

Публикация на тему

# Ajax в Laravel

*Ajax в Laravel. Модальное окно на ajax*

## Анотация

-  
-

## Автор

[Михалькевич Александр Викторович](#)

## Публикация

**Наименование** Ajax в Laravel

**Автор** А.В.Михалькевич

**Специальность** Ajax в Laravel. Модальное окно на ajax,

**Анотация** -

**Anotation in English** -

**Ключевые слова** laravel

**Количество символов** 5903

## Содержание

[Введение](#)

1 [Подготовка html](#)

2 [Файл modal.js](#)

3 [Стили для модального окна](#)

4 [Кнопка закрытия модального окна](#)

5 [Стили для кнопки закрытия](#)

6 [Фон для модального окна](#)

7 [Стили для фона](#)

8 [Закрытие фона](#)

9 [Добавляем Ajax](#)

10 [Ajax на сервере](#)

11 [CsrfToken для ajax](#)

12 [Окончательный вариант файла modal.js](#)

[Заключение](#)

[Список использованных источников](#)

[Приложения](#)

# Введение

В публикации рассматривается создание модального окна с использованием HTML, CSS, jQuery, в качестве серверного языка программирования используется PHP. С помощью технологии Ajax осуществляется вывод данных из базы MySQL.

## 1 Подготовка html

1) Добавить id для элемента

```
<img src="" class="product" id="{{ $one->id }}" />
```

## 2 Файл modal.js

1) Подключаем скрипт modal.js в файлах подшаблона

```
@section('scripts')
@parent
<script src="{{asset('js/modal.js')}}"></script>
@endsection
```

В файлах базового шаблона переменная scripts должна быть определена

```
@section('scripts')
//
@show
```

2) Открываем modal.js

```
$(function(){
    $('.product').click(function(){
        var id = $(this).attr('id');
        console.log(id);
    });
})
```

3) Объектный литерал вывода модального окна

```
$(function(){
    var fx = {
        "initModal" : function(){
            if($(".modal-window").length == 0){
                return $('<div>').addClass("modal-window").appendTo("body");
            }else{
                return $(".modal-window")
            }
        }
    }
    $('.product').click(function(){ var id = $(this).attr('id'); var modal
    fx.initModal();});
```

```
})
```

### 3 Стили для модального окна

Добавим следующие стили для модального окна

```
.modal-window{
  position:fixed;
  top:100px;
  left:50%;
  width:600px;
  margin-left:-300px;
  padding:10px;
  border:solid 1px #ccc;
  background:white;
  border-radius:6px;
  box-shadow:1px 1px 1px black;
  z-index:99;
}
```

### 4 Кнопка закрытия модального окна

В файле modal.js, после создания модального окна, т.е. после определения переменной modal

```
$('<a>').attr('href', '#')
  .addClass('modal-close')
  .html('&times;')
  .click(function(event){
    event.preventDefault();
    modal.remove();
  }).appendTo(modal);
```

### 5 Стили для кнопки закрытия

Для добавим следующие стили для кнопки закрытия:

```
.modal-close{
  position:absolute;
  top:1px;
  right:1px;
  margin:0;
  padding:0;
  text-decoration:none;
```

```
color:red;
font-size:18px;
}
```

Используем фильтр :before для отображения текста

```
.modal-close:before{
position:relative;
top:-1px;
content:"Close";
font-size:12px;
}
```

## 6 Фон для модального окна

JavaScript-код для фона

```
$("<div>").attr('id', 'overlay').fadeIn(2000).appendTo('body')
```

## 7 Стили для фона

В файле стилей для id overlay

```
#overlay{
background:black;
opacity: 0.7;
position:fixed;
height:100%;
width:100%;
left:0;
top:0;
}
```

## 8 Закрытие фона

Код удаления фона пишем там же, где происходит закрытие модального окна

```
$('#overlay').remove();
```

## 9 Добавляем Ajax

В файле modal.js, после объявления переменной modal:

```
$.ajax({
type: "Post",
url: "/ajax",
```

```

data: "id="+id,
success: function(data){
    modal.append(data)
},
error: function(msg){
    console.log(msg);
},
beforeSend: function(){
    //modal.append('')
}
})

```

## 10 Ajax на сервере

### Ajax на сервере

1) Создать контроллер

```
php artisan make:controller AjaxController
```

2) В файле routes/web.php добавить маршрут

```
Route::post('ajax', 'AjaxController@postIndex')
```

3) Экшн контроллера

```

public function postIndex(){
    $id = (int)$_POST['id'];
    $obj = Product::find($id);
    return view('ajax.modal', compact('obj'));
}

```

4) В папке resources/views создаем папку ajax, в ней файл modal.blade.php

```
<h2>{{ $obj->name }}</h2>
```

```
<p>{!! $obj->body !!}</p>
```

## 11 CsrfToken для ajax

Открываем файл middleware VerifyCsrfToken.php

```

protected $except = [
    'ajax'
];

```

## 12 Окончательный вариант файла modal.js

```
$(function () {
```

```

var fx = {
    'initModal': function () {
        if ($('.modal-window').length == 0) {
            $('
').addClass('overlay').appendTo('body');
            return $('
').addClass('modal-window').appendTo('body');
        } else {
            return $('.modal-window');
        }
    }
}
$('.product').click(function () {
    var id = $(this).attr('data-id');
    var modal = fx.initModal();
    $('').attr('href', '#').addClass('modal-close-
btn').html('×').click(function(){
        $('.overlay').remove();
        modal.remove();
    })
    ).appendTo(modal);
    $.ajax({
        data:'id='+id,
        type:'post',
        url:'/ajax/modal',
        success:function(data){
            modal.append(data);
        },
        error:function(msg){
            console.log(msg);
        }
    });
});
});
});

```

## **Заключение**

## **Список использованных источников**

## **Приложения**