

UnicChat

Настройка чатбота

02.03.2024

Разработка бота UnicChat SDK

Этот бот - простой пример того, как можно использовать методы UnicChat JS SDK. Бот является базовым и не обрабатывает ошибки, различные типы сообщений, перезагрузки сервера и другие ситуации возникающие в процессе работы.

Краткое руководство по запуску UnicChat SDK Bot

UnicChat Bot - это пакет для Node.js который работает с Node и npm.

Установка пакета UnicChat SDK

Откройте терминал в директории, в которой вы хотите разработать своего бота, и выполните следующую команду

```
npm init -y
```

```
npm install @unicchat/sdk
```

Создание файлов бота

Для запуска самого простого бота необходимо создать два файла: первый отвечает за работу сервера бота, а второй – за ответы бота на сообщения пользователей.

- создайте файл server.js со следующим содержимым:

```
const { driver } = require('@unicchat/sdk');
const respmap = require('./reply');

//параметры окружения
const HOST = '<UNICCHAT HOST>';
const USER = '<BOT USER NAME>';
const PASS = '<BOT USER PASS>';
const BOTNAME = '<UNIC CHAT BOT ALIAS>';
const SSL = '<SSL USAGE>';
const ROOMS = ['<UNICCHAT CHANNEL>'];
var myUserId;

const runbot = async () => {
  const conn = await driver.connect({ host: HOST, useSsl: SSL })
  myUserId = await driver.login({ username: USER, password: PASS });
  const roomsJoined = await driver.joinRooms( ROOMS );
  console.log('joined rooms');

  const subscribed = await driver.subscribeToMessages();
  console.log('subscribed');
```

```

const msgloop = await driver.reactToMessages( processMessages );
console.log('connected and waiting for messages');

const sent = await driver.sendToRoom( BOTNAME + ' is listening ...', ROOMS[0]);
console.log('Greeting message sent');
}

const processMessages = async(err, message, messageOptions) => {
if (!err) {
  if (message.u._id === myUserId) return;
  const roomname = await driver.getRoomName(message.rid);

  console.log('got message ' + message.msg)
  var response;
  if (message.msg in respmap) {
    response = respmap[message.msg];
  }
  const sentmsg = await driver.sendToRoomId(response, message.rid)
}
}

runbot()

```

Настройте содержимое раздела «параметры окружения» в соответствии с вашим сервером и учетными данными пользователей.

Убедитесь, что <BOT USER NAME> имеет роль BOT на сервере.

- создайте файл reply.js со следующим содержимым:

```

const respmap = {
  "Привет" : "привет",
  "Ты бот?" : "Да, я бот
};

module.exports = respmap;

```

Запуск бота

Запустите следующую команду в терминале:

```
node server.js
```

После выполнения последней команды бот UnicChat SDK попытается подключиться к экземпляру UnicChat и прослушать сообщения в комнате “general”.

Взаимодействие с ботом

Войдите в чат как обычный пользователь (не пользователь типа BOT), перейдите в общую комнату и поговорите с только что созданным ботом.