

# Laravel 大型项目系列教程（二）

## 一、前言

本节教程将大概实现用户的注册、修改个人信息、管理用户功能。

## 二、Let's go

### 1.创建用户注册视图

```
$ php artisan generate:view users.create
```

修改 `app/views/users/create.blade.php` :

```
@extends('_layouts.default')
@section('main')
    <div class="am-g am-g-fixed">
        <div class="am-u-lg-6 am-u-md-8">
            <br/>
            @if (Session::has('message'))
                <div class="am-alert am-alert-{{ Session::get('message')['type'] }}" data-am-alert>
                    <p>{{ Session::get('message')['content'] }}</p>
                </div>
            @endif
            @if ($errors->has())
                <div class="am-alert am-alert-danger" data-am-alert>
                    <p>{{ $errors->first() }}</p>
                </div>
            @endif
            {{ Form::open(array('url' => 'register', 'class' => 'am-form')) }}
            {{ Form::label('email', 'E-mail:') }}
            {{ Form::email('email', Input::old('email')) }}
            <br/>
            {{ Form::label('nickname', 'NickName:') }}
            {{ Form::text('nickname', Input::old('nickname')) }}
            <br/>
            {{ Form::label('password', 'Password:') }}
            {{ Form::password('password') }}
            <br/>
            {{ Form::label('password_confirmation', 'ConfirmPassword:') }}
            {{ Form::password('password_confirmation') }}
            <br/>
            <div class="am-cf">
                {{ Form::submit('Register', array('class' => 'am-btn am-btn-primary am-btn-sm am-fl')) }}
            </div>
        </div>
    </div>
@endsection
```

```
        </div>
        {{ Form::close() }}
        <br/>
    </div>
</div>
@stop
```

修改 `layouts/nav.blade.php` 中的 `@else` 部分：

```
@else
    <div class="am-topbar-right">
        <a href="{{ URL::to('register') }}" class="am-btn am-btn-secondary am-topbar-r-
        btn am-btn-sm topbar-link-btn"><span class="am-icon-pencil"></span> Register
        </a>
    </div>
    <div class="am-topbar-right">
        <a href="{{ URL::to('login') }}" class="am-btn am-btn-primary am-topbar-bt
        n am-btn-sm topbar-link-btn"><span class="am-icon-user"></span> Login</a>
    </div>
@endif
```

在 `routes.php` 中增加：

```
Route::get('register', function(){
    return View::make('users.create');
});
```

启动开发服务器，浏览器中访问 `localhost:8000`，导航条中多了一个 `Register` 按钮：



点击 `Register` 按钮，进入用户注册页面：

ShiYanLou Blog Login Register

E-mail:

NickName:

Password:

ConfirmPassword:

Register

© 2015 By [www.shiyanlou.com](http://www.shiyanlou.com)

## 2.实现用户注册

在 `routes.php` 中增加：

```
Route::post('register', array('before' => 'csrf', function(){
    $rules = array(
        'email' => 'required|email|unique:users,email',
        'nickname' => 'required|min:4|unique:users,nickname',
        'password' => 'required|min:6|confirmed',
    );
    $validator = Validator::make(Input::all(), $rules);
    if ($validator->passes())
    {
        $user = User::create(Input::only('email', 'password', 'nickname'));
        $user->password = Hash::make(Input::get('password'));
        if ($user->save())
        {
            return Redirect::to('login')->with('message', array('type' => 'success',
'content' => 'Register successfully, please login'));
        } else {
            return Redirect::to('register')->withInput()->with('message', array('type' => 'danger', 'content' => 'Register failed'));
        }
    } else {
        return Redirect::to('register')->withInput()->withErrors($validator);
    }
}));
```

上面表单验证规则的 `unique:users,email` 能确保 `users` 表中的 `email` 字段是唯一的，切记 `users` 和 `email` 之间不能有空格，`confirmed` 确保提交的数据必须有一个名为 `password_conformation` 的字段且与 `password` 字段的值相等。

例如当输入已存在的 `email` 时，会出现错误提示：

ShiYanLou Blog

LoginRegister

The email has already been taken.

E-mail:

NickName:

Password:

ConfirmPassword:

Register

© 2015 By [www.shiyancelou.com](http://www.shiyancelou.com)

之后我们再修改两个地方，把 `routes.php` 中 `post login` 内的

```
return Redirect::to('login')->withInput()->with('message', 'E-mail or password error');
```

修改为：

```
return Redirect::to('login')->withInput()->with('message', array('type' => 'danger', 'content' => 'E-mail or password error'));
```

把 `login.blade.php` 中的

```
@if (Session::has('message'))
    <div class="am-alert am-alert-danger" data-am-alert>
        <p>{{ Session::get('message') }}</p>
    </div>
@endif
```

修改为：

```
@if (Session::has('message'))
    <div class="am-alert am-alert-{{ Session::get('message')['type'] }}" data-am-alert>
        <p>{{ Session::get('message')['content'] }}</p>
    </div>
@endif
```

现在你就可以尝试注册，如果注册成功就会跳转到登录页面，并给出成功的提示：

Register successfully, please login

E-mail:

Password:

☐ Remember Me

Login

注册成功之后你可以试试是否能用刚注册的账号成功登录。

### 3.修改个人信息

用户注册之后我们还应该让他能够修改信息，在 `_layouts/nav.blade.php` 中添加修改个人信息的选项：

```
<li><a href="{{ URL::to('user/'. Auth::id() . '/edit') }}"><span class="am-icon-user"></span> Information</a></li>
```

添加视图 `users/edit.blade.php`

```
$ php artisan generate:view users.edit
```

修改 `users/edit.blade.php`：

```
@extends('_layouts.default')
@section('main')
    <div class="am-g am-g-fixed">
        <div class="am-u-lg-6 am-u-md-8">
            <br/>
            @if (Session::has('message'))
                <div class="am-alert am-alert-{{ Session::get('message')['type'] }}" data-am-alert>
                    <p>{{ Session::get('message')['content'] }}</p>
                </div>
            @endif
            @if ($errors->has())
                <div class="am-alert am-alert-danger" data-am-alert>
                    <p>{{ $errors->first() }}</p>
                </div>
            @endif
        </div>
    </div>
```

```

        @endif
        {{ Form::model($user, array('url' => 'user/' . $user->id, 'method' => 'PUT',
        'class' => 'am-form')) }}
        {{ Form::label('email', 'E-mail:') }}
        <input id="email" name="email" type="email" readonly="readonly" value=
        "{{ $user->email }}" />
        <br/>
        {{ Form::label('nickname', 'NickName:') }}
        <input id="nickname" name="nickname" type="text" value="{{ $user->nic
        kname }}" />
        <br/>
        {{ Form::label('old_password', 'OldPassword:') }}
        {{ Form::password('old_password') }}
        <br/>
        {{ Form::label('password', 'NewPassword:') }}
        {{ Form::password('password') }}
        <br/>
        {{ Form::label('password_confirmation', 'ConfirmPassword:') }}
        {{ Form::password('password_confirmation') }}
        <br/>
        <div class="am-cf">
            {{ Form::submit('Modify', array('class' => 'am-btn am-btn-primary a
            m-btn-sm am-fl')) }}
        </div>
        {{ Form::close() }}
        <br/>
    </div>
</div>
@stop

```

在 `routes.php` 中添加：

```

Route::get('user/{id}/edit', array('before' => 'auth', 'as' => 'user.edit', function($id){
    if (Auth::user()->is_admin or Auth::id() == $id) {
        return View::make('users.edit')->with('user', User::find($id));
    } else {
        return Redirect::to('/');
    }
}));

```

上面的 `as` 是命名路由，在生成 URL 时也可以使用别名。例如 `Redirect::route('user.edit', $id)`。

现在登录后在右上角会发现多了一个 `Information` 的选项，点击后会显示用户个人信息的页面：

ShiYanLou Blog

admin

Information

Exit

E-mail:

admin@shiyancelou.com

NickName:

admin

OldPassword:

NewPassword:

ConfirmPassword:

Modify

© 2015 By www.shiyancelou.com

你是不是发现了表单中的 `Form::model($user)`，它会根据 `View::make('users.edit')->with('user', User::find($id))` 传过来的 `user` 进行自动填充。


之后就要实现真正地修改用户信息了，在 `routes.php` 中增加：

```
Route::put('user/{id}', array('before' => 'auth|csrf', function($id){
    if (Auth::user()->is_admin or (Auth::id() == $id)) {
        $user = User::find($id);
        $rules = array(
            'password' => 'required_with:old_password|min:6|confirmed',
            'old_password' => 'min:6',
        );
        if (!(Input::get('nickname') == $user->nickname))
        {
            $rules['nickname'] = 'required|min:4|unique:users,nickname';
        }
        $validator = Validator::make(Input::all(), $rules);
        if ($validator->passes())
        {
            if (!(Input::get('old_password') == "")) {
                if (!Hash::check(Input::get('old_password'), $user->password)) {
                    return Redirect::route('user.edit', $id)->with('user', $user)
                    ->with('message', array('type' => 'danger', 'content' => 'Old password error'));
                } else {
                    $user->password = Hash::make(Input::get('password'));
                }
            }
            $user->nickname = Input::get('nickname');
            $user->save();
        }
    }
});
```

```
        return Redirect::route('user.edit', $id)->with('user', $user)->with('message', array('type' => 'success', 'content' => 'Modify successfully'));
    } else {
        return Redirect::route('user.edit', $id)->withInput()->with('user', $user)->withErrors($validator);
    }
    } else {
        return Redirect::to('/');
    }
    }
    }));
```

现在尝试修改信息，如果失败，就会出现错误提示就像下面这样：

ShiYanLou Blog

 admin1 ▾

The nickname has already been taken.

**E-mail:**

**NickName:**

**OldPassword:**

**NewPassword:**

**ConfirmPassword:**

Modify

© 2015 By [www.shiyancelou.com](http://www.shiyancelou.com)

如果成功就会像下面这样：



ShiYanLou Bloghelloadmin

Modify successfully

**E-mail:**

**NickName:**

**OldPassword:**

**NewPassword:**

**ConfirmPassword:**

Modify

© 2015 By [www.shiyancelou.com](http://www.shiyancelou.com)

这样修改个人信息的功能就完成了。

## 4.管理用户

5.

上面的完成之后,我们就需要管理员能够管理用户,例如可以修改其他用户的昵称、重置它们的密码、锁定用户等。先需要重写下 `_layouts/nav.blade.php` 的 `@if (Auth::check())` 里的内容:

```
@if (Auth::user()->is_admin)<ul class="am-nav am-nav-pills am-topbar-nav">
  <li class=""><a href="#">Users</a></li></ul>
@endif<div class="am-topbar-right">
  <div class="am-dropdown" data-am-dropdown="{boundary: 'am-topbar'}">
    <button class="am-btn am-btn-secondary am-topbar-btn am-btn-sm am-d
ropdown-toggle" data-am-dropdown-toggle><span class="am-icon-users"></span>
{{{ Auth::user()->nickname }}} <span class="am-icon-caret-down"></span></butto
n>
    <ul class="am-dropdown-content">
      <li><a href="{{ URL::to('user/'. Auth::id() . '/edit') }}"><span class="am-icon
-user"></span> Information</a></li>
      <li><a href="{{ URL::to('logout') }}"><span class="am-icon-power-off"></spa
n> Exit</a></li>
    </ul>
  </div>
</div>
```

```
</div>
```

创建用户列表视图：

```
$ php artisan generate:view admin.users.list
```

修改 `views/admin/users/list.blade.php`：

```
@extends('_layouts.default')

@section('main')<div class="am-g am-g-fixed">
    <div class="am-u-sm-12">
        <br/>
        @if (Session::has('message'))
            <div class="am-alert am-alert-{{ Session::get('message')['type'] }}" data-am-alert>
                <p>{{ Session::get('message')['content'] }}</p>
            </div>
        @endif
        <table class="am-table am-table-hover am-table-striped">
            <thead>
                <tr>
                    <th>ID</th>
                    <th>E-mail</th>
                    <th>Nickname</th>
                    <th>Management</th>
                </tr>
            </thead>
            <tbody>
                @foreach ($users as $user)
                    <tr>
                        <td>{{ $user->id }}</td>
                        <td>{{ $user->email }}</td>
                        <td>{{ $user->nickname }}</td>
                        <td>
                            <a href="{{ URL::to('user/' . $user->id . '/edit') }}" class="am-btn am-btn-xs am-btn-primary">Edit</a>
                            {{ Form::open(array('url' => 'user/' . $user->id . '/reset', 'method' => 'PUT', 'style' => 'display: inline;')) }}
                                <button type="button" class="am-btn am-btn-xs am-btn-warning" id="reset{{ $user->id }}">Reset</button>
                            {{ Form::close() }}
                            @if ($user->block)
                                {{ Form::open(array('url' => 'user/' . $user->id . '/unblock', 'method' => 'PUT', 'style' => 'display: inline;')) }}
                                    <button type="button" class="am-btn am-btn-xs am-btn-danger" id="unblock{{ $user->id }}">Unblock</button>
                                {{ Form::close() }}
                            @else
                                {{ Form::open(array('url' => 'user/' . $user->id, 'method' => 'DELETE', 'style' => 'display: inline;')) }}
                                    <button type="button" class="am-btn am-btn-xs am-btn-danger" id="delete{{ $user->id }}">Block</button>
                                {{ Form::close() }}
                            @endif
                        </td>
                    </tr>
                @endforeach
            </tbody>
        </table>
    </div>
</div>
```

```

        </td>
    </tr>
    @endforeach
</tbody>
</table>
</div></div>
<div class="am-modal am-modal-confirm" tabindex="-1" id="my-confirm">
    <div class="am-modal-dialog">
        <div class="am-modal-bd">
            </div>
            <div class="am-modal-footer">
                <span class="am-modal-btn" data-am-modal-cancel>No</span>
                <span class="am-modal-btn" data-am-modal-confirm>Yes</span>
            </div>
        </div></div><script>
        $(function() {
            $('#id^=reset').on('click', function() {
                $('#am-modal-bd').text('Sure you want to reset the password for 12345
6?');
                $('#my-confirm').modal({
                    relatedTarget: this,
                    onConfirm: function(options) {
                        $(this.relatedTarget).parent().submit();
                    },
                    onCancel: function() {
                    }
                });
            });

            $('#id^=delete').on('click', function() {
                $('#am-modal-bd').text('Sure you want to lock it?');
                $('#my-confirm').modal({
                    relatedTarget: this,
                    onConfirm: function(options) {
                        $(this.relatedTarget).parent().submit();
                    },
                    onCancel: function() {
                    }
                });
            });

            $('#id^=unblock').on('click', function() {
                $('#am-modal-bd').text('Sure you want to unlock it?');
                $('#my-confirm').modal({
                    relatedTarget: this,
                    onConfirm: function(options) {
                        $(this.relatedTarget).parent().submit();
                    },
                    onCancel: function() {
                    }
                });
            });
        });
    </script>
    @stop

```

上面的@foreach 相当于 for 循环，可以遍历@users 中的内容。

把 `views/_layouts/default.blade.php` 中的

```
<script src="//cdn.bootcss.com/jquery/2.1.3/jquery.min.js"></script><script src="//cdn.amazeui.org/amazeui/2.1.0/js/amazeui.min.js"></script>
```

移到 `head` 中。

为了保证只有管理员才能管理用户，我们在 `app/filters.php` 中增加一个过滤器：

```
Route::filter('idAdmin', function(){
    if (!Auth::user()->is_admin) {
        return Redirect::to('/');
    }
});
```

在 `routes.php` 中增加：

```
Route::group(array('prefix' => 'admin', 'before' => 'auth|isAdmin'), function(){
    Route::get('users', function()
    {
        return View::make('admin.users.list')->with('users', User::all())->with('page',
        'users');
    });
});

Route::model('user', 'User');

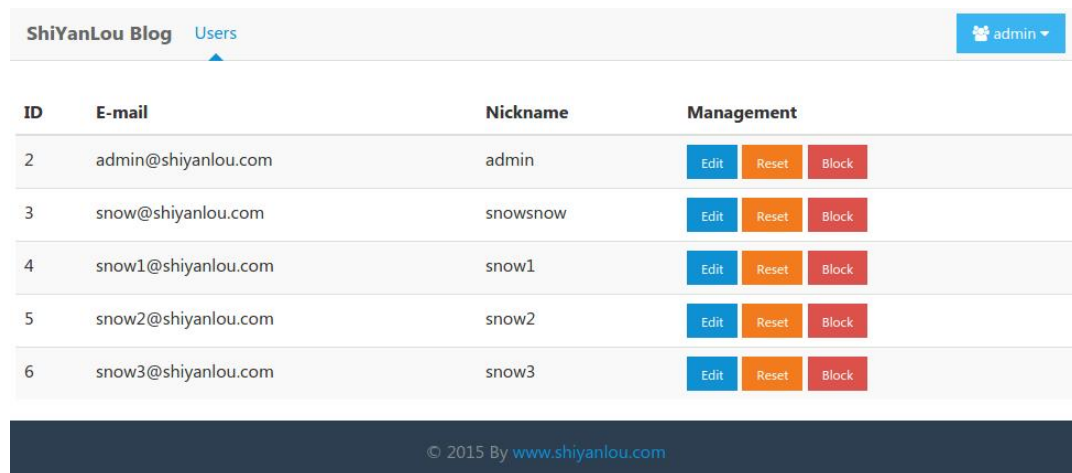
Route::group(array('before' => 'auth|csrf|isAdmin'), function(){
    Route::put('user/{user}/reset', function(User $user)
    {
        $user->password = Hash::make('123456');
        $user->save();
        return Redirect::to('admin/users')->with('message', array('type' => 'success', 'content' => 'Reset password successfully'));
    });

    Route::delete('user/{user}', function(User $user)
    {
        $user->block = 1;
        $user->save();
        return Redirect::to('admin/users')->with('message', array('type' => 'success', 'content' => 'Lock user successfully'));
    });

    Route::put('user/{user}/unblock', function(User $user)
    {
        $user->block = 0;
        $user->save();
        return Redirect::to('admin/users')->with('message', array('type' => 'success', 'content' => 'Unlock user successfully'));
    });
});
```

上面使用了路由组 `Route::group`、路由前缀 `prefix` 和路由与模型绑定 `Route::model`，过滤器是可以有多个的，用 `|` 分隔。

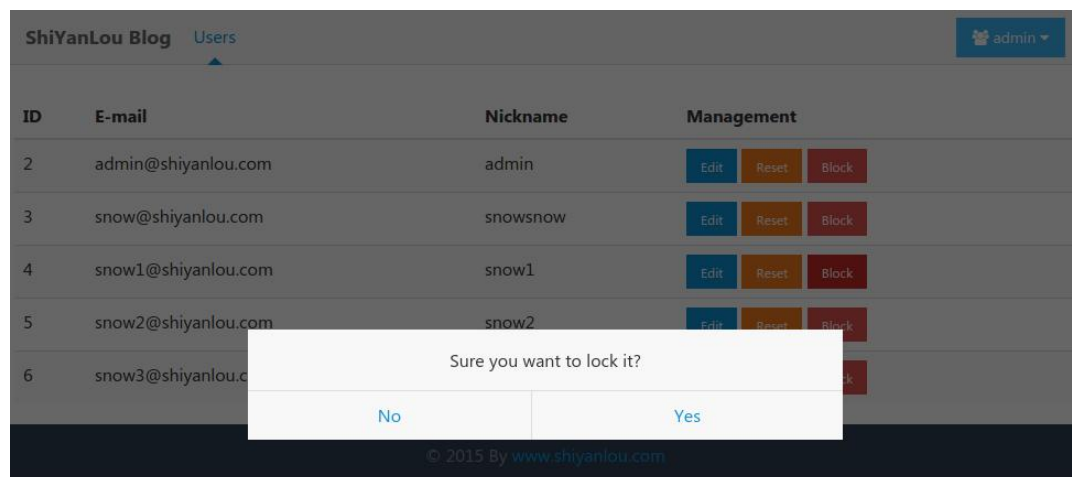
现在用管理员账号登录后会发现导航条多了一个 `Users` 链接，点击 `Users` 超链接会出现下图这样：



ID	E-mail	Nickname	Management
2	admin@shiyanlou.com	admin	<a href="#">Edit</a> <a href="#">Reset</a> <a href="#">Block</a>
3	snow@shiyanlou.com	snowsnow	<a href="#">Edit</a> <a href="#">Reset</a> <a href="#">Block</a>
4	snow1@shiyanlou.com	snow1	<a href="#">Edit</a> <a href="#">Reset</a> <a href="#">Block</a>
5	snow2@shiyanlou.com	snow2	<a href="#">Edit</a> <a href="#">Reset</a> <a href="#">Block</a>
6	snow3@shiyanlou.com	snow3	<a href="#">Edit</a> <a href="#">Reset</a> <a href="#">Block</a>

© 2015 By [www.shiyanlou.com](http://www.shiyanlou.com)

上面的用户数据需要自己添加，当点击 `Block` 的是否会出现确认的提示框：



ID	E-mail	Nickname	Management
2	admin@shiyanlou.com	admin	<a href="#">Edit</a> <a href="#">Reset</a> <a href="#">Block</a>
3	snow@shiyanlou.com	snowsnow	<a href="#">Edit</a> <a href="#">Reset</a> <a href="#">Block</a>
4	snow1@shiyanlou.com	snow1	<a href="#">Edit</a> <a href="#">Reset</a> <a href="#">Block</a>
5	snow2@shiyanlou.com	snow2	<a href="#">Edit</a> <a href="#">Reset</a> <a href="#">Block</a>
6	snow3@shiyanlou.com	snow3	<a href="#">Edit</a> <a href="#">Reset</a> <a href="#">Block</a>

Sure you want to lock it?

[No](#) [Yes](#)

© 2015 By [www.shiyanlou.com](http://www.shiyanlou.com)

点击 Yes 操作成功后会像下面这样：

ShiYanLou Blog

Users

admin

Lock user successfully

ID	E-mail	Nickname	Management
2	admin@shianlou.com	admin	<div>EditResetBlock</div>
3	snow@shianlou.com	snowsnow	<div>EditResetBlock</div>
4	snow1@shianlou.com	snow1	<div>EditResetUnblock</div>
5	snow2@shianlou.com	snow2	<div>EditResetBlock</div>
6	snow3@shianlou.com	snow3	<div>EditResetBlock</div>

© 2015 By [www.shianlou.com](http://www.shianlou.com)

## 5.小结

这节完成了用户管理模块，但是还有很多不完善的地方，你可以在用户列表页面添加按昵称或 Email 查找用户、只显示锁定的用户等功能，还有你是不是发现了在 `routes.php` 中代码显得很零乱，那是因为我们还没有使用 MVC 模式中的 C，在下节教程中就将讲解 Laravel 中的控制器。

本文详细出处：<http://www.shianlou.com/courses/123>