumlah_tag','jumlah_bann
er','jumlah_pending','post','kategori','tag1','banner'));
}

```

Kode 4.4 Jumlah post dashboard

4.1.5 Halaman Admin Post

Gambar 4.6 merupakan tampilan admin bagian post yang berisi mengenai artikel berita. Pada halaman ini berisi gambar artikel, judul artikel, kategori artikel, tag artikel, status validasi artikel, view, edit, dan delete.



Gambar 4.6 Halaman admin post

Kode 4.5 merupakan kode yang digunakan untuk menampilkan data pada halaman admin post. Untuk kode lebih lengkap dapat dilihat di lampiran.

```

public function index()
{
$search="";
if(request()->search)
{
$post =
Post::select('id','judul','slug','gambar','id_kategori','status')-
>where('id_user',Auth::user()->id)->where('judul','LIKE','%'.
request()->search.'%')->latest()->paginate(10);
$search = request()->search;
}
}

```



```

if(count($post)==0){
    request()->session()->flash('search','
<div class="alert alert-success mt-4" role="alert">
    Data yang anda cari tidak ditemukan
</div>
');
}
}
else{
    $post =
Post::select('id','judul','slug','gambar','id_kategori','status')-
>where('id_user',Auth::user()->id)->latest()->paginate(10);
}

return view('admin/post/index',compact('post','search'));
}

```

Kode 4.5 Halaman admin post

4.1.6 Halaman tambah post

Gambar 4.7 menunjukkan tampilan halaman tambah post. Halaman ini menampilkan judul, gambar, kategori, tag, dan konten artikel yang berfungsi untuk diisi oleh jurnalis.

Post

Judul

Gambar

Choose File No file chosen

Kategori

Pilih Kategori

Tag

PC
Console
PS4
PS5

Konten

Paragraph

Submit Back

Gambar 4.7 Halaman tambah post

Kode 4.6 merupakan kode yang berfungsi untuk menampilkan form yang berisi judul dan gambar. Untuk kode yang menampilkan kategori, tag, dan konten dapat dilihat di lampiran.

```

<div class="form-group">
  <label for="Judul">Judul</label>
  <input type="text" class="form-control" id="judul"
  name="judul" value="{{old('judul')}}">
  @error('judul')
    <small> <p class="text-danger">{{ $message
  }}</p></small>
  @enderror
</div>

```

```

<div class="form-group">
  <label for="gambar">Gambar</label>
  <input type="file" class="form-control" id="gambar"
name="gambar">
  @error('gambar')
    <small> <p class="text-danger">{{ $message
}}</p></small>
  @enderror
</div>

```

Kode 4.6 Halaman tambah post

4.1.7 Halaman view artikel

Gambar 4.8 merupakan tampilan view artikel yang berisi cover artikel dan isi artikel.



Gambar 4.8 Halaman view artikel

Kode 4.7 merupakan kode yang berfungsi untuk menampilkan cover dan isi artikel pada halaman view artikel. Untuk lebih lengkapnya dapat dilihat di lampiran.

```

public function show($id)
{
  $post = Post::select('id','judul','gambar','konten','created_at')-
->whereId($id)->firstOrFail();

```

```

return view('admin/post/detail',compact('post'));
}

```

Kode 4.7 Halaman view artikel

4.1.8 Halaman Edit Post

Gambar 4.8 menampilkan form edit post. Pada tampilan ini berisikan judul artikel, gambar artikel, kategori artikel, tag artikel, dan konten artikel.

The screenshot shows a web form for editing a post. The form is titled 'Post' and contains the following fields:

- Judul:** A text input field containing 'Genshin Impact Versi 2.3 jadwal release dan fitur-fitur baru'.
- Gambar:** A file upload field with a 'Choose File' button and the text 'No file chosen'.
- Kategori:** A dropdown menu with 'PC' selected.
- Tag:** A dropdown menu with 'PC', 'Console', 'PS4', and 'PS5' listed.
- Konten:** A rich text editor with a paragraph of text:

Game Genshin Impact akan segera melangsungkan update versi 2.3 pada akhir November 2021 ini. Berdasarkan live streaming Program Khusus Genshin Impact 2.3 yang berlangsung pada Jumat, 12 November 2021, akan ada beberapa konten baru yang ditambahkan. Salah satunya adalah perilsan banner baru dari karakter Itto*Arataki Itto, Gorou, Albedo, serta Eula. Ini akan menjadi debut perdana Itto*Arataki Itto dan Gorou sebagai karakter playable Genshin Impact. Meski sebenarnya kedua karakter ini sudah dikenal oleh sebagian besar pemain karena namanya sering disebut di dalam game. Dalam live streaming Program Khusus versi 2.3, disebutkan bila Albedo dan Eula akan hadir sebagai karakter rerun di banner Event Wish fase pertama.

Gambar 4.9 Halaman edit post

Kode 4.8 merupakan kode yang berfungsi untuk menampilkan data artikel yang dipilih untuk diedit. Untuk kode tampilan UI dapat dilihat pada lampiran.

```

public function edit($id)
{
    $tag = Tag::select('id','nama')->get();
    $kategori = Kategori::select('id','nama')->get();
}

```

```

$post =
Post::select('id','judul','gambar','konten','id_kategori','status')-
>whereId($id)->firstOrFail();

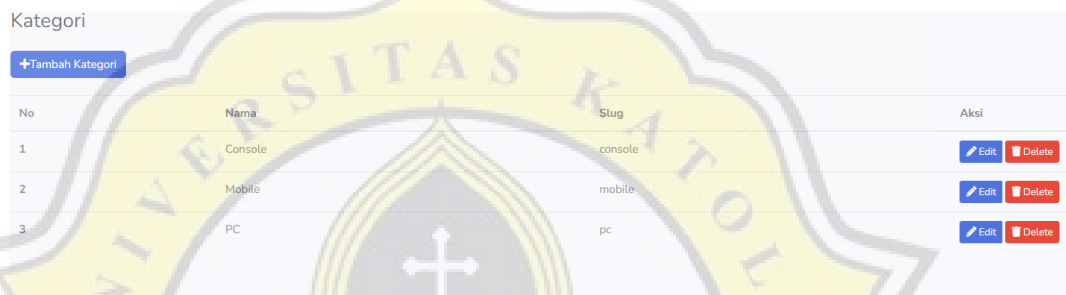
return view('admin/post/edit',compact('post','kategori','tag'));
}

```

Kode 4.8 Halaman edit post

4.1.9 Halaman Daftar Kategori

Gambar 4.10 menampilkan daftar kategori yang berisi nama kategori, slug, edit, dan hapus kategori.



No	Nama	Slug	Aksi
1	Console	console	Edit Delete
2	Mobile	mobile	Edit Delete
3	PC	pc	Edit Delete

Gambar 4.10 Halaman daftar kategori

Kode 4.9 merupakan kode yang digunakan untuk menampilkan daftar kategori yang berisi nama, dan slug kategori.

```

public function index()
{
    $kategori = Kategori::select('id','nama','slug')->latest()-
    >Paginate(10);

    $search="";
    if(request()->search)
    {
        $kategori = Kategori::select('id','nama','slug')-
        >where('nama','LIKE','%. request()->search. %')->latest()-
        >Paginate(10);

        $search = request()->search;

        if(count($kategori) == 0){

```

```

request()->session()->flash('search','
<div class="alert alert-success mt-4" role="alert">
    Data yang anda cari tidak ditemukan
</div>
');
}
}
else{
    $kategori = Kategori::select('id','nama','slug')->latest()-
>Paginate(10);
}
return
view('admin/kategori/index',compact('kategori','search'));
}

```

Kode 4.9 Halaman daftar kategori

4.1.10 Halaman Tambah Kategori

Pada gambar 4.11 menampilkan form tambah kategori yang berisi nama kategori saja.

The image shows a web form with the title 'Kategori'. Below the title is a text input field with the placeholder text 'Kategori'. At the bottom of the form, there are two buttons: a blue 'Submit' button and a grey 'Back' button.

Kode 4.10 merupakan kode yang digunakan untuk menampilkan form tambah kategori. Untuk kode validasi dan add kategori dapat dilihat di lampiran.

```

@extends('sb-admin/app')
@section('title','Kategori')

```



```

@section('kategori','active')
@section('main','show')
@section('main-active','active')
@section('content')
<!-- Page Heading -->
<h1 class="h3 mb-4 text-gray-800">Kategori</h1>

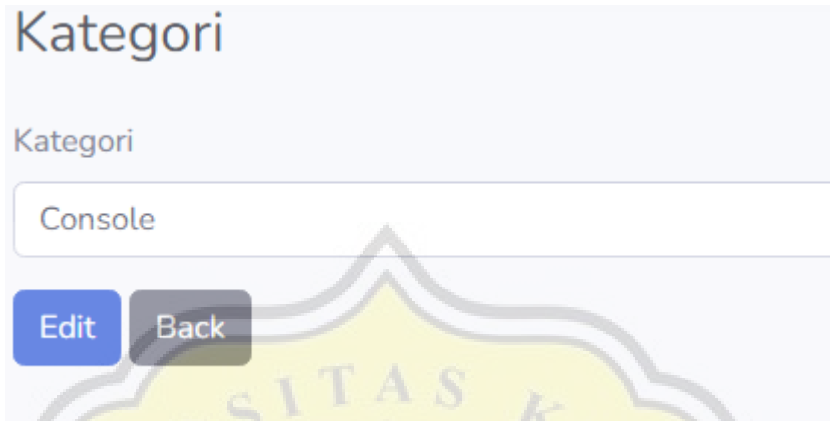
<form action="/kategori" method="POST">
@csrf
<div class="form-group">
<label for="nama">Kategori</label>
<input type="text" class="form-control" id="nama"
name="nama">
@error('nama')
<small> <p class="text-danger">{{ $message
}}</p></small>
@enderror
</div>
<button type="submit" class="btn btn-
primary">Submit</button>
<a href="/kategori" class="btn btn-secondary">Back</a>
</form>
@endsection

```

Kode 4.10 Halaman tambah kategori

4.1.11 Halaman Edit Kategori

Pada gambar 4.12 merupakan tampilan form edit kategori yang hanya berisi nama kategori.



Gambar 4.11 Halaman edit kategori

Kode 4.11 merupakan kode yang berfungsi untuk menampilkan data kategori yang dipilih untuk diedit. Kode UI dan update dapat dilihat pada lampiran.

```
public function edit($id)
{
    $kategori = Kategori::select('id','nama')->whereId($id)->first();
    return view('admin/kategori/edit',compact('kategori'));
}
```

Kode 4.11 Halaman edit kategori

4.1.12 Halaman Daftar Tag

Pada gambar 4.13 menampilkan daftar tag yang datanya berisi nama tag, slug, tombol edit, dan tombol delete.



No	Nama	Slug	Aksi
1	PC	pc	Edit Delete
2	Console	console	Edit Delete
3	PS4	ps4	Edit Delete
4	PSS	ps5	Edit Delete
5	Mobile	mobile	Edit Delete
6	XBOX ONE	xbox-one	Edit Delete

Gambar 4.13 Halaman daftar tag

Kode 4.12 merupakan kode yang berfungsi untuk menampilkan data daftar tag. Untuk kode UI dapat dilihat pada lampiran.

```
public function index()
{
    $search="";
    if(request()->search)
    {
        $tag = Tag::select('id','nama','slug')-
        >where('nama','LIKE','%'. request()->search. '%')->paginate(10);
        $search = request()->search;
        if(count($tag) == 0){
            request()->session()->flash('search','
            <div class="alert alert-success mt-4" role="alert">
                Data yang anda cari tidak ditemukan
            </div>
            ');
        }
    }
    else{
        $tag = Tag::select('id','nama','slug')->paginate(10);
    }
}
```

```

    }

    return view('admin/tag/index', compact('tag','search'));
}

```

Kode 4.12 Halaman daftar tag

4.1.13 Halaman Tambah Tag

Pada gambar 4.14 menampilkan form tambah tag yang hanya menampilkan nama tag saja untuk diisi.



Gambar 4.14 Halaman tambah tag

Kode 4.13 merupakan kode yang berfungsi untuk menampilkan form tambah. Untuk kode validasi dan tambah tag dapat dilihat di lampiran.

```

@extends('sb-admin/app')

@section('title','Tag')

@section('tag','active')

@section('main','show')

@section('main-active','active')

@section('content')

<!-- Page Heading -->

```

```
<h1 class="h3 mb-4 text-gray-800">Tag</h1>
```

```
<form action="/tag" method="POST">
```

```
@csrf
```

```
<div class="form-group">
```

```
<label for="nama">Tag</label>
```

```
<input type="text" class="form-control" id="nama" name="nama">
```

```
@error('nama')
```

```
<small> <p class="text-danger">{{ $message }}</p></small>
```

```
@enderror
```

```
</div>
```

```
<button type="submit" class="btn btn-primary">Submit</button>
```

```
<a href="/tag" class="btn btn-secondary">Back</a>
```

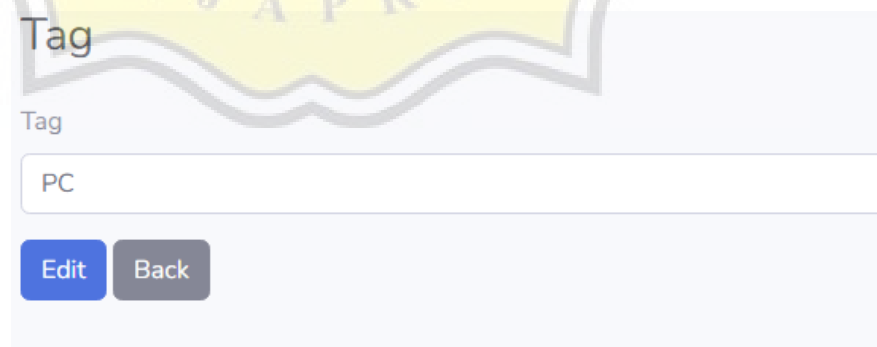
```
</form>
```

```
@endsection
```

Kode 4.13 Halaman tambah tag

4.1.14 Halaman Edit Tag

Pada gambar 4.15 menampilkan form edit tag, pada form ini hanya menampilkan nama tag untuk diedit.



```
Tag
Tag
PC
Edit Back
```

Gambar 4.15 Halaman edit tag

Kode 4.14 merupakan kode yang berfungsi untuk menampilkan data yang dipilih untuk diedit. Untuk kode UI dan update tag dapat dilihat di lampiran.




```
public function edit($id)
{
    $tag = Tag::select('id','nama')->whereId($id)->firstOrFail();
    return view('admin/tag/edit',compact('tag'));
}
```

Kode 4.14 Halaman edit tag

4.1.15 Halaman Daftar Banner

Pada gambar 4.16 merupakan tampilan data daftar banner. Tampilan ini menampilkan data berupa gambar banner, judul banner, slug banner, tombol detail, tombol edit, dan tombol hapus.



No	Gambar	Judul	Slug	Aksi
1		GTA Trilogy Definitive Edition	gta-trilogy-definitive-edition	Detail Edit Hapus
2		Genshin Impact Versi 2.3 jadwal release dan fitur-fitur baru	genshin-impact-versi-23-jadwal-release-dan-fitur-fitur-baru	Detail Edit Hapus
3		Banner1	banner1	Detail Edit Hapus

Gambar 4.16 Halaman Daftar Banner

Kode 4.15 merupakan kode yang berfungsi untuk menampilkan data pada halaman daftar banner. Untuk kode tampilan UI dapat dilihat di lampiran.

```
public function index()
{
    $footer = Footer::select('konten')->first();
    $banner = Banner::select('id','slug','gambar','judul')->latest()->paginate(10);
}
```



```

$search="";
if(request()->search)
{
    $banner = Banner::select('id','slug','gambar','judul')-
>where('judul','LIKE','%'. request()->search.'%')->latest()-
>paginate(10);

    $search = request()->search;

    if(count($banner) == 0){
        request()->session()->flash('search','
<div class="alert alert-success mt-4" role="alert">
    Data yang anda cari tidak ditemukan
</div>
');
    }
    else{
        $banner = Banner::select('id','slug','gambar','judul')-
>latest()->paginate(10);
    }

    return
view('admin/banner/index',compact('banner','footer','search'));
}

```

Kode 4.15 Halaman daftar banner

4.1.16 Halaman Tambah Banner

Pada gambar 4.17 menampilkan form tambah banner yang berisi judul banner, gambar banner, dan konten banner.



Banner

Judul

Gambar

Choose File No file chosen

Konten

Submit Back

Gambar 4.17 Halaman tambah banner

Kode 4.16 merupakan kode untuk memasukan data ke database. Untuk kode yang menampilkan UI tambah banner dapat dilihat di lampiran.

```
public function store(Request $request)
{
    $request->validate([
        'judul' => 'required',
        'gambar' => 'required|mimes:jpg,jpeg,bmp,png',
        'konten' => 'required',
    ]);
}
```

```
$gambar = $request->gambar->getClientOriginalName();  
$request->gambar->move('uploads/banner',$gambar);
```

```
Banner::create([  
    'gambar'=>$gambar,  
    'judul'=>$request->judul,  
    'konten'=>$request->konten,  
  
    'slug' => Str::slug($request->judul,'-'),  
]);
```

```
Alert::success('Success', 'Data berhasil ditambahkan');
```

```
return redirect('/banner');
```

```
}
```

Kode 4.16 Halaman tambah banner

4.1.17 Halaman Detail Banner

Pada gambar 4.18 menampilkan cover banner dan isi banner. Pada tampilan ini sebenarnya sama pada detail artikel hanya saja berbeda pada penempatan artikel.



Gambar 4.18 Halaman detail banner

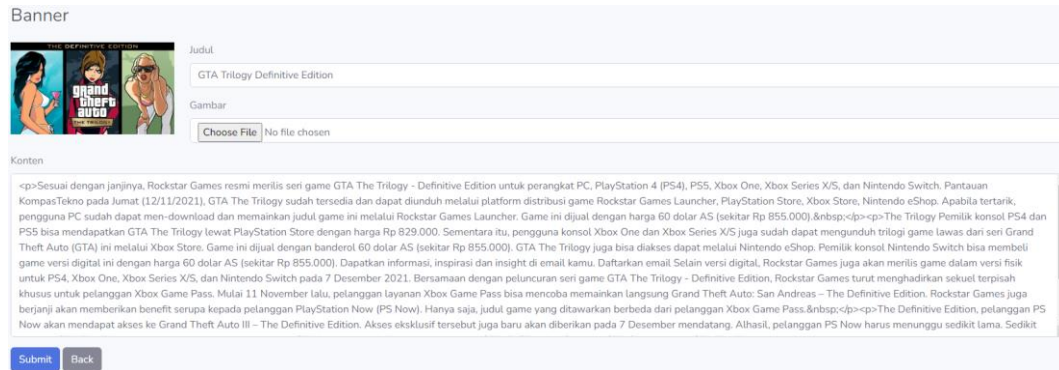
Pada kode 4.17 merupakan kode untuk menampilkan cover banner dan isi banner. Untuk kode UI detail banner dapat dilihat pada lampiran.

```
public function show($id)
{
    $footer = Footer::select('konten')->first();
    $banner = banner::select('id','judul','gambar','konten','created_at')->whereId($id)->firstOrFail();
    return view('admin/banner/detail',compact('banner','footer'));
}
```

Kode 4.17 Halaman detail banner

4.1.18 Halaman Edit Banner

Pada gambar 4.19 merupakan tampilan edit banner yang menampilkan form berisi judul banner, gambar banner, dan konten banner. Ketiga data tersebut dapat diupdate.



Gambar 4.19 Halaman Edit Banner


Kode 4.18 merupakan kode untuk menampilkan data yang dipilih untuk diedit, Untuk kode tampilan UI dan Update dapat dilihat di lampiran.

```
public function edit($id)
{
    $footer = Footer::select('konten')->first();
    $banner =
    banner::select('id','judul','gambar','konten','created_at')-
    >whereId($id)->firstOrFail();
    return view('admin/banner/edit',compact('banner','footer'));
}
```

Kode 4.18 Halaman edit banner

4.1.19 Halaman Daftar Validasi

Pada gambar 4.20 menampilkan daftar artikel yang belum divalidasi. Ketika artikel telah divalidasi, maka artikel akan langsung terpublish di website. Data yang ditampilkan pada daftar validasi yaitu gambar, judul artikel, kategori artikel, tag artikel, status validasi, dan tombol validasi.

No	Gambar	Judul	Kategori	Tag	Status	Aksi
1		coba5a	PC	PC	Berita belum divalidasi	Validasi

Gambar 4.20 Halaman daftar validasi

Pada kode 4.19 merupakan kode yang berfungsi untuk menampilkan daftar artikel yang belum divalidasi. Untuk kode yang lebih lengkap dapat dilihat di lampiran.

```

public function index()
{
    $search="";
    if(request()->search)
    {
        $post = Post::select('id','judul','slug','gambar','id_kategori')-
        >where('status',0)->where('id_user',Auth::user()->id)-
        >where('judul','LIKE','%'. request()->search.'%')->latest()-
        >paginate(10);

        $search = request()->search;

        if(count($post)==0){
            request()->session()->flash('search','
            <div class="alert alert-success mt-4" role="alert">
                Data yang anda cari tidak ditemukan
            </div>
            ');
        }
    }
    else{
        $post = Post::select('id','judul','slug','gambar','id_kategori')-
        >where('status',0)->where('id_user',Auth::user()->id)->latest()-
        >paginate(10);
    }
}

```



```
return view('admin/validasi/index',compact('post','search'));  
}
```

Kode 4.19 Halaman validasi

4.2 Hasil Pengujian

Setelah membagikan kuesioner dan mendapatkan 30 responden maka dilakukan uji data menggunakan aplikasi SPSS untuk uji validitas, uji reliabilitas, dan uji korelasi. Uji validitas berfungsi untuk menguji ketepatan variabel dalam penelitian. Uji realibilitas adalah uji yang digunakan untuk mengetahui konsistensi alat ukur sebuah alat uji. Uji korelasi digunakan untuk menguji korelasi antara variabel.

4.2.1 Uji Validitas

Validitas diambil dari kata *validity* yang mempunyai arti sejauh mana kecermatan dan keakuratan suatu alat ukur dalam melakukan fungsi ukuranya[7]. Sedangkan menurut Sugiharto dan Sitinjak, validitas suatu peubah mengukur untuk apa yang seharusnya diukur[8]. Validitas dalam penelitian bertujuan untuk menyatakan derajat ketepatan alat ukur dalam mengukur isi suatu penelitian.

Uji validitas digunakan untuk mengukur valid atau tidaknya suatu kuesioner. Suatu kuesioner dinyatakan valid jika kuesioner tersebut mampu mengungkapkan sesuatu yang diukur oleh kuesioner itu[9].

Pada tabel 4.1 menunjukkan tabel hasil uji pertama yaitu variabel PE, EE, E, FC, BI. Dikarenakan indikator PE3, PE4, dan E1 tidak mengelompok dan bernilai dibawah 0.4 maka harus dihapus dan dilakukan uji ulang.

Tabel 4.1 Hasil uji validitas pertama

Component Matrix^a

	Component				
	1	2	3	4	5
PE1	.701	.041	.314	-.280	-.185
PE2	.763	-.193	-.056	-.424	.039
PE3	.140	.027	.753	-.055	.468
PE4	.338	.616	.354	-.281	-.089
EE1	.509	-.515	-.179	.174	.181
EE2	.554	.010	.025	-.430	-.324
EE3	.686	-.275	-.117	-.058	-.100
E1	.351	.451	.334	.396	-.491
E2	.543	-.593	.234	-.013	.166
E3	.648	.293	-.040	-.190	.295
FC1	.436	.327	.270	.591	.109
FC2	.524	-.357	.239	.413	.109
FC3	.584	-.508	-.113	.258	-.404
BI1	.448	.349	-.514	.122	.244
BI2	.602	.478	-.313	.177	.010
BI3	.647	.345	-.341	.026	.198

Extraction Method: Principal Component Analysis.

a. 5 components extracted.

Setelah menghapus variabel PE3, PE4, dan E1 maka dilakukan pengujian ulang. Ditemukan bahwa FC1 memiliki nilai dibawah 0.4 dan tidak mengelompok sehingga harus dihapus dan melakukan uji ulang.

Tabel 4.2 Hasil uji validitas kedua

Component Matrix^a

	Component		
	1	2	3
PE1	.666	-.073	-.294
PE2	.781	-.118	-.394
EE1	.574	-.361	.295
EE2	.551	.036	-.543
EE3	.717	-.196	.004
E2	.585	-.559	.006
E3	.626	.385	-.173
FC1	.382	.295	.570
FC2	.537	-.374	.468
FC3	.629	-.438	.129
BI1	.443	.555	.106
BI2	.571	.610	.151
BI3	.633	.511	.082

Extraction Method: Principal Component Analysis.

a. 3 components extracted.

Setelah menghapus variabel FC1 maka dilakukan uji ulang dan menemukan hasil bahwa semua variabel telah mengelompok dan memiliki nilai diatas 0.4 sehingga dinyatakan valid.

Tabel 4.3 Hasil uji validitas ketiga

Component Matrix^a

	Component		
	1	2	3
PE1	.671	-.039	-.371
PE2	.803	-.045	-.284
EE1	.573	-.375	.261
EE2	.564	.097	-.564
EE3	.724	-.176	-.040
E2	.605	-.531	.072
E3	.616	.409	-.159
FC2	.533	-.402	.462

FC3	.639	-.429	.133
B11	.442	.593	.447
B12	.544	.603	.210
B13	.614	.516	.140

Extraction Method: Principal Component Analysis.

a. 3 components extracted.

4.2.2 Uji Reliabilitas

Reliabilitas berasal dari kata reliability yang memiliki arti keajegan pengukuran[10]. Reliabilitas juga menunjukkan pada suatu pengertian bahwa instrumen yang digunakan untuk mendapatkan informasi dan dapat dipercaya sebagai alat pengumpulan data dan dapat mengungkapkan informasi yang ada dilapangan[11]. Reliabilitas adalah alat untuk mengukur kuesioner yang merupakan indikator dari peubah[12]. Reliabilitas juga menunjukkan hasil pengujian dengan alat harus dapat dipercaya[13]. Hasil pengukuran harus reliabel dan harus memiliki tingkat konsistensi yang tinggi. Uji reliabilitas dapat dilakukan secara bersamaan terhadap semua variabel dalam kuesioner[14]. Dasar pengambilan keputusan dalam uji reliabilitas sebagai berikut:

1. Jika nilai Cronbach's alpha $>0,60$ maka kuesioner dinyatakan reliabel.
2. Jika nilai Cronbach's alpha $< 0,60$ maka kuesioner dinyatakan tidak reliabel.

Tabel 4.4 adalah hasil uji reliabilitas terhadap variabel PE yang memiliki nilai 0.695.

Tabel 4.4 Hasil uji reliabilitas PE

Reliability Statistics	
Cronbach's Alpha	N of Items
.695	2

Tabel 4.5 adalah hasil uji reliabilitas terhadap variabel EE yang memiliki nilai 0.555.

Tabel 4.5 Hasil uji reliabilitas EE

Reliability Statistics	
Cronbach's Alpha	N of Items
.555	3

Tabel 4.6 adalah hasil uji reliabilitas terhadap variabel E yang memiliki nilai 0.350.

Tabel 4.6 Hasil uji reliabilitas E

Reliability Statistics	
Cronbach's Alpha	N of Items
.350	2

Tabel 4.7 adalah hasil uji reliabilitas terhadap variabel FC yang memiliki nilai 0.630.

Tabel 4.7 Hasil uji reliabilitas FC

Reliability Statistics	
Cronbach's Alpha	N of Items
.630	2

Tabel 4.8 adalah hasil uji reliabilitas terhadap variabel BI yang memiliki nilai 0.783.

Tabel 4.8 Hasil uji reliabilitas BI

Reliability Statistics	
Cronbach's Alpha	N of Items
.783	3

Tabel dibawah merupakan nilai rentang pengujian dari uji reliabilitas terhadap suatu variabel.

Cronbach's Alpha	Internal Consistency
$\alpha \geq 0.9$	Excellent
$0.8 \leq \alpha < 0.9$	Good
$0.7 \leq \alpha < 0.8$	Acceptable
$0.6 \leq \alpha < 0.7$	Questionable
$0.5 \leq \alpha < 0.6$	Poor
$\alpha < 0.5$	Unacceptable

Tabel 4.9 Tabel rentang uji reliabilitas

Pada tabel 4.10 menunjukkan bahwa PE bernilai questionable, EE bernilai poor yang berarti tidak dapat digunakan kembali pada penelitian sebelumnya, E bernilai Unacceptable, FC bernilai questionable, dan BI bernilai acceptable yang berarti konsisten dan dapat digunakan kembali dalam penelitian berikutnya.

Tabel 4.10 Tabel nilai pengujian reliabilitas

Variabel	Cronbach's alpha	Internal Consistency
PE	0.695	Questionable
EE	0.555	Poor
E	0.350	Unacceptable
FC	0.630	Questionable
BI	0.783	Acceptable

4.2.3 Uji Korelasi

Korelasi dalam matematika dapat diartikan ukuran dari seberapa dekat antara dua variabel. Analisis korelasi dalam ilmu statistika memiliki 9 jenis, yaitu: pearson product moment (r), korelasi ratio (y), korelasi spearman rank atau rho (rs atau p), korelasi point biserial (rpb), korelasi biserial (rb), korelasi phi (0), korelasi tetrachoric (rt), korelasi contingency (C), dan korelasi kendall's tau (8) [12]. Dalam penelitian ini menggunakan uji korelasi pearson product moment.

Tabel 4.11 menunjukkan hasil uji korelasi variabel, dimana variabel PE, EE, E, dan FC akan dikorelasikan dengan variabel BI

Tabel 4.11 Tabel uji korelasi

		TotalPE	TotalEE	TotalE	TotalFC	TotalBI
TotalPE	Pearson Correlation	1	.469**	.596**	.295	.320
	Sig. (2-tailed)		.007	.000	.101	.074
	N	32	32	32	32	32
TotalE	Pearson Correlation	.596**	.434*	1	.596**	.420*
	Sig. (2-tailed)	.000	.013		.000	.017
	N	32	32	32	32	32
TotalFC	Pearson Correlation	.295	.583**	.596**	1	.328
	Sig. (2-tailed)	.101	.000	.000		.067
	N	32	32	32	32	32
TotalBI	Pearson Correlation	.320	.351*	.420*	.328	1
	Sig. (2-tailed)	.074	.049	.017	.067	
	N	32	32	32	32	32

** . Correlation is significant at the 0.01 level (2-tailed).

* . Correlation is significant at the 0.05 level (2-tailed).

- Dari tabel uji, nilai arah hipotesis (Sig. (2-tailed) menunjukkan $PE < 0.374$ dan nilai Pearson Correlation 0.320 yang berarti PE berkorelasi signifikan dengan BI.
- Dari tabel uji, nilai arah hipotesis (Sig. (2-tailed) menunjukkan $E > 0.374$ dan nilai Pearson Correlation 0.420 yang berarti E tidak berkorelasi signifikan dengan variabel BI

- Dari tabel uji, nilai arah hipotesis (Sig. (2-tailed) menunjukkan $FC < 0.374$ dan nilai Pearson Correlation 0.328 yang berarti FC berkorelasi signifikan dengan variabel BI.

