

Актуальные
зарплаты в IT

110k

100k

150k

120k

180k

**empenoso**

10 фев в 03:25

Как бесплатно разослать всем членам ТСЖ персональное сообщение о завтрашней уборке снега через WhatsApp

Простой

6 мин

3.2K

Open source*, Node.JS*, Инженерные системы*

Кейс

Уборка снега требует своевременного оповещения жителей о необходимости убрать автомобили. В противном случае возникают сложности с «подснежниками» – машинами, которые простаивают всю зиму, а также с жильцами которые игнорируют объявления на подъездах. Обзвон всех жильцов отнимает много времени.



Пример двора

WhatsApp выглядит привлекательным способом оповещения. К примеру, в нашем ТСЖ недавно создали WhatsApp-группу и предполагается что WhatsApp есть у всех жителей. А ещё бывают люди, подключающие домовый чат в качестве эпизодического развлечения: отключают для него уведомления и просто копят непрочитанные.

Хотя скрипт полностью рабочий и протестированный, он ещё не был опробован в реальной жизни при рассылке сообщений - когда я подошёл с таким предложением к нашему председателю, то он не особо впечатлился, предпочтя более привычный обзвон.

Установка и настройка

Чтобы настроить автоматическую рассылку сообщений, нужно подготовить рабочую среду. Это не займет много времени, даже если вы раньше не занимались программированием.

Первым шагом является установка Node.js - инструмента, который позволяет запускать JavaScript-код на компьютере. Без него скрипт попросту не заработает. Перейдите на [официальный сайт Node.js](https://nodejs.org/), скачайте рекомендованную версию (LTS) и установите её, следуя подсказкам на экране. Убедиться, что всё установлено правильно, можно, открыв терминал и введя команду `node -v`. Она покажет текущую версию Node.js.



whatsapp-web.js

Клиентская библиотека WhatsApp для NodeJS,
которая подключается через браузерное
приложение WhatsApp Web

[Руководство →](#)[Документация ↗](#)

Автоматизация для всех

Программно управляйте WhatsApp независимо от того, используете ли вы пользовательские или бизнес-аккаунты.

Создано на базе WhatsApp Web

Whatsapp-web.js подключается к официальной версии WhatsApp Web под капотом, снижая риски блокировки.

Простота в использовании

Объектно-ориентированный подход позволяет легко и быстро приступить к работе.

Лицензия Apache версии 2.0.

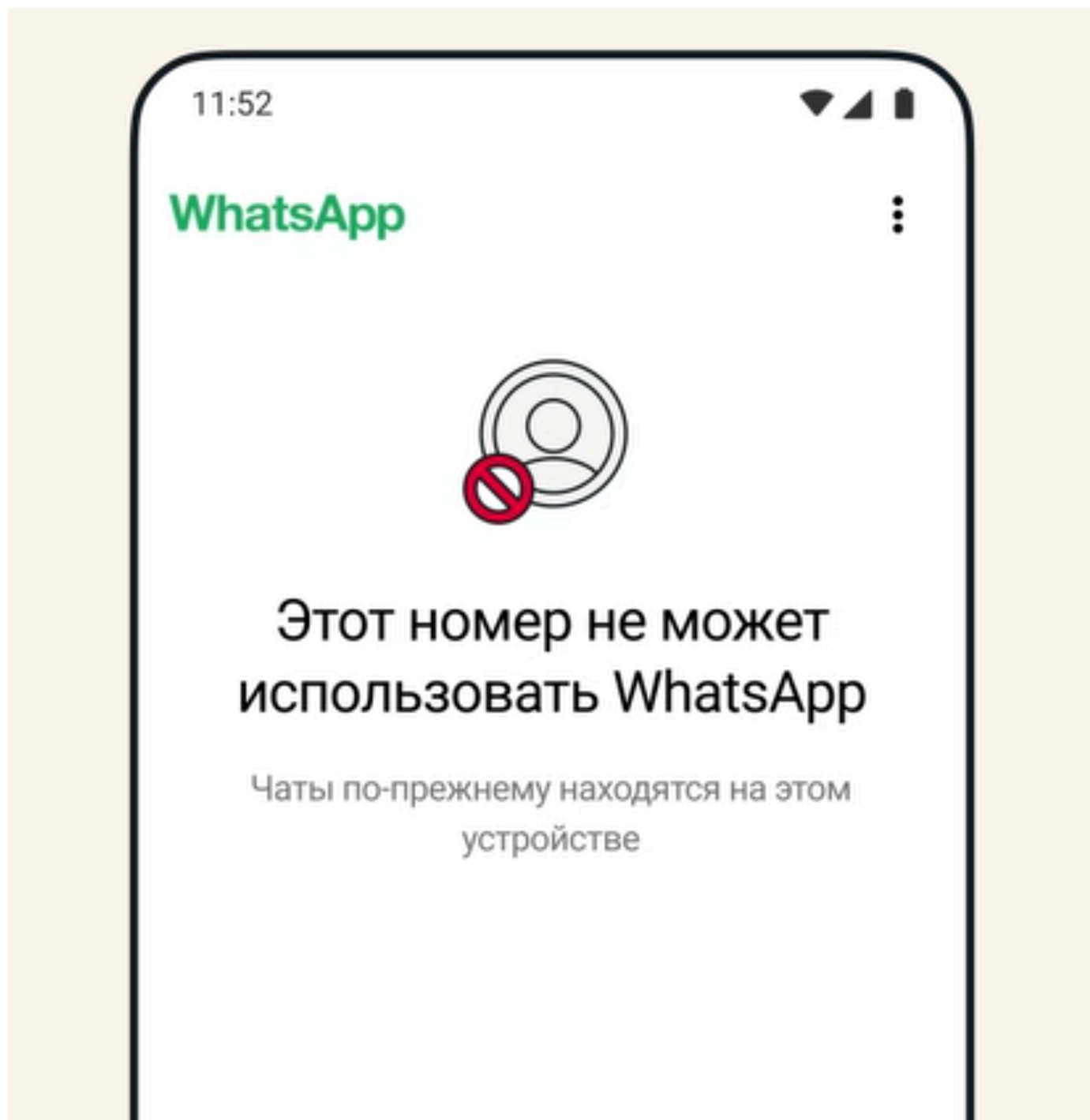
Следующий важный элемент — библиотека **whatsapp-web.js**. Она обеспечивает взаимодействие с веб-версией WhatsApp, позволяя автоматически отправлять сообщения. Это основной инструмент, который делает весь процесс рассылки возможным.

Помимо этого, мы будем использовать библиотеку **qrcode-terminal**, чтобы вывести QR-код прямо в консоли. Он потребуется для подключения вашего аккаунта WhatsApp к скрипту. Это быстрое и удобное решение, чтобы не прибегать к сложным настройкам.

Для обработки списка номеров жильцов пригодится библиотека **xlsx**. Она позволяет читать и работать с Excel-файлами, из которых скрипт будет брать контакты для рассылки. Это значительно упрощает подготовку данных и исключает необходимость вручную переносить номера.

Соблюдение лимитов для предотвращения бана аккаунта

Автоматическая рассылка сообщений через WhatsApp — отличное решение, но важно помнить о существующих ограничениях, чтобы избежать блокировки аккаунта. WhatsApp внимательно отслеживает подозрительную активность, и [массовая рассылка может быть воспринята как нарушение правил](#).



Ключевое ограничение — суточный лимит на количество отправленных сообщений. С одного аккаунта нельзя отправить больше 1000 сообщений в день (**у меня нет**

официальной информации на этот счёт). Но для ТСЖ это не должно стать проблемой, ведь даже при большом количестве жильцов обычно достаточно 200-300 сообщений.

Еще один важный момент - скорость отправки. Если отправлять сообщения слишком быстро, система может заподозрить неестественное поведение. Рекомендуется не превышать 5-15 сообщений в минуту. Для этого в скрипте можно встроить задержки между отправками, чтобы рассылка выглядела максимально естественно.

Также стоит позаботиться об уникальности текста сообщений. Если отправлять однотипные фразы, это может вызвать подозрения у алгоритмов WhatsApp. Добавление персонализации - например, имени, отчества или информации о марке машины — делает сообщения индивидуальными.

Простое сообщение вроде « Уважаемый(ая) `{person.Name}`,

завтра будет проводиться уборка снега с 9:00 до 15:00. Просим вас заранее убрать автомобиль `{person.Car || ' '}` с парковки, чтобы работа прошла быстро и качественно.

Спасибо за понимание! » выглядит естественнее и снижает риск блокировки.

Соблюдение этих правил поможет использовать WhatsApp для рассылок без риска потери доступа к аккаунту.

Сам скрипт рассылки всем членам ТСЖ сообщений о завтрашней уборке снега через WhatsApp

Для организации автоматической рассылки сообщений через WhatsApp потребуется выполнить несколько шагов. Первым делом нужно установить необходимые зависимости. Используйте команду `npm install whatsapp-web.js xlsx qrcode-terminal`. Она загрузит библиотеку для работы с WhatsApp, модуль для работы с Excel-файлами и генератор QR-кодов.

Затем нужно подготовить список контактов в формате Excel. Файл должен включать три обязательные колонки: `Name` (имя), `Phone number` (номер телефона) и `Car` (марка автомобиля). Убедитесь, что данные заполнены корректно, так как скрипт будет использовать их для отправки сообщений.

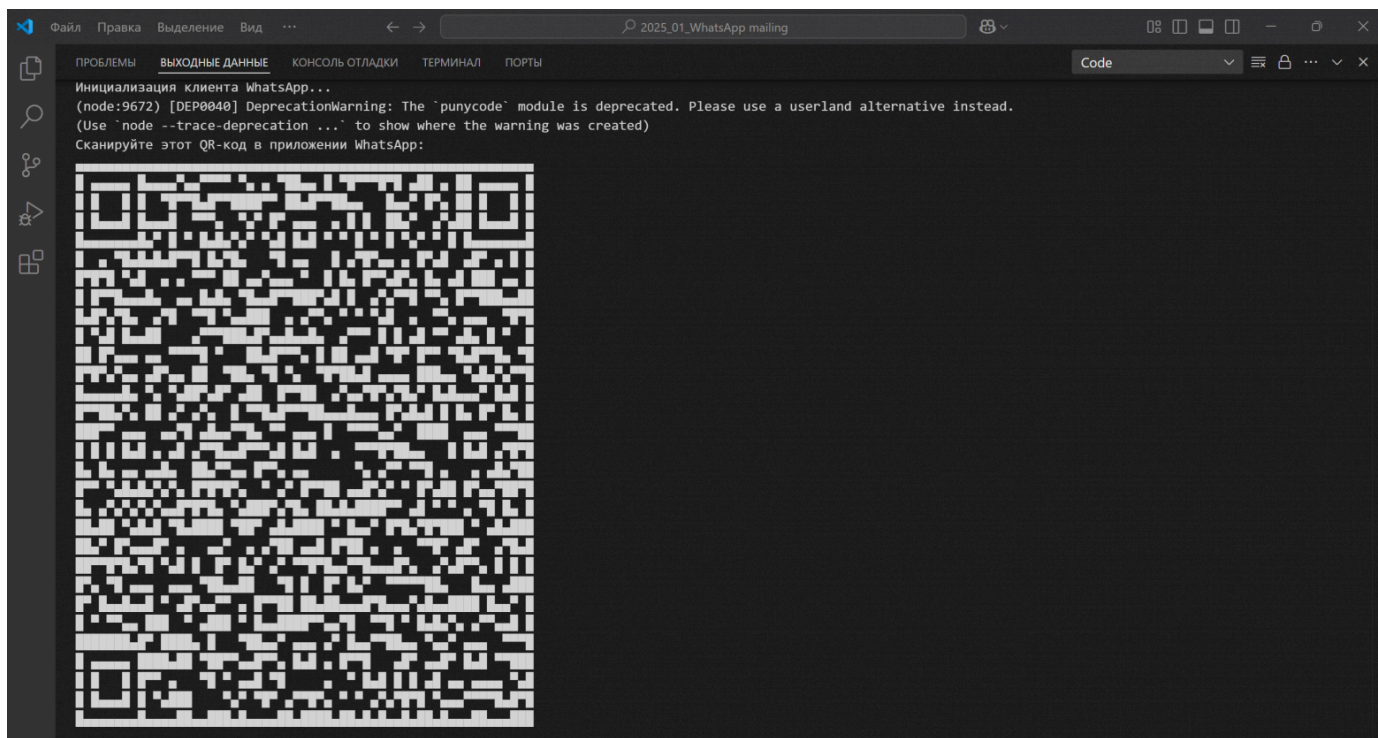
Следующий этап — редактирование шаблона сообщений. Код уже включает базовый текст, например:

Уважаемый(ая) Иван Иванович,

Завтра будет проводиться уборка снега с 9:00 до 15:00.

Просим вас заранее убрать автомобиль Toyota Camry с парковки, чтобы работа прошла быстро.
Спасибо за понимание!

Подключение WhatsApp осуществляется через QR-код. После запуска скрипта на экране появится QR-код, который необходимо отсканировать через приложение WhatsApp. Это позволит синхронизировать аккаунт со скриптом. Важно учитывать, что для рассылки лучше использовать аккаунт, который не жалко потерять, чтобы избежать рисков, связанных с возможной блокировкой.



НЕ надо сканировать, это просто скриншот

После настройки можно запускать программу, и она автоматически отправит всем жильцам персонализированные уведомления с учетом задержек между отправками. Это обеспечит соблюдение лимитов WhatsApp и сделает процесс оповещения быстрым и удобным.

```
/**
 * Система автоматических уведомлений через WhatsApp для ТСЖ и УК
 *
 * Данный скрипт загружает список жильцов из Excel-файла и отправляет им сообщения чере
 * Используется библиотека whatsapp-web.js для взаимодействия с WhatsApp Web.
```

```
*
* Подготовка окружения *
* 1. Установите Node.js:
*   - Скачайте LTS версию с официального сайта nodejs.org
*   - Проверьте установку командой: `node -v`
*
* 2. Установите необходимые библиотеки:
*   npm install whatsapp-web.js xlsx qrcode-terminal
*
* Подготовка данных *
* Создайте Excel-файл `TSZH.xlsx` со следующими колонками:
* - Name (Имя жильца)
* - Phone number (Номер телефона)
* - Car (Марка автомобиля, опционально)
*
* Подробнее о скрипте: https://habr.com/ru/articles/876216/
*
*/

// Импорт необходимых библиотек
const { Client, LocalAuth } = require('whatsapp-web.js');
const qrcode = require('qrcode-terminal');
const xlsx = require('xlsx');
const path = require('path');

// Путь к Excel-файлу с данными
const excelFilePath = path.join(__dirname, 'TSZH.xlsx');

// Читаем данные из Excel-файла
console.log("Чтение данных из Excel-файла...");
const workbook = xlsx.readFile(excelFilePath);
const sheetName = workbook.SheetNames[0];
const data = xlsx.utils.sheet_to_json(workbook.Sheets[sheetName]);

// Вывод данных в консоль сразу после чтения
console.log("Данные из Excel-файла:", data);

// Настройка клиента WhatsApp
console.log("Инициализация клиента WhatsApp...");
const client = new Client({
  authStrategy: new LocalAuth()
});
```

```

// Генерация QR-кода
client.on('qr', qr => {
  console.log("Сканируйте этот QR-код в приложении WhatsApp:");
  qrcode.generate(qr, { small: true });
});

// Когда клиент готов
client.on('ready', async () => {
  console.log("Клиент WhatsApp готов к отправке сообщений.");

  // Функция для отправки сообщения с задержкой
  const sendMessageWithDelay = async (person, delay) => {
    return new Promise(resolve => {
      setTimeout(async () => {
        // Проверка наличия имени и номера телефона, чтобы избежать ошибок
        if (!person.Name || !person['Phone number']) {
          console.error(`Ошибка: Отсутствует имя или номер телефона для записи`);
          return resolve(); // Переходим к следующей записи
        }

        // Преобразуем номер телефона в строку, если он число
        let phoneNumber = person['Phone number'];
        if (typeof phoneNumber === 'number') {
          phoneNumber = String(phoneNumber);
        }

        // Удаляем все нецифровые символы из номера и добавляем суффикс WhatsApp
        phoneNumber = phoneNumber.replace(/\D/g, '') + '@c.us';

        const message = `Уважаемый(ая) ${person.Name},

```

Завтра будет проводиться уборка снега с 9:00 до 18:00. Просим вас заранее убрать автомо

Спасибо за понимание!`;

```

  try {
    await client.sendMessage(phoneNumber, message);
    console.log(`Сообщение успешно отправлено на номер ${person['Phone number']}`);
  } catch (err) {
    console.error(`Ошибка при отправке сообщения на номер ${person['Phone number']}`);
  }
  resolve();
}

```



```
    }, delay);
  });
};

// Персонализированная рассылка сообщений с задержкой
for (let i = 0; i < data.length; i++) {
  // Задержка 4 секунды между сообщениями (15 сообщений в минуту)
  await sendMessageWithDelay(data[i], i * 4000);
}

});

// Обработка ошибок
client.on('auth_failure', () => {
  console.error("Ошибка авторизации. Проверьте свои учетные данные.");
});

client.on('disconnected', () => {
  console.log("Клиент WhatsApp был отключен.");
});

// Запуск клиента
client.initialize();
```

Завершение работы и проверка отправки

Чтобы убедиться, что сообщения успешно отправлены, можно использовать два способа. Во-первых, обратите внимание на логи в консоли. Программа выводит подробную информацию о процессе отправки: номера телефонов, текст сообщения и возможные ошибки. Если возникли проблемы, лог поможет понять, на каком этапе произошел сбой. Во-вторых, выполните ручную проверку. Зайдите в WhatsApp на подключённом аккаунте и убедитесь, что сообщения действительно отправлены указанным получателям.

Задел на будущее

В перспективе скрипт можно расширить, добавив более сложные сценарии: опросы, напоминания или уведомления об авариях. Такой подход позволит сделать общение с жильцами удобнее и эффективнее.

Заключение

Использование WhatsApp для автоматической рассылки сообщений в ТСЖ - это бесплатное, быстрое и удобное решение. Даже начинающий пользователь легко настроит скрипт благодаря понятным библиотекам и простому коду.

Однако стоит учитывать риски. Массовая рассылка может привести к блокировке аккаунта, поэтому начните с небольших списков и внимательно следите за скоростью отправки.

Попробуйте предложить эту систему в своём ТСЖ уже сегодня и оцените, насколько чище будет ваш двор!

Автор: Михаил Шардин

 [Моя онлайн-визитка](#)

 [Telegram «Умный Дом Инвестора»](#)

10 февраля 2025 г.

Теги: [whatsapp-web.js](#), [тсж](#), [снег](#), [уборка снега](#)

Хабы: [Open source](#), [Node.JS](#), [Инженерные системы](#)

Редакторский дайджест ✕

Присылаем лучшие статьи раз в месяц



183

87.1

Карма

Рейтинг

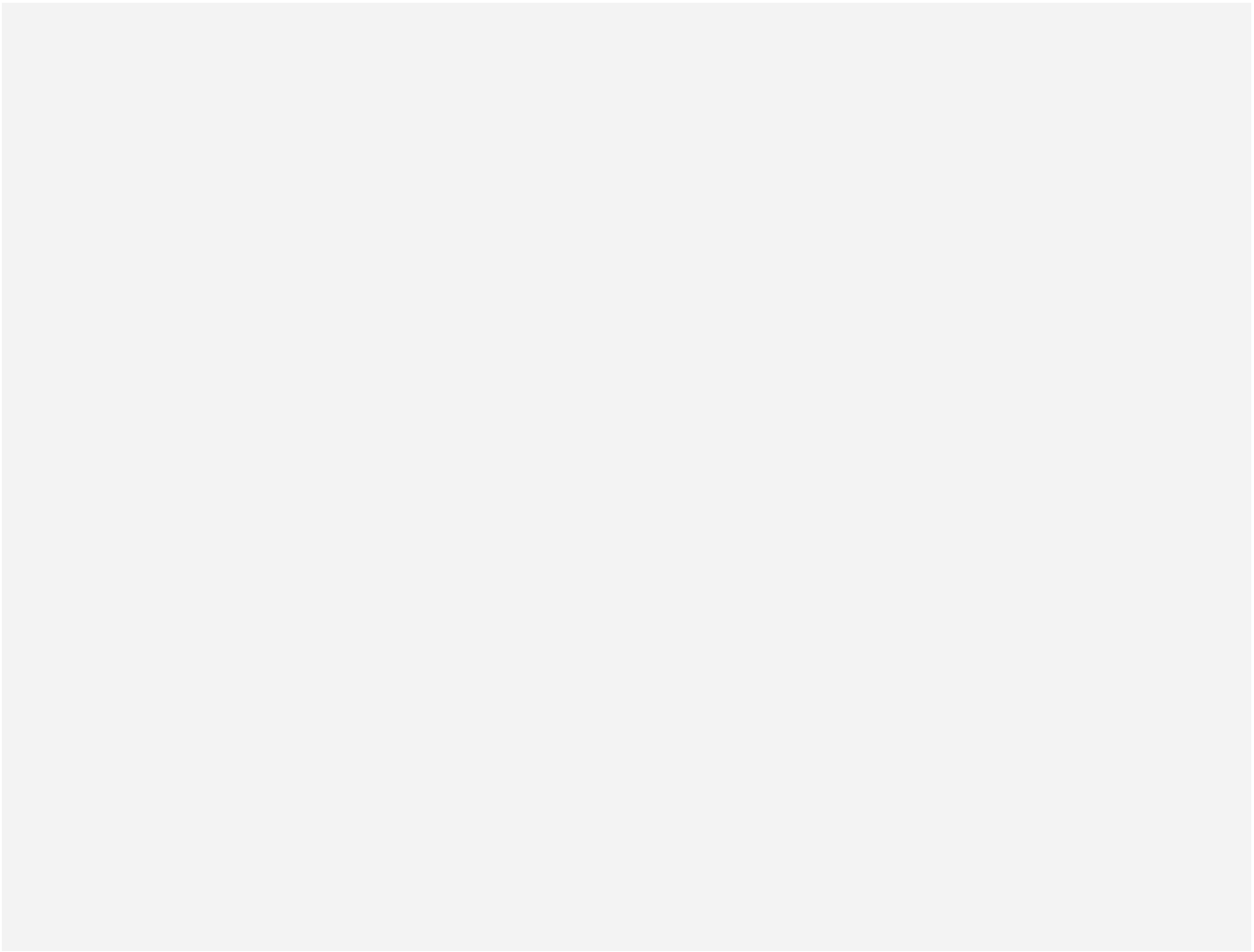
Михаил Шардин [@empenoso](#)

[Автоматизация](#) / [Данные](#) / [Финансы](#) / [Умные дома](#)

[Подписаться](#)



[Сайт](#) [Сайт](#) [Github](#)



 Комментарии 28

Публикации

ЛУЧШИЕ ЗА СУТКИ

ПОХОЖИЕ



rssdev10

22 часа назад

Почему въехав по «визе талантов» в США я с радостью вернулся в Россию



Средний



32 мин



34K

Мнение

 +169

 108

 491

**melnik909**

18 часов назад

Вы не знаете CSS. Мои вопросы о CSS с ответами. Часть 2

Средний 7 мин 2K

Обзор

+38

31

1

**DAN_SEA**

16 часов назад

Генерация случайных чисел

Средний 10 мин 2.3K

Обзор

+31

23

33

**OrkBiotecnologist**

22 часа назад

VPS за 139 рублей — дом для вашего резюме на основе Hugo

Простой 7 мин 8.2K

Тutorial

+28

45

12

**nebykoff_anton**

23 часа назад

До и после: оптимизация изображений для Lighthouse и не только

Простой 8 мин 941

Tutorial

+28




24

2




**Корчениу**


23 часа назад

Имитатор касаний. Ч2: Железная часть




 Средний  9 мин  1.6K

Кейс




 +24  20  12


 tertiumnon
17 часов назад

Минимум книг, которые нужно прочитать начинающему или продолжающему свою кривую обучения программисту




 Простой  3 мин  8.2K

Обзор




 +16  181  16


 PatientZero
1 час назад

Пишем стек TCP/IP с нуля: Ethernet, ARP, IPv4 и ICMPv4




 Простой  13 мин  747

Тutorial Перевод




 +15  21  1


 Ibkanter
48 минут назад

Бэкдор Auto-color: разбор угрозы, технический анализ и способы защиты

 Средний  4 мин  216

Обзор

 +13  4  2

 alexander-shustanov
18 часов назад

В поисках идеального Database-клиента для IDE: Amplicode выбирает DBeaver

Простой 6 мин 2.5K

+13

13

7

Как избежать цифрового хаоса при бизнес-миграции

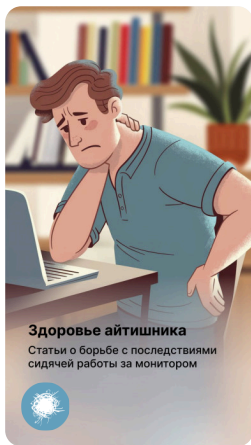
Турбо

Показать еще

ИСТОРИИ



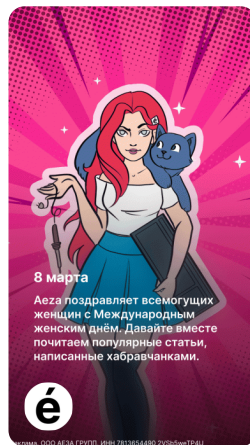
Как расти на работе?



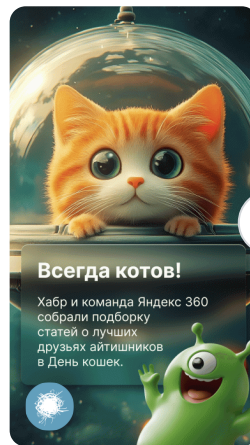
Здоровье айтишника



Угадайте будущее в новом сезоне



С праздником весны!



Всегда котов!

СРЕДНЯЯ ЗАРПЛАТА В ИТ

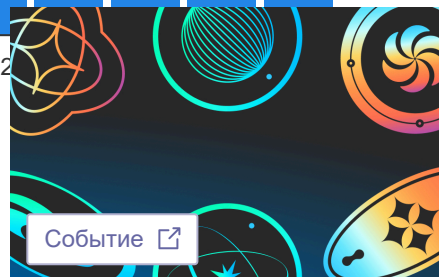
212 662 ₺/мес.

— средняя зарплата во всех ИТ-специализациях по данным из 33 637 анкет, за 1-ое пол. 2025 года. Проверьте «в рынке» ли ваша зарплата или нет!



Турбо

«Люк, я твой фактор!» — защищаем подключения с MFA



Событие [↗](#)

Как писать про здоровье на Хабре? Расскажем на вебинаре



Турбо

Сезон футурологии на Хабре: какой будет жизнь 3.0

РАБОТА

JavaScript разработчик

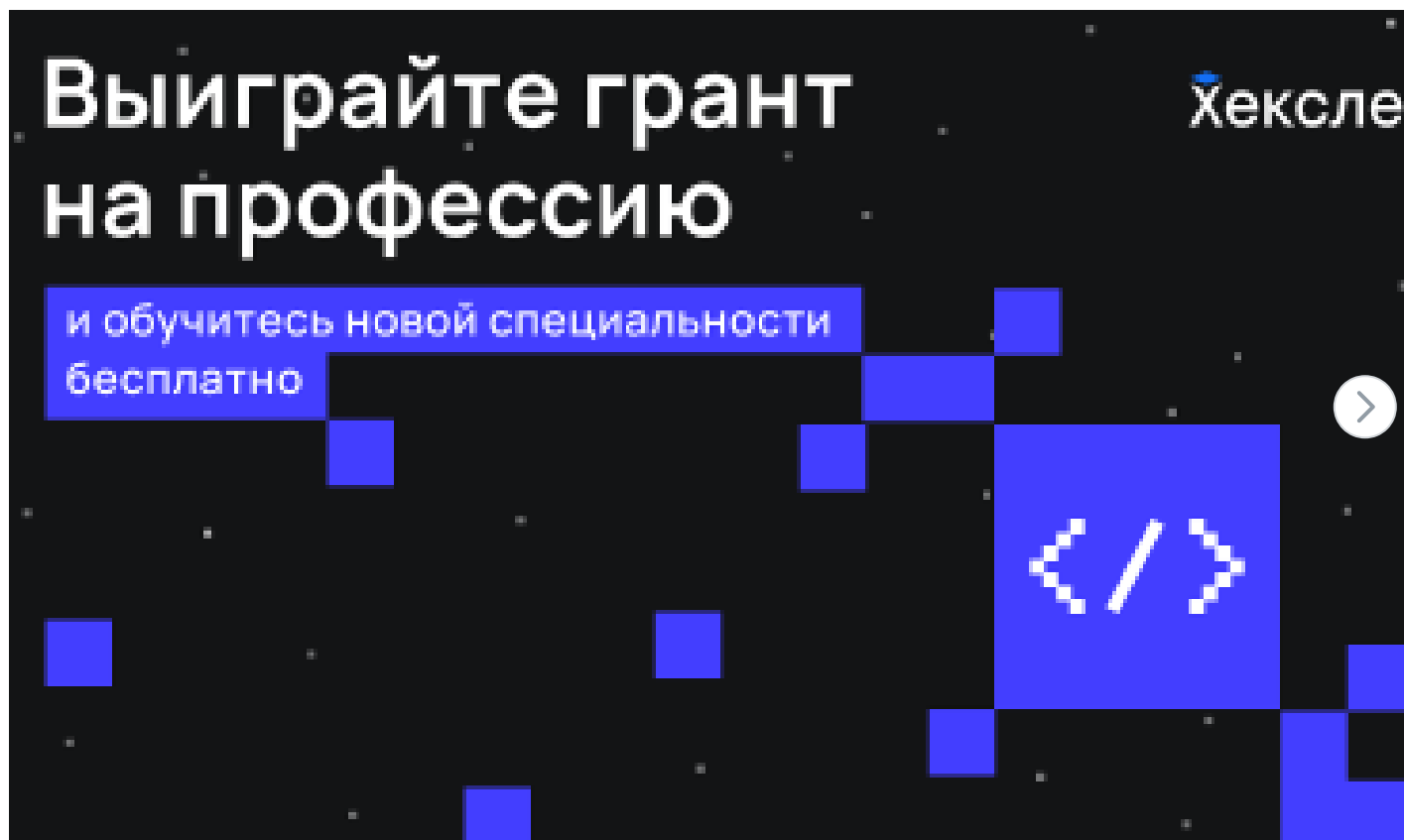
134 вакансии

Node.js разработчик

56 вакансий

[Все вакансии](#)

БЛИЖАЙШИЕ СОБЫТИЯ



17 февраля – 24 марта

Конкурс «Снежный код» от Хекслета. Три гранта на бесплатное 10-месячное обучение

Онлайн

Разработка

[Больше событий в календаре](#)

Хабр



🌐 [Настройка языка](#)

[Техническая поддержка](#)

© 2006–2025, Habr