

## Идея. «ИИ-переговорщик», который делает всё за пользователя

— Универсальный AI-ассистент, способный находить любые сервисы, товары и услуги (от отелей и барбершопов до ресторанов и вещей), связываться с поставщиками/продавцами и вести с ними переговоры «от лица» пользователя, обеспечивая лучшие условия (цену, время, качество).

Впервые: **главная ценность** — не просто поиск, а именно **полностью автоматизированный диалог и договорённость** (конфигурация заказа, уточнение наличия, бронирование времени, согласование цены, доставка/услуга в нужный срок).

---

### 1. Актуальность и потребность рынка

#### 1. Рост «покупательской лени» и дефицит времени.

- В современных мегаполисах люди не хотят тратить 30–60 минут на обзвон отелей, кафе и салонов.
- Сантехник, барбер, консьерж в отеле и так дают сервис, но приходится самому договариваться, уточнять детали, ждать ответа.

#### 2. Повышение уровня ожиданий «безболезненного сервиса».

- Люди готовы платить за то, чтобы не переключаться между мессенджерами, сайтами, телефонами.
- В сегменте премиальных услуг (отели, рестораны, салоны) — консьерж-службы стоят \$50–\$200 в месяц, но большинство пользователей не готовы платить столько, им нужен «универсальный» AI-консьерж за более демократичную цену.

#### 3. Переговоры как наибольший барьер.

- Многие пользователи не любят говорить напрямую: «Хочу скидку», «Можно перенести время?», «Есть ли свободный мастер?»
  - Уровень коммуникации: не все умеют добиваться лучших условий — AI-ассистент берёт это на себя.
- 

### 2. Целевая аудитория

1. **Занятые профессионалы и руководители** (возраст 25–45 лет, доход  $\geq$  \$1 500/мес), которые ценят время и готовы платить за экономию даже 10–15 минут.
  2. **Молодые семьи** (тоже 25–40 лет), у которых 2–3 ребёнка, и нужно успеть «усилить» все бытовые задачи (записать ребёнка в школу, найти врача, забронировать такси).
  3. **Тревел-туристы (business & leisure)**, которые часто ездят: командировочные, digital nomads.
  4. **Люди с низким уровнем терпимости к «бумажной волоките»**: не любят «ходить по сайтам».
  5. **Сегмент «multi-service seekers»**: хотят один бот на все услуги — от поиска пиццы до махинаций по доставке товара.
- 

### 3. Существующие решения (конкуренты) и их ограничения

#### 1. Консьерж-службы (Bellboy, Quintessentially, Gepäck):

- Работают через человека, стоят от \$50/мес до нескольких сотен (абонентка), ограничены по количеству запросов.
- Не все услуги в пакете, часто надо отдельно звонить.

## 2. Чат-боты-агрегаторы (TripAdvisor, [Booking.com](https://www.booking.com), 2GIS-чат-боты):

- Есть базовый AI-ассистент для поиска, но не ведёт полноценные переговоры.
- Показывают «список» отелей или барберов, но не пишут «от лица» под твои требования.

## 3. Локальные Telegram-каналы и боты (например, «Запиши меня к барберу»):

- Ограничены одним видом услуги.
- Не умеют «держат диалог» и согласовывать время повторно, если что-то меняется.

## 4. Ремесленные агенты (Airbnb-concierge, персональные ассистенты на фрилансе):

- Как правило, берут 10–20 % комиссии, ждут 1–2 дня, чтобы выполнить запрос.
- Не всегда есть 24/7-поддержка, невозможно мгновенно изменить планы.

**Итого:** нет «универсального AI-агента», который мог бы «сам вмешаться» и довести любую задачу до результата, начиная от сложного (найти + забронировать хостел) до мелкого (купить наушники в ближайшем магазине с доставкой завтра к 9 утра).

---

## 4. Концептуальная архитектура продукта

### 4.1. Основные модули AI-ассистента

#### 1. NLP/NLU-слой (обработка запросов пользователя):

- Понимание intent'а («найди отель на завтра недалеко от Кремля, с парковкой»).
- Извлечение сущностей: дата, локация, услуга, бюджет, персональные предпочтения (звёздность, рейтинг, «только с завтраком»).

#### 2. Knowledge Graph & Ontology (база знаний о категориях услуг):

- Словарь «услуг»: отели, бары, рестораны, салоны, такси, товары.
- Типовые атрибуты для каждой категории (отель: район, звёзды, Wi-Fi; парикмахер: пол, стиль, цена).
- Связи: «отель — ближайшие бары», «барбершоп — район» и т.д.

#### 3. Поисковый и агрегационный модуль (data-scraping + API-интеграции):

- Интеграция с публичными API ([Booking.com](https://www.booking.com) API, 2GIS, Яндекс.Рестораны, Zoon, Google Maps, AliExpress/CJ Dropshipping).
- Web-scraping (если нет официального API) + обновление кэша (каждые 1–2 часа).
- Unit-инициализация источников (как fallback, если API недоступен).

#### 4. Диалоговый движок (Multi-Turn Dialogue Manager):

- Обеспечивает несколько раундов переговоров (диалог «бот ↔ поставщик»).
- Умеет возвращаться назад (если поставщик спрашивает уточнения), multi-steps («мне нужны 2 кровати, а не 1»).
- «Клиентские профили»: хранит preferences, тон разговора, как просить скидку.

#### 5. AI-переговорщик (Negotiation Agent):

- Использует LLM (GPT-4o, Claude 3 или локальная Llama X/Falcon) с fine-tuning на переписках (человеческих e-mail/push/Telegram) и «как договариваться о скидке».

- Умеет «Soft-negotiation»: просить «лучше цену» без хамства, цифры предложений, сравнительный анализ рынка.

- Управляет «офферами»: если отель дал слишком дорогую цену, запрашивает «лучшую ставку, ваши конкуренты дают 15 % скидку» (на основе глобальных цен).

#### 6. Transaction & Booking Backend (Оформление и подтверждение):

- Интеграция с платёжными шлюзами (Stripe, Payme /Kaspi).
- Генерация подтверждений брони, счетов, ссылок для оплаты, чек-ин email/SMS.
- Трекер статуса: «запрос отправлен», «ожидаем ответа», «бронирование подтверждено», «оплата получена», «услуга оказана».

#### 7. User Profile & Personalization (данные о пользователе):

- История: прошлые запросы и успешные «переговоры».
- Предпочтения: локации, бюджеты, «любимые отели», «любимый стиль».
- Напоминания и триггеры: «каждый год в марте бронировать отель у моря», напоминать о стрижке каждые 4 недели.

#### 8. Analytics & Reporting (метрики и отчёты):

- Отслеживание KPI: % успешных договорённостей, средняя стоимость услуг, «сэкономлено денег», LTV, Retention, NPS.
- Дашборд для админов и партнёров (отели/рестораны «сколько бронирований пришло через нас», средний чек).

---

### 5. Примеры сценариев использования (Use Cases)

#### 1. Поиск и бронирование отеля

- Пользователь: «Найти на 20–22 июня отель 4\* в центре Алматы, с завтраком и парковкой, не дороже 20 000Т/ночь».
- ИИ-ассистент:
  - через Booking API получает 10 вариантов, фильтрует по «есть парковка» и «завтрак включён».
  - выбирает 3 лучших, проверяет цены через сайт отеля напрямую (доп. скидка на предоплату).
  - открывает веб-чат или e-mail отеля, пишет: «Найдите, пожалуйста, тариф с учётом предоплаты и скажите, можно ли получить скидку 10 % при оплате сразу».
  - получает ответ, бронирует, присылает пользователю подтверждение и QR-код.

#### 2. Запись к барберу (парикмахеру-мужскому)

- Пользователь: «Мне нужна мужская стрижка близко к дому, рейтинг  $\geq 4.5$ ».
- ИИ-ассистент:
  - через 2GIS парсит ближайшие «барбершопы» в радиусе 3 км, проверяет рейтинги и расписание.
  - выбирает 2–3, пишет в Telegram-чат или WhatsApp владельцам: «Привет, нужен точно локальный мастер с опытом работы 5 лет. Хочу стрижку завтра в 18:00. Какая цена? Есть ли мастер?»
  - на основе ответа подбирает лучшее время, записывает, напоминает за час до визита.

### 3. Заказ любимого торта к празднику

- Пользователь: «Нужен шоколадный торт на 10 человек с оформлением «звёздным небом», доставить 14 июня в 10 утра».
- ИИ-ассистент:
  - через Yelp (или локальные кондитерские) выбирает 5-6 заказных кондитерских, проверяет отзывы.
  - обращается в чат-боты кондитерских: «Здравствуйте, нужен торт 2 кг, тематика «звёздный», вкус шоколадный, доставка 14 июня 10:00, адрес...», уточняет цену.
  - сравнивает сроки и стоимость, бронирует у лучшего.
  - отправляет уведомление клиенту с фото эскиза и суммой.

### 4. Покупка электроники (по фото) с торгом

- Пользователь загружает фото смартфона или ноутбука в Telegram-боте.
- ИИ-ассистент:
  - распознаёт марку/модель по фото (Computer Vision + Yolo/CLIP)
  - парсит ближайшие локальные объявления (OLX, Авито, [ceo.kz](https://www.ceo.kz))
  - пишет продавцам: «Я заинтересован в покупке MacBook Pro 2020 в хорошем состоянии. Готов забрать завтра, какой минимальный ценник?»
  - сравнивает ответы, ведёт торг («если вы вчера просили 300 000Т, я могу взять за 280 000Т сегодня»), и после согласования даёт контакт и инструкции по оплате и встрече.

### 5. Резерв столика в ресторане с учётом личных предпочтений

- Пользователь: «Нужен итальянский ресторан, веган-меню, живая музыка, завтра в 20:00 на 4 человека».
- ИИ-ассистент:
  - агрегирует рестораны (TripAdvisor + Google Maps + локальные Telegram-каналы).
  - пишет в WhatsApp администратору: «Здравствуйте, нас будет 4 человека, веган-меню, нам нужен столик в зоне с живой музыкой».
  - при подтверждении брони шлёт QR-код, предлагает меню на выбор (тоже генерирует PDF-лист с ценами).

---

## 6. Ключевые возможности и «киллер-фичи»

### 1. Auto Discovery + Auto Outreach

- Автоматический поиск в нужной категории, локальном радиусе, с учётом рейтинга и цены.
- Сразу же «пробой» контакт — не ссылка на сайт, а «написание от твоего имени».
- Переход от «хи-уки» (“я посмотрю”) к «точным запросам» („сценарий из нескольких параграфов“).

### 2. Переговорный AI-движок

- Промежуточная переписка «AI ↔ поставщик» (в email, Telegram, WhatsApp, Viber, VK).
- Soft-negotiation: запрос “скидки”, «лучших условий», «гибкого кейса» (например, «будет 2 ночи, но выписка поздняя»).
- Умная аргументация: «Вашей семье дали такой тариф — я хотел бы получить такой же».

### 3. Multi-Modal Input

- Фото: загрузка картинки (товар, меню, помещение).
- Голос: «Хочу шашлык, но без лука» — AI распознаёт intent.
- Текст: поддержка естественной речи (даже сленга).

### 4. Персональные профили и контекст

- История прошлых запросов: «каждый март ищу волейбольную площадку с тренером»
- User-preference matrix: «не люблю высоту», «люблю позднюю выписку», «только cashless»
- Сохранение «тональности» и «уровня торга» (от «я просто уточняю» до «нам нужна скидка»).

### 5. Самодиагностика отказов и fallback

- Если поставщик не отвечает за 12 часов, AI переключается на backup-вариант (другой ресторан/отель).
- Автоматическая перераспределение запросов (escalation): «администратор не взял трубку — ведём ещё одного участника диалога» (например, менеджера более высокого уровня).

### 6. Instant Booking + Payment

- Одно-кликковое подтверждение, интеграция с платежами (Stripe, Apple Pay, Kaspi).
- Генерация e-ticket, QR-приглашение, календарная запись.

### 7. Географическая и временная оптимизация

- Учитывается трафик, дорожная обстановка, загруженность (Google Traffic)
- Автобронирование такси до места («если еду из офиса в 18:00, пересчитай время дороги»).

---

## 7. Архитектура MVP

Чтобы за 8–10 недель запустить **первую версию**, хватит минимального набора:

### 1. Frontend (мобильный + веб-чат-бот)

- Telegram/WhatsApp-бот или мобильное приложение (React Native), plus Web-интерфейс (Next.js).
- Простой UI: поле «Что нужно?» + несколько кнопок-шаблонов (Отель, Барбер, Ресторан, Товар).

### 2. Backend (Node.js/Express + Python-микросервисы)

- API-роуты для создания задач, отложенных задач, истории.
- Микросервис «Negotiation Agent» (Python + FastAPI) с LLM (GPT-4o) для ведения переписки и логики «как торговаться».

### 3. Data Layer

- PostgreSQL (юзер-профили, история запросов, настройки)
- Redis (кеширование свежих товаров/вакансий/моделей)
- MongoDB (лог переписок “AI ↔ поставщик”)

### 4. Интеграции (через API и scraping)

- [Booking.com](https://booking.com) API (или Amadeus API) → поиск отелей
- 2GIS/Google Maps API → локальный поиск кафе, барбершопов

- Защищённый web-scraper (Puppeteer) для сайтов, где нет API (например, локальные СПА-салоны)
- Сообщения через встроенные чаты (WhatsApp Cloud API, Telegram Bot API, Viber Business API).
- Платежи: Stripe, Payme/Kaspi.

## 5. LLM-модели

- GPT-4o / Claude 3 Sorbet: основной диалог «пользователь ↔ AI».
- Fine-tuning на конкретных шаблонах писем-запросов (этикет, переговоры, скидки, брони) для «Negotiation Agent».
- Whisper (или Vosk) для конвертации голосовых запросов в текст.
- CLIP/YOLO для распознавания товаров по фото.

## 6. Диалоговый менеджер (Rasa или собственная логика)

- Хранит состояния диалога (когда ожидать ответ, триггерить fallback-сценарий).
- Переключается между «поиском» → «переговоры» → «оплата» → «завершение».

## 7. Уведомления и напоминания

- Push/Push-оповещения (Firebase Cloud Messaging) + Email/SMS (Twilio/SendPulse).
- “Статусы” и логика пересылки: «ждём ответа», «ответ получен, нужен твой финальный confirm», «услуга оказана».

---

## 8. Технологии и модель разработки

### 1. Бэкенд:

- Node.js (Express) — API.
- Python (FastAPI) — AI-микросервисы (Negotiation Agent, Vision).
- PostgreSQL + Redis + MongoDB.

### 2. Фронтенд:

- React (Next.js) для веб-версии.
- React Native (Expo) для мобильного приложения.
- Telegram/WhatsApp-бот на базе node-библиотек.

### 3. AI/ML:

- OpenAI GPT-4o (Dialog & Negotiation).
- Hugging Face Inference (при необходимости локальных моделей).
- Whisper (Speech-to-Text).
- CLIP/YOLO (Object Detection для фото-товаров).
- Fine-tuning с Python + PEFT (Parameter Efficient Fine-Tuning).

### 4. Интеграции:

- [Booking.com](https://booking.com) API / Amadeus.
- 2GIS / Google Maps Places API.

- Yandex.Market API (для поиска товаров).
- Telegram Bot API / WhatsApp Cloud API / Viber Business API.

#### 5. **DevOps:**

- Docker + Kubernetes (кластеры для AI-микросервисов).
- GitHub Actions → CI/CD (build/test/deploy).
- AWS/GCP/Hetzner (серверы + GPU инстансы).
- Sentry (логирование ошибок), Prometheus/Grafana (метрики).