

Универсальный модуль аутентификации пользователей на базе PHP-фреймворка Symfony

Вальчук Виталий Юрьевич

УО "Брестский государственный университет имени А.С. Пушкина"

Большинство современных веб-приложений предназначено для взаимодействия не только с анонимными пользователями (например, в случае новостных ресурсов), но и аутентифицированными пользователями (блоги, системы управления ресурсами предприятий и др.). При проектировании и создании таких приложений требуется реализовать функции входа пользователей в систему (аутентификации), хранения пользовательских данных в базе данных и в сессии веб-приложения, выхода из системы, регистрации пользователей и управления перечнем пользователей.

Современные PHP-фреймворки (в частности, Laravel [1] и Symfony [2]) функции аутентификации пользователей уже реализованы в той или иной степени. В Laravel достаточно выполнить две консольные команды [3] для построения каркаса аутентификации, включающего модели, контроллеры, представления (шаблоны веб-страниц). В Symfony также есть сервисные функции поддержки аутентификации пользователей, но лишь в виде незавершенного каркаса [4], завершение которого предлагается разработчикам конкретных веб-проектов с учетом их назначения и функциональности. В то же время, документация фреймворка Symfony рекомендует использовать стороннюю реализацию – модуль FOSUserBundle [5], документация FOSUserBundle располагается на официальной странице документации Symfony.

К сожалению, собственный неудачный опыт использования FOSUserBundle, а также ряд экспертных мнений [6] не позволяют рекомендовать FOSUserBundle для задач аутентификации пользователей, несмотря на популярность этого модуля. Более того, популярность (число загрузок) за месяц выглядят парадоксально на фоне неудачных архитектурных решений, принятых в FOSUserBundle:

- ряд полей в классе сущности User явно излишни: `username_canonical` (ведь уже есть `username`), `email_canonical`, `salt` (при хешировании паролей алгоритмом `bcrypt` соль не требуется);
- поле `roles` содержит массив (несколько значений), чем явно нарушает уже первую нормальную форму [7] в реляционной модели данных.

В связи со сказанным, целью настоящей работы является создание универсального модуля аутентификации пользователей для веб-приложений на базе фреймворка Symfony.

В соответствии с общими рекомендациями, принятыми разработчиками фреймворка Symfony, модуль авторизации реализован в виде разделяемого бандла (`reusable bundle` [8]). Все классы размещены в пространстве имен `triguk/AuthenticationBundle`.

В настоящее время модулем предусмотрены:

- контроллер, форма и шаблон веб-страницы для входа пользователя в систему;
- контроллер, формы и шаблоны веб-страниц административного раздела, предназначенного для создания записей о пользователях (регистрации новых пользователей) и редактирования сведений (в том числе для смены пароля);
- контроллер, формы и шаблоны веб-страниц самостоятельной регистрации пользователей.

В созданном нами модуле отсутствуют функции авторизации пользователей (контроля и ограничения доступа пользователей к ресурсам веб-приложения), поскольку с этой целью создавался другой модуль – `triguk/AuthorizationBundle`. Отсутствуют также функции сторонней аутентификации с помощью социальных сетей, необходимость добавления такой функции находится в стадии рассмотрения.

Созданный нами модуль (`bundle`) является разделяемым, с возможностью повторного использования, пригоден к использованию в несложных веб-приложениях, не требующих входа в систему через сторонние ресурсы (LDAP-серверы или социальные сети). Отсутствует ряд недостатков, присущих популярному модулю `FOSUserBundle` (усложненность излишними полями в базе данных и функциями, нарушение первой нормальной формы и др.). Предлагаемый нами модуль может использоваться как без изменений (путем подключения к имеющемуся Symfony-проекту), так и в качестве открытой кодовой базы, путем дополнения требуемым в конкретном веб-приложении функционалом (хранение дополнительных сведений о пользователях, подключение иных методов аутентификации).

В настоящее время модуль проходит апробацию в веб-проекте `TransitControl`, предназначенном для контроля за техническим состоянием и состоянием документации грузового транспорта логистических предприятий.

СПИСОК ЛИТЕРАТУРЫ

1. Laravel - The PHP Framework For Web Artisans [Electronic resource]. – Mode of access: <https://laravel.com/>. – Date of access: 20.12.2016.
2. Symfony, High Performance PHP Framework for Web Development [Electronic resource]. – Mode of access: <https://symfony.com/>. – Date of access: 20.12.2016.
3. Authentication [Electronic resource] // Laravel - The PHP Framework For Web Artisans. – Mode of access: <https://laravel.com/docs/master/authentication>. – Date of access: 24.12.2016.
4. Authentication [Electronic resource] // Symfony, High Performance PHP Framework for Web Development. – Mode of access: <http://symfony.com/doc/current/components/security/authentication.html>. – Date of access: 24.12.2016.
5. Getting Started With FOSUserBundle [Electronic resource] // Symfony, High Performance PHP Framework for Web Development. – Mode of access: <http://symfony.com/doc/current/bundles/FOSUserBundle/index.html>. – Date of access: 24.12.2016.
6. Do not use FOSUserBundle [Electronic resource] // JoliCode - Réalisation de projets Web et mobiles de qualité. – Mode of access: <https://jolicode.com/blog/do-not-use-fosuserbundle>. – Date of access: 24.12.2016.
7. Дейт, К. Дж. Введение в системы баз данных / Дж. К. Дейт. – М.: Вильямс, 2005. – 1328 с.
8. The Bundle System [Electronic resource] // Symfony, High Performance PHP Framework for Web Development. – Mode of access: <http://symfony.com/doc/current/bundles.html>. – Date of access: 24.12.2016.