

## Рассмотрение способа аутентификации в PHP фреймворке Yii2

*Кочитов Михаил Евгеньевич*

*Приамурский государственный университет им. Шолом-Алейхема*

*студент*

### **Аннотация**

В данной статье рассматривается аутентификация в PHP фреймворке Yii2, являющаяся основной функцией входа в веб-систему, которая позволяет провести проверку подлинности пользователя. Также будут рассмотрены предлагаемые PHP фреймворком Yii2 функции, которые дают возможность идентифицировать пользователя и разрешить ему вход в веб-систему.

**Ключевые слова:** аутентификация, PHP фреймворк Yii2, вход в систему, подлинность

## Considering Authentication Method in PHP Yii2 Framework

*Kochitov Mikhail Evgenevich*

*Sholom-Aleichem Priamursky State University*

*student*

### **Abstract**

This article discusses authentication in the Yii2 PHP framework, which is the main web login function that allows user authentication. We will also consider the functions offered by the PHP framework Yii2, which make it possible to identify the user and allow him to log into the web system.

**Keywords:** authentication, PHP Yii2 framework, login, authenticity

Аутентификация используется во всех веб-сайтах, у которых имеется возможность зарегистрироваться или авторизоваться. Многие веб-сайты содержат базу зарегистрированных пользователей и если им понадобится веб-сайт, то они зайдут в саму систему, тем самым пройдя аутентификацию, которая проверит подлинность пользователя на наличии правильно введенных данных - логина и пароля. Веб-сайты снабжены аутентификацией для того, чтобы во время авторизации пользователя в саму веб-систему и на сервере были добавлены идентификационные данные его аккаунта, которые дают получить определенный доступ пользователю к возможностям сайта.

В статье В.С. Титова рассматривается идентификация и аутентификация, способы и методы аутентификации [1]. Рассматривая статью Е.В. Цебука можно увидеть проблему аутентификации информации в локальных сетях [2]. Н.А. Балашов рассмотрел аутентификацию для веб-приложений с использованием сгенерированных на автономном устройстве

одноразовых паролей [3]. А.Е. Шафер, А.В. Черников рассмотрели в своей статье двухфакторную аутентификацию с использованием СМС-сервиса [5].

Целью данной статьи является аутентификацию, реализованная в PHP фреймворке Yii2, которая позволяет провести проверку подлинности пользователя для входа в саму веб-систему. Также будут рассмотрены все функции аутентификации, интегрированные в сам PHP фреймворк Yii2.

Чтобы PHP фреймворк Yii2 [6] работал полноценно со всеми его возможностями, то необходимо установить локальный веб-сервер OpenServer [7], который предоставляется в свободном доступе. Теперь рассмотрим PHP код, в котором представлены основные функции идентификации пользователя, который проходит этап аутентификации. Для начала перейдем к содержимому PHP фреймворка Yii2 и перейдем в папку Models, в котором откроем файл User.php. В нем рассмотрим основные важные методы аутентификации пользователей, прописанные в ядре PHP фреймворка Yii2, который реализует класс User

```
public static function findIdentity($id)
{
    return isset(self::$users[$id]) ? new static(self::$users[$id]) : null;
}
```

Рисунок 1 – метод findIdentity()

На рисунке 1 представлен метод findIdentity(), который получает идентифицированный класс пользователя, отправляя ему аргумент в виде ID пользователя. Данный метод предлагается использовать в случае, когда нужна необходимость в поддержке состояния аутентификации через сессии.

```
public static function findIdentityByAccessToken($token, $type = null)
{
    foreach (self::$users as $user) {
        if ($user['accessToken'] === $token) {
            return new static($user);
        }
    }

    return null;
}
```

Рисунок 2 - метод findIdentityByAccessToken()

На рисунке 2 изображен метод findIdentityByAccessToken(), который в отличие от метода findIdentity() получает идентификационные класс пользователя с помощью токена. Этот метод аутентификации нужен только в том случае, когда надо аутентифицировать пользователя по секретному токену.

```
public function getId()  
{  
    return $this->id;  
}
```

Рисунок 3 - метод getId()

На рисунке 3 продемонстрирован метод getId(), который возвращает ID пользователя, проходящий аутентификацию.

```
public function getAuthKey()  
{  
    return $this->authKey;  
}
```

Рисунок 4 - метод getAuthKey()

На рисунке 4 можно увидеть метод getAuthKey(), который возвращает ключ авторизации, использующийся для аутентификации, основанной на cookie. Сам ключ сохраняется в cookie аутентификации для того, чтобы позже его сравнить с версией, которая находится на сервере для удостоверения того, что cookie аутентификация валидная.

```
public function validateAuthKey($authKey)  
{  
    return $this->authKey === $authKey;  
}
```

Рисунок 5 - метод validateAuthKey()

На рисунке 5 представлен метод validateAuthKey, который реализует логику проверки ключа для аутентификации, основанной на cookie.

```
$isGuest = Yii::$app->user->isGuest;  
$identity = User::findOne(['username' => $username]);  
Yii::$app->user->login($identity);  
Yii::$app->user->logout();
```

Рисунок 6 - Функции аутентификации на проверку гостя, получение идентификации пользователя и авторизация, выход пользователя

На рисунке 6 продемонстрированы функции аутентификации. isGuest - проверяет текущее состояние аутентификации, если пользователь не авторизован в веб-системе, то он является гостем и эта функция становится истинной. findOne() - получает идентифицированного пользователя по его

логину, `login()` - собственно данная функция авторизует пользователя (заходит в свой аккаунт), а функция `logout()` - наоборот производит выход из своего аккаунта в веб-системе.

В самом PHP фреймворке Yii2 имеется два зарегистрированных пользователя - `admin` и `demo`. Особенность аутентификации в этом фреймворке в том, что ей не обязательна база данных зарегистрированных пользователей, для тестирования новых пользователей достаточно добавить в файл `User.php` новые идентификационные данные, что в дальнейшем и будет рассмотрено

```
private static $users = [  
    '100' => [  
        'id' => '100',  
        'username' => 'admin',  
        'password' => 'admin',  
        'authKey' => 'test100key',  
        'accessToken' => '100-token',  
    ],  
    '101' => [  
        'id' => '101',  
        'username' => 'demo',  
        'password' => 'demo',  
        'authKey' => 'test101key',  
        'accessToken' => '101-token',  
    ],  
    '102' => [  
        'id' => '102',  
        'username' => 'test_user',  
        'password' => '12345',  
        'authKey' => 'test_auth_key',  
        'accessToken' => 'test_access_token',  
    ],  
];
```

Рисунок 7 - Массив `$users`, в котором добавлены пользователи системы в файле `User.php`

На рисунке 7 продемонстрирован начальный фрагмент PHP кода файла `User.php` в котором имеется массив переменной `$users`, содержащий пользователей системы и их идентификационные данные: `id` - идентификатор, `username` - имя пользователя или логин, `password` - пароль, `authKey` - ключ авторизации, `accessToken` - токен доступа. Сейчас на рисунке 7 добавлен третий пользователь с идентификатором 102 и логином - `test_user`.

Далее необходимо перейти в саму веб-страницу формы авторизации и в соответствующие поля ввода вбить логин и пароль добавленного на предыдущем рисунке третьего пользователя `test_user`

[Home](#) / [Login](#)

## Login

Please fill out the following fields to login:

**Username**

**Password**

**Remember Me**

Рисунок 8 - форма авторизации в PHP фреймворке Yii2 с введенными данными третьего пользователя test\_user

Рассматривая рисунок 8, можно увидеть представленную форму авторизацию, в которую в соответствующие поля написан логин - "test\_user" и пароль "12345". По желанию можно оставить галочку "Remember Me" (запомнить меня) и нажать на синюю кнопку "Login" и после проверить, как прошла аутентификация пользователя "test\_user"

## User

User

## User Info

<b>Access Token</b>	'test_access_token'
<b>Auth Key</b>	'test_auth_key'
<b>Id</b>	'102'
<b>Password</b>	'12345'
<b>Username</b>	'test_user'

Рисунок 9 - Отладка с отображением успешной аутентификации и идентификации пользователя "test\_user"

На рисунке 9 изображен отладка PHP фреймворка Yii2, в котором открыт раздел User, содержащий информацию об авторизованном пользователе test\_user и его идентификационными данными. Если в отладке отображается данная информация, то значит аутентификация прошла успешно и данный пользователь получил доступ к своему аккаунту.

Таким образом, была рассмотрена аутентификация в PHP фреймворке Yii2, позволяющая провести проверку на подлинность пользователя, тем самым идентифицировав самого пользователя, который войдет в саму веб-систему. Также были рассмотрены функции и методы аутентификации, которыми обладает PHP фреймворк Yii2 для регистрации и авторизации пользователей.

### **Библиографический список**

1. Титов В.С. Идентификация и аутентификация. Способы и методы аутентификации // В сборнике: Международная научно-техническая конференция молодых ученых БГТУ им. В.Г. Шухова. 2017. С. 4514-4519.
2. Цебрук Е.В. Проблема аутентификации информации в локальных сетях // В сборнике: Современные тенденции в научной деятельности. VII Международная научно-практическая конференция. Научный центр "Олимп". 2015. С. 1531-1533.
3. Балашов Н.А. Аутентификация для веб-приложений с использованием сгенерированных на автономном устройстве одноразовых паролей // В сборнике: Проблемы правовой и технической защиты информации. Сборник научных статей. В.В. Поляков - ответственный редактор. 2015. С. 20-24.
4. Кошельков И.Н. Методы статической и динамической аутентификации на основе клавиатурного почерка // В сборнике: Роль и место информационных технологий в современной науке. Сборник статей Международной научно-практической конференции. Ответственный редактор: Сукиасян Асатур Альбертович. 2016. С. 28-34.
5. Шафер А.Е., Черников А.В. Двухфакторная аутентификация с использованием СМС-сервиса // Вестник Пермского университета. Серия: Математика. Механика. Информатика. 2015. № 1 (28). С. 79-85.
6. PHP Framework Yii2 URL: <https://www.yiiframework.com/> (дата обращения 16.01.2021)
7. Локальный веб-сервер OpenServer URL: <https://ospanel.io/> (дата обращения 16.01.2021)