



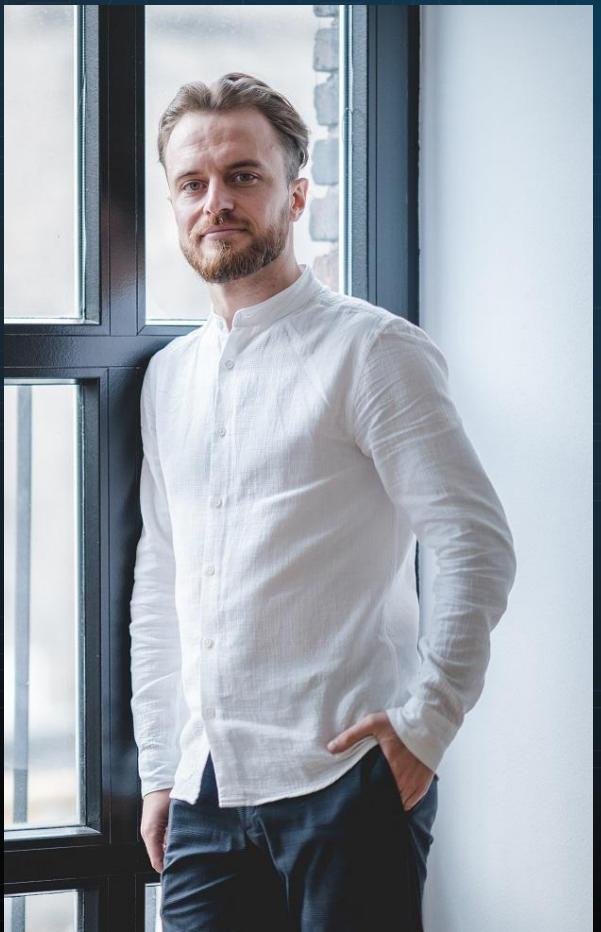
LEADTEХ

ЧАТ-БОТЫ ДЛЯ СТАРТАПОВ

www.leadteh.ru

О ЧЕМ ПОГОВОРИМ

- Знакомство
- Что такое чат-боты
- Какие типы чат-ботов бывают
- Как можно использовать чат-боты
- Вопросы - ответы



Касьянов Ярослав

31 год, г. Санкт-Петербург

ceo LEADTEX

Окончил:

ХНЭУ 2014, Мультимедийные издания

ВЭШ СПбГУ 2018, Менеджмент-маркетинг

Карьера в найме:

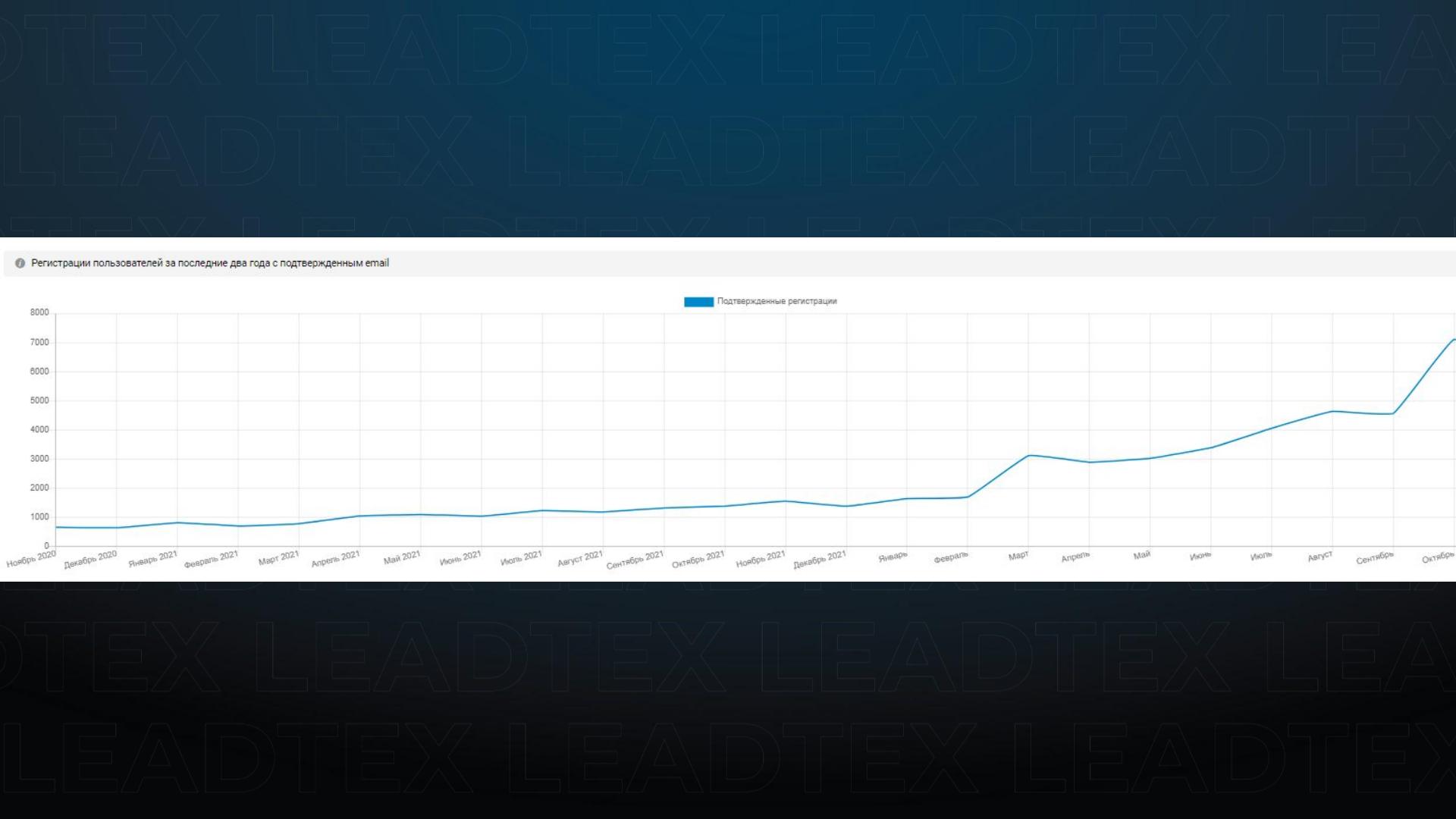
Графический дизайн: 2 года

Маркетинг: 4 года

Карьера в бизнесе:

LEADTEX: 3 года

DILIAPI: скоро



Количество пользователей

61 448

Количество ботов

73 539

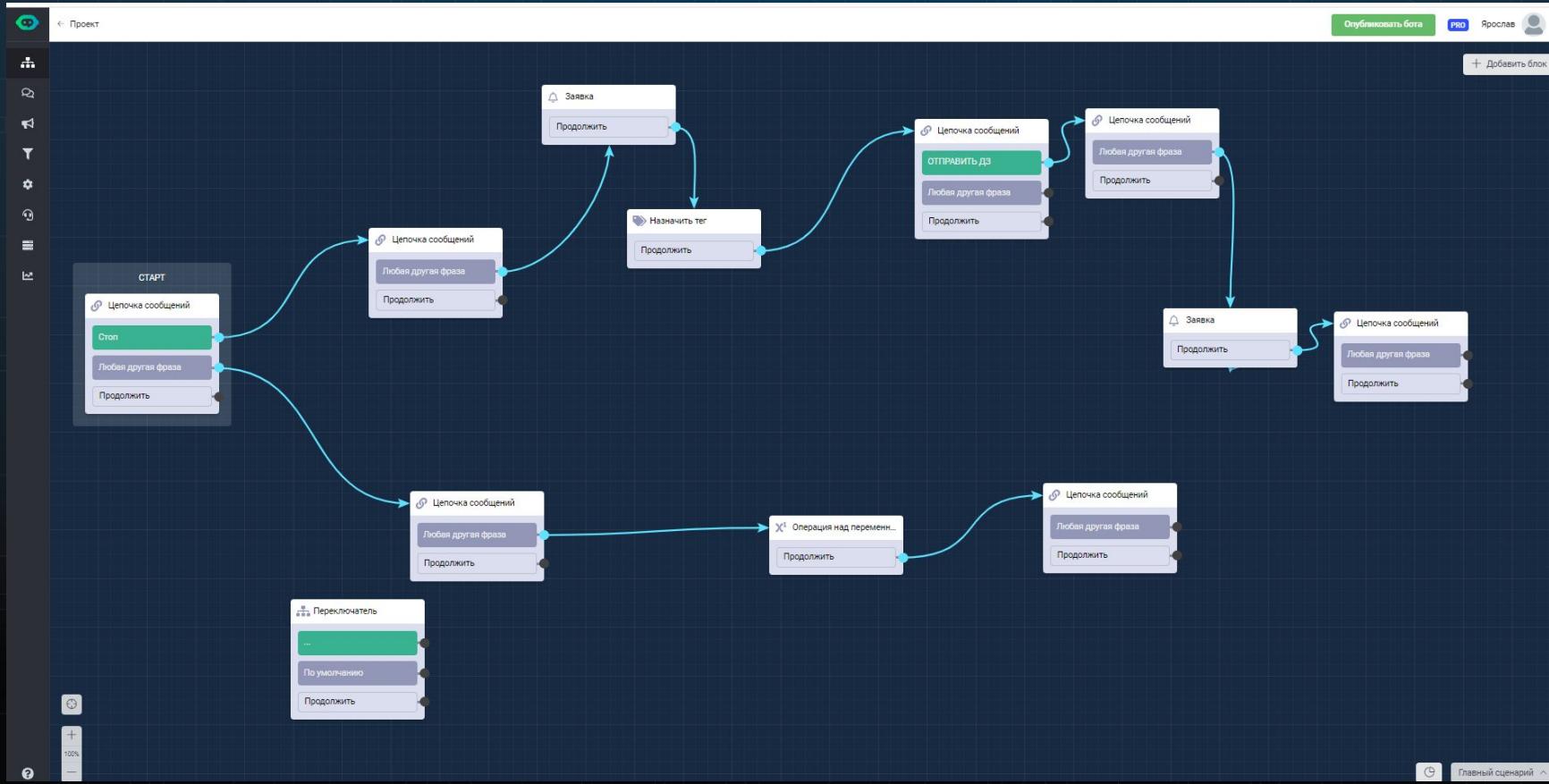
Количество сообщений

116 640 153

Количество подписчиков

3 001 868

LEADTEX - конструктор чат-ботов



ЧТО ТАКОЕ ЧАТ-БОТЫ

Чат-бот - это программа, которая выясняет потребности пользователей, а затем помогает удовлетворить их. Автоматическое общение с пользователем ведется с помощью текста. Чат-бот ведет коммуникацию от лица компании или бренда с целью упростить онлайн-общение (предоставить актуальную информацию в наиболее оперативные сроки), используется как альтернатива переписке с живым оператором или звонку менеджеру компании.

ЧАТ-БОТЫ СЕГОДНЯ

Это не просто текстовое общение робота с человеком.

Это сущность, которая не имеет границ применения.

Всё зависит от целей и задач.

ПРИМЕНЕНИЕ ЧАТ-БОТОВ

ПРОДАЖИ

СИСТЕМА

ПРОДАЖИ

ЧАТ-БОТЫ ДЛЯ ПРОДАЖИ ТОВАРОВ И УСЛУГ

СИСТЕМА

АВТОМАТИЗАЦИЯ БИЗНЕС-ПРОЦЕССОВ

ВИДЫ ЧАТ-БОТОВ

1. Бот-магазин (продажа товаров, доставка еды, услуг)
2. Бот-собеседник (техническая поддержка, тренинг)
3. Лид-бот (сбор контактов на сайте)
4. Бот-портал (корпоративный портал для компаний)
5. Автоворонка (автоворонка продаж товаров и услуг)
6. Бот-квиз (сбор контактов в виде теста или выявления потребности)
7. Бот для квестов и игр (квесты, корпоративы, игры)
8. Бот-приложение (аналог мобильного приложения)
9. Бот-онлайн школа (инструмент для обучения)

САМОПИСНЫЕ ЧАТ-БОТЫ

Чат-боты разработанный с помощью языков программирования, то есть написан с помощью кода.

САМОПИСНЫЕ ПРЕИМУЩЕСТВА

- реализация необычных и сложных опций;
- использование технических инструментов и платформ на ваш выбор.

САМОПИСНЫЕ НЕДОСТАТКИ

- длительная пауза между стартом проекта и запуском;
- необходимо оплачивать сервер и хостинг, а также вовремя обслуживать всю ИТ-инфраструктуру;
- цена стартует от \$250 и заканчивается десятками тысяч долларов;
- возможные трудности при смене исполнителя: разработчики редко берутся за доработку чужого кода, им выгоднее и проще написать собственный с нуля.

ЧАТ-БОТЫ НА КОНСТРУКТОРЕ

Чат-боты разработанные с помощью специальных no-code конструкторов.

ПРЕИМУЩЕСТВА КОНСТРУКТОРА

- быстрый запуск для тестирования гипотез и автоматизации процессов внутри бизнеса;
- легкая интеграция с внешними системами, если она включена в тарифный план;
- возможность самообучения по инструкциям, с помощью службы поддержки или базы знаний;
- бесплатное внедрение новых функций с каждым обновлением сервиса.
- скорость разработки в разы выше.
- удобная админ панель для редактирования и управления проектом.

НЕДОСТАТКИ КОНСТРУКТОРА

- реализация специализированных опций разработчиками только в том случае, если они необходимы большому количеству клиентов;
- в случае смены платформы экспорт базы подписчиков не всегда доступен;
- количество интеграций с внешними системами и мессенджеров ограничено.

СРАВНЕНИЕ

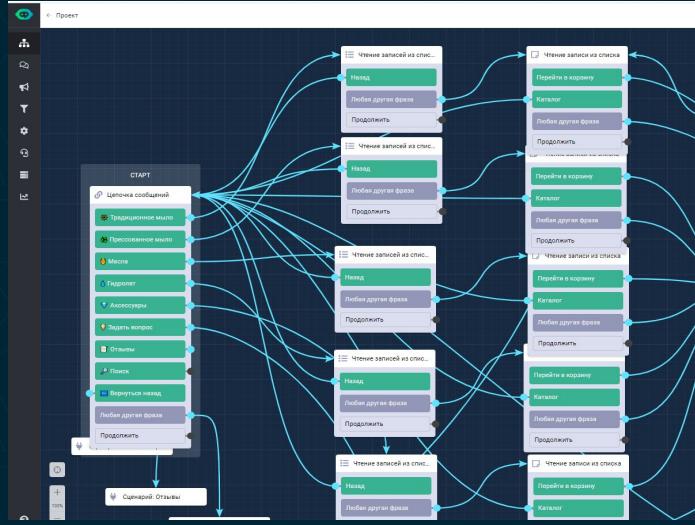
Screenshot of PyCharm IDE showing a Python script for a Telegram bot. The script uses the aiogram library to handle messages and interact with a database. It includes handlers for '/start' and '/help' commands, and a handler for '/email' which updates the user's email in the database.

```
from aiogram.dispatcher import FSMContext
from aiogram.dispatcher.filters import Command

from loader import dp, db

@dp.message_handler(Command("email"))
async def add_email(message: types.Message, state: FSMContext):
    await message.answer("Пришли мне свой имейл")
    await state.set_state("email")

@dp.message_handler(state="email")
async def enter_email(message: types.Message, state: FSMContext):
    email = message.text
    db.update_email(email=email, id=message.from_user.id)
    user = db.select_user(id=message.from_user.id)
    await message.answer(f"Данные были обновлены. Запись в бд: {user}")
    await state.finish()
```



ПЛЮСЫ ДЛЯ СТАРТАПОВ

- Быстрая сборка MVP
- Сохранение базы потенциальных клиентов
- Быстрая проверка гипотез
- Экономия на разработке
- Различные варианты использования

Спасибо за внимание!

