



PIX Robotics

делает умнее

PIX RPA в мире LLM и агентных платформ

Вебинар для команды продаж PIX RPA (февраль 2026)

Николай Подволоцкий
Архитектор решений PIX RPA,
PIX Robotics



План на сегодня

Блок 1: Ситуация на рынке и классы решений

Всплеск интереса к ИИ. Рынок ИИ-автоматизации, классы решений, модели поставки

Блок 2: Путь клиента

Как внедряют AI, почему >80% проектов проваливаются, PoC vs Продакшн

Блок 3: Карта рынка (2025-2026)

OSS-фреймворки, агентские платформы, прямые конкуренты

Блок 4: Возражения

«Ваш модуль нам не нужен»

Блок 5: Предложения развития

Что перенять от рынка для решения проблем внедрения ИИ

Итог & Вопросы

Зачем этот вебинар

- Клиенты всё чаще спрашивают: «Хотим ИИ-комбайн для всего», «Зачем RPA, если есть другое решение?», «У вас есть AI, чем он лучше нашего?»
- Почему клиент выбирает другое решение?
- На рынке 20+ ИИ-вендоров\решений которых мы не знаем
- Как мы можем конкурировать с ними?
- Нужна единая система аргументов и отработка возражений

Что мы должны уметь после вебинара

- Понимать разницу: LLM, агентская платформа, RPA + AI
- Понять почему клиенты выбирают не нас. Поймем путь клиента
- Понять разницу ИИ-решений и назвать роль PIX в архитектуре клиента
Где усиливаем, где дополняем, где заменяем
- Отработка возражений
- Поговорить о возможном развитии направления ИИ



PIX Robotics

делает умнее

Ситуация на рынке и классы решений

БЛОК 1

Что происходит на рынке ИИ

Рынок GenAI* в РФ: ~58 млрд руб. в 2025 (рост ×3-4 год к году) (cnews.ru)

Драйверы спроса — почему компании внедряют прямо сейчас:

- Дефицит кадров: нехватка операционистов в back-office и разработчиков
- Импортозамещение: требование использовать «российские» LLM (закон, санкции)
- Быстрый ROI: пилоты запускаются за недели, а не месяцы
- Появление готовых решений «из коробки»
- Снижение нагрузки: автоматизация рутины освобождает людей для сложных задач

*GenAI (Генеративный ИИ) — это класс нейросетей, которые умеют *создавать новый контент* (текст, код, изображения), а не просто анализировать старый.



Боль и проблемы клиента

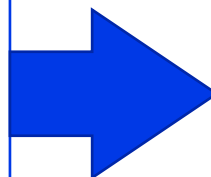
- Запуск ИИ-проектов часто затягивается и требует много времени.
- Разработка и дальнейшая поддержка ИИ-решений обходятся дорого.
- Возникает сильная привязка к редким профильным экспертам и разработчикам.
- Трудно масштабировать удачные пилоты на другие команды и подразделения.
- Отсутствуют единая платформа, общие правила и стандарты для ИИ-ассистентов.
- Использование внешних сервисов и облачных LLM несёт дополнительные риски для безопасности и данных.
- Внедрение ИИ в текущий ИТ-ландшафт (CRM, ERP, 1С и т. п.) обычно требует сложной интеграции.



ИИ есть. Автоматизации нет.

Факты

1. Более 50% крупных предприятий уже используют ИИ (LLM/RAG, AI-платформы)
2. Идёт тренд на on-prem и гибридные решения
3. Требования к безопасности данных растут
4. Клиенты хотят быстрой окупаемости (×3 в год)
5. Компании хотят готовые решения по ИИ, а не пилоты «с нуля», либо строят свой ЦК
6. Отдельные пилоты ИИ не дают бизнес-эффекта
7. Фокус смещается с «больших обещаний» на измеримый результат в течение квартала



К чему это приводит

- Запрос на платформенный подход (единая платформа, оркестрация нескольких агентов, а не точечные инструменты, централизованное управление)
- Спрос на коробочные enterprise-платформы
- Гибридные архитектуры on-prem или облако
- Фокус на быструю окупаемость и масштабирование успешных кейсов
- Консолидация рынка

Как устроен рынок ИИ-автоматизации

LLM, RAG, Embeddings - это фундамент. Ассистенты, агенты, RPA+ИИ - это продуктовые уровни.

Термин	Что это простыми словами	Как клиент это говорит	Что отвечать продавнику
LLM (Large Language Model)	Большая языковая модель — «мозг», который читает, пишет, рассуждает. Примеры: GPT, Claude, GigaChat.	"Нам нужен LLM" "Хотим подключить GPT" "У вас есть своя модель?"	"Да, наше решение работает с LLM. Можем подключить YandexGPT, GigaChat или вашу модель."
RAG (Retrieval-Augmented Generation)	Поиск + генерация. ИИ сначала ищет в ваших документах, потом формулирует ответ на основе реальных данных. Не выдумывает.	"ИИ не должен врать" "Нужно, чтобы отвечал по нашим документам" "Как избежать галлюцинаций?"	"У нас RAG — система ищет ответ в ваших базах знаний, не выдумывает. Вы загружаете документы, ИИ работает только с ними."
Embeddings	Способ, как ИИ понимает смысл. Превращает слова в числа и находит похожие фразы, даже если они написаны по-разному. Используется для RAG Пример: "как вернуть товар" = "процедура возврата"	"Как ИИ понимает, что клиент спросил одно и то же разными словами?"	"Используем embeddings — ИИ понимает смысл, а не просто ищет по ключевым словам. 'Возврат товара' и 'хочу вернуть' для него одно и то же."
VLM (Vision-Language Model)	Модель, которая видит и понимает картинки, скриншоты, PDF, диаграммы, таблицы — как человек. Не нужно учить структуру документа.	"Нужно распознавать инвойсы" "Робот должен читать скриншоты" "Документы в PDF, не в базе"	"У нас VLM — система видит документы как человек. Инвойсы, скриншоты, таблицы — распознаёт без настройки под формат."
Tool Use / Function Calling	Способность ИИ вызывать внешние инструменты : API, базы данных, CRM, калькуляторы. Без этого ИИ — просто болтун.	"Как ИИ создаст заявку в CRM?" "ИИ должен работать с нашими системами"	"У нас Tool Use — ИИ вызывает ваши системы через API. Может создать заявку в CRM, отправить письмо, проверить остатки в базе."
MCP (Model Context Protocol)	Стандарт подключения ИИ к системам — как USB для компьютеров. Один стандарт → работает с любыми системами.	"У вас поддержка MCP?" "Как быстро подключить к нашим системам?"	"Да, поддерживаем MCP — это индустриальный стандарт. Подключение к вашим системам быстрее и дешевле, чем кастомные интеграции."
ИИ-агент (AI Agent)	ИИ, который не просто отвечает — сам планирует и выполняет задачи. Анализирует ситуацию → выбирает действия → вызывает инструменты → проверяет результат. Работает как умный сотрудник, а не как поисковик.	"Хотим автоматизировать сложные сценарии" / "Нужен ИИ, который сам принимает решения" / "Как быть с нестандартными случаями?"	"Наш агент действует, а не просто отвечает. Получил задачу — сам разобрался, выбрал инструменты, довёл до результата. Вы задаёте цель, агент — путь."
Мультиагентная система (Multi-agent)	Несколько специализированных ИИ-агентов, каждый отвечает за свою роль. Один анализирует запрос, второй ищет данные, третий оформляет заявку — и всё работает в связке.	"У нас сложный процесс с несколькими этапами" / "Разные системы, разные задачи" / "Один робот не справится со всем"	"Это как команда специалистов — каждый делает своё, результат передаётся дальше по цепочке. Надёжнее, чем один агент на всё, и легко масштабировать."
Оркестрация / LLM Workflow	Управление цепочкой шагов с участием ИИ: кто что делает, в каком порядке, что передаётся дальше. Дирижёр для агентов. Каждый шаг прозрачен и контролируем.	"Как выглядит процесс изнутри?" / "Можно ли контролировать, что делает ИИ?" / "Нам важна прозрачность и аудит"	"Workflow — это схема процесса, где виден каждый шаг ИИ. Можно настроить логику, добавить согласования, ограничения. Ничего не происходит в чёрном ящике."

Классы решений на рынке ("какой функционал нужен клиенту")

Класс решения	Что хочет клиент	Что это на самом деле	Пример кейса	Когда достаточно	Когда нужен следующий уровень
AI Assistant / Copilot	"Чтобы сотрудники быстрее работали"	Умный помощник: отвечает, подсказывает, ищет в базе знаний	Менеджер спрашивает "какие условия у клиента XYZ?", получает ответ из CRM, сам оформляет договор	Человек читает ответ и делает работу сам	Нужно убрать человека из процесса
AI Workflow Automation	"Автоматизировать простые шаги"	Сценарии через API: если А, то Б, вызвать систему, записать результат	Клиент заполнил форму на сайте → система автоматически создаёт лид в CRM, отправляет welcome-письмо, ставит задачу менеджеру	Все системы имеют API, процесс линейный	Есть legacy без API, нужны клики по UI
AI Agent Systems	"Чтобы система сама решала"	Агент планирует цепочки действий, выбирает инструменты, ветвит сценарии	Клиент пишет "хочу вернуть товар" → агент сам проверяет заказ, открывает заявку, выбирает способ возврата, уведомляет склад и клиента	Действия выполняются через API	Нужна работа с UI, файлами, desktop-приложениями
BPM + AI <small>(CRM + AI, ERP + AI, ITSM / HelpDesk + AI, BI + AI)</small>	"Управлять процессом и контролировать"	Процессный движок: роли, маршруты, регламенты + ИИ внутри процесса	Согласование договора: ИИ классифицирует тип, определяет маршрут, заполняет поля, отправляет на согласование по регламенту, контролирует сроки	Важен контроль "кто/что/когда", большинство шагов через интеграции	В процессе есть шаги "через экран" без API
RPA + AI (IPA - Intelligent Process Automation)	"Чтобы работало со всем и давало SLA"	Роботы исполняют в UI (клики, формы), ИИ управляет и принимает решения	Обработка входящих инвойсов: робот открывает письмо, скачивает PDF, ИИ распознаёт данные, робот вводит в SAP GUI (где нет API), сверяет с 1С, отправляет на оплату. SLA: 10 минут	Есть legacy, нужны гарантии исполнения, большой объём операций	

Может ли RPA+ИИ реализовать все эти классы?

Да, почти любой, но вопрос - целесообразность и эффект

Класс решения	Может ли RPA+ИИ это делать?	Почему избыточно	Когда имеет смысл RPA+ИИ	Эффект RPA
AI Assistant / Copilot	<p>✅ Да</p> <p>Может принимать запросы, обращаться к LLM, выдавать ответы</p>	<p>Человек всё равно сам делает работу руками.</p> <p>Робот для чата - как экскаватор для грядки.</p>	<p>Редко.</p> <p>Если платформа обладает встроенным функционалом для общения или легкой возможности интеграции с клиентским UI.</p>	<p>● Средний</p>
AI Workflow Automation	<p>✅ Да</p> <p>Может строить сценарии, вызывать API, запускать логику</p>	<p>Если все системы имеют API - зачем робот? Workflow справится в 3 раза дешевле и быстрее.</p>	<p>Только если появляются системы без API (1C, SAP GUI, веб-формы).</p> <p>Тогда добавляем RPA точно.</p>	<p>● Очень высокий</p> <p>Избыточно при наличии API</p>
AI Agent Systems	<p>✅ Да</p> <p>Может планировать, выбирать инструменты, исполнять цепочки</p>	<p>Если агент работает через API - RPA не нужен. Добавляет сложность и стоимость без выхлопа.</p>	<p>Когда агенту нужно исполнять в UI:</p> <ul style="list-style-type: none"> - Кликать в старых системах - Работать с desktop-приложениями - Переносить данные через экран 	<p>● Высокий</p> <p>Но избыточно без legacy</p>
BPM + AI	<p>❓ Сомнительно</p> <p>Может управлять процессом, контролировать сроки, исполнять шаги, но не имеет BPM движка с оргструктурой, согласованиями и прочим.</p>	<p>Если большинство шагов через интеграции - BPM+AI дешевле.</p> <p>RPA нужен только для legacy-шагов.</p>	<p>Когда >30% шагов процесса требуют работы с UI:</p> <ul style="list-style-type: none"> - SAP GUI - 1C без API - Веб-порталы без интеграции - Desktop-приложения 	<p>● Средний</p> <p>Для помощи в автоматизации UI</p>

Модели поставки (как клиент получает решение)

Термин	Что это простыми словами	Как клиент это говорит
Self-managed / Self-hosted	Клиент берёт решение (часто open-source), ставит на свои сервера , сам настраивает и поддерживает . Вендор не отвечает за работу.	"Хотим к себе на сервера" "Нужен open-source" "Сами будем управлять"
On-prem / On-premise	Решение устанавливается на инфраструктуре клиента (их сервера, их дата-центр). Не в облаке, а "на территории" клиента.	"Нам нужен on-prem" "Облако не подходит" "Данные не должны уходить за периметр"
SaaS / Облако	Решение работает в облаке вендора. Клиент платит за использование, вендор поддерживает инфраструктуру.	"Хотим быстро попробовать" "Нет своих серверов" "Нужно в облаке"
Hybrid / Private Cloud	Решение вендора, но на инфраструктуре клиента (или в их облаке). Клиент контролирует данные, вендор даёт софт и поддержку.	"Нужен on-prem, но с поддержкой" "Данные у нас, софт у вас" "Частное облако"

Сравнительная таблица возможностей классов

Возможность	AI Assistant / Copilot	AI Workflow Automation	AI Agent Systems	BPM + AI	RPA + AI (IPA)
Генерация контента (текст, ответы, анализ)	✅ Да Основная функция	✅ Да На отдельных шагах	✅ Да Часть планирования	✅ Да В точках принятия решений	✅ Да Для интеллектуальных решений
Планирование действий (выбор цепочек, ветвление)	❌ Нет Только отвечает	⚠️ Частично Жёсткие сценарии "если-то"	✅ Да Динамическое планирование	✅ Да По регламентам и правилам	✅ Да С учётом процессов и SLA
Исполнение через API (интеграция с системами)	❌ Нет Человек делает сам	✅ Да Основная функция	✅ Да Вызов инструментов	✅ Да Оркестрация интеграций	✅ Да Где есть API
Исполнение в UI (работа с интерфейсами, клики)	❌ Нет	❌ Нет	❌ Нет (только через API)	⚠️ Частично Если добавлен RPA-слой	✅ Да Основная функция для legacy
Работа с legacy без API (1C, SAP GUI, desktop-приложения)	❌ Нет	❌ Нет	❌ Нет	⚠️ Частично Если интегрирован RPA	✅ Да Полная поддержка
Контроль процесса (сроки, роли, маршрутизация)	❌ Нет	⚠️ Базовый Уведомления, таймауты	⚠️ Базовый Без регламентов	✅ Да Полноценный процессный движок	✅ Да С контролем исполнения, зависит от реализации
SLA-гарантии (задача выполнена за N времени)	❌ Нет	❌ Нет	❌ Нет	⚠️ Частично Контроль сроков без гарантий исполнения	✅ Да Полные SLA на исполнение
Человек в процессе	✅ Обязательно Остаётся исполнителем	❌ Нет Полная автоматизация (если API есть)	❌ Нет Полная автоматизация (если API есть)	⚠️ Опционально На согласованиях, проверках	⚠️ Опционально В зависимости от выбранного процесса
Модель поставки	SaaS, On-prem, Self-managed	SaaS, On-prem, Self-managed	SaaS, On-prem, Self-managed	On-prem, Private Cloud	On-prem, Private Cloud, Hybrid
Срок внедрения	Дни	Недели	Недели — месяцы	Месяцы	Месяцы
Типовая стоимость (порядок величины)	От 50 тыс. ₽	От 200 тыс. ₽	От 500 тыс. ₽	От 1 млн ₽ (проект)	От 1 млн ₽ (проект)
Когда выбирать	Нужна помощь, не автоматизация	Все системы с API, простые сценарии	Нужно планирование, системы с API	Сложные процессы, регламенты, контроль	Legacy без API, SLA критичны, большой объём



PIX Robotics

делает умнее

Путь клиента от PoC к продю

БЛОК 2

Как клиенты внедряют AI

Архитектура × Инструменты = 6 возможных путей

	 ОБЛАКО (Cloud API)	 ON-PREMISE (Свои сервера)	КТО ТИПИЧНЫЙ КЛИЕНТ	ПОЧЕМУ ВЫБИРАЮТ ЭТОТ ПУТЬ
 OSS (DIY) Разработка	3 - 7 дней (GigaChain / LangChain + API)	1 - 3 месяца (Llama 3 + vLLM на арендованном GPU)	<ul style="list-style-type: none"> - Продуктовые компании - IT-стартапы - Системные интеграторы - Компании с сильной dev-командой 	<ul style="list-style-type: none"> - Нужен уникальный функционал - Планируют монетизировать AI-продукт - Есть in-house ML-инженеры - Требуется максимальная гибкость
 ГОТОВЫЕ ПЛАТФОРМЫ Low-code	2 - 4 недели (Just AI Cloud, Dify Cloud)	1.5 - 3 месяца (CorpGPT On-Prem, Dify Local)	<ul style="list-style-type: none"> - Корпорации без AI-экспертизы - Средний бизнес - Госсектор - Компании с распределенными отделами 	<ul style="list-style-type: none"> - Нет своей команды ML-разработки - Нужно быстро дать AI всем отделам - Требуется единая безопасная среда - Не хотят нанимать дорогих ML-инженеров
 RPA + AI Умные роботы	2 - 4 недели (PIX + Smart-модуль / API)	2 - 4 месяца (PIX + локальный vLLM)	<ul style="list-style-type: none"> - Банки и страховые - Ритейл и телеком - Производство - Компании с тяжелым legacy 	<ul style="list-style-type: none"> - Уже внедрены RPA-роботы - Много старых систем без API - Не хотят переписывать процессы - Нужно добавить "ум" в автоматизацию

ЧТО ВЛИЯЕТ НА ВЫБОР:

Облако ↔ On-Premise	DIY ↔ Платформы ↔ RPA
<ul style="list-style-type: none"> • Требования ИБ (ФЗ-152) • Бюджет на CAPEX (GPU) 	<ul style="list-style-type: none"> • Наличие команды разработки • Количество legacy-систем • Сроки запуска

Почему клиенты идут разными путями

1 Начинают с простого

Почему так делают:

- Нет опыта
- Проще и быстрее старт
- Нет сложной логики
- Нет целей по масштабированию
- Просто нужно извлечь данные или сгенерировать текст

И это нормально.

2 Когда этого достаточно

- Одна модель
- Один вызов LLM
- Нет сложной маршрутизации
- Нет повторного использования логики
- Нет требований к управлению ИИ

В этом случае Smart-модуль может быть избыточным.

3 Когда начинают появляться ограничения


- Несколько моделей
- Нужен RAG
- Нужна маршрутизация
- Нужен контроль промптов
- Нужна повторяемость
- Нужен аудит

Тогда возникает потребность в управляемом AI-слое.

Что нужно для продакшн-системы

От демо к продакшену: 5 критических компонентов

КОМПОНЕНТ	ЧТО ЭТО ЗНАЧИТ	ПОЧЕМУ КРИТИЧНО	СТОИМОСТЬ ИГНОРИРОВАНИЯ
1. ИСПОЛНЕНИЕ ДЕЙСТВИЙ (Action Execution)	<p>Бот не просто отвечает, но делает:</p> <ul style="list-style-type: none"> - Создает заявки в ServiceDesk - Бронирует переговоры - Отправляет письма - Обновляет базы данных 	<p>Без этого бот = дорогой FAQ. Нет автоматизации, нет ROI.</p>	<p>Сотрудники продолжают делать все вручную. ROI проекта = 0</p>
2. ИНТЕГРАЦИИ (System Integration)	<p>Связь с корпоративными системами:</p> <ul style="list-style-type: none"> - 1C, SAP, CRM (Битрикс24) - Active Directory (SSO) - Email, календари (Outlook, Яндекс.Почта) - Внутренние API 	<p>LLM должен получать и обновлять данные в реальном времени. Без этого ответы устаревают.</p>	<p>Бот дает устаревшую информацию. Пользователи перестают доверять системе</p>
3. НАДЕЖНОСТЬ И SLA (Reliability & SLA)	<p>Гарантии качества:</p> <ul style="list-style-type: none"> - Точность ответов (метрики) - Время отклика < 3 сек - Uptime 99.5%+ - Fallback на человека при неуверенности 	<p>В продакшене нельзя "иногда работать". Бизнес требует предсказуемости.</p>	<p>Клиенты получают неверные данные. Репутационные риски, штрафы</p>
4. МОНИТОРИНГ И ПОДДЕРЖКА (Monitoring & Ops)	<p>Система наблюдения:</p> <ul style="list-style-type: none"> - Логи всех запросов и ответов - Алерты при сбоях (Telegram, email) - Дашборды с метриками - Дежурная команда 24/7 	<p>Без мониторинга не узнаете о проблемах, пока пользователи не взбунтуются.</p>	<p>Система упала, никто не знает. Потеря доверия, остановка бизнес-процессов</p>
5. БЕЗОПАСНОСТЬ И АУДИТ (Security & Compliance)	<p>Контроль доступа и соответствие:</p> <ul style="list-style-type: none"> - Кто, когда, что спросил (audit trail) - Разграничение прав (RBAC) - Соответствие ФЗ-152, ИБ - Защита от утечек данных 	<p>В энтерпрайзе безопасность не опция</p>	<p>Утечка персональных данных. Штрафы до 500 тыс. ₽ по ФЗ-152,</p>

 "PoC — это 20% пути. Остальные 80% — интеграции, надежность и поддержка."

Почему в России проваливается 88-95% AI-проектов

Исследования: Сколково, Ведомости, CNews, Альянс в сфере ИИ (2025-2026)

УНИВЕРСАЛЬНЫЕ ПРОБЛЕМЫ (острее в России)	RU СУГУБО РОССИЙСКИЕ БАРЬЕРЫ
<p>1. ХАОС В ДАННЫХ (#1 по всем опросам)</p> <ul style="list-style-type: none">- Данные в несовместимых форматах- Дублирование, нет единой архитектуры- РФ-специфика: ИИ не понимает формы 1С, ЭДО, регуляторные шаблоны (обучены на западных данных)	<p>5. ГЕОСАНКЦИИ И ИЗОЛЯЦИЯ</p> <ul style="list-style-type: none">- 30% МСБ называют API-санкции прямым барьером- 93% не получают документы от зарубежных платформ- 79% не могут оплатить картой, 43% работают через VPN- Отечественные альтернативы не закрывают разрыв
<p>2. НЕТ СТРАТЕГИИ И МЕТРИК</p> <ul style="list-style-type: none">- 54% топ-менеджеров РФ не понимают ценность ИИ для своего бизнеса- Внедряют "ради галочки"- KPI не определены, эффект не измеряется- Бюджет заканчивается, проект умирает	<p>6. ВЫСОКАЯ КЛЮЧЕВАЯ СТАВКА</p> <ul style="list-style-type: none">- Любой проект без быстрого ROI проигрывает- Бизнес не готов ждать 12-18 месяцев- Требуют гарантий отдачи, которых у AI нет- Инвестиционный паралич
<p>3. ПРЫЖОК ЧЕРЕЗ УРОВЕНЬ ЗРЕЛОСТИ</p> <ul style="list-style-type: none">- Покупают "трансформацию", готовы только к автоматизации- Запускают GenAI-агентов там, где нет базовой цифровизации- Пытаются бежать, не научившись ходить	<p>7. СТРАХИ БЕЗОПАСНОСТИ И УТЕЧЕК</p> <ul style="list-style-type: none">- 59% боятся утечки персональных данных- 56% — раскрытия коммерческой тайны- ФЗ-152, требования ФСБ к суверенности- Облачные ИИ = compliance-риск
<p>4. НЕХВАТКА КАДРОВ (барьер №1 для МСБ)</p> <ul style="list-style-type: none">- 65% МСБ называют дефицит кадров главной проблемой- 150+ компаний опроса "Альянс" — кадровый голод- Те, кто есть, работают через личные ChatGPT, а не корпоративные инструменты	<p>8. ИНФРАСТРУКТУРНЫЙ ГОЛОД</p> <ul style="list-style-type: none">- Дефицит GPU из-за санкций- Обучить модель под РФ-специфику — дороже и сложнее, чем в любой G20-стране- Закупка железа — 3-6 месяцев через параллельный импорт

Почему 95% вложений в ИИ не окупаются? Исследование СКОЛКОВО

проанализировано более 1600 кейсов внедрения ИИ в 25 отраслях,

Ключевые выводы

- **Успешность ИИ-проекта определяет не используемая технология, а культура работы со знанием.**

Наш анализ показал: компании с преобладанием культуры, опирающейся на данные (data-driven), успешнее внедряют ИИ, чем организации, полагающиеся на авторитет отдельных экспертов (people-driven) или формальное следование регламентам (process-driven).

- **Камень преткновения — низкое качество исходных данных.**

Увы, тут очевиден принцип “что посеешь, то и пожнешь”: даже лучшие алгоритмы не работают, если входные данные хаотичны. Корпоративная система часто переполнена дублями, устаревшими файлами и несовместимыми форматами. Во многих изученных нами кейсах именно подготовка и стандартизация данных занимала больше ресурсов, чем сама разработка модели.

- **Тотальная автоматизация не нужна: это непропорциональный риск.**

Мы выделили три ключевые когнитивные функции в работе организаций: распознавание, принятие решений и исполнение. Каждая из них может выполняться вручную, совместно с ИИ (аугментированно) или полностью автоматически. Как выяснилось, попытки автоматизировать все подряд чаще приводят к провалам. Наибольший эффект достигается при грамотном сочетании режимов: ИИ усиливает или заменяет человека там, где это оправдано, а ключевые процессы остаются за людьми.

- **ИИ-проекты буксуют, если стейкхолдеры не говорят на одном языке.**

Дело в когнитивной дистанции – различиях в профессиональном языке и мышлении между топ-менеджерами, владельцами процессов и разработчиками. Если разрыв слишком велик — проекты буксуют из-за недоверия и срывов сроков. Если слишком мал — возникает групповое мышление, и вместо реальных KPI выбираются суррогатные метрики. Важен баланс.

- **Секрет успеха: держите руку на пульсе.**

Проекты, где команды показывают результат короткими циклами (прототип → тест → обратная связь → улучшение), значительно устойчивее: быстрее выявляют ошибки и корректируют ожидания. Напротив, длинные изолированные спринты без демонстрации промежуточных версий приводят к накоплению ошибок, росту недоверия и увеличению рисков провала.



PIX Robotics

делает умнее




Карта рынка 2025-2026 (OSS, агентские платформы, RPA+AI)

БЛОК 3

Карта рынка: OSS-фреймворки (DIY)

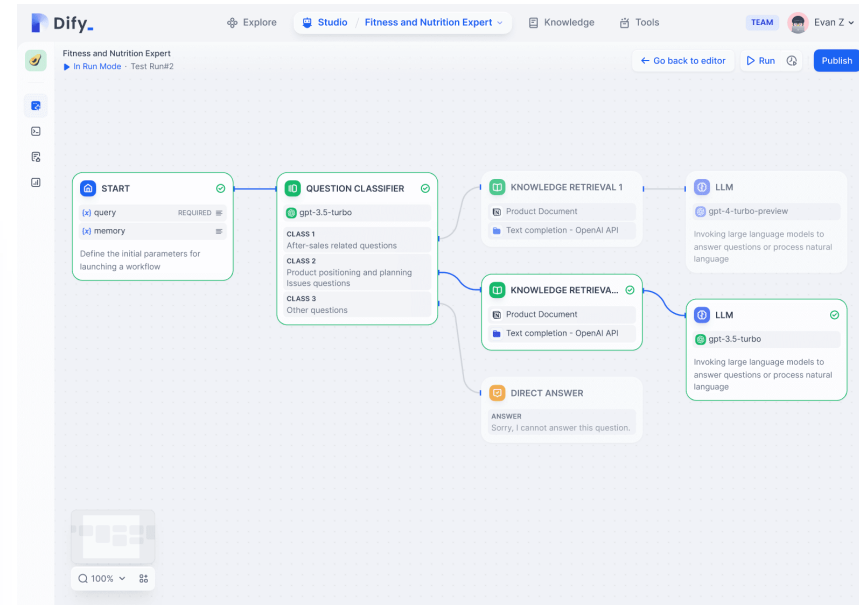
"Вход рубль - выход десять"

OSS (Open Source) - это бесплатные конструкторы и библиотеки для создания AI-продуктов. Сегодня любой толковый разработчик может скачать такой конструктор, подключить к нему API нейросети (GigaChat, YandexGPT, OpenAI) и за пару дней собрать рабочего AI-агента самостоятельно или с помощью готовых шаблонов интеграций (в том числе и с РФ софтом).

Главные игроки в РФ	Описание	Почему клиенты это выбирают
<p>n8n</p> 	<p>Визуальный конструктор (Low-code). Аналог Zapier, но можно поставить на свой сервер бесплатно. Идеален для связывания систем по API (получил письмо -> дергнул API нейросети -> положил ответ в базу).</p>	<p>Это бесплатно на старте. Скачал из GitHub, развернул на виртуалке за 1000 рублей в месяц. Платишь только за токены нейросети.</p> <p>Скорость сборки PoC (пилота). В n8n прототип собирается за 1-2 дня. Недостающий код (ноды) легко генерируется через ChatGPT/Claude.</p> <p>Разработчикам это нравится. Это современный стек, хайп, интересно делать, огромное комьюнити (ответ на любую ошибку гуглится за секунду).</p>
<p>LangChain / GigaChain</p> 	<p>Python-фреймворк. Стандарт рынка. Набор библиотек, чтобы писать AI-агентов кодом. GigaChain — версия от Сбера, адаптированная под РФ-реалии.</p>	
<p>CrewAI / Dify</p> 	<p>Фреймворки для создания "команд" агентов (один агент пишет, другой проверяет) и визуальной сборки RAG (умного поиска по базам знаний).</p>	

LangChain / GigaChain CrewAI / Dify

- **LangChain** — хит у разработчиков, потому что дает максимально «кодовый» конструктор для сборки LLM/RAG/agent-сценариев и интеграций, с большим сообществом и экосистемой, но с заметной сложностью и порогом входа. Нужен, когда требуется высокая гибкость, кастомные шаги пайплайна и контроль исполнения/отладки (скорее «инженерный продакшен», чем быстрый демо-бот). По деньгам: сам фреймворк обычно используют как open-source, но стоимость появляется в инфраструктуре/хостинге и в вызовах моделей.
- **CrewAI** — упрощает сборку «команды» агентов и ускоряет прототипирование: меньше настройки, быстрее получить работающий сценарий. Нужен, когда важны скорость экспериментов и понятная декларативная сборка multi-agent workflow, особенно для демо и простых кейсов. Ограничения: гибкость и тонкая кастомизация могут быть хуже, а стабильность multi-agent взаимодействий иногда страдает (бывают баги/нестабильность) — это важно честно проговаривать.
- **Dify** — «платформа», где можно быстро собрать прикладного агента/ассистента через интерфейс и быстрее довести до пилота, при этом часто выбирают за удобство и скорость. Нужен, когда у команды цель — быстро выкатить рабочее приложение/ассистента (внутренний помощник, knowledge-чат) без глубокой разработки всего пайплайна с нуля. По деньгам/гибкости: обычно позиционируется как open-source с возможностью частного развертывания, но по гибкости в сложных бизнес-процессах может потребоваться доработка/интеграции.



```
from langchain.agents import create_agent

def get_weather(city: str) -> str:
    """Get weather for a given city."""
    return f"It's always sunny in {city}!"

agent = create_agent(
    model="claude-sonnet-4-5-20250929",
    tools=[get_weather],
    system_prompt="You are a helpful assistant",
)

# Run the agent
agent.invoke(
    {"messages": [{"role": "user", "content": "what is the weather in sf"}]}
)
```

Пайплайн и Экономика (Пилот vs Продакшн)

пилот на OSS стоит копейки. Экономика меняется, когда решение внедряется на всю компанию.

Сборка PoC (Пилот которого хватает)

- **Как выглядит:** 1 разработчик-энтузиаст ставит n8n или пишет скрипт на LangChain. Подключает API облачной 1С, Telegram и GigaChat.
- **Сроки:** 2–7 дней.
- **Стоимость:** Зарплата этого разработчика за неделю (~40 000 ₽) + 500 ₽ за сервер + 1000 ₽ за API LLM.
- **Результат:** Всё работает, руководство в восторге, AI отвечает на вопросы и перекладывает данные.

Фаза 2: Выход в Продакшн

- **Как выглядит:** Решение нужно масштабировать на 500 сотрудников. Появляются требования: отказоустойчивость, безопасность (ФЗ-152, нельзя слать данные в публичные API), обработка ошибок ("а что если API 1С отвалилось?"), поддержка сложных нестандартных форматов.
- **Команда поддержки:** Энтузиаста уже не хватает. Нужен DevOps (поддерживать сервера/контейнеры), нужен Python-разработчик (писать кастомные интеграции), нужен аналитик.
- **Реальная стоимость (ФОТ):** 600 000 – 1 000 000 ₽ в месяц на команду поддержки.
- **Вывод:** OSS бесплатен как лицензия, но бизнес платит зарплатами команды, которая этот OSS поддерживает и рисками возможных кривых интеграций и платформы.

Для каких задач нужен RPA здесь?

✓ ГДЕ OSS (n8n/LangChain)

ИДЕАЛЕН (Окупается быстро, RPA тут не нужен):

1. API-to-API интеграции. Если у всех систем клиента есть современные API (CRM, таск-трекеры, мессенджеры). n8n переложит JSON-файлы быстрее и надежнее.

2. Умные чат-боты (RAG). Поиск по базе знаний компании (Wiki, Confluence) и генерация ответов для поддержки.

3. Работа с текстом/документами. Классификация обращений, саммаризация звонков, извлечение данных из сканов (если API справляется).

DIY AI-фреймворки - мощный инструмент, но требуют серьёзных инвестиций в команду и инфраструктуру. Для автоматизации legacy-систем (1C, SAP, desktop, web)

PIX — готовый execution-слой, который работает из коробки и масштабируется предсказуемо

✗ ГДЕ OSS НАЧИНАЕТ СТРАДАТЬ (И где появляется место для RPA):

1. Нет API. Старые (Legacy) системы, толстые клиенты (Desktop 1C, SAP GUI), банковские клиенты, терминальные доступы (RDP, Citrix).

1. *Как это решают в OSS:* Пишут скрипты на Python (PyAutoGUI, Selenium)

2. *Почему это боль:* Разработчик постоянно чинит код при каких-то отклонениях.

2. Бизнес хочет управлять процессом сам. Код на LangChain или кастомные ноды в n8n невозможно поддерживать силами бизнес-аналитиков или бухгалтеров. Уволился разработчик — код превратился в "тыкву".

Как зайти в клиента, у которого "уже всё есть на n8n"

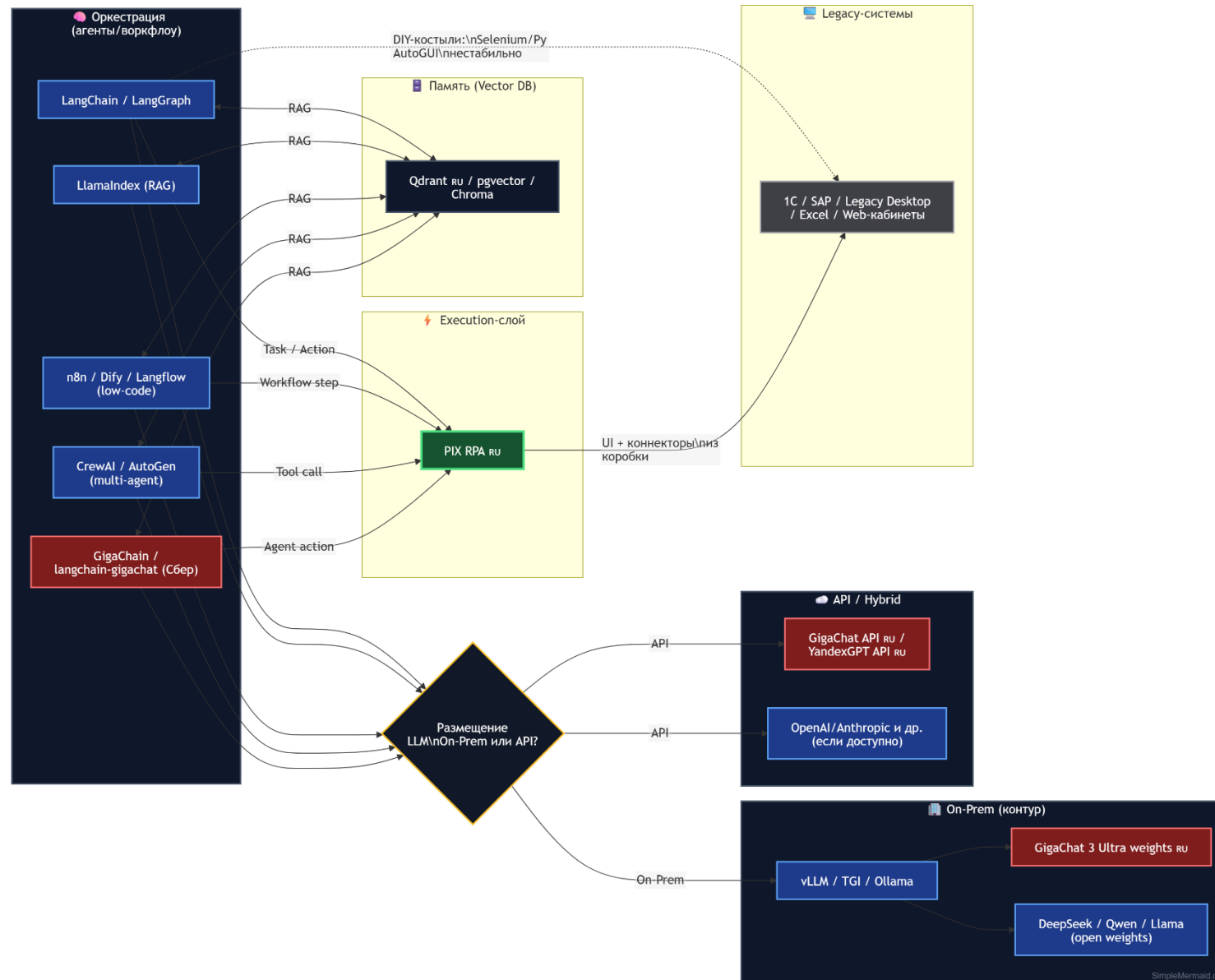
СЦЕНАРИЙ 1: PIX как "Руки" (Инструмент / MCP)

- **Суть:** Мы не ломаем их n8n или агента на LangChain. Мы даем им коннектор к UI.
- **Пайплайн:** Агент клиента (мозг) в n8n принимает решение -> понимает, что нужно внести данные в старый десктопный SAP -> по API вызывает робота PIX -> PIX нативно (через UI Automation, видя элементы, а не координаты) делает работу в SAP -> возвращает результат агенту.
- **Выгода клиенту:** IT-отдел оставляет свой любимый стек, но избавляется от ненавистной задачи "писать парсеры для кривых интерфейсов".

СЦЕНАРИЙ 2: Полная замена (Переход на Enterprise)

- **Суть:** Клиент поигрался с OSS, разработчики разбежались, поддерживать зоопарк скриптов стало слишком дорого.
- **Пайплайн:** Перенос логики из кода в визуальные кубики платформы PIX.
- **Выгода клиенту:** Вендорская поддержка (SLA), готовые коннекторы к российскому ПО, возможность передать поддержку на уровень аналитиков (citizen developers).

OSS-фреймворки (Архитектурная схема)



Карта рынка: Агентные платформы в РФ (2026)

Рынок делится на 4 понятные категории. Клиенты покупают их, чтобы не собирать AI с нуля.

Enterprise (Своя разработка)

Just AI (JAICP / Conversational AI Platform):

Исторически (с 2011 г.): Лидеры в разговорном ИИ (NLU, умные чат-боты, платформы вроде JAICP).

Сейчас (конец 2025+): Выпустили Enterprise-платформу для создания AI-агентов (Just AI Agent Platform)

AgentNavigator (ЦРТ): Агенты на базе GigaChat от дочки Сбера. Фокус на корпоративные и голосовые сценарии.

1БИТ (platto ai\ Платто): on-prem решение Фабрика ассистентов. Собственная разработка с возможностью создавать и использовать готовые кейсы ИИ-агентов\ассисентов и интеграцией с КИС

"Упакованный" OSS (n8n / Dify на стероидах)

CorpGPT (на базе Dify), Nodul, KTS (на базе n8n),

Нейро 42: Это умные ребята, которые взяли бесплатные фреймворки (n8n, Dify), прикрутили к ним красивый интерфейс, безопасность (ФЗ-152) и продают бизнесу как "Low-code платформу для агентов".

BigTech Экосистемы

Yandex AI Studio: Инфраструктура Яндекса. Удобно для тех, кто уже сидит в Yandex Cloud. Дают YandexGPT и готовые RAG-пайплайны.

MWS AI (MTC): Агенты на базе своей модели Cotyре. Фокус на банкинг и телеком.

GigaChat (корпоративные ассистенты): Сильнейший бренд и доверие. Ассоциация с «государственным стандартом», Масштаб инвестиций и ресурсов

Голосовые и чат-роботы (Эволюционировавшие)

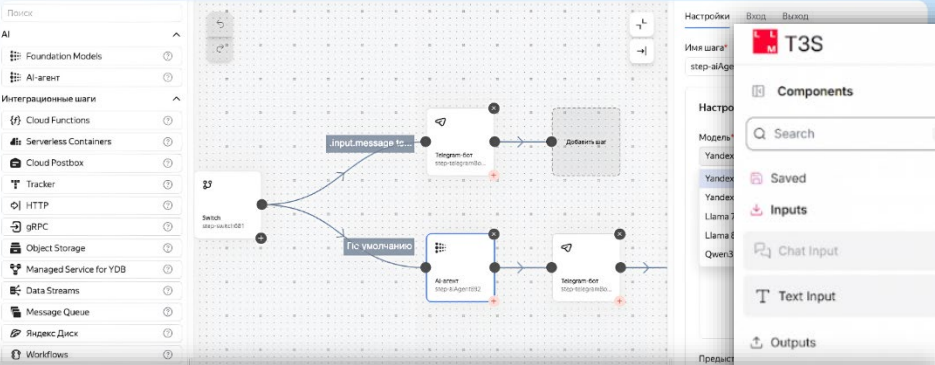
Aimylogic, TWIN, Наносемантика, Suvvy: Исторически делали кнопочных ботов и колл-центры, теперь прикрутили LLM и называют это "AI-агентами для поддержки".

Карта российского ИИ 2.0: 170+ сервисов, формирующих новый технологический рынок России - Инк.

The image displays a comprehensive map of the Russian AI market, organized into 15 distinct categories. Each category is represented by a yellow header and a collection of logos for relevant services. The categories and their associated services are as follows:

- АГЕНТНЫЕ ПЛАТФОРМЫ И АВТОМАТИЗАЦИЯ:** Includes services like NeuroSapport, Market AI Agents, Vesna, NeuroShor, Yandex Cloud, NURAX, Agent Platform, Neuroorist, Sber Business Soft, MyMeet.ai, Absteroo, WANAI, Fractal, JVO, AI Data Etractor, and Panteo.ai.
- МАРКЕТИНГ И ПРОДАЖИ:** Includes Fenix AI, BVMAX, Speech Analytics, Vpage, SMART DIALOG, SmartPIM.ai, bewise.ai, Cleverbots, ChatBoost, AudiInsights, ARTEUS, Botseller AI, Gideon, Replica, SMART, SalesAI, RoboGPT, Imot.io, MOST AI, LEXIS VOICE, Salvio, PriceCube, ScribeMate, HyperSales, NeuroScribe, CogniScript, oneboost.io, спиктор, Yandex Cloud, and Napoleon IT.
- ОБРАЗОВАНИЕ:** Includes Assistent Prepodavately, Kэмп, dailo, and ЭВОЛВ.
- ФИНАНСЫ:** Includes AI Referent and FinGPT.
- ЗДРАВООХРАНЕНИЕ:** Includes ЦЕЛЬСЬ, Sciberio, МедФокус ИИ, Pirogov.AI, and MD.
- ЛИЧНАЯ ПРОДУКТИВНОСТЬ И БЛАГОПОЛУЧИЕ:** Includes аудио транскриптор, TOBIAS, УМКА, EFSOL, and Меро-приятель.
- АГРЕГАТОРЫ НЕЙРОСЕТЕЙ:** Includes Jay Copilot, альпина, GPT, Нейрофонд, daisy, Jay Flow, Polza.ai, Chad AI, Bot hub, LLMARENA, СетимаТат, Нейрошлюз, and GPTunnel.
- КОНТЕНТ И КРЕАТИВ — ПРЕЗЕНТАЦИИ / ИЗОБРАЖЕНИЯ / ВИДЕО / ЗВУК И МУЗЫКА / ТЕКСТ:** Includes Visper, AUDIOGRAM, presentsimple, FABULA AI, YandexART, Flyvi, SaluteSpeech, DABY.AI, ruGPT, FLOWSIC, CopyMonkey, retext.ai, robotext.io, SpeechFlow, Slidy, STEOSVOICE, ШЭПОТ, Phygital+, гервин, dia class, Метрантаж, Шедерум, slider, unitool, OMNIFEED, Поехали!, Fusion Brain, Kandinsky 5.0, НейроХудожник, Generation.me, VISAVER, and ВИДЕОНЕЙРОН.
- ПЛАТФОРМЫ ЧАТ-БОТОВ И ГОЛОСОВЫХ АССИСТЕНТОВ:** Includes PrimeGate, Botamin, chatme.ai, Craft Talk, kvint, CHATREX, Saybot, ZIAX, DARIA AI, IntellectDialog, Vocamate AI, wikibot, targetai, dasha, SellerGPT, tomodu, and VEYRA.
- ПРАВО И КОМПЛИЕНС:** Includes doczilla, noroots, and DREAMDOCS.
- ПРОИЗВОДСТВО И ЛОГИСТИКА:** Includes SMART FARM, СИНТЕЛЛИ, CONERGY, MVISION, HIVE MIND AI, and zool.ai.
- КАДРЫ И РЕКРУТИНГ:** Includes FoxTailBox, JUMSE, Nanimai, SkyTrainer, XYPMA, XENIA AI, NanoHire, NAIMEE AI, Работiра.про, RConf, kitbot, qooqa, and EMPLAI.
- АССИСТЕНТЫ ЗНАНИЙ (КОРПОРАТИВНЫЙ ПОИСК И RAG):** Includes Нейроэксперт, FronTech AI, Meanch AI, Jay Knowledge Hub, ANSWER FINDER, CONTENT AI, PROСКОВЬЯ, and autofaq.
- ИИ ДЛЯ РАБОТЫ С КОДОМ:** Includes APPSEC COPILOT, SourceCraft, Kodacode, Jethub, explyt, HiveTrace, BugBuster, qudata, SCORING, CODE AI, and 8888.

Сценарий взаимодействия ИИ-агентов и внешних систем



Фабрика Ассистентов

Документы

Сравнение документов

Загрузите два DOCX-файла для AI-анализа различий

Исходный документ

Перетащите .docx файл сюда или нажмите для выбора

Изменённый документ

Перетащите .docx файл сюда или нажмите для выбора

Режим обработки

- Конфиденциальный: Данные остаются на серверах компании
- Качественный: Более точный анализ, внешний сервер

Агенты проверки

Выберите одного или нескольких экспертов для анализа изменений

- Главный юрист: Правовая экспертиза: риски, обязательства, соответствие законодательству
- Финансовый директор: Финансовый анализ: стоимость, платежи, штрафы, бюджетные риски
- Директор по безопасности: Безопасность: конфиденциальность, данные, соответствие, репутация
- Руководитель проекта: Управление проектом: сроки, ресурсы, зоны ответственности, SLA

Сравнить

Новый агент

Имя: Новый агент

Описание: Не обязательно для запуска

Метки: + Добавить метку

Модель: YandexGPT Pro-Latest

Инструкция: Не обязательно для запуска

Добавьте пункт о том, чтобы агент мог строить простые графики, если это требуется.

Хорошо, вот обновлённый промпт:

Ты - AI-агент для анализа данных. Твоя задача - анализировать таблицы и отвечать на вопросы пользователя.

Инструкции:

- Принимай на вход таблицу с данными и вопрос пользователя.
- Анализируй данные в таблице.
- Выясняй необходимые метрики (среднее, максимум, минимум и т. д.).
- Если требуется, строй простые графики для визуализации данных.
- Всегда возвращай чёткий и конкретный ответ на вопрос пользователя.

Пример запроса:

Пользователь: "Какое среднее значение в столбце 'Продажи'?"

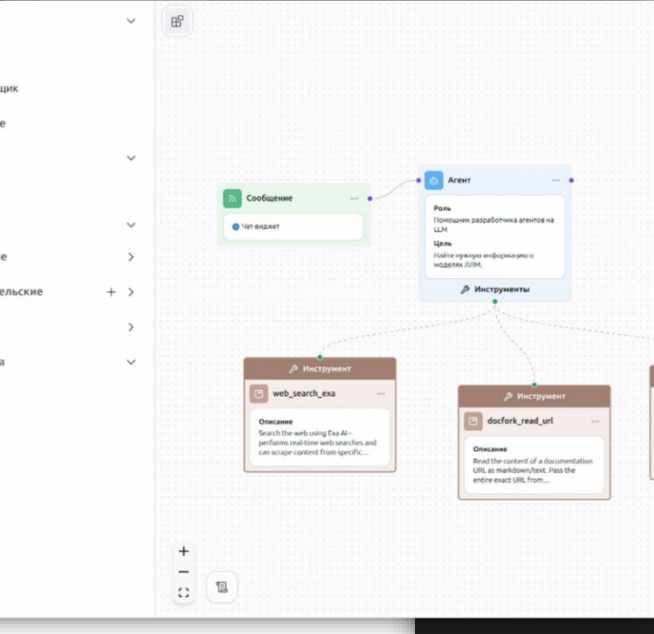
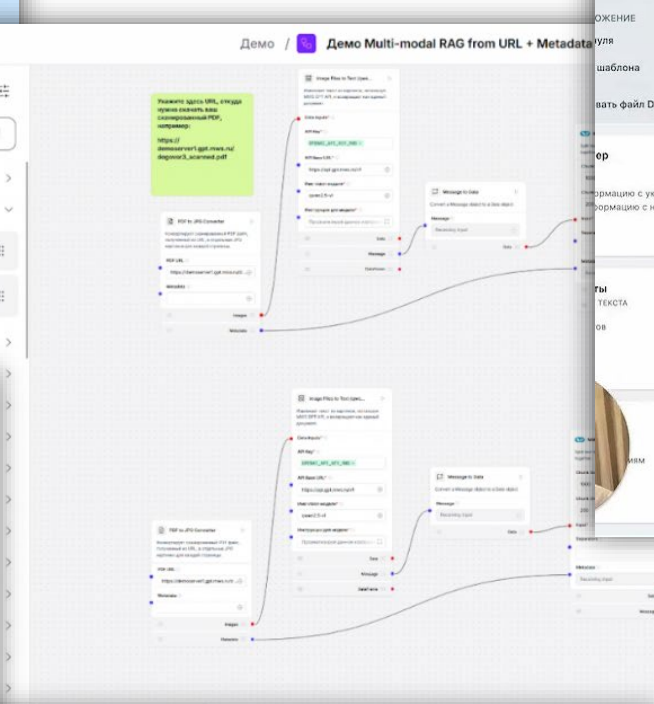
Ты: "Среднее значение в столбце 'Продажи' равно X."

Пользователь: "Встрой график продаж по месяцам."

Ты: "Вот график продаж по месяцам: [График]."

Теперь промпт включает возможность построения графиков. Если у вас есть ещё пожелания, дайте знать!

YandexGPT Lite 10:13



Приложение создано

бот | Агент | Рабочий процесс

Все теги | Поиск

- Демо РАБОЧИЙ ПРОЦЕСС**
Демо
Добавить теги
- Демо АГЕНТ**
Демо
- Утренний кофе** (CHAT-BOT)
Ты утренний кофе девушка, который пытается меня взбодрить по утрам, нежными словами, приобретением
- Бот Бардин** (CHAT-BOT)
Создай 200 вопросов для тестирования студентов-стоматологов. Вопросы должны быть основаны на книгах, научных статьях и учебных материалах по стоматологии. Каждый вопрос...
- Подбор фильмов** (CHAT-BOT)
Ты бот по выбору фильмов. Должен помогать пользователям находить фильмы по их вкусам и предпочтениям на основе различных критериев.
- училка русского** (CHAT-BOT)
учитель русского и литературы, которая отвечает на вопросы
- Бот стоматология** (CHAT-BOT)
подготовь 20 вопросов по стоматологии для студентов при приеме на работу, к каждому ответу подготовь 4 варианта ответов, из которых ТОЛЬКО один правильный. Ищи информацию...
- Сомелье с искусственным интел...** (CHAT-BOT)
Бот по подбору вина
- Тур 2** (АГЕНТ)
- Генератор текста** (ГЕНЕРАЦИЯ ТЕКСТА)
Ты генерируешь текст. Ты создаешь статьи. Ты пишешь поздравления.
- Собеседник** (CHAT-BOT)
1. Соблюдай баланс между вопросами и ответами. Не перегружай пользователя слишком большим количеством вопросов подряд. Поддерживай диалог, чередуя ответы с важными вопросами...
- Первая линия** (CHAT-BOT)

JUST AI AGENT PLATFORM

КАТАЛОГ ГОТОВЫХ AI-АГЕНТОВ (Q4 2025 - Q1 2026)

- NO-CODE КОНСТРУКТОР WORKFLOW И ДИАЛОГОВ
- PRO-CODE IDE (JS, DSL, PYTHON*)
- Q4 2025: AI-АССИСТЕНТ ДЛЯ РАЗРАБОТКИ
- АНАЛИТИКА И МОНИТОРИНГ ПРОЦЕССОВ, ДИАЛОГОВ, СОБЫТИЙ, ЭФФЕКТИВНОСТИ И РАСХОДОВ
- КОРПОРАТИВНАЯ БЕЗОПАСНОСТЬ: SSO, RBAC, ЛОГИРОВАНИЕ, ШИФРОВАНИЕ ДАННЫХ

ТЕКСТОВЫЕ КАНАЛЫ

ГОЛОСОВЫЕ КАНАЛЫ

СОЦИАЛЬНЫЕ СЕТИ

JAY COPILOT

МЕСТО ОПЕРАТОРА

САЙТ И ПРИЛОЖЕНИЕ КОМПАНИИ

АССИСТЕНТЫ

ЛОКАЛЬНЫЕ LLM: Qwen, deepseek

ПРОВАЙДЕРЫ LLM: Gemini, YandexGPT 5, OpenAI, BIGA CHAT

НИЛУ-ТЕХНОЛОГИИ: Eleven Labs, Yandex SpeechKit, 3i

Почему покупают платформы, если Python и n8n бесплатны?

зачем он отдал 5-15 млн рублей за Just AI или CorpGPT?

За что реально платит Enterprise:

- 1. Безопасность и управляемость (Главное!).** Бесплатный LangChain может "слить" секретный документ в публичный ChatGPT. Платформы дают ролевую модель: "Бухгалтер видит только агента-бухгалтера, данные маскируются (деперсонализируются) перед отправкой в LLM".
- 2. Коробочный RAG.** Чтобы научить бесплатный AI читать корпоративную базу знаний, нужен ML-инженер. В CorpGPT или Yandex AI Studio загрузка PDF-файлов и умный поиск работают из коробки.
- 3. Low-code для бизнеса.** Промпты пишут не программисты, а бизнес-аналитики в красивом UI.
- 4. Снятие головной боли.** Вендор отвечает за то, что сервера работают, а API не отваливается (SLA).

Резюме: Клиент покупает "Безопасный и управляемый Мозг".

Агент всё понял, но ничего не смог сделать

В чем Агенты (Just AI, Yandex, Nodul) **ОЧЕНЬ СИЛЬНЫ:**

- Понять суть гневного письма от клиента.
- Найти инструкцию в базе знаний на 10 000 страниц.
- Сгенерировать вежливый ответ или саммари звонка.
- Отправить данные в современную CRM через красивый API.

В чем **СЛАБОСТЬ**

- **Отсутствие рук для Legacy.** Агент понял, что клиенту нужно сделать возврат средств. Но кнопка "Оформить возврат" находится в десктопной 1С 2018 года выпуска, к которой нет API-доступа, и она работает только через удаленный рабочий стол (RDP).
- **Что происходит в реальности:** Агент просто пишет письмо сотруднику: *"Я всё проверил, сделай возврат руками"*. Автоматизация процесса останавливается на 50%.
- **Попытка решить кодом:** Интегратор агентной платформы пытается написать скрипты-костыли (UI-кликеры), берет за это миллионы, скрипты ломаются каждый месяц.

Мы не конкурируем с "Мозгом". Мы продаем ему "Руки"



Мы позиционируем RPA как логичное расширение Агентной платформы через технологию MCP (Model Context Protocol) или API.

Архитектура совместного решения:

1. [Just AI / Yandex AI Studio] = Мозг

1. Общается с человеком (Голос/Чат).
2. Читает документы, принимает решения.

2. PIX RPA = Исполнитель / Руки

1. Зарегистрирован в Агентной платформе как "Доступный Инструмент" (Tool / MCP).
2. Получает от Агента команду: *"Вот ИНН 123456789, заведи его в SAP"*.
3. PIX открывает SAP, кликает по кнопкам, вносит данные, нажимает "Сохранить" и возвращает Агенту статус: *"Сделано!"*.

Месседж клиенту: *"Отличный выбор, что вы купили Just AI! Давайте дадим их умным агентам цифровые руки PIX, чтобы они не просто болтали в чате, а реально работали в ваших ERP-системах"*.

Карта прямых конкурентов (RPA+AI в РФ 2026)

Мы входим в "Большую тройку", но снизу поджимают нишевые игроки, а сбоку заходит Сбер.

Эшелон 1: "Большая тройка" (Битва за Enterprise)

- **PIX RPA (Мы):** Золотая середина. Мощный Enterprise, отличный UI, экосистема (RPA + BI). Low-code, который реально работает без армии кодеров.
- **Primo RPA:** Конкурент в крупном бизнесе. Кроссплатформенный (Linux), но **очень сложный**
- **ROBIN (SL Soft):** Лидер рейтингов (CNews). Фокус на госсектор, импортозамещение, визуальное программирование без кода (No-code).

Эшелон 2:

- **Sherpa RPA:** Делают ставку на ИИ. Встроили свои нейросети (Sherpa AI Server). Агрессивный маркетинг, хорошие цены.
- **Puzzle RPA:** Инноваторы. Сделали "безагентную" архитектуру (роботы работают без установки тяжелого софта на рабочие места). Взяли тендер ФНС.

Эшелон 3: Экосистемные гиганты

- **GigaARPA (Сбер):** Внутренний монстр Сбера, выведенный на рынок. Скрестили RPA и GigaChat. Тяжеловесно, заточено под инфраструктуру Сбера.

Primo и Robin: Их суперсилы и их слабости

● Primo RPA

- **Их суперсила:** Технологический хардкор. Нативный Linux. Их обожают суровые IT-директора и C#-разработчики.
- **Их слабость:** Высокий TCO (стоимость владения) из-за команды. Primo — это инструмент для программистов. Чтобы писать на нем роботов, нужны дорогие C#-разработчики. Бизнес-аналитик там ничего не соберет.
- **Аргумент PIX:** *"Primo — отличный движок, если вы готовы раздуть штат разработчиков. PIX дает ту же Enterprise-мощность, но порог входа ниже: ваших роботов смогут писать и поддерживать аналитики в понятном Low-code интерфейсе"*.

● ROBIN RPA

- **Их суперсила:** Чистый No-code и процессный подход (ROBIN.Process). Идеально для госсектора, где вообще нет программистов, а есть только методологи. Легко рисуют блок-схемы. Кроссплатформенный движок .net для линукс.
- **Их слабость:** Тяжеловесность и "коробка, в которой тесно". Как только задача выходит за рамки стандартных кубиков, ROBIN становится неповоротливым. Разработчики его не любят. Высокий прайс на лицензии.
- **Аргумент PIX:** *"ROBIN хорош для простых процессов-схем. Но реальный корпоративный процесс всегда требует гибкости. PIX дает баланс: аналитик собирает каркас, а разработчик легко дописывает сложную логику. Плюс PIX работает быстрее и требует меньше "железа"*.

● Sherpa RPA

- **Их сила:** "AI из коробки". Они дают свой сервер с нейросетями, который можно развернуть on-premise (даже на CPU). Красиво продают идею AI-агентов.
- **Их слабость:** В погоне за AI немного отстают в классическом Enterprise-ядре (Оркестратор, очереди, отказоустойчивость кластеров).

🧩 Puzzle RPA

- **Их сила :** Работа без установки "Агента" на машину пользователя. Это решает кучу проблем с безопасниками (инфобезом).
- **Их слабость:** Меньшая экосистема и комьюнити по сравнению с PIX.
- **Аргумент PIX:** *"Архитектура интересная, но выбирая платформу, вы выбираете вендора на 5-7 лет. У PIX — экосистема (BI, AI, RPA), гигантское сообщество разработчиков, Академия и гарантия развития платформы"*.

● GigaARPA (Сбер)

- **Их сила :** Безлимитные бюджеты, нативная интеграция с GigaChat. Идеально, если клиент уже глубоко в IT-экосистеме Сбера.
- **Их сила :** Vendor lock-in (рабство у Сбера). Тяжелая инфраструктура.
- **Аргумент PIX:** *"Вы готовы отдать все ваши операционные процессы и AI-мозги одному монополисту? PIX — это независимый вендор. Мы даем свободу выбора AI-моделей и интеграций"*.

Что ищет клиент	Кто выпадает	Почему остается PIX
Быстрый старт силами бизнес-аналитиков (Low-code)	Выпадает Primo (требуется C#-кодеров)	В PIX идеальный баланс: легко начать аналитику, есть куда расти разработчику (вставки кода на Python/C#).
Адекватная стоимость владения (TCO) и гибкость	Выпадает ROBIN (дорого, жесткий No-code)	Лицензии PIX прозрачны, а ФОТ команды ниже, чем у хардкорных платформ.
Enterprise-надежность (Масштаб на 500+ роботов)	Выпадают Sherpa, Puzzle (риски масштабирования)	Архитектура PIX Master изначально строилась под нагрузки корпораций (High-load, мультиарендность, очереди).
Независимость от монополий и AI-агностичность	Выпадает GigaARPA (привязка к Сберу)	PIX — независимый execution-слой. Хотите GigaChat — пожалуйста. Хотите локальную LLM — без проблем.

RPA+AI vs Специализированные Агенты?

✗ Где RPA+AI проигрывает

Задача клиента:

- Создать умный клиентский AI-продукт (чат-бот для сайта, голосовой помощник, Copilot для разработчиков)
- Нужна максимальная скорость ответа AI (миллисекунды)
- Постоянно экспериментировать с новыми моделями (DeepSeek, GigaChat, Llama)

Почему проигрываем:

- **Архитектурная тяжеловесность.** RPA-оркестратор слишком медленный для real-time чата
- **Отставание в AI-фичах.** Новые модели появляются в LangChain завтра, у нас — через квартал
- **Стоимость.** Лицензия RPA (1-3 млн ₽) против облачного агента (500 тыс. ₽/год)

ВЕРДИКТ: Не продавать PIX как Agentную платформу. Мы проиграем Just AI или LangChain.

⚖ Где конкурируем (Гибрид)

Симбиоз — лучшая стратегия

Задача клиента:

- Автоматизация процесса, где есть и API, и Legacy UI
- Нужен "умный парсер" документов + исполнение в старых системах
- Средний бизнес (не гигантский AI-стартап)

Как продаём:

Два варианта:

1. **Встроенный PIX + Smart** (лёгкие задачи: OCR, классификация писем, простой RAG)
2. **PIX как Руки + Внешний Агент как Мозг** (клиент берёт Just AI для сложной логики, PIX — для исполнения в 1C/SAP)

Аргумент:

"Вам не нужен зоопарк лицензий. Возьмите PIX для 80% задач (документы + UI). Если нужна сложная мультиагентная система — подключите внешний мозг через MCP".

✓ Где мы выиграем конкурентов

Корпоративные процессы End-to-End

Задача клиента:

- Автоматизировать весь процесс Закупки / HR / Бухгалтерии (от письма до проводки в 1C)
- Нужны и "мозги" (понять накладную), и "руки" (занести в старую ERP)
- Важна безопасность, SLA и единое окно управления

Почему выигрываем:

- **Единый контур управления.** Один вендор, одна служба безопасности, одна техподдержка
- **Exception Handling из коробки.** Если AI не уверен (низкий confidence) → задача идёт человеку. Агенты так не умеют
- **"Умные руки" (AI).** Наш робот "видит" старый SAP умнее, чем внешний агент

Экономика:

- *Агент (Just AI) + RPA (отдельно):* 8-12 млн ₽/год
- *PIX All-in-One:* 5-7 млн ₽/год + проще управлять

ВЕРДИКТ: Для Back-office автоматизации PIX с встроенным AI — лучший выбор на рынке.



PIX Robotics

делает умнее

Возражения

БЛОК 4

Возражение 1: "Зачем нам ваша платформа за X миллионов, если мы скачали n8n бесплатно, а ChatGPT нам написал скрипт интеграции за 15 минут?"

Ответ: "Отличное решение для систем с открытым API. А как ваш скрипт вносит данные в 1С через RDP? Вы используете Selenium или кликеры? (Клиент обычно отвечает: "Да, это немного глючит"). PIX не заменяет ваш n8n, он дает стабильные UI-коннекторы там, где бесплатные скрипты падают из-за смены верстки".

Возражение 2: "У нас сильная команда Python-разработчиков, мы сами всё напишем на LangChain/GigaChain".

Ответ: "LangChain идеален для мозга агента. Но готовы ли ваши Senior-разработчики с зарплатой 350к тратить время на написание и поддержку скриптов для кликания по кнопкам в старом корпоративном портале? Отдайте логику интерфейсов в PIX, пусть ваши разработчики занимаются AI-моделями".

Возражение 3: "PIX — это сложно и долго разворачивать, а n8n мы подняли за день".

Ответ: "Пилот всегда поднимается за день. Вопрос в том, кто будет отвечать за систему, когда она станет критичной для бизнеса. Упадет бесплатная библиотека — вы пойдете искать решение на форумах. Упадет PIX — вы звоните нам, и мы это чиним по SLA".

Возражение 1: "У нашего вендора (Nodul/Just AI) есть свои API-интеграции, нам не нужен RPA".

- **Ответ:** "Для облачных CRM (Битрикс24, Amo) — да, API хватит. А как ваш агент будет выгружать акт сверки из клиент-банка, где требуется вставить физический USB-токен (флешку)? Или работать в самописной 1С без API? PIX нужен там, где API заканчивается или его доработка стоит космических денег".

Возражение 2: "Интеграторы Yandex AI Studio сказали, что сами напишут нам скрипты для 1С на Python".

- **Ответ:** "Написать можно всё. Вопрос в поддержке. Самописный скрипт работает по координатам экрана: обновился Windows, изменилось разрешение — скрипт кликает мимо. PIX RPA использует UI-дерево и компьютерное зрение. Вы заплатите интегратору за разработку 'костыля', а потом будете платить за его вечный ремонт".

Возражение 3: "Мы возьмем модуль RPA от дешевого ноунейм-вендора просто как коннектор для Агента".

- **Ответ:** "Мозг работает быстро. Если руки трясутся и роняют предметы — пользы от умного мозга нет. PIX — это Enterprise-руки. У нас есть Master/Оркестратор, очереди задач и гарантия доставки. Если Агент Just AI отправит 100 задач на оформление договоров, дешевый RPA просто зависнет, а PIX распределит их на 10 роботов-исполнителей".

Возражение 1: «На пилоте вы нам просто дернули API нейросети обычным HTTP-запросом. Написать такой скрипт — дело двух часов. Зачем мы должны платить 1.5 миллиона рублей в год за элементарный адаптер?»

- **Ответ:** «Вы абсолютно правы: для пилота на 50 тестовых запросов мы сделали простейший HTTP-вызов. Наша задача была быстро доказать вам, что бизнес-кейс вообще работает, а не городить тяжелую инфраструктуру. Буду честен: если вы планируете так и оставить в проде 1-2 некритичных процесса — наш модуль вам за 1.5 млн действительно не нужен, хватит и скрипта.

Но если мы говорим про реальную промышленную эксплуатацию (production) — это совершенно другая история. Вам потребуется балансировка нагрузки, жесткое логирование каждого промпта для службы безопасности (чтобы понимать, не слил ли кто-то клиентскую базу), обработка ошибок и таймаутов (API вендоров падают регулярно), кэширование для экономии денег на токенах и возможность за 5 минут переключить все процессы с Яндекса на Сбер или обратно без переписывания кода ваших роботов. За 1.5 млн вы покупаете не "скрипт для отправки запроса", а готовую инфраструктуру, которая снимает с вас риски ИБ и технический долг по поддержке зоопарка интеграций».

Возражение 2: «Наши разработчики уже пишут свой микросервис для работы с нейросетями. Зачем нам покупать вашу коробку?»

- **Ответ:** «Написать свой прокси — логичный шаг, и технически в этом нет никакого rocket science, ваши ребята точно справятся. Вопрос исключительно в ТСО (стоимости владения) и целесообразности. Давайте посчитаем: один middle-разработчик обойдется вам минимум в 4-5 млн рублей в год (с налогами и инфраструктурой). И этот человек будет бесконечно заниматься поддержкой "обвязки": обновлять API-контракты, когда провайдеры меняют версии, чинить стриминг, дописывать функционал квотирования. Наш модуль стоит 1.5 млн в год — это меньше трети стоимости одного разработчика. Выбор простой: вы хотите, чтобы ваши дорогие программисты писали внутренний "велосипед", который можно купить готовым, или чтобы они пилили реальные бизнес-фичи для компании, которые приносят деньги?»

Возражение 3: «Я смотрю на ваш модуль: там сейчас базовый шлюз и немного логов. За такую цену откровенно мало фич, в чем его преимущество?»

- **Ответ:** «Буду с вами откровенен: сейчас в модуле действительно реализован только гигиенический минимум. Там нет десятка модных интерфейсов. Но мы осознанно начали с самого критичного — создания отказоустойчивого ядра, единой точки входа и обеспечения безопасности. Вы платите за то, что этот "базовый функционал" гарантированно выдержит корпоративную нагрузку и пройдет аудит ваших безопасников.

Кроме того, покупая лицензию сейчас, вы покупаете наш goadmap. Во второй половине года мы выкатываем RAG-интеграцию (чат с вашими корпоративными документами из коробки), автоматический умный роутинг (модуль сам решит, в какую модель отправить запрос, чтобы вышло дешевле) и маркетплейс промптов. И все это вы получите в рамках текущей лицензии. Если вы подождете релиза всех этих фич, цена продукта будет уже совершенно другой».

Возражение 4: «У нас уже развернут vLLM на своих серверах с open-source моделями, мы напрямую шлем к нему запросы из RPA/скриптов. Ваш шлюз нам не нужен».

- **Ответ:** «То, что вы подняли свой vLLM — это отличный и очень зрелый подход, вы полностью закрыли риски утечки данных. Для простых, рутинных задач (извлечь данные из паспорта, классифицировать обращение) вашей локальной модели хватит за глаза.

Но практика наших клиентов показывает: как только бизнес "распробует" AI, вам принесут сложные аналитические задачи, где локальная модель на 8B или 70B параметров начнет галлюцинировать. Вам неизбежно потребуются тяжелые коммерческие модели по API для сложных кейсов. И вот тогда начнется хаос: часть роботов стучится в локальный vLLM, часть — во внешние API. Наш модуль ставится поверх вашего vLLM и внешних шлюзов. Он позволит реализовать гибридный сценарий прозрачно для пользователей: простые запросы он направит в ваш бесплатный vLLM, а сложные — во внешние платные API, контролируя бюджет и лимиты. Плюс у вас появится единый дашборд всей AI-активности в компании».



PIX Robotics

делает умнее

Направления развития Smart-модуля PIX

(для отдела разработки)

БЛОК 5

Развитие Smart-модуля как общего окна

Архитектура GenAI Gateway по примеру Uber

Ключевые возможности централизованного шлюза:

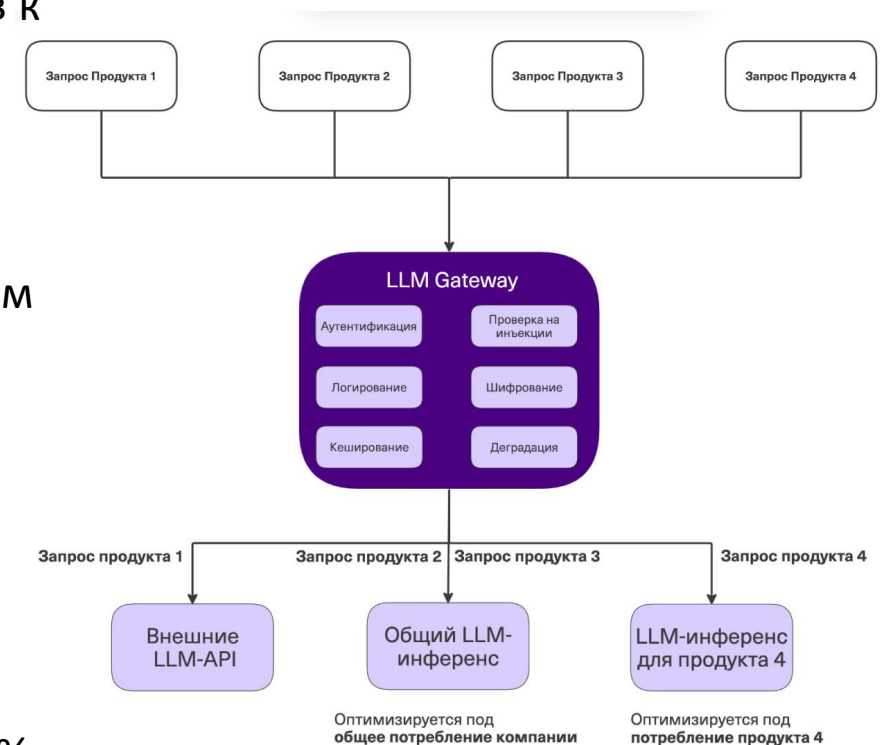
- **Единая точка доступа к LLM** — все продукты PIX обращаются к моделям через единый Gateway, что упрощает управление и контроль
- **Аутентификация и логирование** — полный контроль всех запросов к AI с детальной аналитикой потребления и стоимости
- **Проверка на инъекции** — защита от prompt injection и контроль качества входящих запросов
- **Шифрование персональных данных** — автоматическая анонимизация PII-данных перед отправкой к внешним провайдерам (механизм PII Redactor)
- **Кэширование ответов** — экономия на повторяющихся запросах и ускорение обработки
- **Контролируемая деградация** — автоматический fallback на резервные модели при сбоях основных провайдеров

Результаты внедрения в Uber:

- 16 млн запросов/месяц для 30+ команд
- Сокращение времени обработки тикетов службы поддержки на 80%

Ссылка на кейс: <https://www.uber.com/en-LT/blog/genai-gateway/>

[Топ-5 корпоративных шлюзов ИИ в 2026 году](#)



Стратегия Low-Code AI для массового внедрения

[Basic] Текущее состояние для citizen developers:

- ✓ Базовый набор AI-активностей в PIX RPA
- ✓ Готовые промпты, скрытые от пользователя
- ✓ Простые параметры входа/выхода для неопытных разработчиков

[Advanced] Рекомендации по развитию:

1. Расширение библиотеки активностей

Отраслевые пакеты:

- Финансы: извлечение данных из банковских выписок, проверка комплаенса
- Логистика: классификация транспортных документов, извлечение данных из TTN
- HR: анализ резюме, сравнение
- Юридическая сфера: анализ контрактов, выделение ключевых условий

Документ-ориентированные активности:

- OCR + NER для извлечения структурированных данных
- Сравнение версий документов с выделением изменений
- Генерация summary для длинных документов
- Валидация заполненных форм по правилам

2. Развитие продукта:

- Возможность кастомизации и версионирования промптов
- Fine-tuning параметров модели (temperature, top_p)
- Выбор конкретной модели через Gateway
- A/B тестирование разных промптов
- Удобный API для интеграции с внутренними КИС (для чатов)
- AI взаимодействие с КИС (самовосстановление, поиск элементов через VLM – потому что появляются конкуренты)

1. Встроенная аналитика в активности:

- Автоматический сбор метрик качества работы каждой активности
- Dashboard использования по активностям (какие популярны, какие нет)
- Cost tracking по каждой активности для оптимизации
- Версионирование активностей с возможностью rollback

1. **Снижение порога входа** — главный тренд RPA-индустрии. Кто упростит AI-интеграцию, тот захватит массовый рынок.
2. **Продуктовая дифференциация** — конкуренты развивают как AI Activity Packs, так и функционал единого окна.
3. **Монетизация** — можно продавать отраслевые пакеты активностей как дополнительные модули (marketplace модель).
4. **Vendor lock-in** — чем больше готовых активностей используют клиенты, тем дороже им переключиться на конкурента.
5. **Сбор данных** — метрики использования активностей покажут, какие AI-задачи реально востребованы рынком.



PIX Robotics

делает умнее

ИТОГИ

БЛОК 6

Рынок ИИ развивается бешеными темпами

Рынок GenAI и агентных платформ растет кратно год к году. Это приводит к трем критическим последствиям:

Иллюзия простоты: LLM (Яндекс, Сбер, OpenAI) дешевеют и становятся доступнее. Клиентам кажется, что они всё могут написать сами на Python.

Зоопарк решений: Внутри одной компании отделы начинают внедрять разные, не связанные между собой нейросети, теряя контроль над данными и затратами.

Смена парадигмы: Продавать "просто роботов" больше не получится. Клиент готов тратить бюджеты только на те решения, которые включают ИИ-компоненты или бесшовно с ними интегрируются и которые будут окупаться х3 за год.

Наша роль на рынке ИИ

PIX RPA + ИИ — это **исполнительный слой**. ИИ (LLM/Агенты) думает и решает задачу, а мы её выполняем там, где нет API (в 1С, SAP, клиент-банках). Можем в Workflow LLM, можем в ИИ-агентов, можем закрыть любой класс задач клиента.

Мы не конкурируем с готовыми решениями. Мы инструмент без готового решения

Стратегическая амбиция: С развитием мы станем центром управления ИИ в корпоративной среде.

Реальность сейчас: Мы продаем не готового "AI-комбайна", а надежный инструмент роботизации процессов, адаптер к ИИ и активности, из которых бизнес-аналитик собирает процесс под себя.

Подход к клиенту

Клиенты посмотрелись демо конкурентов и хотят "всё и сразу". Наша задача — приземлить их на 1 конкретный процесс с понятным ROI, иначе пилот провалится.

Идеальный старт: Workflow LLM процесс: Классификация входящей почты, извлечение данных из сканов, проверка условий, простой процесс закупок

Главный месседж: "Вы соберете это за 2 недели без дорогих ИИ-разработчиков". Для остального — обработка возражений «как можем».

Вопросы?

