

Техническое задание на разработку Web API доски объявлений пользователей.

Используемые технологии

- Приложение должно быть написано строго на языке C# либо Java.
- СУБД для приложения должен быть MsSql/MySql/Postgres.

Общая информация

Разработать приложение, которое обеспечивает функциональность:

- регистрацию, аутентификацию, авторизацию пользователей
- разделение прав доступа пользователей по ролям
- реализовать матрицу действий по управлению объявлениями
- реализовать матрицу действий по управлению пользователями
- категоризацию объявлений (авто/недвижимость/техника/...)
- типизацию объявлений (куплю/продам/найдено..)
- просмотр объявлений с фильтрацией по категории, по типу
- формирование и ведение базы данных зарегистрированных пользователей, объявлений, ролей, типов и категорий объявлений
- предоставление API клиентам сервера

Используемые обозначения:

- Пользователь - авторизованный пользователь, который может публиковать объявления
- Объявление - текст, опубликованный пользователем
- Роль – права доступа пользователя (администратор, модератор, пользователь) по управлению сервером объявлений

Матрица действий по управлению объявлениями:

Доступные действия/Роль	Не зарегистрированный пользователь	Пользователь	Автор объявления	Модератор	Администратор
Публикация объявления	-	+	+	+	+
Редактирование объявления	-	-	+(своих)	+	+
Удаление объявления	-	-	+(своих)	-	+
Просмотр объявлений	+	+	+	+	+

Матрица действий по управлению пользователями:

Доступные действия/Роль	Не зарегистрированный пользователь	Пользователь	Модератор	Администратор
Редактирование роли пользователя	-	-	-	+
Редактирование имени пользователя	-	+(свое имя)	+	+
Удаления пользователя	-	-	-	+
Ограничение возможности публикации объявлений	-	-	+	+

База данных должна содержать следующую информацию:

- зарегистрированные пользователи
- объявления зарегистрированных пользователей
- роли пользователей
- категории/подкатегории объявлений
- типы объявлений
- фотографии или видео объявлений

Требования:

- проект должен компилироваться и запускаться без ошибок.
- обеспечить автоматическое формирование и первичное заполнение БД.
- не хранить строку подключения к БД в коде, использовать конфиги.
- реализовать валидацию модели пользователя (логин и пароль не пустые и не должны быть одинаковыми).
- реализовать авторизацию пользователей по ролям.
- реализовать аутентификацию пользователей любым из перечисленных способов (jwt-токен, cookie, клаймы, роли).
- таблица „Пользователи“ должна содержать: имя пользователя и пароль.
- таблица „Роли“ должна содержать: наименование роли.
(пользователь/модератор/администратор).
- таблица „Объявления“ должна содержать: текст, дату формирования, тип, категорию, **ссылки на фото и видео (ссылки на ресурсы Api)**.
- таблица „Категории объявлений“ должна содержать: наименование категории (авто/недвижимость/бытовая техника).
- таблица „Подкатегории объявлений“ должна содержать: наименование подкатегории (дом, дача, иномарки, отечественные автомобили) и ссылку на категорию.
- таблица „Тип объявлений“ должна содержать: наименование типа (куплю/продам/меняю/найдено/пропажа/сдам в аренду).
- удаление пользователя не должно приводить к удалению его роли, объявлений и т.д. (мягкое удаление)
- удаление объявления не должно приводить к удалению автора, категории и типа объявления. (мягкое удаление)

Ограничения:

- к объявлению можно загрузить не более 5 файлов.
- общий объем файлов не должен превышать 250 Мб (при превышении лимита Api должна корректно работать).
- файлы для объявлений должны быть либо картинками либо видеофайлами и ничем другим.

Будет плюсом:

- использование миграций для изменения структуры БД и заполнения данными.
- использование встроенных механизмов валидации моделей.
- пароль в базе данных хранится в зашифрованном виде.
- работа с базой данных в приложении реализована через ORM.
- чистый код (логика работы api и работа с БД в отдельных классах и сервисах).
- использование паттернов.
- использование Unit-тестов.