

Protegendo rotas com autorização

Só podemos acessar a página de novo produto quando estamos logados, mas se eu tentar acessar a uri `/mercado/produtos/formulario` consigo visualizar o formulário e também cadastrar o produto! Na verdade não queria deixar o usuário fazer isso se ele está deslogado. De alguma maneira preciso verificar tal condição para proibi-lo de acessar a página.

Primeiro queremos proteger o formulário, então vamos no controller de `produtos`, na função `formulario`. Pegamos o usuário logado:

```
public function formulario() {  
    $usuarioLogado = $this->session->userdata("usuario_logado");  
    $this->load->view("produtos/formulario");  
}
```

Se o usuário não existe, devemos redirecionar para a página inicial com uma mensagem de erro:

```
$usuarioLogado = $this->session->userdata("usuario_logado");  
if(!$usuarioLogado) {  
    $this->session->set_flashdata("danger", "Você precisa estar logado");  
    redirect("/");  
}
```

Tentamos acessar o formulário deslogado e ele mostra a mensagem de erro. Temos que agora colocar esse mesmo código em todo lugar que precisa do usuário logado, fazemos o que? Copy e paste? Seria muito ruim pois é difícil de manter copy e paste. Vamos extrair isso para uma função de ajuda, um helper de autorização e autenticação, `auth_helper`:

```
<?php  
function autoriza() {  
    $usuarioLogado = $this->session->userdata("usuario_logado");  
    if(!$usuarioLogado) {  
        $this->session->set_flashdata("danger", "Você precisa estar logado");  
        redirect("/");  
    }  
    return $usuarioLogado;  
}
```

Como esse helper será usado em muitos lugares, autoload nele:

```
$autoload['helper'] = array('url', 'form', 'text', 'date', 'auth');
```

Vamos no formulário e chamamos a função `autoriza`, agora bem mais simples:

```
public function formulario() {  
    autoriza();  
}
```

```
$this->load->view("produtos/formulario");  
}
```

Mas ao testarmos ele fala que não existe o `$this`. O que acontece? Nosso helper tem a função 'voando', fora de um objeto, então não existe o `$this` e nenhuma das duas linhas que precisam dele funcionam. Como acessar a session então? O Code Igniter disponibiliza uma função chamada `get_instance()`:

```
$ci = get_instance();
```

Com ela podemos acessar a `session` normalmente:

```
<?php  
function autoriza() {  
    $ci = get_instance();  
    $usuarioLogado = $ci->session->userdata("usuario_logado");  
    if(!$usuarioLogado) {  
        $ci->session->set_flashdata("danger", "Você precisa estar logado");  
        redirect("/");  
    }  
    return $usuarioLogado;  
}
```

Podemos usar agora o `autoriza` no novo, onde queremos ter certeza que só se o usuário está logado pode adicionar um novo produto:

```
public function novo() {  
    $usuarioLogado = autoriza();  
    // resto do código  
}
```

Por fim, no `Vendas` desejamos também conferir se o usuário está logado:

```
public function nova() {  
    $usuario = autoriza();  
    // resto do código  
}  
  
public function index() {  
    $usuario = autoriza();  
    // resto do código  
}
```