

Описание системы

Необходимо разработать информационную систему для создания, просмотра и редактирования постов.

Роли в системе:

- гость, не авторизованный пользователь;
- автор, авторизованный пользователь;
- администратор системы, авторизованный пользователь с правами управления системой.

Функционал гостя:

- 1) просматривать ленту постов;
- 2) открывать страницу со всеми постами;
- 3) открывать страницу поста с детальной информацией по посту;
- 4) регистрироваться в системе;
- 5) авторизоваться, после авторизации переходит в статус автора.

Функционал автора:

- 1) просматривать ленту постов;
- 2) открывать страницу со всеми постами;
- 3) открывать страницу поста с детальной информацией по посту;
- 4) написание поста;
- 5) редактирование поста и удаление поста. Удалять посты возможно только те, у которых нет комментариев;
- 6) написание комментариев на любой пост (кроме своих), ответить на чей-то комментарий у своих постов.

Функционал администратора:

- 1) просматривать ленту постов;
- 2) открывать страницу со всеми постами;
- 3) открывать страницу поста с детальной информацией по посту;
- 4) удаление любого поста. Если у поста были комментарии, то они так же удаляются;
- 5) удаление любого комментария к посту;
- 6) просмотр списка пользователей в системе;
- 7) блокировка входа пользователя в систему на определенное время либо на всегда. Если доступ пользователю блокируется на всегда, то удаляются все посты (с веткой комментариев к посту) этого пользователя.

Страницы информационной системы.

По мимо страниц, предусмотреть кнопки «Регистрация» и «Вход». Данные кнопки доступны для не авторизованного пользователя. Кнопка «Выход» для авторизованного пользователя.

Страница "О нас".

На странице представлена информация по описанию системы, включая информацию о разработчике.

Главная страница.

Страница содержит ленту превью 10 последних добавленных постов, с указанием даты и времени создания поста, автора (логина) и анонса (превью) поста.

Страница с постами.

Страница содержит ленту превью всех постов, с указанием даты и времени создания поста, автора (логина) и анонса (превью) поста.

Для автора поста доступна кнопка редактирования поста.

Для администратора доступна кнопка удаления поста.

Страница просмотра поста.

Страница содержит текст поста, ветку комментариев к посту, форму создания комментария к посту с учетом условий комментирования поста. Элементы для: перенаправления на редактирование поста и удаления поста либо комментария с учетом разрешений.

Страница создания (редактирования) поста.

Страница содержит форму создания поста. На форме должны присутствовать элементы: создания превью поста, самого текста поста, а также, должна быть возможность удаления поста с учетом прав и разрешений.

Регистрация нового пользователя.

Форма регистрации нового пользователя в системе должна содержать следующие поля:

- name – обязательное поле;
- surname – обязательное поле;
- patronymic – не обязательное поле;
- login – обязательное и уникальное поле;
- email – обязательное и уникальное поле, проверка на email;
- password – обязательное поле, не менее 6-ти символов;
- password_repeat – обязательное поле, должно совпадать с полем password;
- rules - согласие с правилами регистрации.

При ошибках валидации пользователю должны выводиться сообщения. Валидация полей должна происходить на сервере.

Аутентификация

Форма авторизации пользователя в системе должна содержать следующие поля:

- login;
- password.

При ошибках валидации пользователю должны выводиться сообщения.

Страница со списком пользователей в системе.

Страница содержит список пользователей в системе с элементами управления пользователями.

Логин и пароль для администратора системы определяет разработчик системы.

Технические требования к системе.

Информацию о пользователях, постах и комментариях хранить в базе данных. Базу данных реализовать на MySQL или MariaDB. Роли пользователей должны храниться в отдельной таблице. Пароли пользователей должны храниться в БД в захешированном виде.

В случае, если пользователь зарегистрированный, ко всем ссылкам в системе добавить параметр – токен пользователя. Токен представляет собой строку из случайных символов. Длину строки определяет разработчик. Токен должен быть уникальный в системе для каждого пользователя. Токен должен генерироваться при авторизации пользователя и удаляется при его выходе из системы. Передача токена происходит через параметр в url.

На пример: `http://xxxxxxx.php?token=jhkEwRbJIk34ub`.

Может быть активен только один токен, который был выдан пользователю при последней авторизации. В случае если пользователь был авторизован на нескольких устройствах (браузерах), то при действиях в системе с недействительным токеном, пользователь должен быть разлогинен в системе (без удаления токена в БД) и перенаправлен на главную страницу.

Если необходимо передать информацию о посте, то можно передавать ее через параметр, на пример: `http://xxxxxxx.php?token=jhkEwRbJIk34ub&post=10`, где `post=10` данные о редактируемом/удаляемом посте. Аналогично можно передавать ссылку о пользователе, при реализации функционала администратора.

*) Дополнительно функционал системы может быть расширен: могут быть реализованы изображения для постов и аватарки для пользователей.