

Laravel

- Laravel 15-repository ☐☐☐☐☐☐

Laravel I5-repository



I5-repository Laravel Repository

1.



- Repository
-
-
-
-



2020

2.



```
composer require prettus/l5-repository
```



```
php artisan vendor:publish --provider="Prettus\Repository\Providers\RepositoryProvider"
```



```
// config/repository.php
return [
    'repository' => [
        'cache' => [
            'enable' => env('REPOSITORY_CACHE', true),
            'minutes' => 30,
            'clean' => true
        ]
    ]
];
```

3.

Repository

```
php artisan make:repository PostRepository
```

Repository

```
namespace App\Repositories;

use App\Models\Post;
use Prettus\Repository\Eloquent\BaseRepository;

class PostRepository extends BaseRepository
```

```

{
    public function model()
    {
        return Post::class;
    }

    public function published()
    {
        return $this->scopeQuery(function($query){
            return $query->where('status', 'published');
        })->all();
    }
}

```



```

public function index(PostRepository $repository)
{
    $posts = $repository->with(['author', 'comments'])->paginate(10);
    return view('posts.index', compact('posts'));
}

```

4.



```

$posts = $repository
    ->where('category_id', 1)
    ->orderBy('created_at', 'desc')
    ->paginate(15);

```



```
// []
$repository->skipCache()->all(); // []
$repository->withCache(60)->find(1); // []
```

Criteria

```
namespace App\Repositories\Criteria;

use Prettus\Repository\Contracts\CriteriaInterface;
use Prettus\Repository\Contracts\RepositoryInterface;

class ActiveUsersCriteria implements CriteriaInterface
{
    public function apply($model, RepositoryInterface $repository)
    {
        return $model->where('active', true);
    }
}

// []
$repository->pushCriteria(new ActiveUsersCriteria())->all();
```

5. []

	l5-repository	Spatie Query Builder	
Repository []	[]	[]	[]
[][]	[]	[]	[]
[][]	[]	[]	[]
[][]	[]	[]	[]
[][]	[][]	[][]	[][]
[][][]	[][]	[][]	[][]

6. []

1. `PHPUnit`
 - `PHPUnit`
 - `CRUD`
2. `PHPUnit`

Controller -> Service -> Repository -> Model

3. `PHPUnit`
 - `PHPUnit`
 - `PHPUnit`
4. `PHPUnit`

```
// PHPUnit Repository
$mock = $this->mock(PostRepository::class);
$mock->shouldReceive('find')->andReturn(new Post());
```

7. `PHPUnit`

`PHPUnit`

1. **Spatie Laravel Query Builder**

composer require spatie/laravel-query-builder

2. `PHPUnit`

```
interface RepositoryInterface {
    public function all();
    public function find($id);
    // ...
}
```

3. `PHPUnit` **Repository**

- `rinvex/repository`
- `bosnadev/repositories`

`PHPUnit`

l5-repository

`PHPUnit`

`PHPUnit` Repository