2602070383 - Bernard Bereness
LC01

Routing

```
Route::get('/', [NavigationController::class, 'indexPage'])->name('indexPage');
Route::get('/category/{category_name}', [NavigationController::class, 'categoryPage'])->name('categoryPage');
Route::get('/course/{course_name}', [NavigationController::class, 'courseDetail'])->name('courseDetail');
Route::get('/writers', [NavigationController::class, 'writersPage'])->name('writersPage');
Route::get('/writer/{writer_id}', [NavigationController::class, 'writerDetailPage'])->name('writerDetailPage');
Route::get('/about-us', [NavigationController::class, 'aboutUsPage'])->name('aboutUsPage');
Route::get('/popular', [NavigationController::class, 'popularPage'])->name('popularPage');
```

Route “/”, akan mengarahkan page ke bagian index dengan memanggil function indexPage dari class NavigationController.

Route “/category/{category_name}”, akan mengarahkan page ke bagian detail category dengan memanggil function categoryPage dari class NavigationController dengan parameter category_name.

Route “/course/{course_name}”, akan mengarahkan page ke bagian detail dari sebuah course dengan memanggil function courseDetail dari class NavigationController.

Route “/writers”, akan mengarahkan page ke bagian writers dengan memanggil function writersPage dari class NavigationController.

Route “/writer/{writer_id}”, akan mengarahkan page ke bagian detail dari writer dan course yang dia tulis dengan memanggil function writerDetailPage dari class NavigationController.

Route “/about-us”, akan mengarahkan page ke bagian about us dengan memanggil function aboutUsPage dari class NavigationController.

Route “/popular”, akan mengarahkan page ke bagian popular dengan memanggil function popularPage dari class NavigationController.

Controller

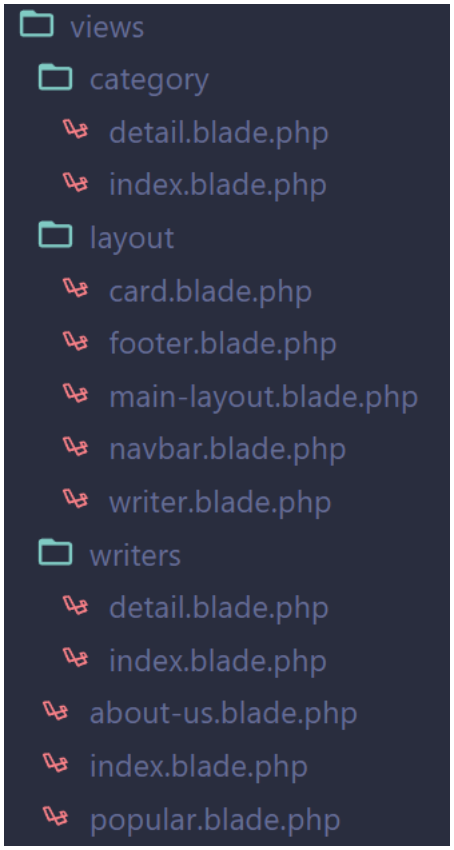
```
public function indexPage()
{
    $categories = Category::all();
    $courses = Course::all();

    return view('index', compact('categories', 'courses'));
}

public function categoryPage($category_name)
{
    $categories = Category::all();
    $searchedCategoryID = $categories->where('name', $category_name)->first();
    $searchedCategoryID = $searchedCategoryID->id;
    $courses = Course::with('category', 'writer')->where('category_id', $searchedCategoryID)->get();

    return view('category.index', compact('categories', 'courses', 'category_name'));
}

public function courseDetail($course_name)
{
    $categories = Category::all();
    $course = Course::with('writer', 'category')->where('name', $course_name)->first();

    return view('category.detail', compact('categories', 'course'));
}
```

```
public function writersPage()
{
    $categories = Category::all();
    $writers = Writer::all();

    return view('writers.index', compact('categories', 'writers'));
}

public function writerDetailPage($writer_id)
{
    $categories = Category::all();
    $courses = Course::with('writer')->where('writer_id', 'LIKE', $writer_id)->get();
    $writer = Writer::findOrFail($writer_id);

    return view('writers.detail', compact('categories', 'courses', 'writer'));
}

public function aboutUsPage()
{
    $categories = Category::all();

    return view('about-us', compact('categories'));
}
```

```php
public function popularPage()
{
    $categories = Category::all();
    $courses = Course::with('writer')->orderBy('name')->paginate(3);

    return view('popular', compact('categories', 'courses'));
}
```

1. indexPage => Berfungsi untuk mengembalikan page index dengan semua course dan semua category.
2. categoryPage => Berfungsi untuk mengembalikan detail category page dengan semua course yang dipilih menggunakan parameter category_name.
3. courseDetail => Berfungsi untuk mengembalikan detail dari course dengan parameter course_name.
4. writersPage => Berfungsi untuk mengembalikan detail dari semua data writer.
5. writerDetailPage => Berfungsi untuk mengembalikan data dari writer yang dipilih dan menampilkan semua course yang ditulisnya.
6. aboutUsPage => Berfungsi untuk mengembalikan page about us.
7. popularPage => Berfungsi untuk mengembalikan popular page dengan urutan ascending dengan pagination.

View



Model

1. Category memiliki relation hasMany dengan Course, yang artinya satu Category bisa memiliki banyak Course.
2. Course memiliki relation belongsTo dengan Category dan Writer, yang artinya satu Course hanya bisa memiliki satu Category dan juga Writer.
3. Writer memiliki relation hasMany dengan Course, yang artinya satu Writer bisa memiliki banyak Course.

```php
public function course() : HasMany
{
    return $this->hasMany(Course::class);
}
```

```php
public function category() : BelongsTo
{
    return $this->belongsTo(Category::class);
}

public function writer() : BelongsTo
{
    return $this->belongsTo(Writer::class);
}
```

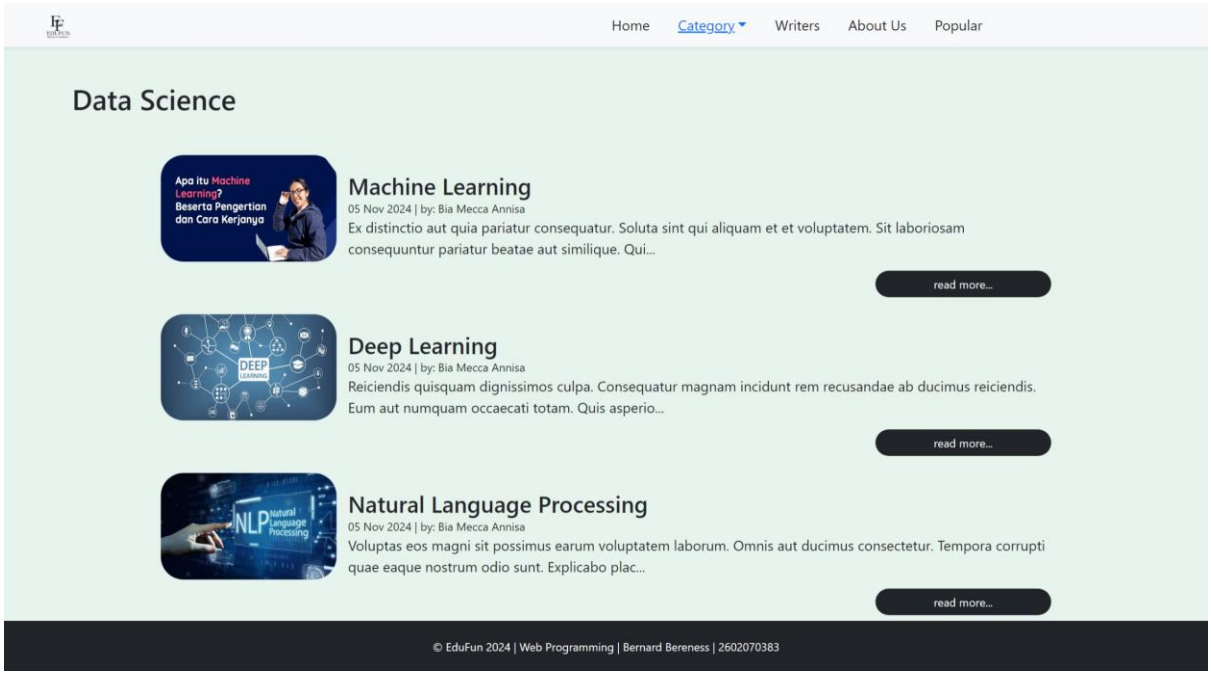
```php
public function course() : HasMany
{
    return $this->hasMany(Course::class);
}
```

1. Home Page

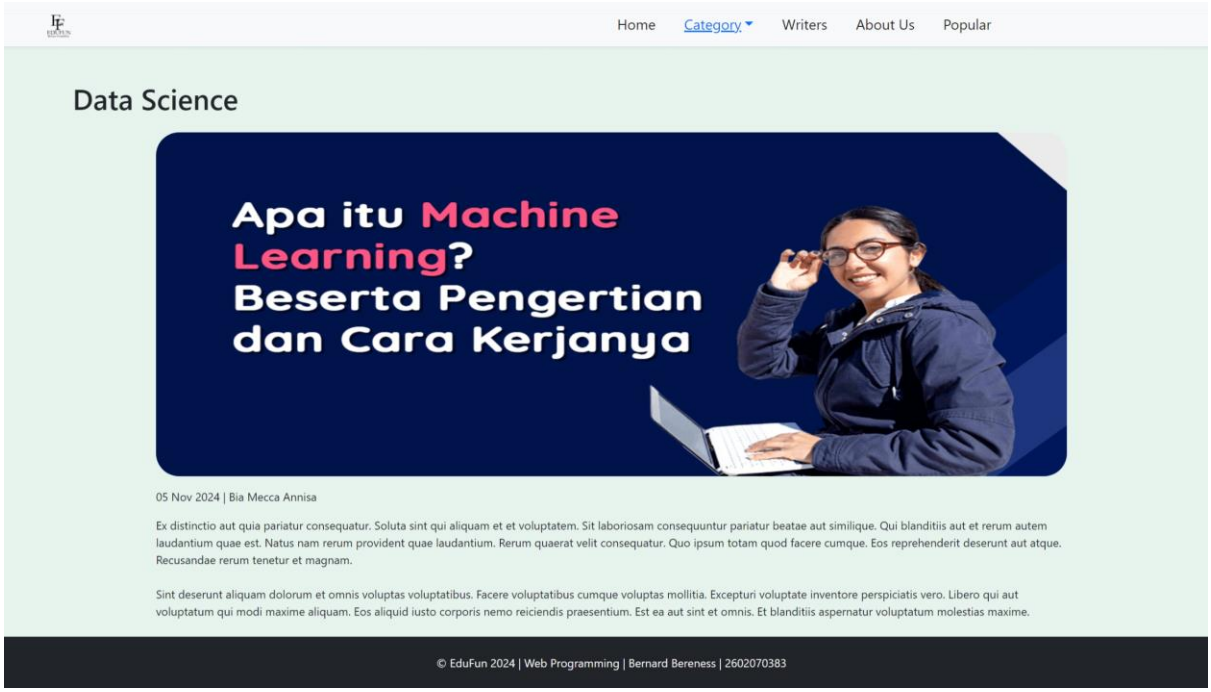
Home
Category
Writers
About Us
Popular
Machine Learning
05 Nov 2024 | by: Bia Mecca Annisa
Ex distinctio aut quia pariatur consequatur. Soluta sint qui aliquam et et voluptatem. Sit laboriosam consequuntur pariatur beatae aut similique. Qui...
read more...
Deep Learning
05 Nov 2024 | by: Bia Mecca Annisa
Reiciendis quisquam dignissimos culpa. Consequatur magnam incidunt rem recusandae ab ducimus reiciendis. Eum aut numquam occaecati totam. Quis asperio...
read more...
Natural Language Processing
05 Nov 2024 | by: Bia Mecca Annisa
Voluptas eos magni sit possimus earum voluptatem laborum. Omnis aut ducimus consectetur. Tempora corrupti quae eaque nostrum odio sunt. Explicabo plac...
read more...
Software Security
05 Nov 2024 | by: Abi Firmansyah
Non repellendus debitis veniam omnis. Eum deleniti cum quis autem. Mollitia itaque et corrupti ipsam possimus quasi architecto. Corporis eligendi ea a...
read more...
Network Administration
05 Nov 2024 | by: Abi Firmansyah
Vel distinctio necessitatibus nesciunt cum consequatur non assumenda. Dolorem enim magnam dolores ut eveniet laudantium architecto. Ad ut ut culpa asp...
read more...
Popular Network Technology
05 Nov 2024 | by: Abi Firmansyah
Molestiae aut ipsa qui sed. Distinctio debitis quod error qui et quia quia. Officiis maxime ipsa qui laudantium aut magnam quisquam. Rerum exercitatio...
read more...
© EduFun 2024 | Web Programming | Bernard Bereness | 2602070383

2. Category Page

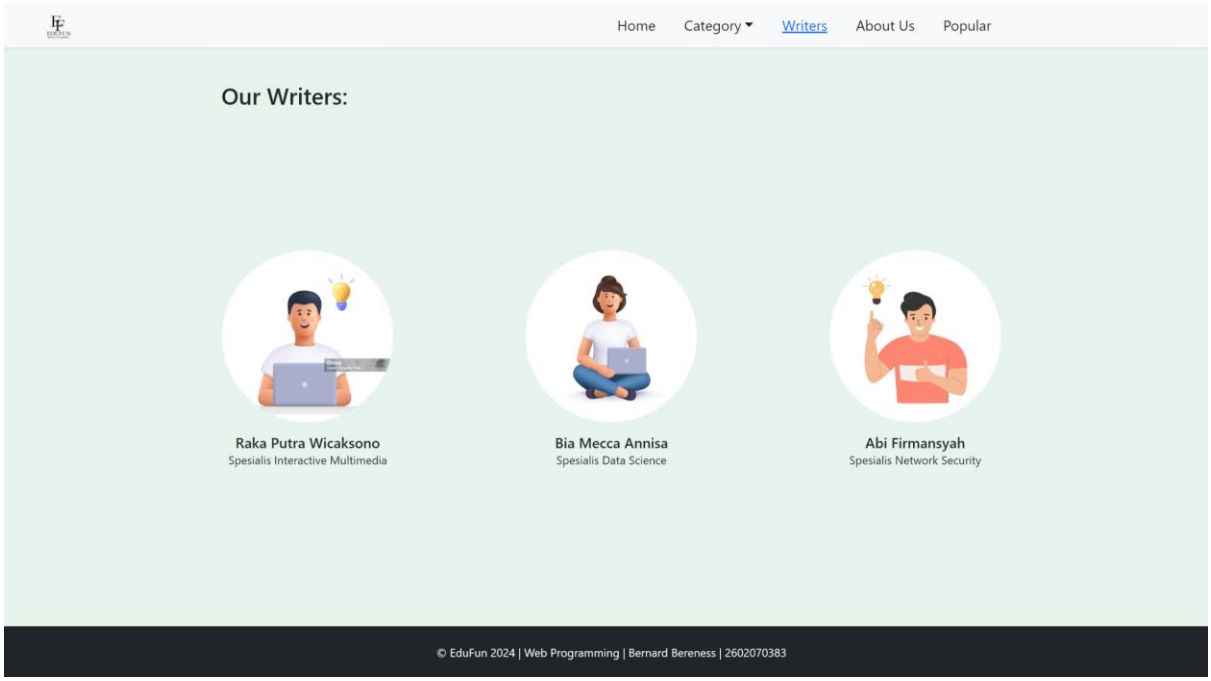
3. Detail Page

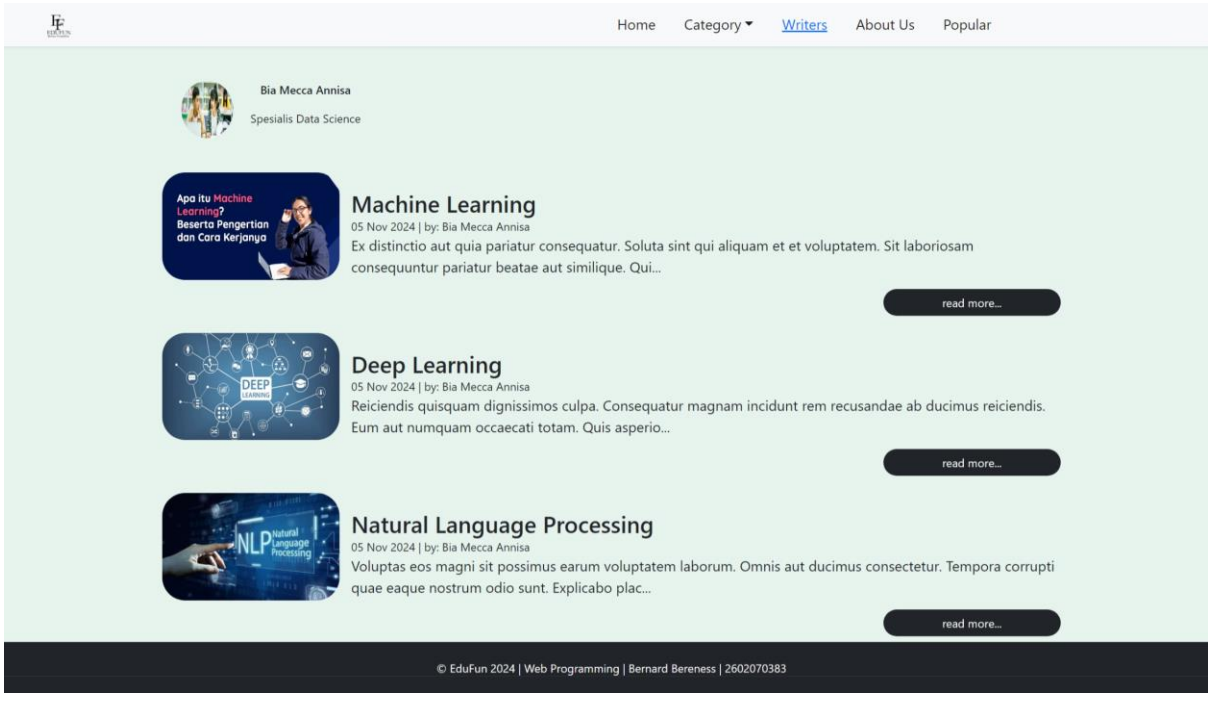


Read more…

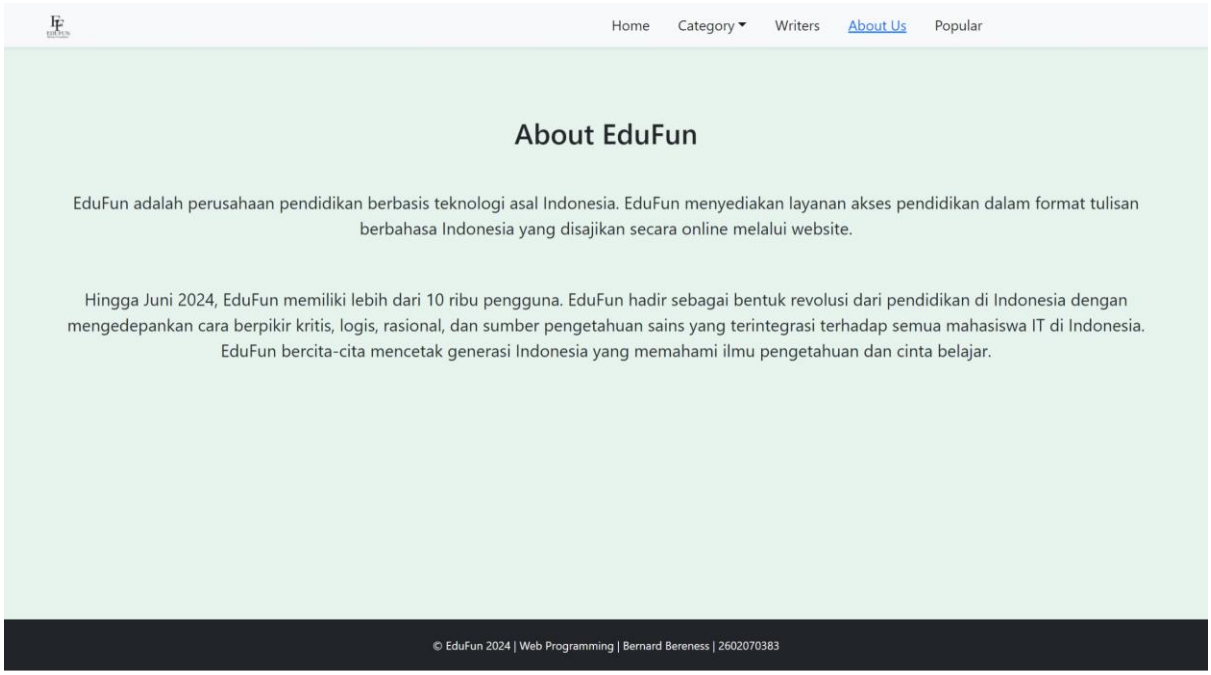


4. Writers Page

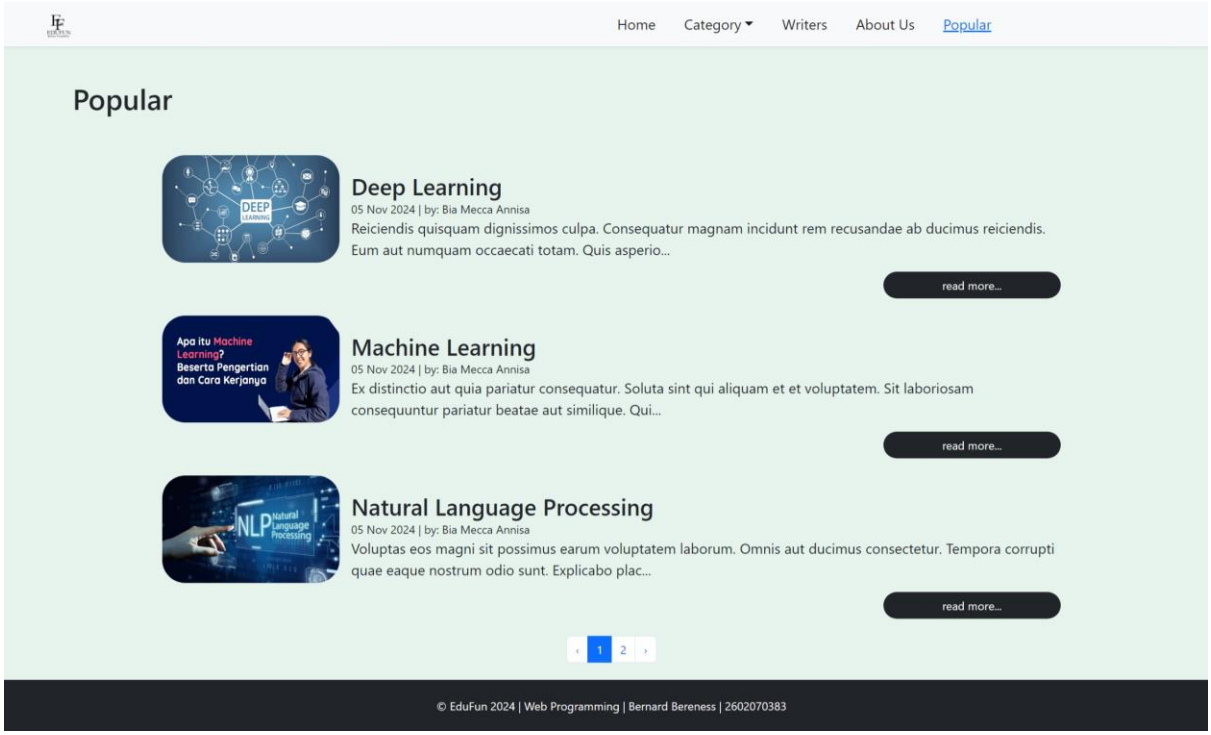




5. About Us Page



6. Popular (Page 1)



Home Category Writers About Us Popular

Popular

Network Administration

05 Nov 2024 | by: Abi Firmansyah

Vel distinctio necessitatibus nesciunt cum consequatur non assumenda. Dolorem enim magnam dolores ut eveniet laudantium architecto. Ad ut ut culpa asp...

read more...

Popular Network Technology

05 Nov 2024 | by: Abi Firmansyah

Molestiae aut ipsa qui sed. Distinctio debitis quod error qui et quia quia. Officiis maxime ipsa qui laudantium aut magnam quisquam. Rerum exercitatio...

read more...

Software Security

05 Nov 2024 | by: Abi Firmansyah

Non repellendus debitis veniam omnis. Eum deleniti cum quis autem. Mollitia itaque et corrupti ipsam possimus quasi architecto. Corporis eligendi ea a...

read more...

‹ 1 2 ›

© EduFun 2024 | Web Programming | Bernard Bereness | 2602070383