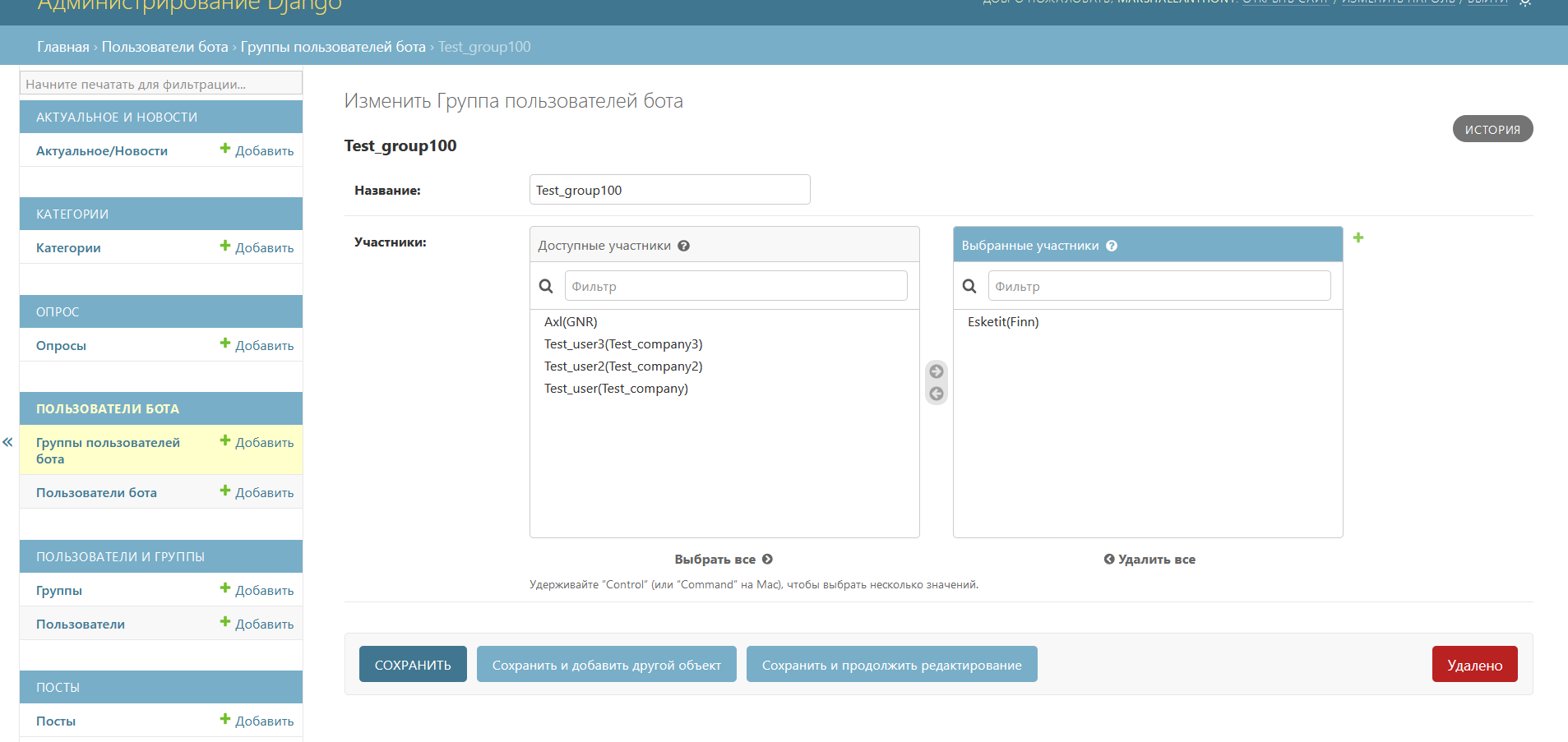
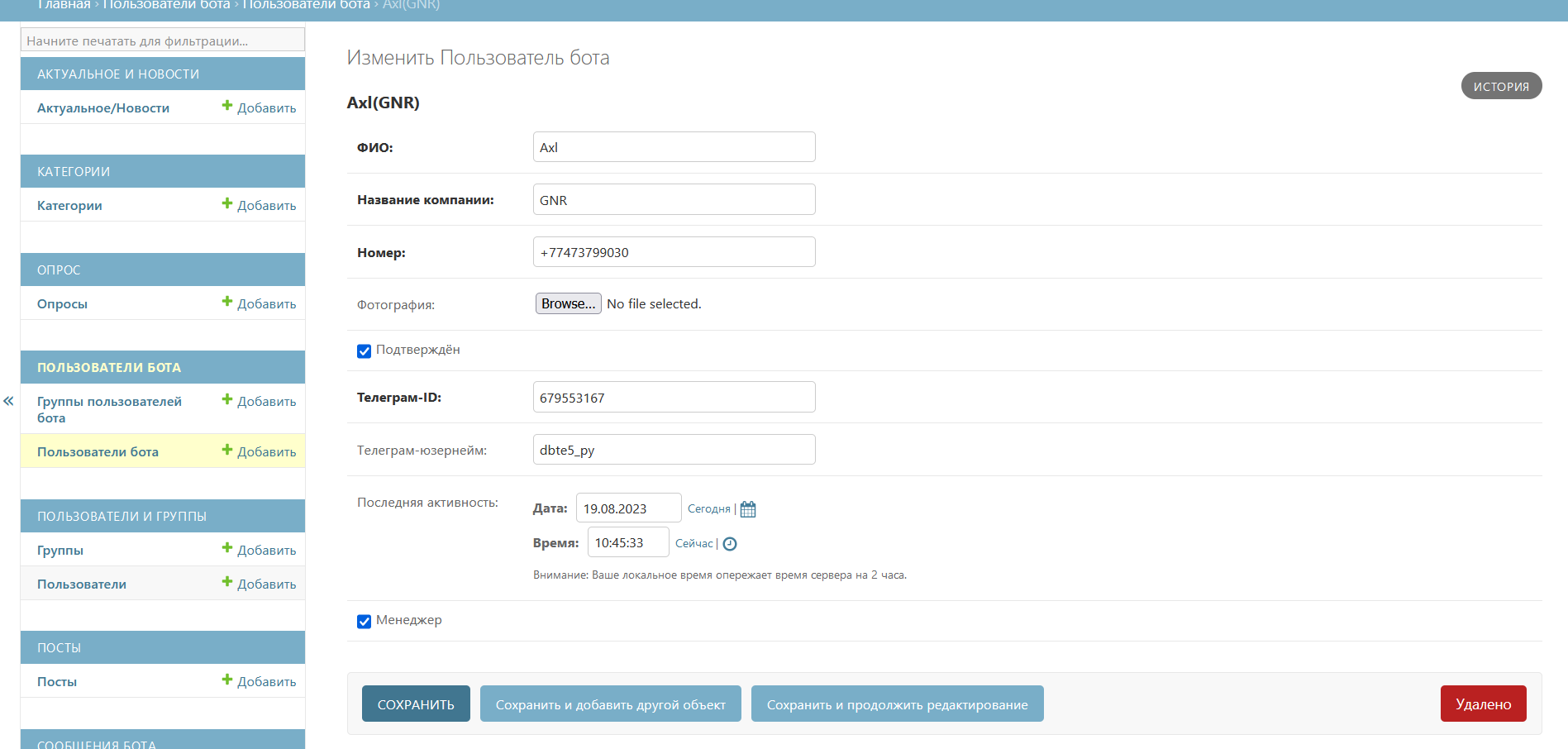
# Пользователи бота

### 1.Группы пользователей бота



В админ-панели можно создать группу для пользователей бота, куда можно добавить одного или нескольких пользователей для их сегментации и рассылок по группам.

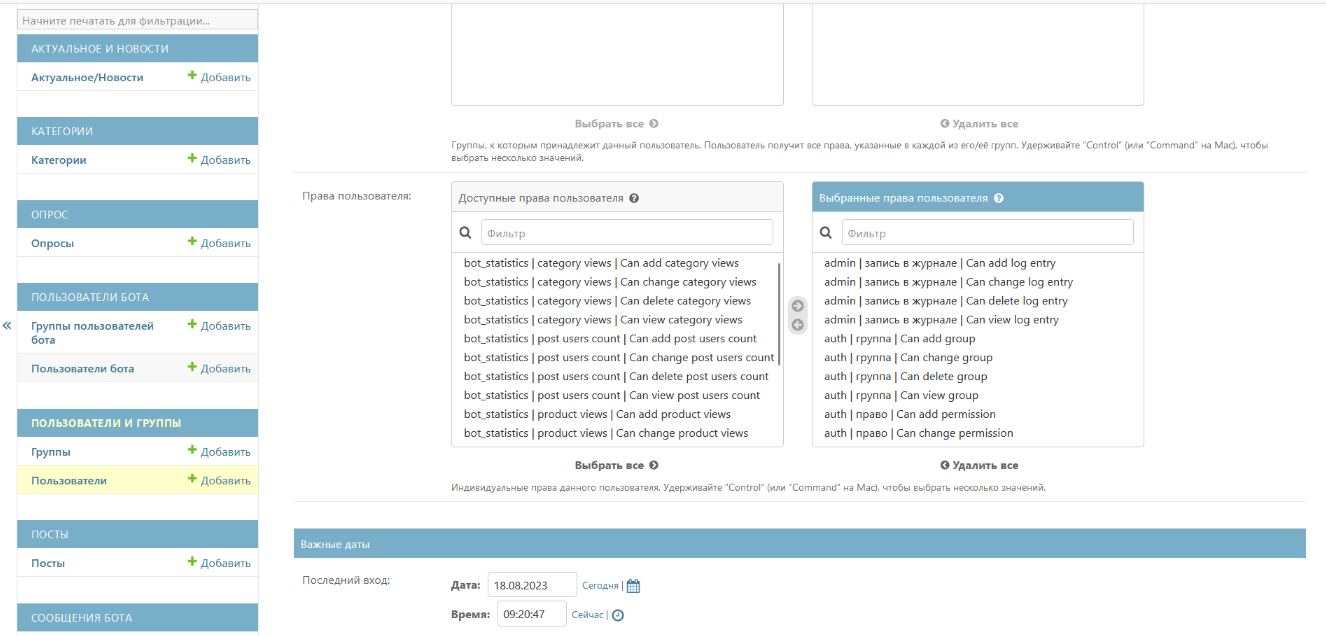
### 2.Пользователи бота



**Пользователи бота** – это телеграм-пользователи, которые зарегистрировались через бота, с помощью команды /start.

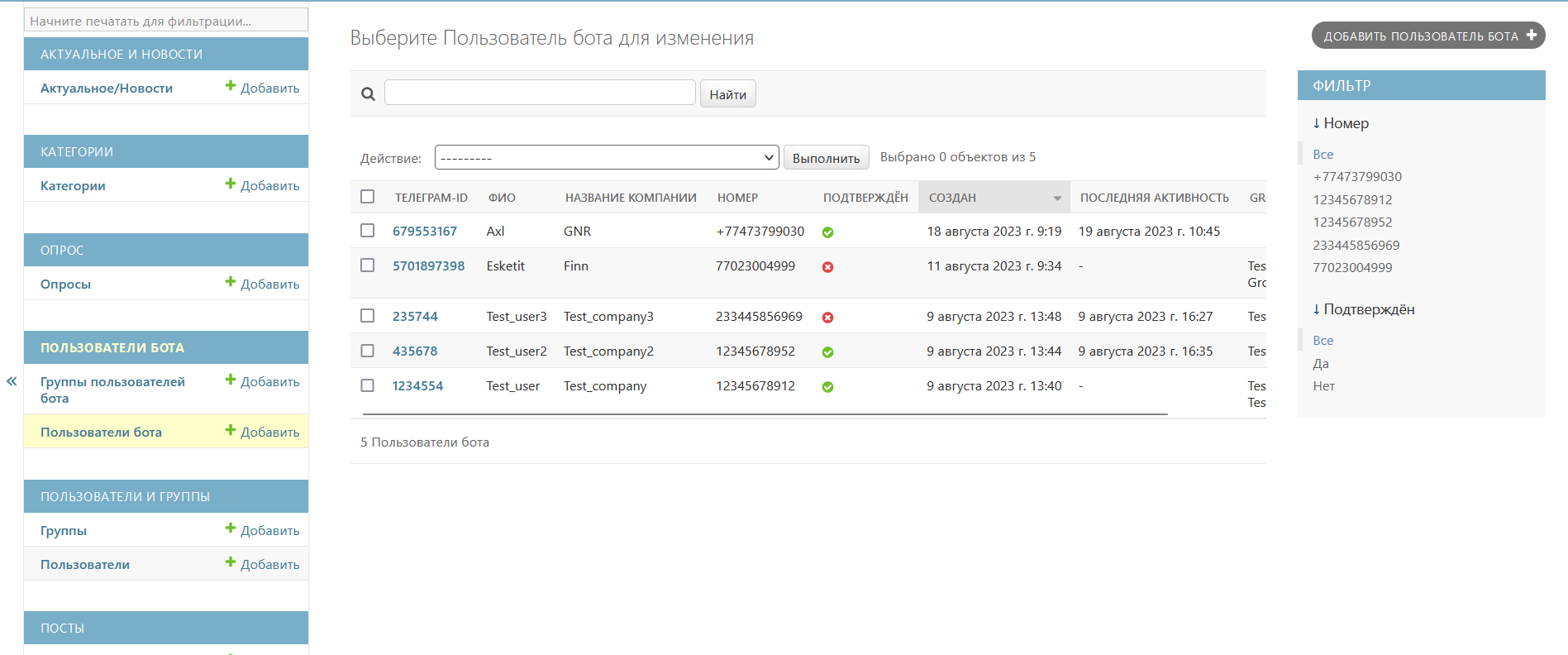
**После завершения регистрации**, аккаунт отправляется админу на подтверждение (о подтверждении аккаунта говорит галочка у поля “Подтверждён”).  
Если аккаунт подтверждён, то пользователь бота сможет им пользоваться.

**Админ может назначать менеджерами** пользователей бота, поставив галочку у поля “Менеджер”. После этого, в личные сообщения пользователя придёт логин и пароль для входа в админ-панель с кнопкой “Подтвердить”, которую должен будет нажать для подтверждения аккаунта.   
После этого, админ должен зайти в “Пользователи и группы”->Пользователи, где появится этот пользователь бота. В этой вкладке менеджеру можно раздать разрешения на определённые действия:



Экспорт пользователей:

**Чтобы экспортировать пользователей** нужно нажать на выпадающее меню “Действие” и выбрать “Экспортировать выбранные Пользователи бота”, после чего нажать на “Выполнить”



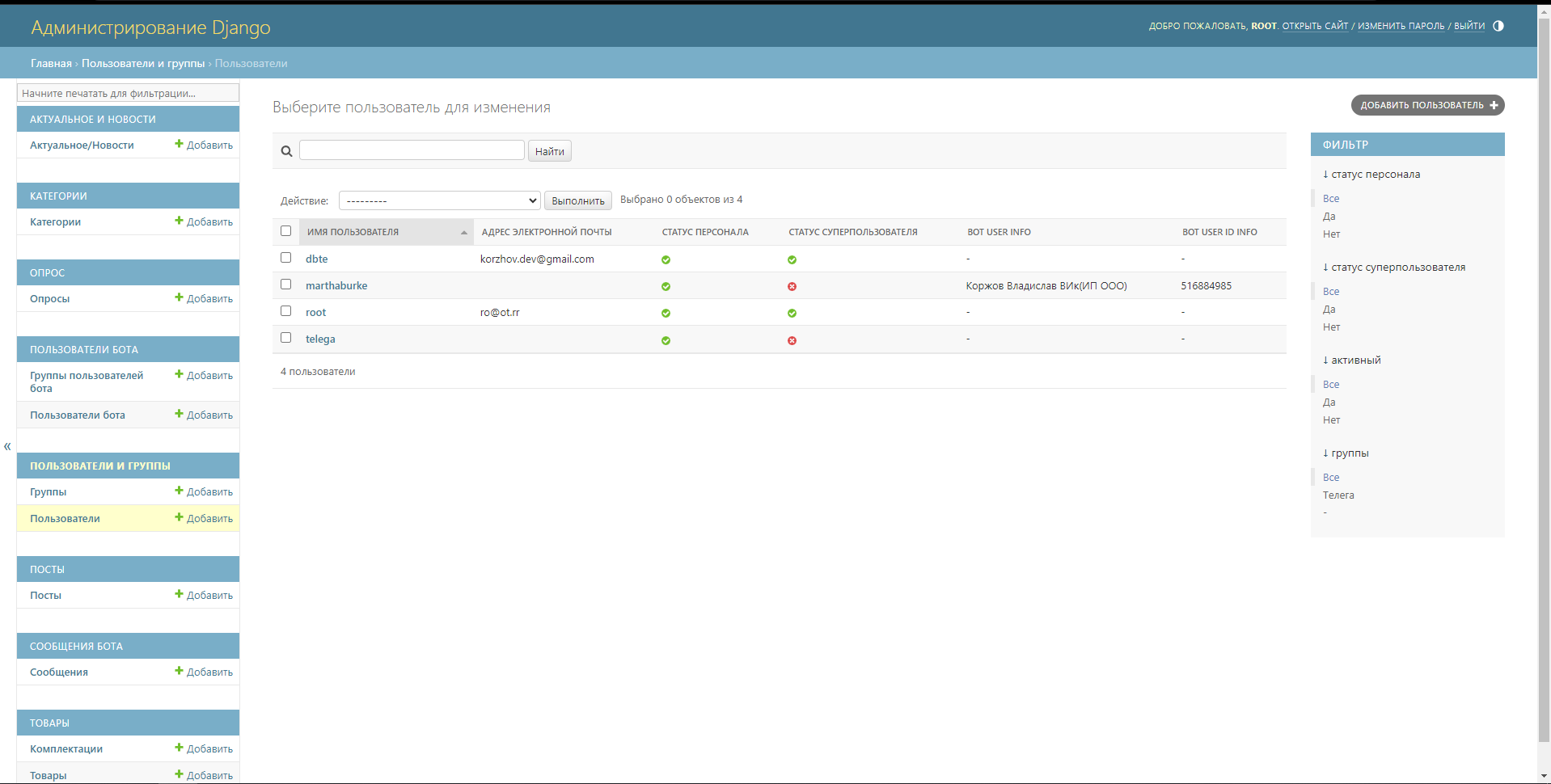
(“Действие” находится под полем поиска)

# Пользователи и группы

## 1. Группы

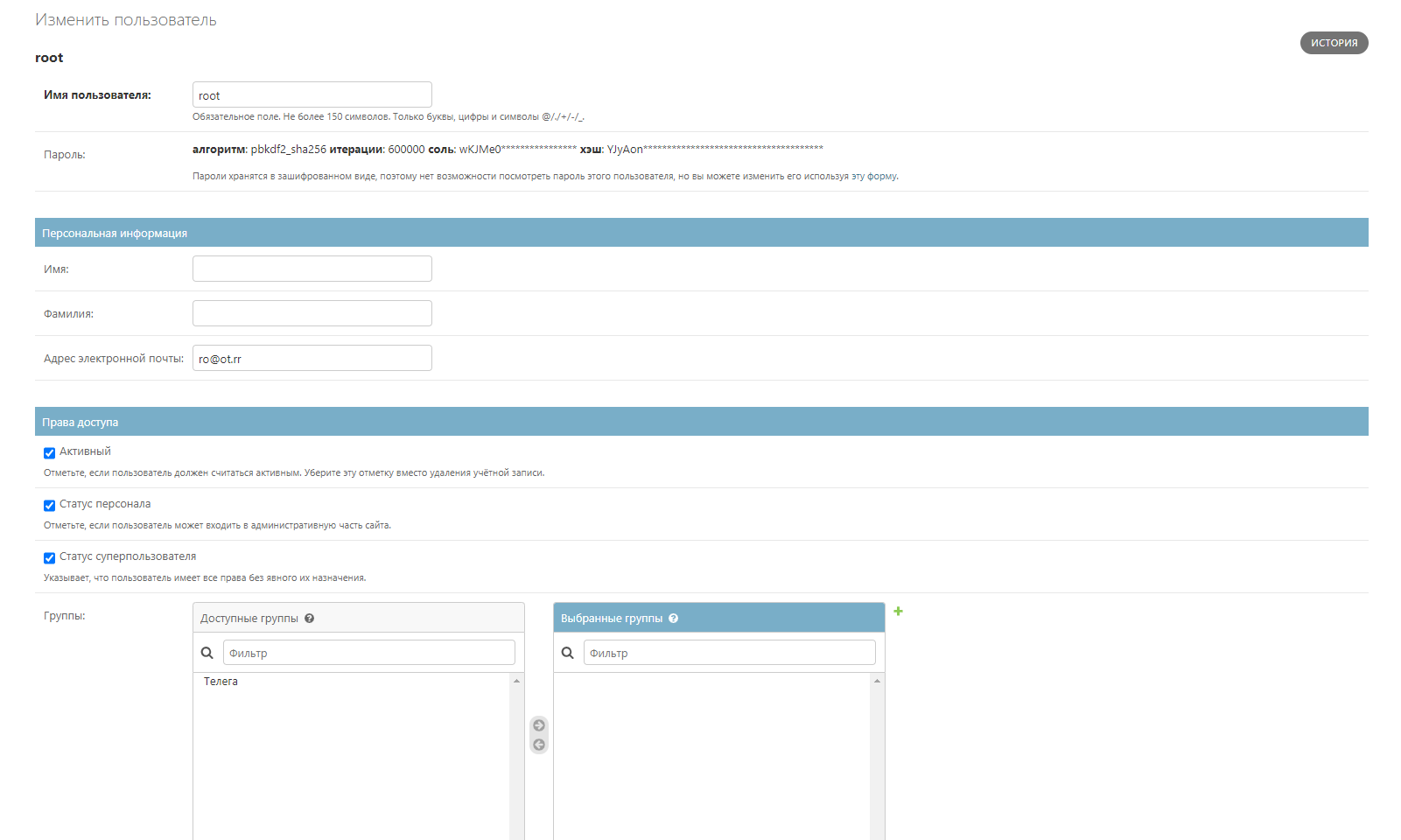
Позволяют создать группу с определенным набором правил для дальнейшего назначения конкретному пользователю.

## 2. Пользователи



**Как было написано выше**, пользователи – это либо менеджеры, либо админ (и ещё Django-superuser, который имеет пустые Bot User Info и Bot User ID. Он не используется).

Здесь им можно раздать разрешения на определённые действия, сделать кого-то суперпользователем и т.д.



* **Имя пользователя –** Имя для доступа в админ панель
* **Пароль** – Пароль для доступа в админ панель
* **Права доступа –** Активный (Показывает статус, пользователь входил в админ панель), статус персонала (Позволяет пользователю входить в админ панель), Статус суперпользователя (Дает пользователю все права на бота).
* **Группы\* –** Позволяет наградить пользователя определенным набором прав доступа, созданными для группы
* **Права пользователя\* -** Позволяет наградить пользователя определенным набором прав доступа, созданными для него лично.

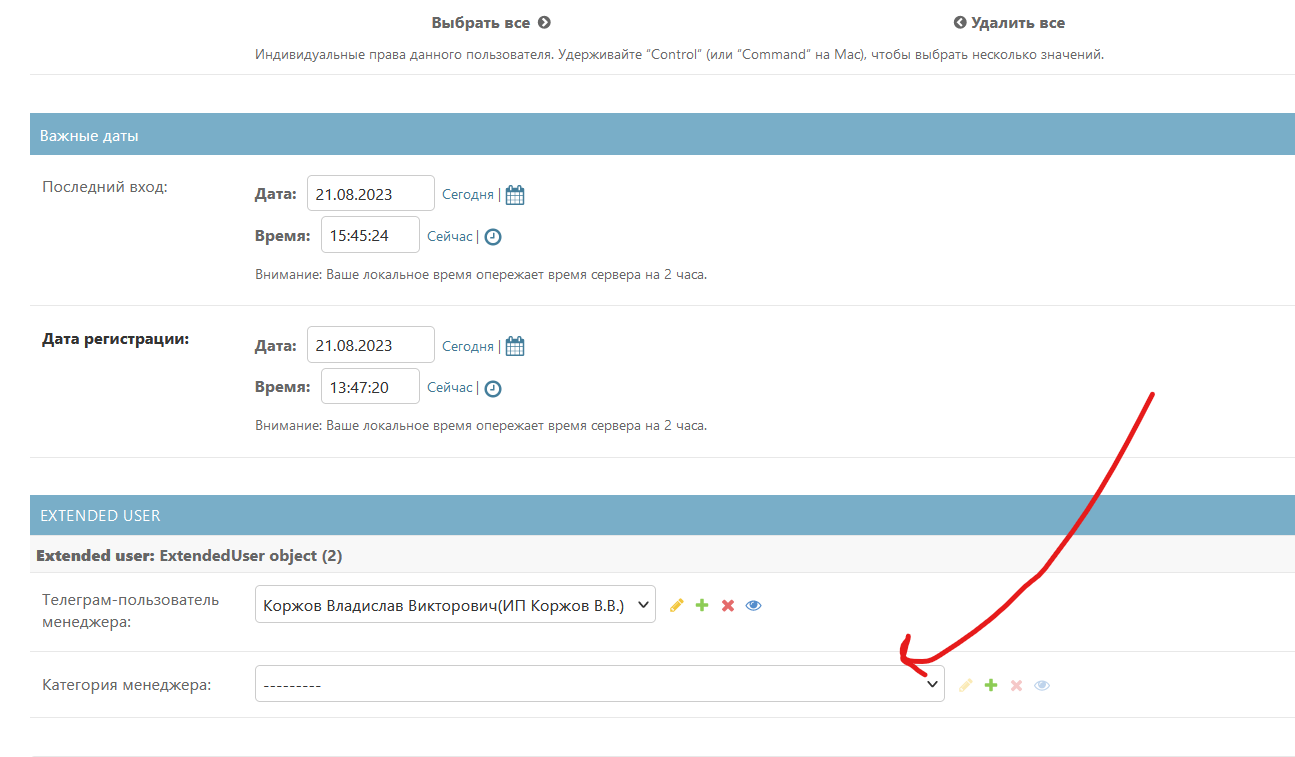
\* Can add – Может добавлять элемент модели

\* Can change – Может изменять элемент модели

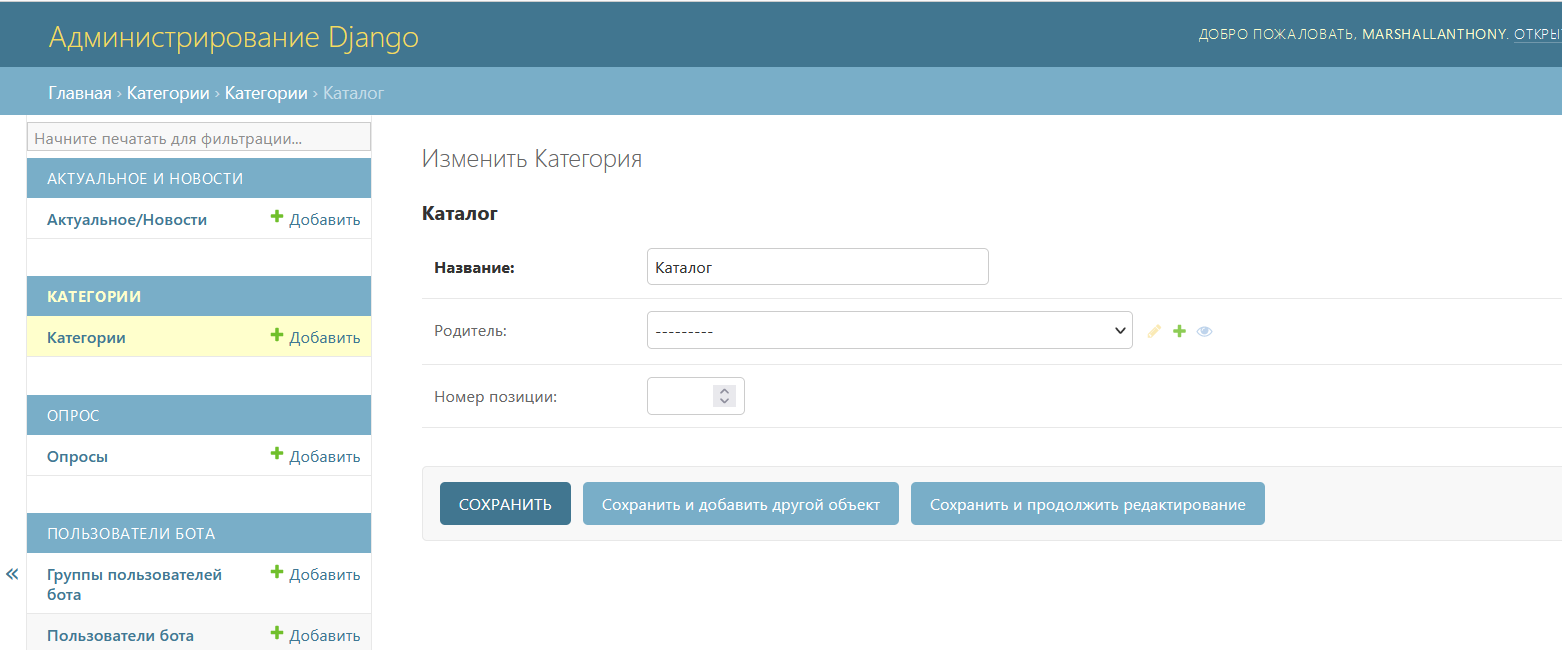
\* Can delete – Может удалять элемент модели

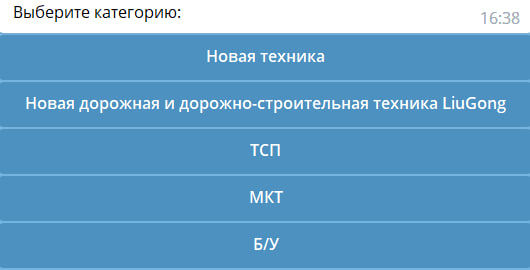
\* Can view – Может смотреть элемент модели

Также, у менеджера есть поле “Категория менеджера”. Это поле, которое показывает, за какую категорию ответственен тот или иной менеджер. Именно благодаря этому полю, при нажатии на “Чат с менеджером” или “Запрос на КП” у товара, будет находиться нужный менеджер.



# Категории

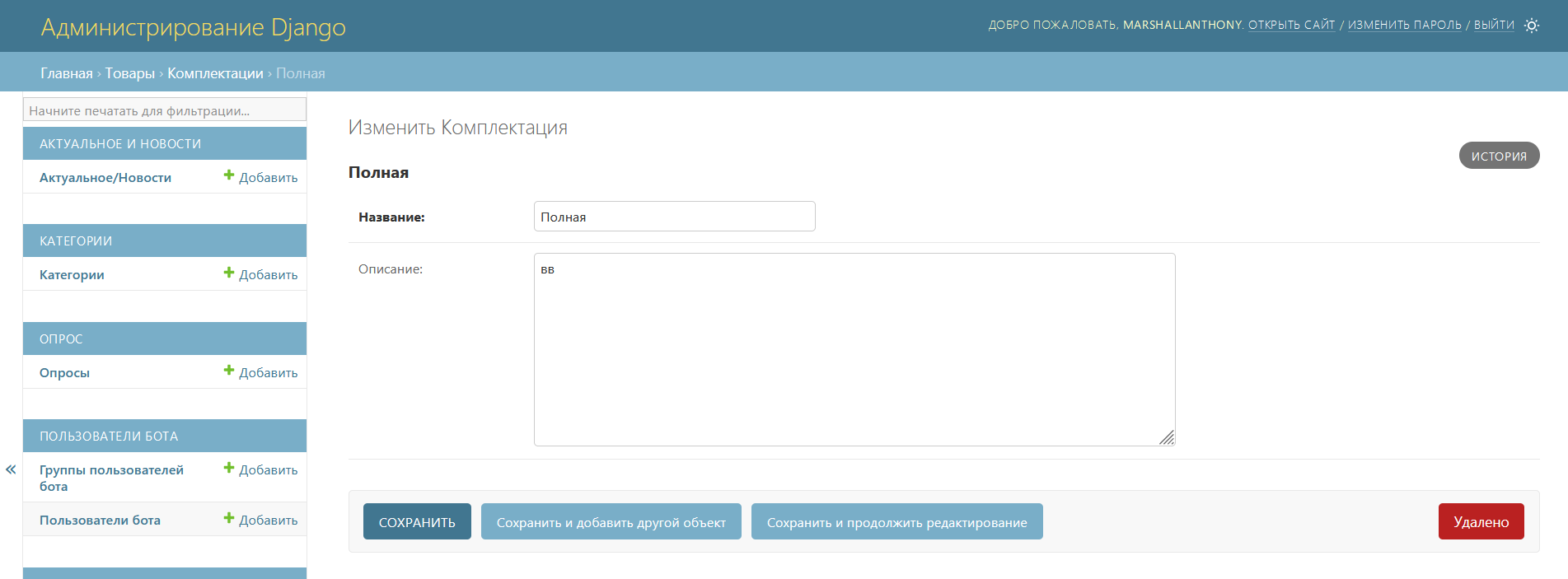
  
**Для создания** категории нужно указать её родителя, а также можно указать номер позиции, на которой вы бы хотели, чтобы находилась эта категория:

  
Поле можно оставить пустым, а можно заполнить номером 1,2,3 и т.д.

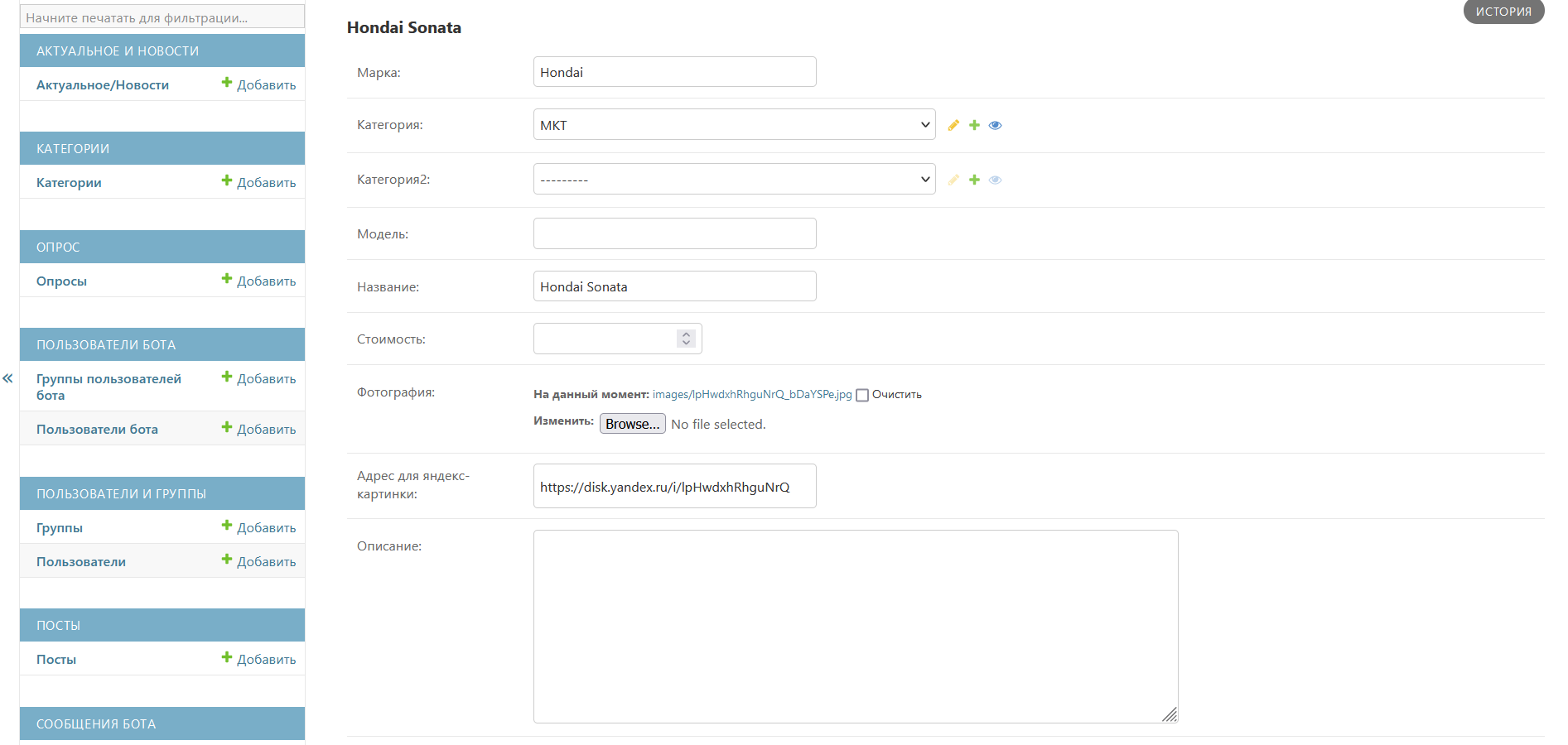
# Товары

### Комплектации

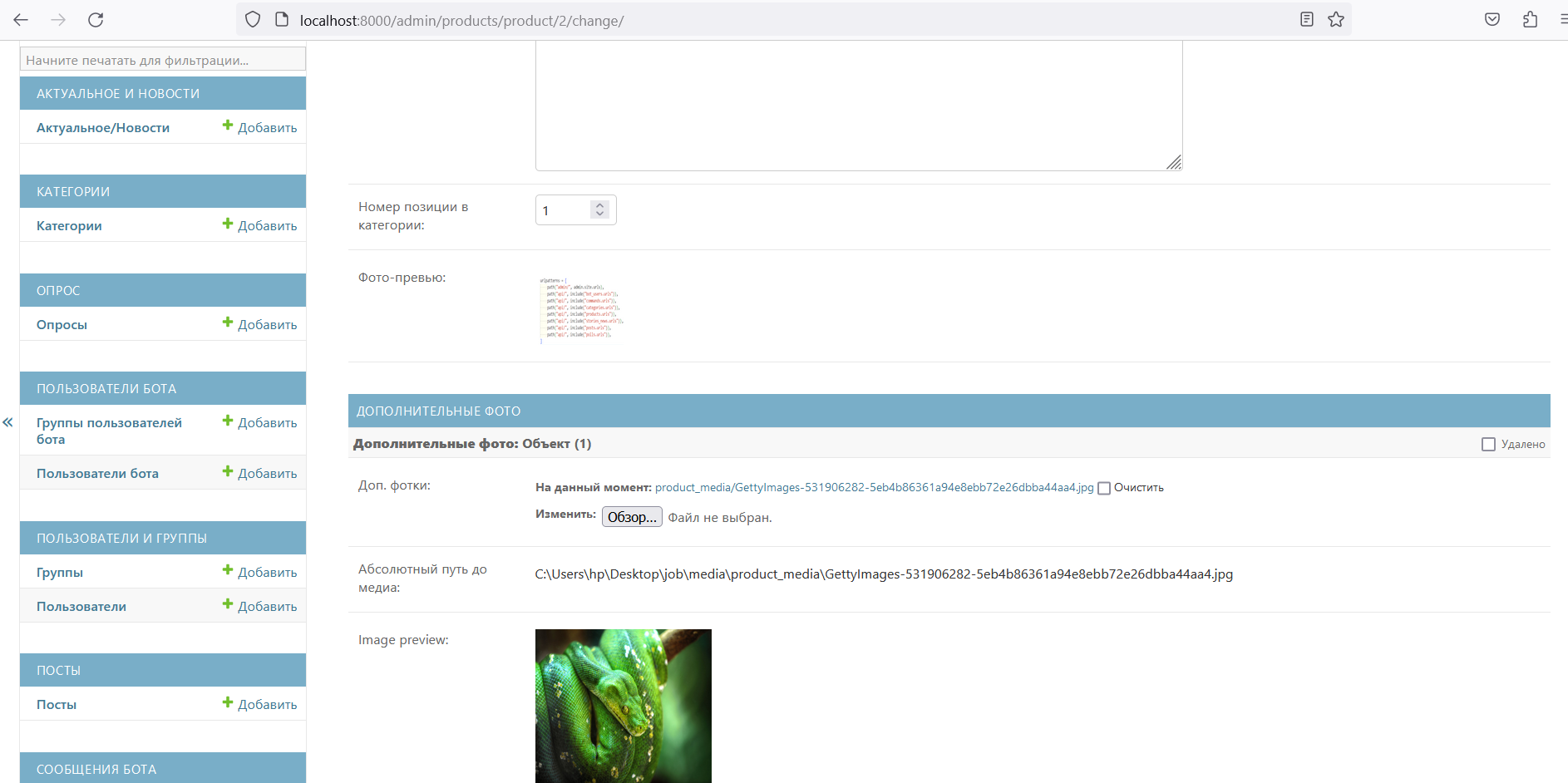
Комплектация, которую потом можно будет указать у товара



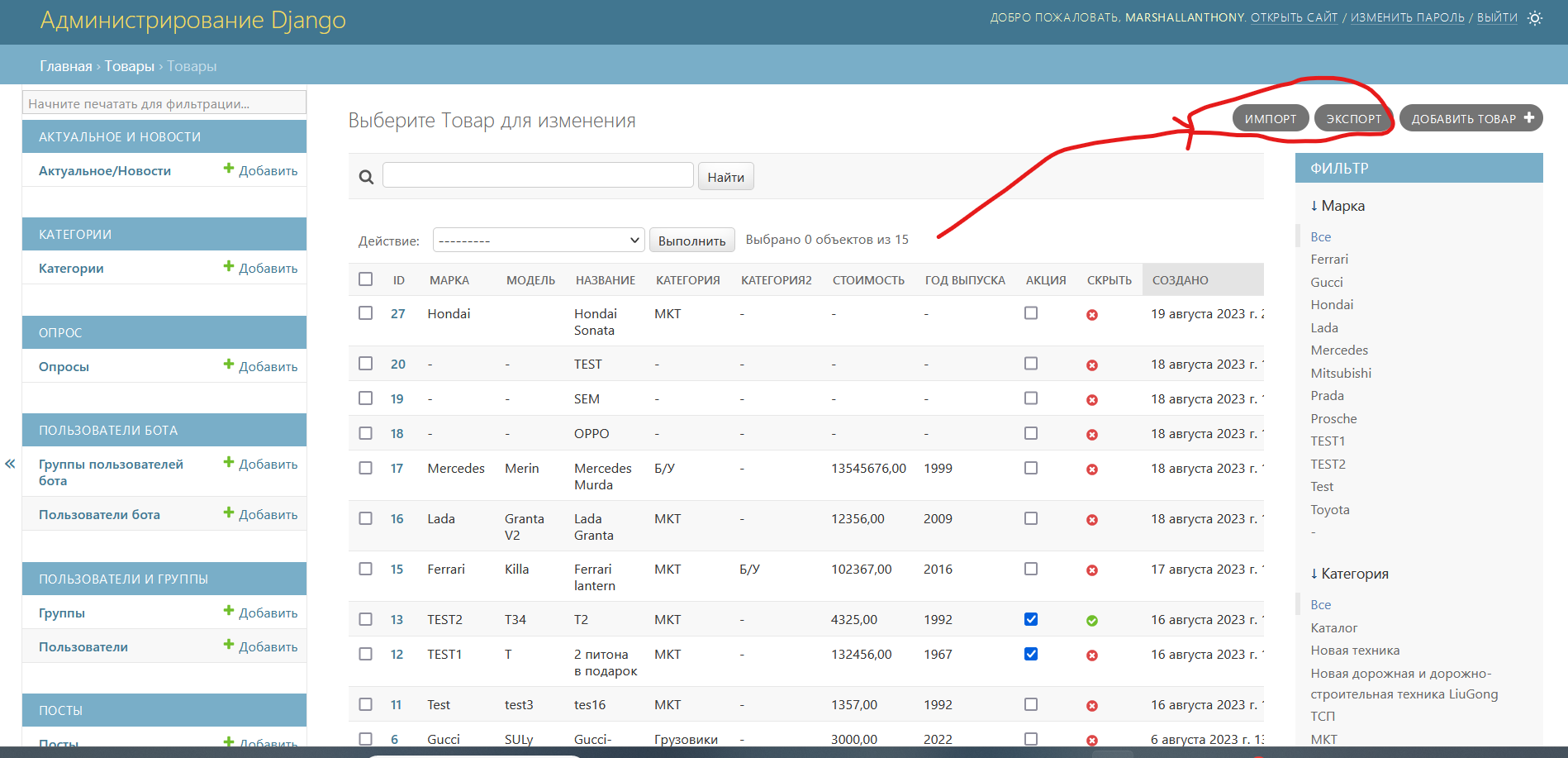
### Товары

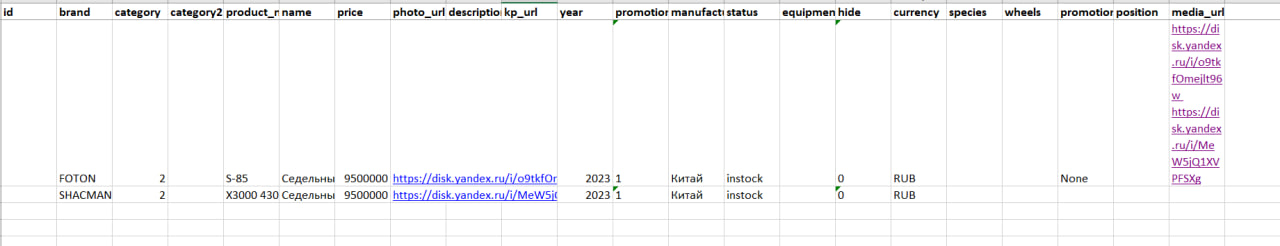
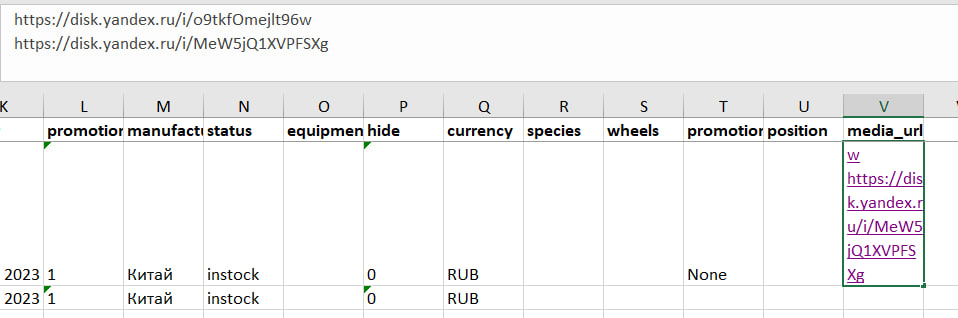


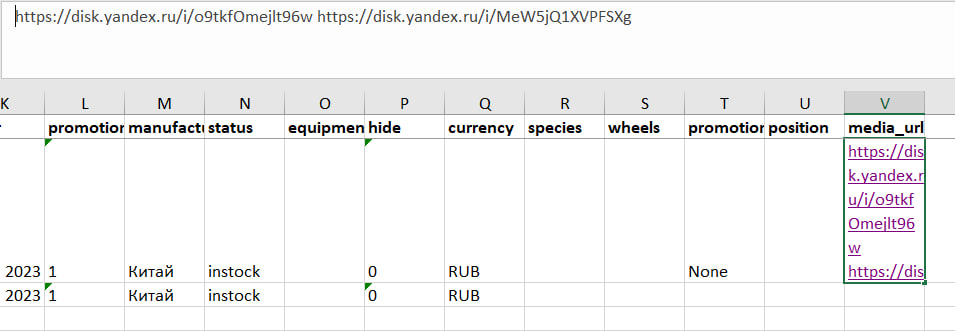
**У товара** можно указать две категории, которым он будет в одинаковой степени принадлежать  
**Фотографию можно** добавить либо через “Browse…” с компьютера, либо указав ссылку на открытую фотографию в поле photo\_url с Яндекс.Диска, например: <https://disk.yandex.ru/i/lpHwdxhRhguNrQ> (в таком случае, поле photo оставляем пустым)

**Для добавления дополнительных фото**, которые будут доступны в карточке товара по кнопке “Показать ещё”**,** нужно загрузить фото (максимум 10) в разделе “Дополнительные фото” в админ-панели, который находится внутри товара:  
  
  
**Если поставить галочку** у поля “Акция”, то товар будет отображаться после нажатия на кнопку “Акции” и не будет в остальных категориях. Так же, если выставлена галочка у “Акции”, нужно добавить “Описание акции”, т.к. обычное “Описание” не относится к акциям.  
Номер позиции в категории – какой по счёту будет товар в карусели (можно оставить пустым, также значения могут повторяться от товара к товару)

Импорт и экспорт:

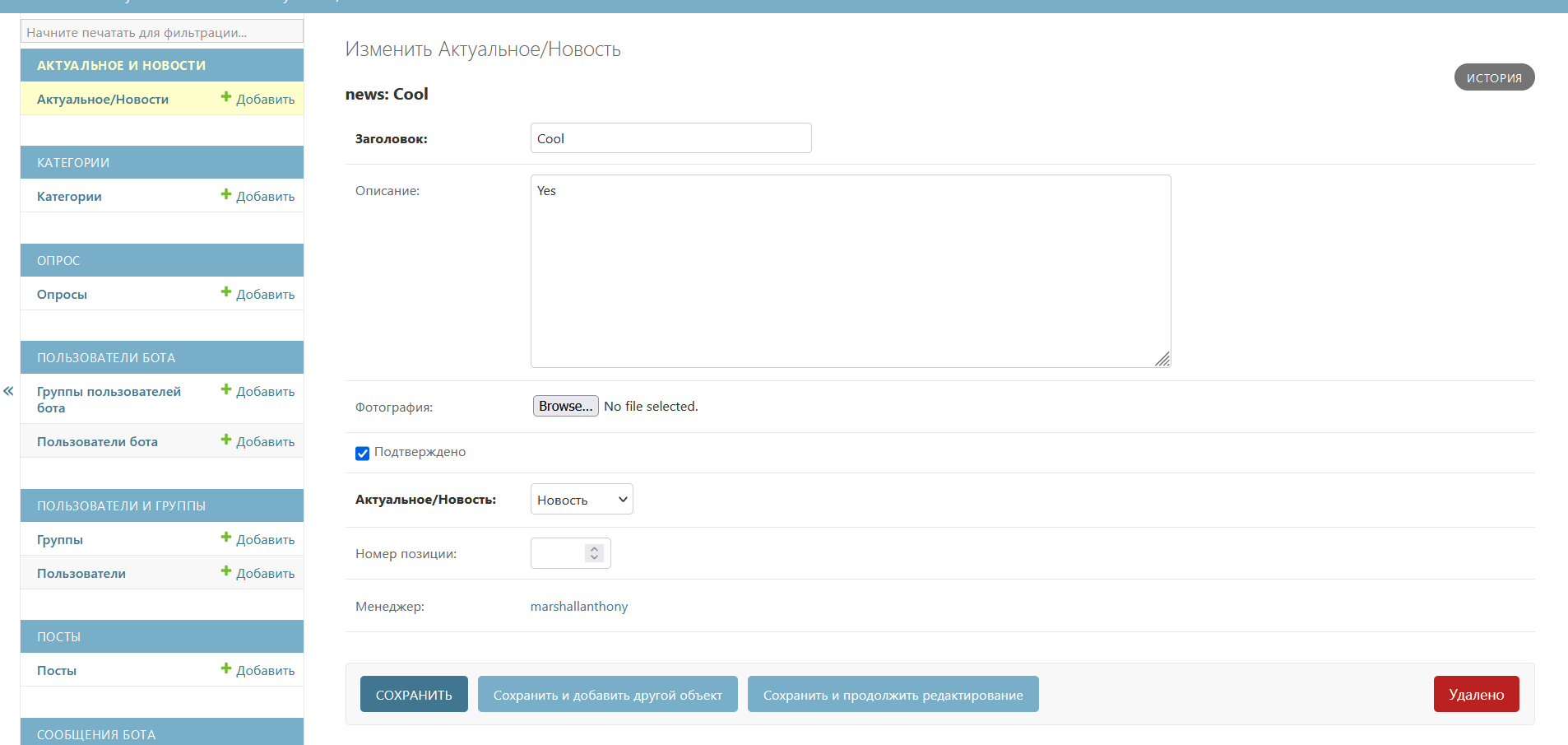


**Экспорт**: для экспорта нужно нажать на кнопку экспорт, выбрать формат и скачать  
**Импорт**: для импорта надо сделать экспорт, чтобы получить пример файла для импорта и изменить его под себя. Вот пример импорт-файла:  
  
**Поле id оставляем пустым, если хотим, чтобы создался новый товар. А если хотим, чтобы изменился определённый товар, то ставим id нужного товара. При подтверждении импорта, изменяемые товары подсвечиваются жёлтым, а создаваемые – зелёным.**  
В плане фотографии, при excel импорте, работает только ссылка на фотографию, хранящуюся на Яндекс.Диске. То есть, вписываем ссылку в поле **photo\_url**  
**Для дополнительных фоток**, нужно заполнить поле **media\_url** (оно обязательно должно быть, даже если оно будет пустым). Если у нас одна фотка, то просто вставляем её, без лишних пробелов. Если же дополнительных фоток для одного товара несколько, то их можно вставлять либо через пробел, либо с новой строки (alt+enter). Обя примера:  




# Актуальное и новости

### Актуальное/новости

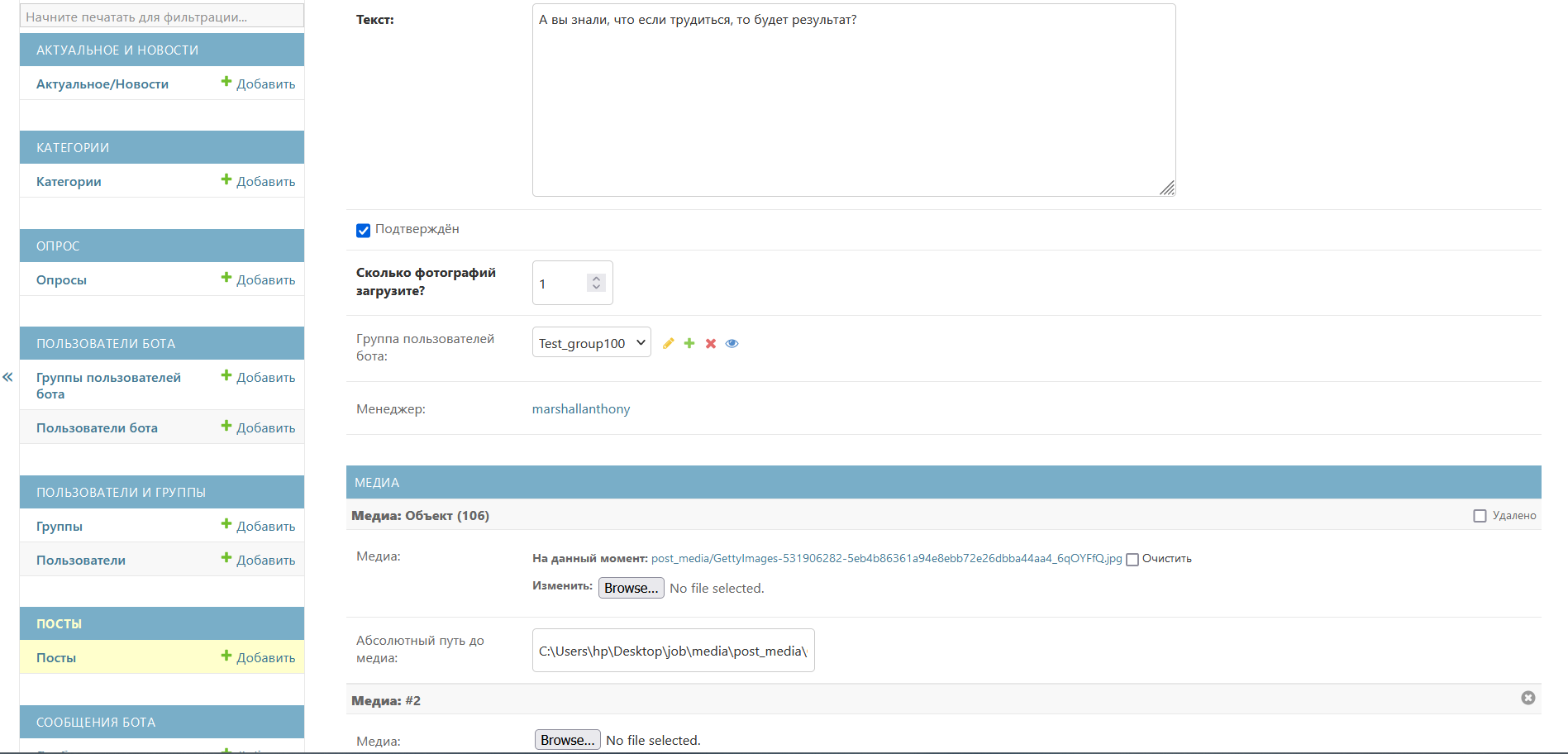


**Актуальное и новость создаются одинаково**. В поле “Актуальное/Новость” нужно указать, создаёте ли вы новость или же актуальное.  
**Номер позиции** – это порядок, в котором будут отображаться новости/актуальное в карусели (можно оставить пустым, также могут быть одни и те же значения у разных новостей/актуальных – 1,2,3 и т.д.)

**После создания новости/актуального, оно отправляется админу в** лс в телеграм, где он должен подтвердить или отклонить.   
Если отклоняет – менеджеру, создавшему эту новость/актуальное, отправляется сообщение об отклонении и он может пойти в админ-панель и как-нибудь исправить эту новость/актуальное, после чего она вновь отправится на подтверждение админу.  
Если админ подтверждает новость/актуальное, то она добавляется в карусель новостей/актуальных и становится доступной для пользователей.

# Посты

### Посты

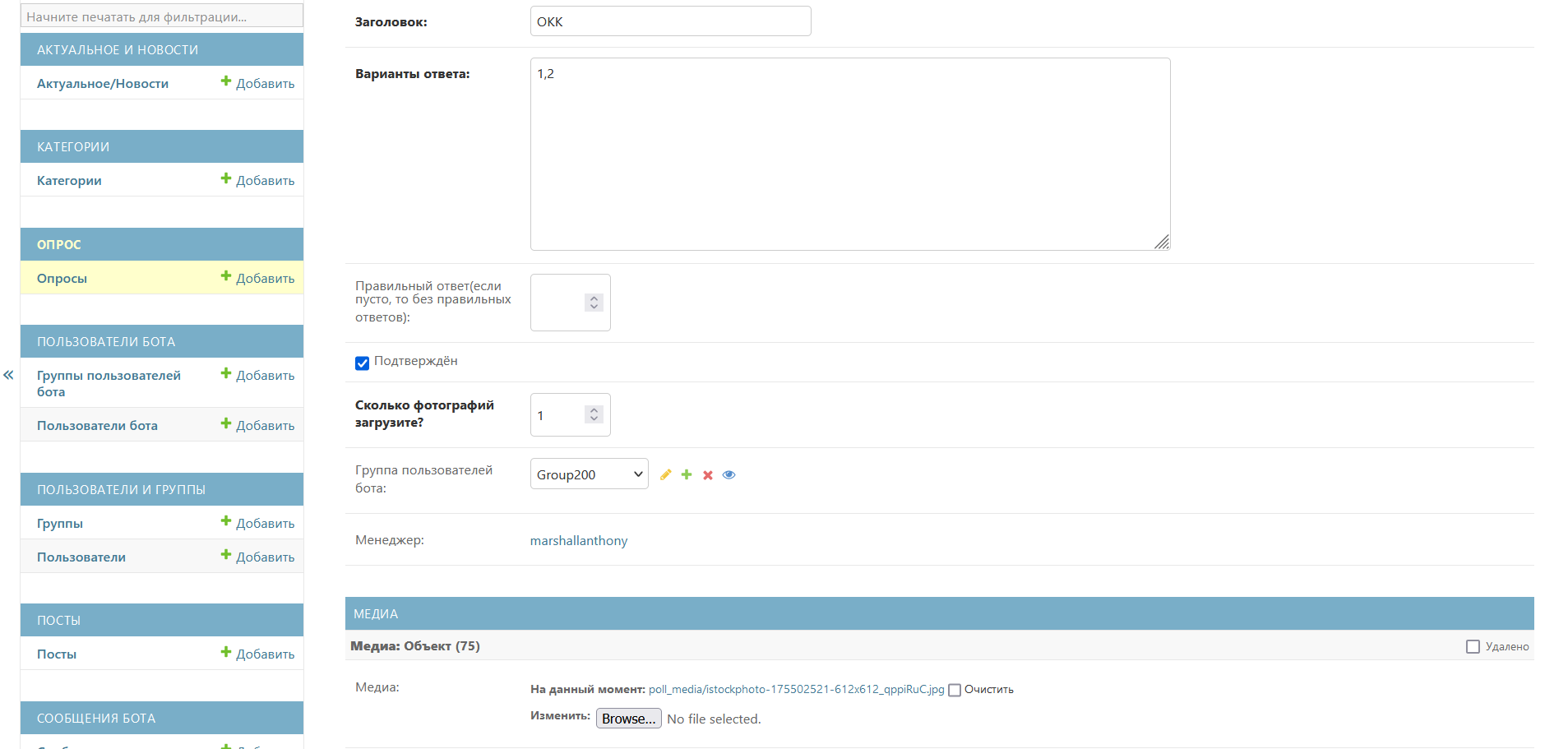


**Текст** – текст поста  
**“Дата и время рассылки”** – можно указать, когда именно будет произведена рассылка. Если оставить пустым, то рассылка произойдёт мгновенно   
**“Подтверждён”** – автоматически заполняется после подтверждения в лс админом  
**“Сколькой фотографий загрузите**” – сколько медиа собираетесь отправить вместе с текстом  
**“Группа пользователей”** – группа, которой вы хотите сделать рассылку. Если пустая, то отправляется всем пользователям бота  
**“Медиа”** – медиа, которое вы хотите отправить вместе с постом, нужно загрузить его, нажав на “Browse…”. “Абсолютный путь до медиа” оставляем пустым, он заполнится сам. Добавляем столько медиа, сколько указали в поле “Сколько фотографий загрузите”.

**Менеджер создаёт пост -> он отправляется на подтверждение админу, который либо отклоняет его, либо подтверждает.**   
**При отклонении поста** админом, менеджеру, создавшему его, приходит сообщение об отклонении, после чего он может пойти в админ-панель и исправить пост. После исправления, пост отправится на подтверждение вновь.  
**В случае подтверждения**, пост отправится всем пользователям бота.

# Опрос

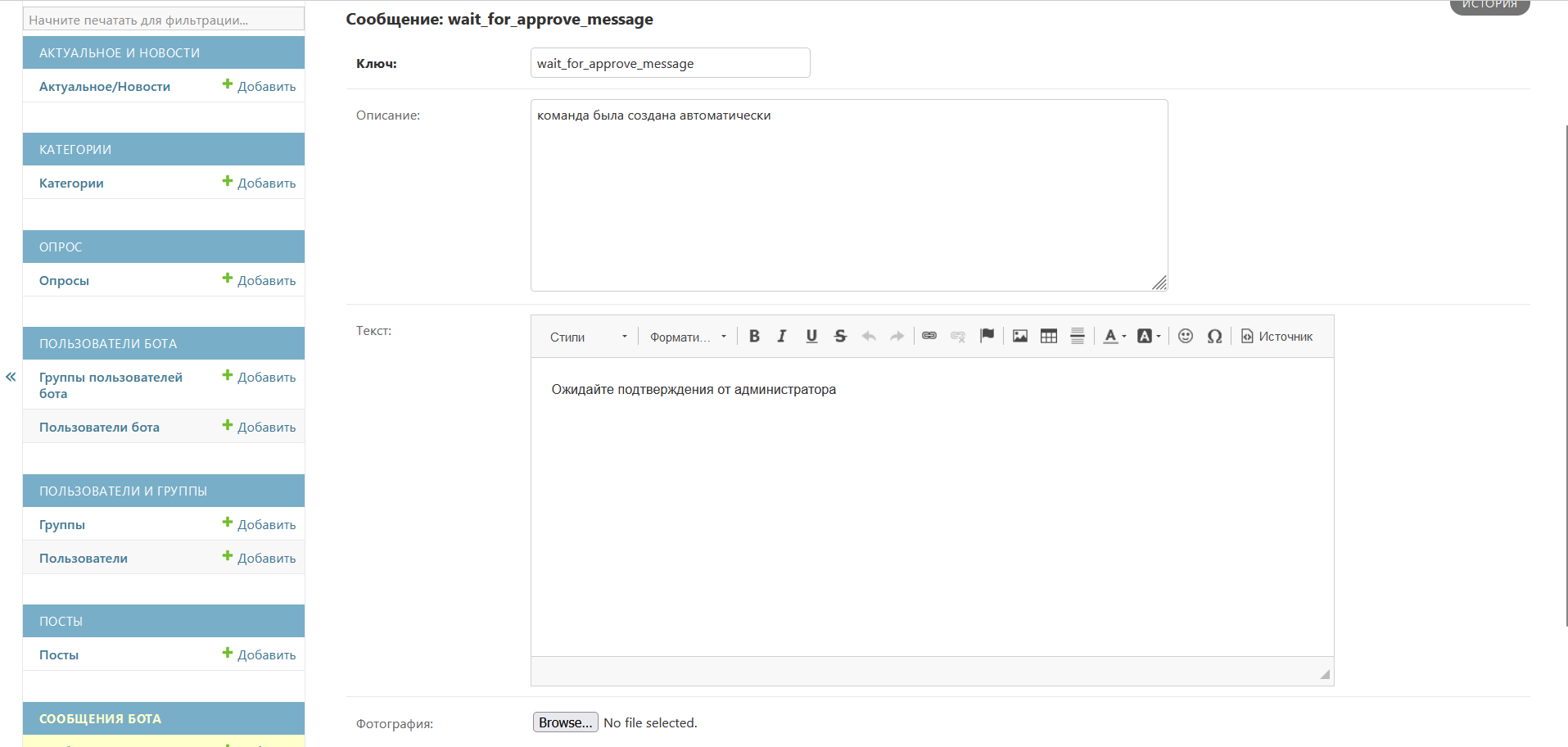
### Опросы

  
**То же самое, что и с постом**. Единственное отличие – это поля “Варианты ответа” и “Правильный ответ”.  
“Варианты ответа” указываем через запятую без пробелов

**“Правильный ответ”** – это то, какой вариант ответа будет правильным (1 – первый, 2 – второй и т.д.). Если оставить пустым, то будет не викторина с правильным ответом, а просто опрос.

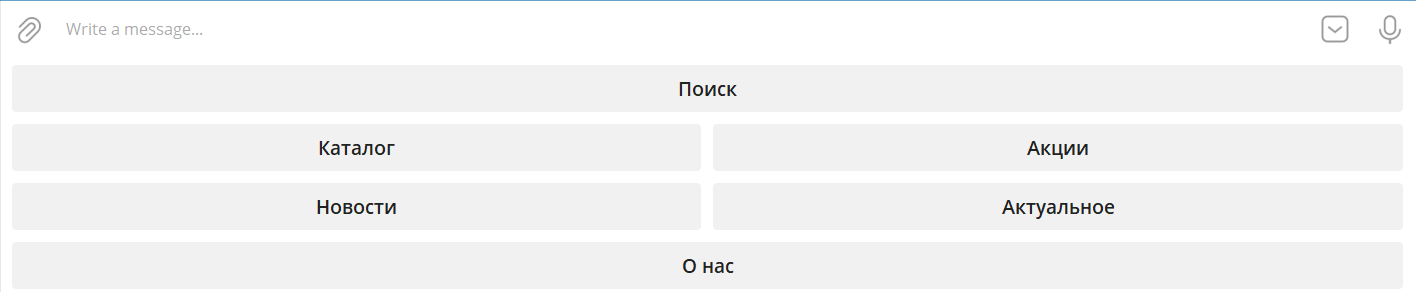
# Сообщения бота

### Сообщения



**Сообщения, которые отправляет бот**. Список фиксированный. При добавлении своих, ничего не изменится.  
Сообщения можно менять (поле “Текст”), но без изменения поля “Ключ”.   
**Встроенные в редактор текста смайлы не работают**, для их использования нужно копировать их с, например, телеграма.  
Если сообщение по какой-то причине удалится, то оно автоматически создастся с описанием, что “команда была создана автоматически”

Поиск



Функция поиска, в отличие от Каталога, позволяет получить товары не только по их категории, но и по нескольким дополнительным фильтрам (страна производителя, вид, колёсная формула, год, марка).  
После нажатия на поиск выбираем нужную категорию (как в Каталоге), далее нажимаем на нужные фильтры. Варианты каждого фильтра создаются на основе товаров выбранной категории. Если у какого-то фильтра не заполнено поле для фильтра, то появляется кнопка “Пропустить”. При выборе “Пропустить”, в конечном итоге, вернутся товары, у которых не заполнено это поле.   
После успешной фильтрации, вернётся карусель из товаров, отвечающих заданной фильтрации.

Команда /city

Позволяет задать/изменить свой город пользователю бота. Город устанавливается либо через просьбу поделиться геопозицией при регистрации, от которой можно отказаться, либо через команду /city

# Статистика бота

Статистика находится по адресам:

[/api/statistics/](http://localhost:8000/api/statistics/) - **общая статистика**, можно указать месяц, за который нужно посмотреть эту статистику и скачать excel-файл с этой статистикой за указанный месяц

/api/manager\_results/ - **статистика по менеджерам**. Можно выбрать одного/нескольких менеджеров и увидеть, сколько постов, актуального/новостей и т.д. сделал тот или иной менеджер.

# Первоначальная настройка бота.

**Виртуальное окружение и установка зависимостей:**

* Поднять виртуальное окружение
* Загрузить зависимости из requirements.txt

**Создание суперпользователя:**

* python manage.py createsuperuser

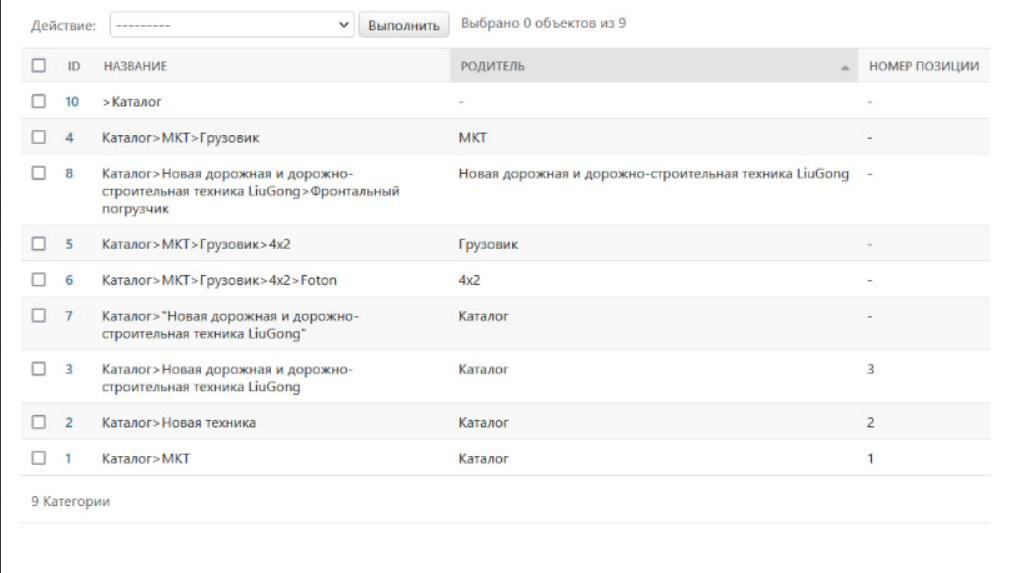
**Создаём файл .env** (он должен быть на одном уровне с папками приложений, manage.py и т.д.). Шаблон такого файла можно найти, перейдя в .env\_template, где всё, что нужно – это заполнить пустые строки переменных своими значениями. Вот переменные, которые нужно заполнить:

* Токен бота в переменную TOKEN = ‘’
* Телеграм-id администратора, которому будут приходить вещи на подтверждение, в admin\_id = ‘’
* Телеграм администратора, которому будут приходить запросы на КП и Чат с менеджером, в случае отсутствия менеджера у категории, в admin\_tg = ‘’
* Домен (к примеру, <http://localhost:8000/>) на котором будет располагаться сайт в domen = ‘’
* Юзернейм crера (python manage.py createsuperuser), с помощью которого можно зайти в админ-панель, добавляем в admin\_username = ‘’
* Пароль суперюзера из admin\_username в admin\_password = ‘’
* DB\_NAME = ‘’ – имя базы данных PostgreSQL
* DB\_USER\_NAME = ‘’ – имя пользователя PostgreSQL
* DB\_USER\_PASSWORD = ‘’ – пароль пользователя PostgreSQL

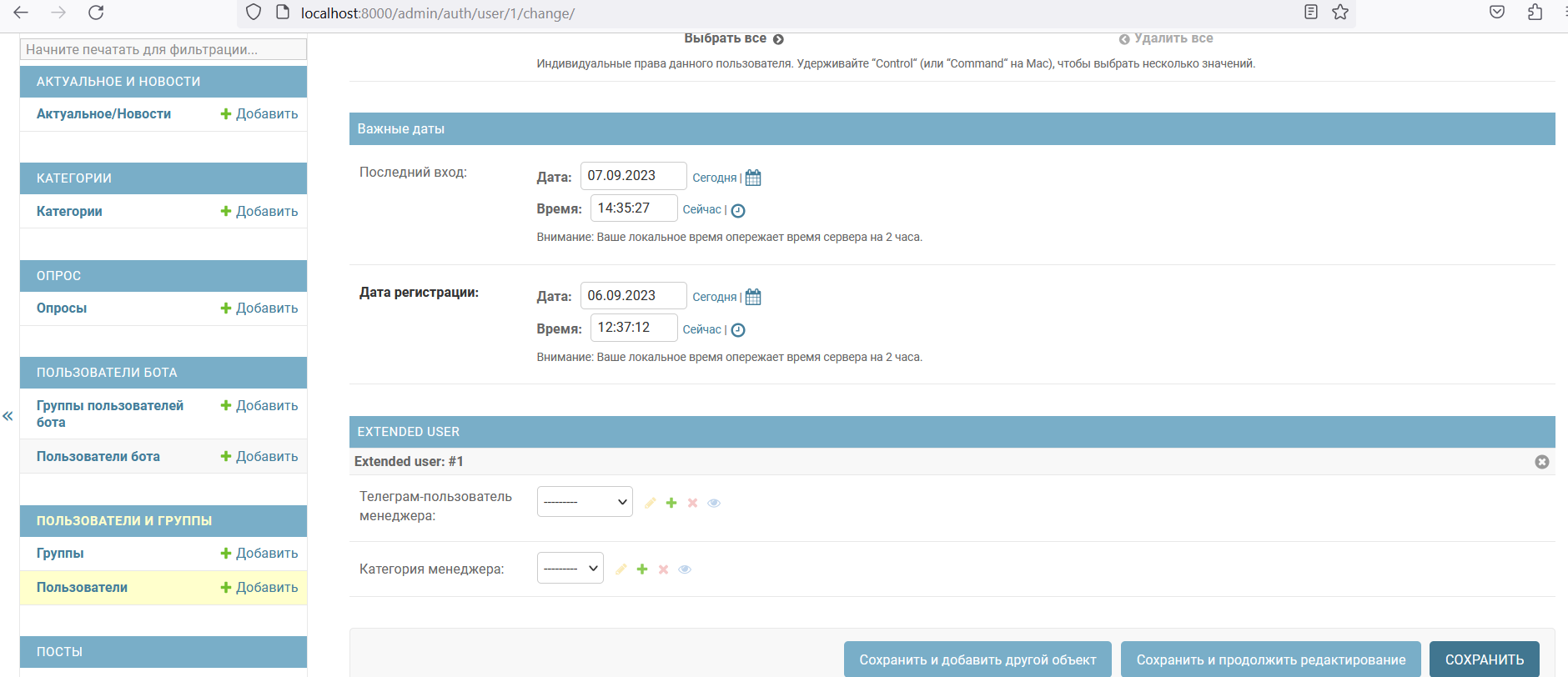
**Миграции:**

* python manage.py migrate

**Для работы каталога:**

****

Нужна одна корневая категория с именем "Каталог", от которой уже будут наследоваться следующие

**Для последующей работы с админ-панелью нужно** создать админа через /start. Т.е. createsuperuser нужен только для того, что описывалось выше и для того, чтобы дать статус superuser реальному админу, который будет зарегистрирован через /start и сделан менеджером (это нужно, потому что джанговский superuser никак не связан с телеграмом, а это необходимо). Ну либо можно связать джанговского superuser в админке с пользователем, созданным через /start вручную (выставить поле “Телеграм-пользователь менеджера”): ****

Т.е. либо делаем менеджера админом, с помощью суперюзера, либо определяем поле “Телеграм-пользователь менеджера” у самого суперюзера и не забываем поставить галочку у “Статус суперпользователя”