

## Инструкция по эксплуатации

### Установка бота

На машине должны быть установлены следующие пакеты:

- 1) Java 17 и выше;
- 2) СУБД PostgreSQL.

### 1.2. Конфигурация application.properties

Открываем проект переходим по пути “src/main/resources” находим файл “application.properties” и открываем его, там будет что по типу:

```
spring.application.name=promo

spring.datasource.driver-class-name=org.postgresql.Driver
spring.datasource.username=USERDB
spring.datasource.password=PASSDB
spring.datasource.url=jdbc:postgresql://SERVER:PORT/DATABASE
spring.jpa.hibernate.ddl-auto=update

vk.token=VK-ACCESS-TOKEN
vk.group=VK-GROUP-ID
vk.admin=VK-ADMIN-IDS
vk.manager=VK-MANAGER-ID
vk.confirm.code=CONFIRMATION-CODE
vk.secret.key=SECRET-KEY
```

USERDB – Логин пользователя СУБД

PASSDB – Пароль пользователя СУБД

SERVER – IP-адрес СУБД (если на этой же машине, то localhost)

PORT – Порт СУБД (по умолчанию 5432)

DATABASE – Название базы данных

VK-ACCESS-TOKEN – Токен доступа для бота (Скриншот 5)

VK-GROUP-ID – Id сообщества ВК (Скриншот 6)

VK-ADMIN-IDS – Id или теги админов бота (через пробел)

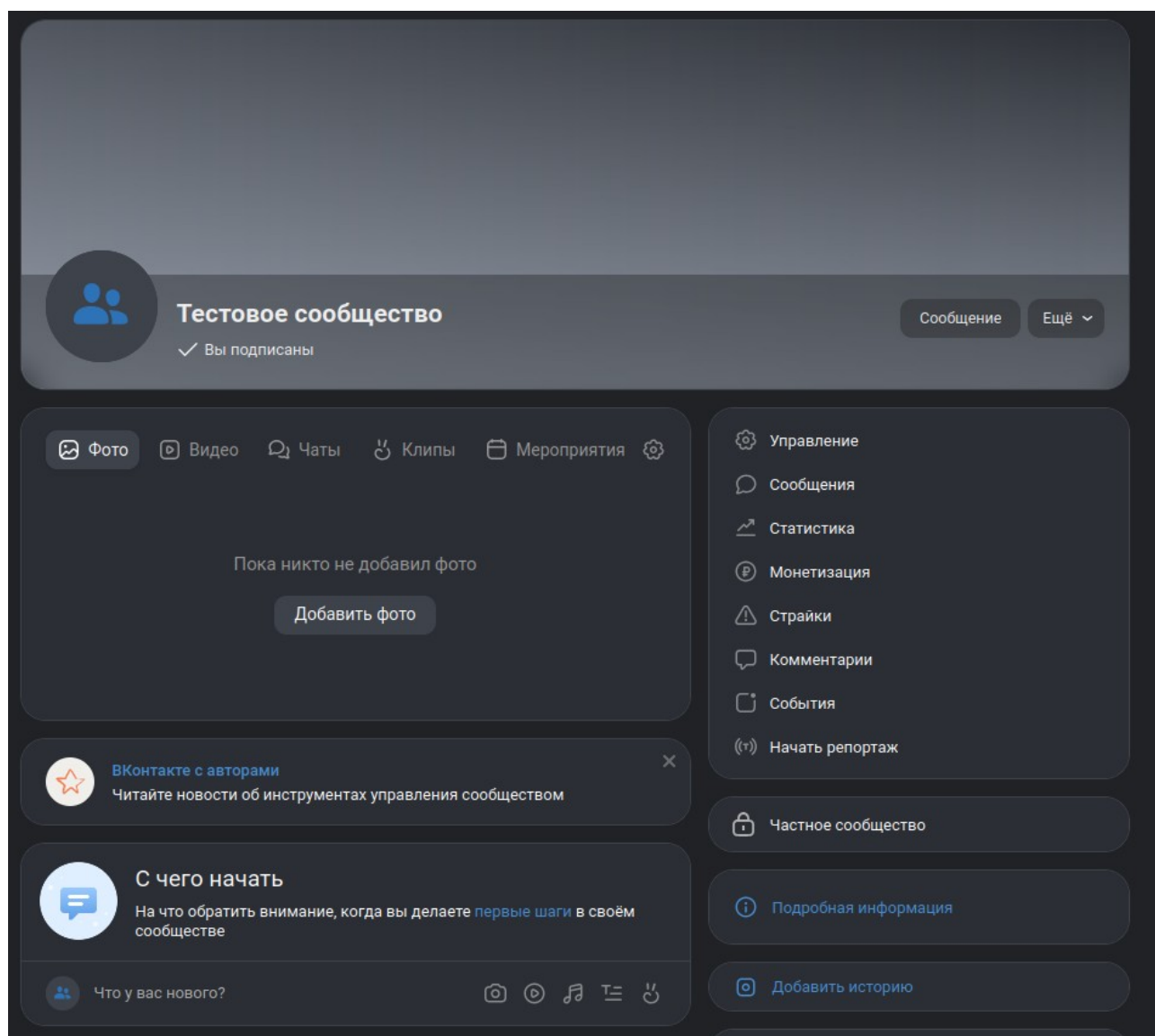
VK\_MANAGER-ID – Id или тег менеджера (1 ID без пробелов)

CONFIRMATION-CODE – Строка, которую должен вернуть сервер

(Скриншот 6)

SECRET-KEY – Секретный ключ (Скриншот 6)

### 1.2.1 Настройка бота в сообществе ВК



Скриншот 1. Открываем сообщество ВК и во вкладку “Управление”

Основная информация

Название:

Описание:

Тип сообщества: Частное

Обложка: [Добавить](#)

Адрес: 

Как правильно выбрать адрес и можно ли использовать уже занятый, читайте [здесь](#).  
Вы можете создать наклейки для вашего сообщества, добавив странице короткий адрес.  
[Создать наклейки для других сообществ](#)

Верификация: [Подать заявку](#)

Сохранить

Кнопка действия

Кнопка действия: [Выключена](#)

Кнопка вверху вашего сообщества помогает людям выполнить действие. Кнопку можно отредактировать в любой момент.

Сохранить

Тестовое сообщество

вернуться к странице

Настройки

Разделы

Комментарии

Ссылки

Адреса

Меню

Работа с API

Истории

Подписчики

Сообщения

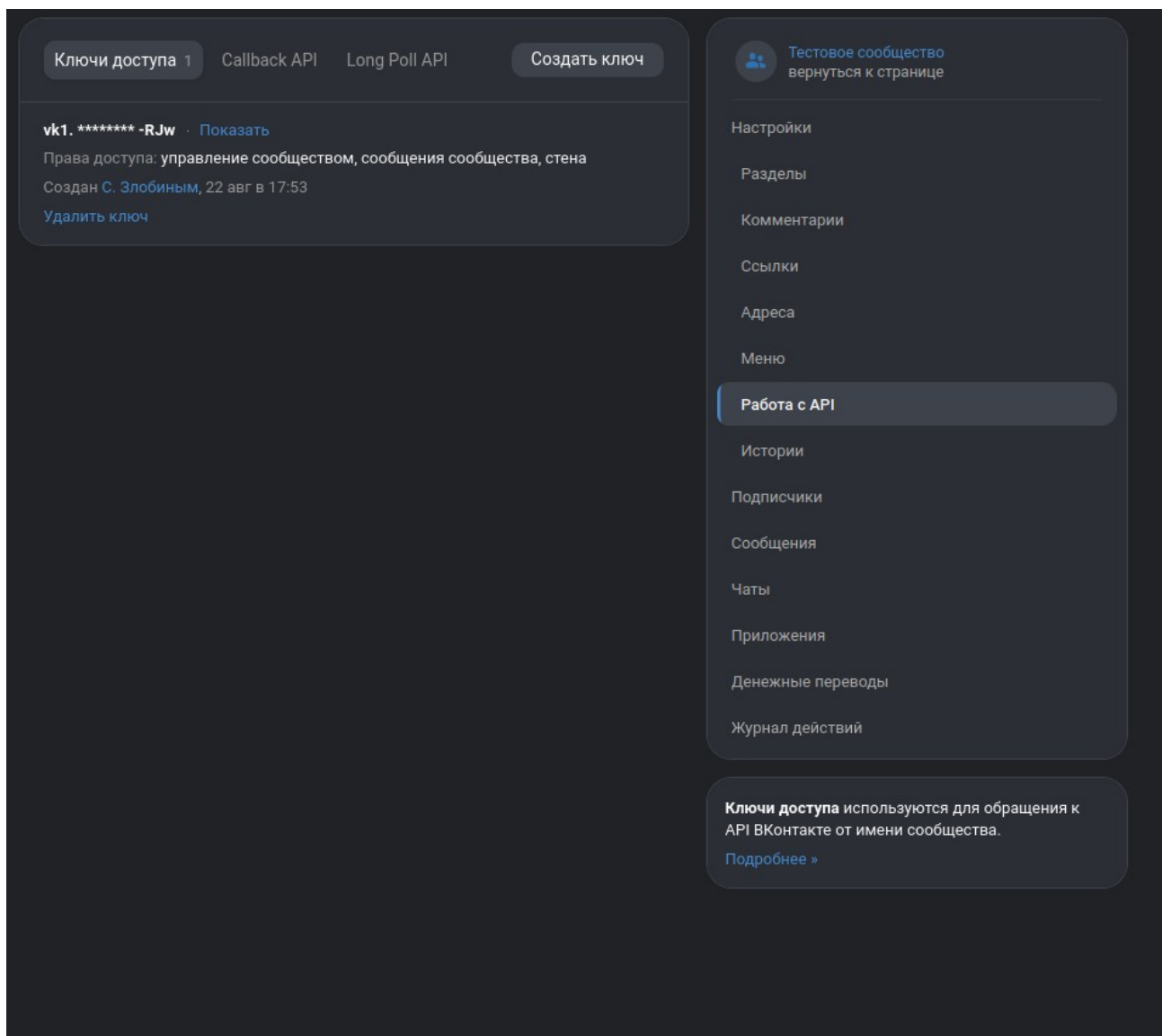
Чаты

Приложения

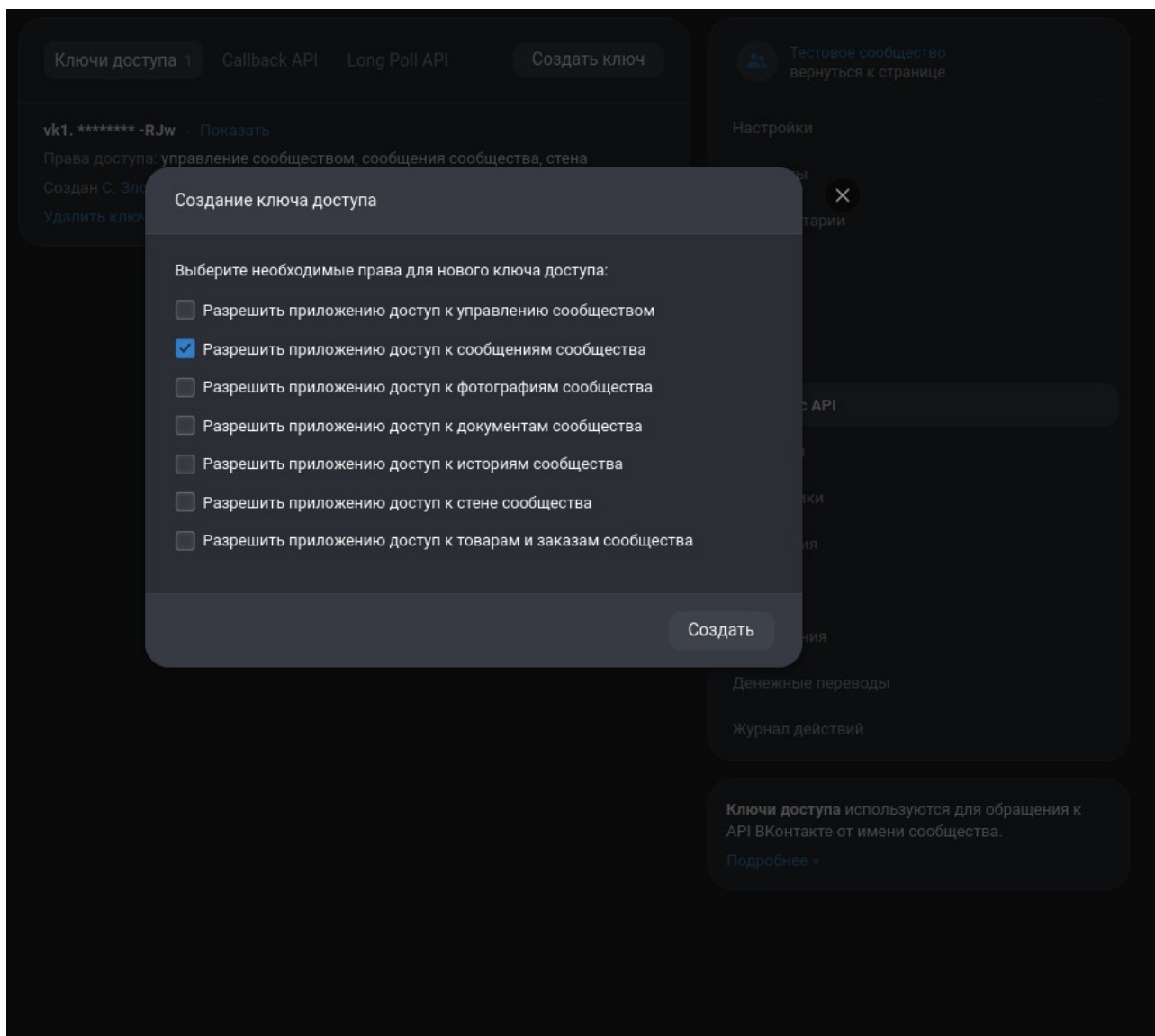
Денежные переводы

Журнал действий

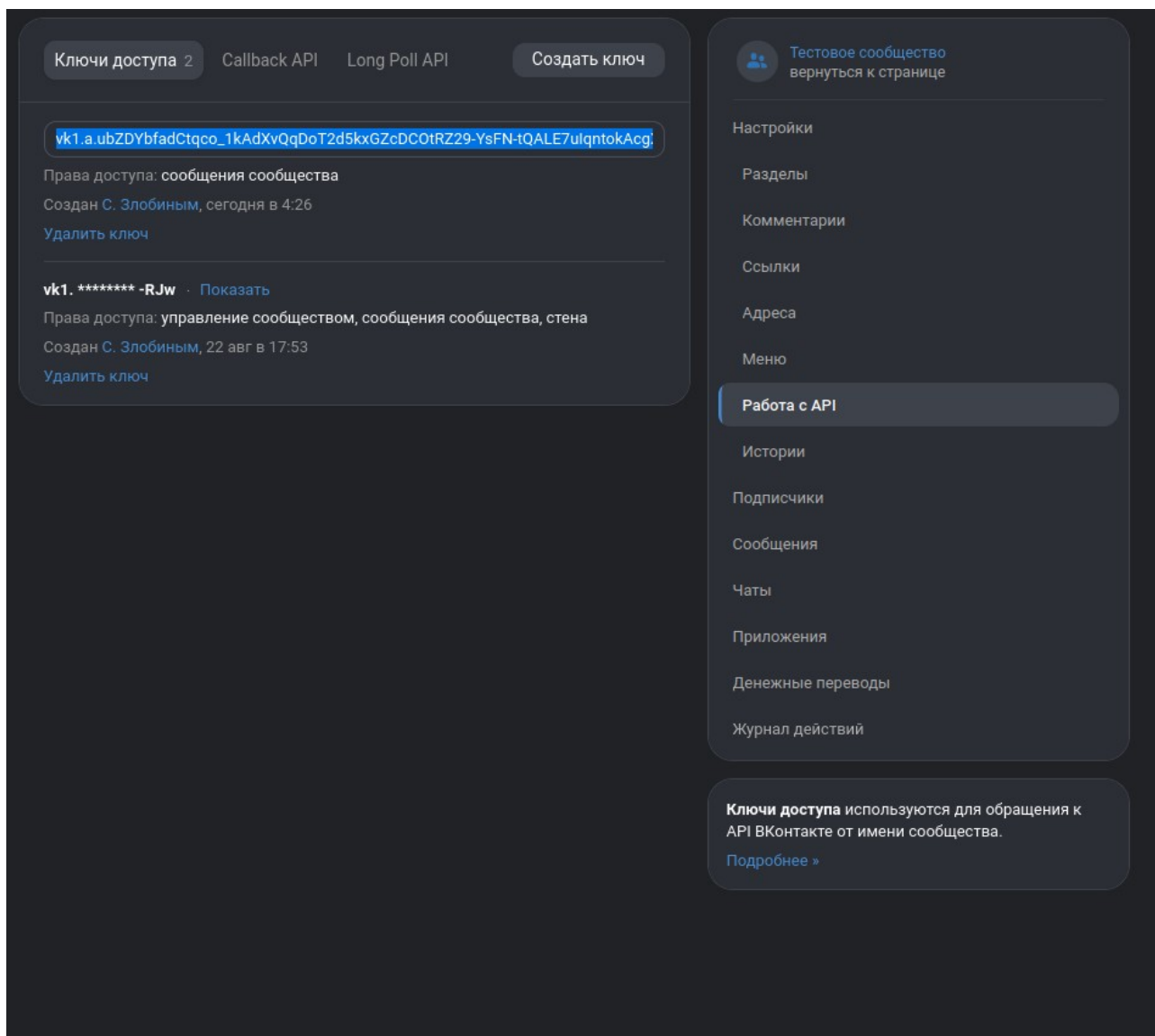
Скриншот 2. Переходим во вкладку Работа с API



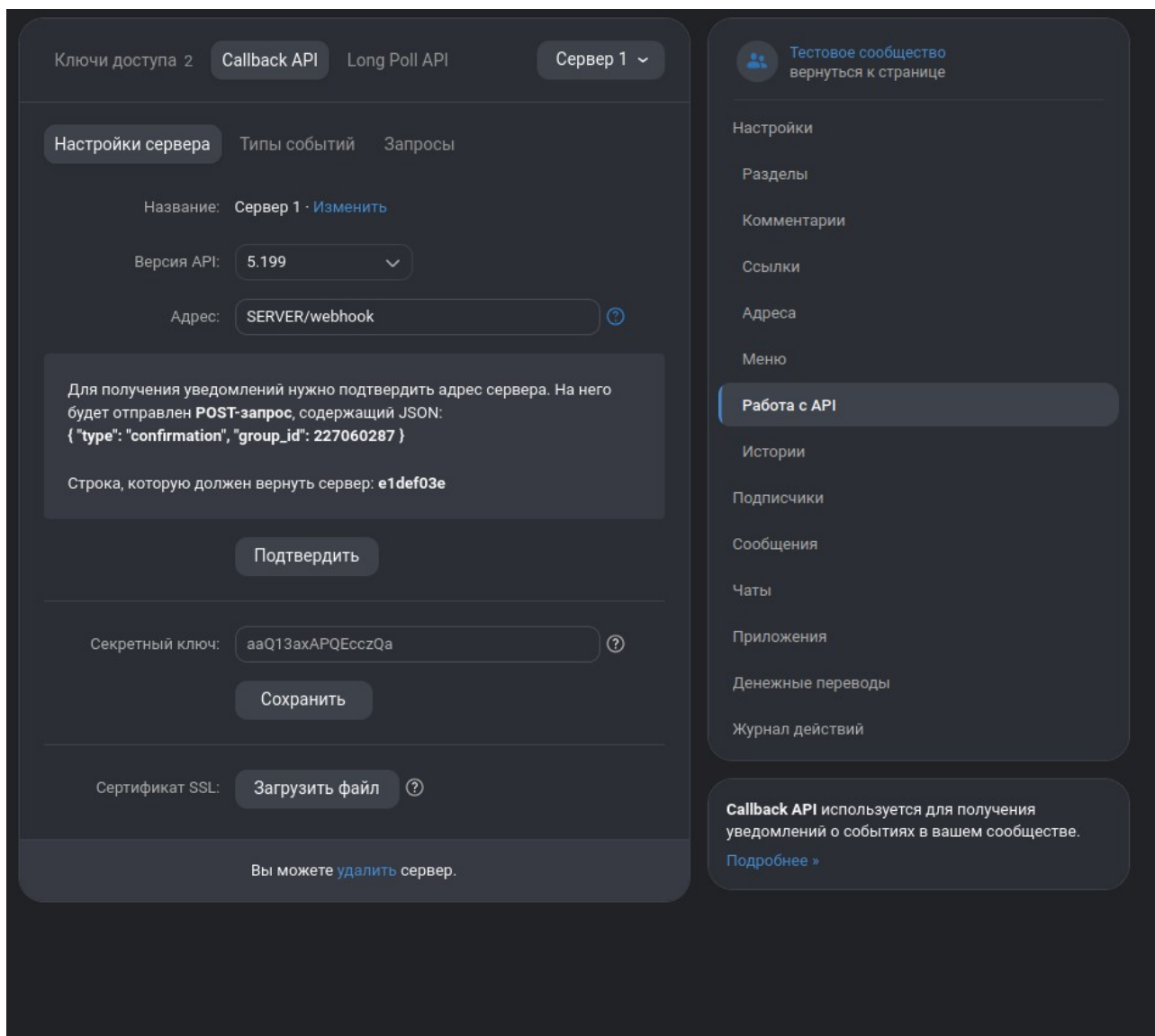
Скриншот 3. Создаём ключ для бота



Скриншот 4. Так как этот чат-бот, то даем доступ только к сообщениям



Скриншот 5. У нас появится новый ключ и копируем его в application.properties  
И переходим во вкладку Callback API



Скриншот 6. Здесь в application.properties копируем:

GROUP-ID

CONFIRMATION-CODE (Строка, которую должен вернуть сервер)

SECRET-KEY (Секретный ключ, его лучше сгенерировать до 50 латинских букв и цифр)

В поле Адрес вместо SERVER пишем белый IP-адрес сервера и порт 8080, либо же домен сервера.

После того как будет заполнен application.properties, открываем терминал в корне нашего проекта и прописываем команду

```
java -jar target/promo-0.0.1-SNAPSHOT.jar
```

Ждем когда проект запуститься, и переходим на окно как на скриншоте 6 и нажимаем кнопку Подтвердить. После этого на этом окне должно быть написано, что Адрес сохранен.

Все бот запущен и работает в сообществе.