

**podlodka.**

AI Crew

16–20 февраля

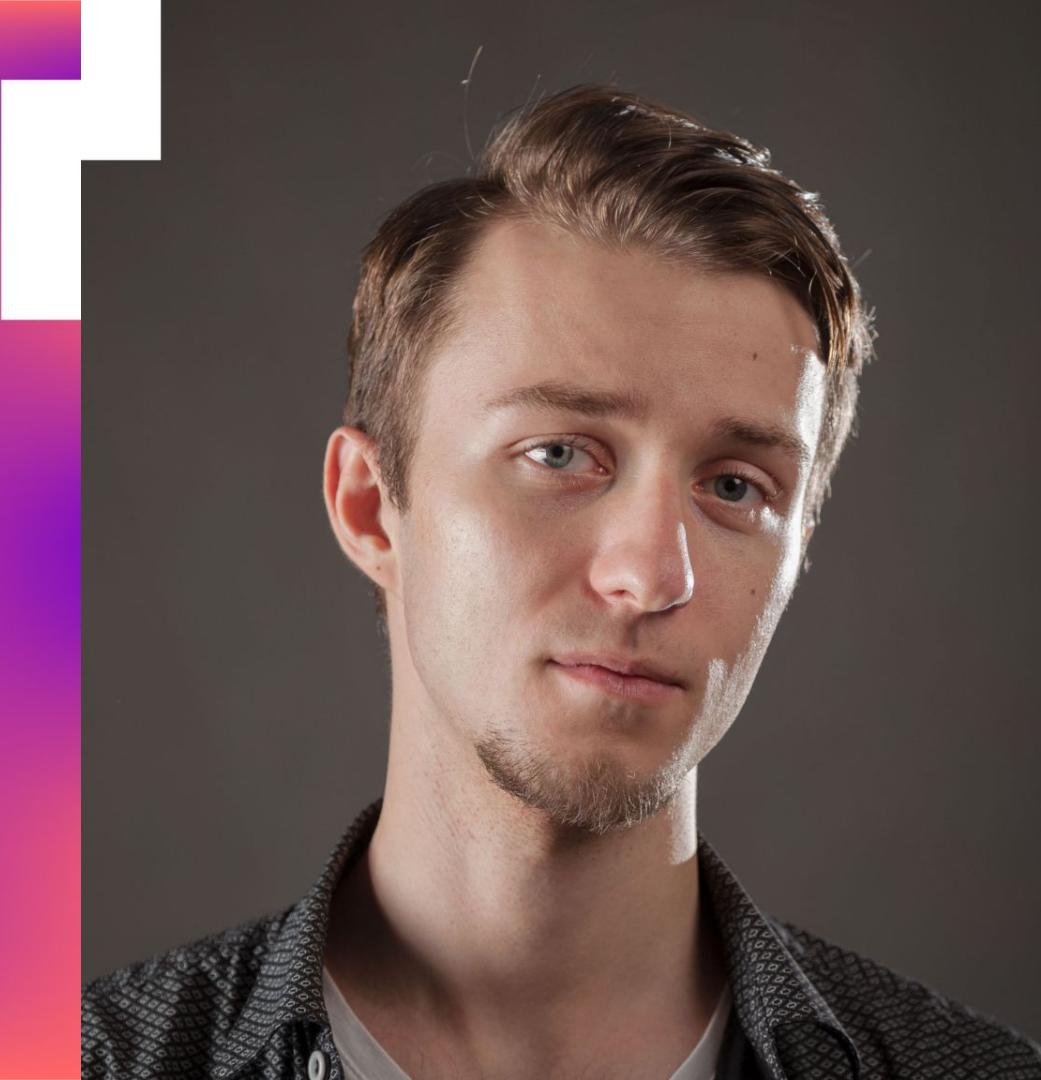
AI-агенты в разработке

**Влад Рыжков**

QA at Reliability team, inDrive

**Демо**

AI-Driven QA: Claude,  
тестирование API и  
визуальные тесты



## Зачем AI для QA?

Разработчики с ИИ пишут больше и кода быстрее → QA без ускорения или автоматизации может стать узким горлышком

## Зачем QA для AI?

Результаты AI генераций могут быть нестабильны и пайплайны генерации тоже нужно тестировать. Подходы к оценки качества применимы в любой сфере



# AI ошибается без контекста

## Без контекста

- Использует требования с противоречиями
- Генерирует меньшее число тестов ~30-50%
- Может терять проверки даже граничных значений
- Создаёт множество поясняющих md файлов (Haiku)

## С контекстом

- Легко выполнить несколько небольших операций преднастройки
- Лучше покрывает требования
- Генерирует API тесты в соответствии с указанной структурой и правилами

# **Слишком много контекста за один раз - плохо**

Context Usage	
👤	claudie-sonnet-4-6 · 34k/200k tokens (17%)
👤	<i>Estimated usage by category</i>
👤	👤 System prompt: 5.1k tokens (2.6%)
👤	👤 System tools: 16.2k tokens (8.1%)
👤	👤 MCP tools: 2k tokens (1.0%)
👤	👤 Memory files: 1.7k tokens (0.9%)
👤	👤 Skills: 757 tokens (0.4%)
👤	👤 Messages: 9.6k tokens (4.8%)
👤	👤 Free space: 132k (65.8%)
👤	👤 Autocompact buffer: 33k tokens (16.5%)



\* 8 044 строки в \*.md файлах

## Деградация поведения

- При превышении 50% контекстного окна
- При нерелавантном контексте
- При противоречивых инструкциях
- После достижения 100% контекстного окна и 'compact'

## Эффективное использование контекста:

- Отдельные небольшие инструкции
- Очистка контекста (/clear)
- Сохранение артефактов перед закрытием сессии
- Вывод в.md или файл кода дешевле чем в чат
- Декомпозиция операций
- Пополнять Claude.md / SKILL.md чтобы не повторять ошибки

# Как выбирать в какой слой добавлять инструкции

CLAUDE.md (rules.md) < 300-500 строк (Readme for AI)

- Стек, Архитектура, Критические Правила

agent.md < 300-500 строк

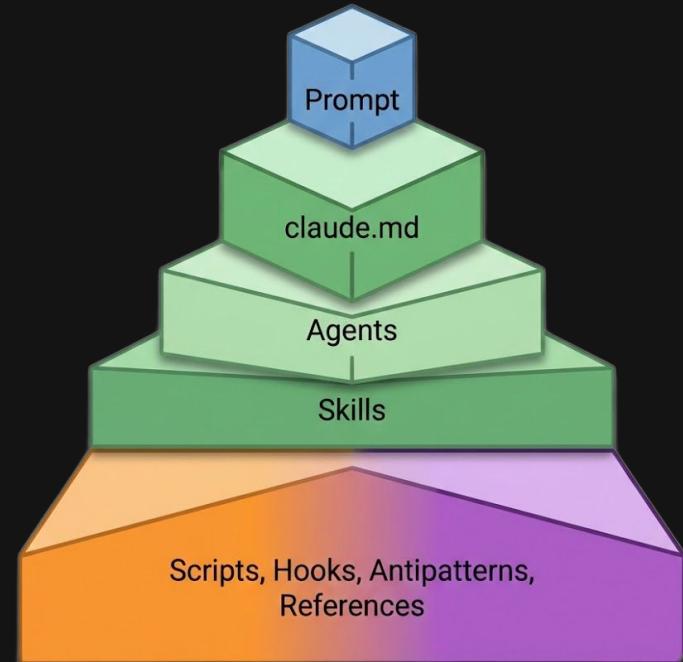
- Mindset + Карта Скиллов + Ссылки на Паттерны и Антипаттерны:
- Definition of Done: Что считается завершенной задачей (код, тест, отчет).

SKILL.md < 500 строк или больше - пошаговые гайды:

- Проверяем требования
- Пишем мануальные тест-кейсы
- Пишем API-тесты

При превышении лимита

- /references.
- /antipatterns.



## **Всё можно автоматизировать, но не нужно**

- Проверка документа на непротиворечивость
- Проверка правильной связности нескольких документов
- Автоматические предложения по исправлению документов и вынесению отдельных частей на нижние слои при превышении объёма
- Автоматически обновлять документацию “для себя” что есть в сетапе и как работает
- Цикл: запустить автотесты, решить ошибки (не более X попыток)

# Не плодим агентов без необходимости

```
✓  └── .claude
    └── agents
        M↓ auditor.md
        M↓ sdet.md
        └── audit
    >   └── hooks
    >   └── protocols
    >   └── qa-antipatterns
    >   └── skills
        M↓ qa_agent.md
    { } settings.json
```

- Оркестратор, руководитель, проверятор
- Аналитик, проверятор
- Писатель



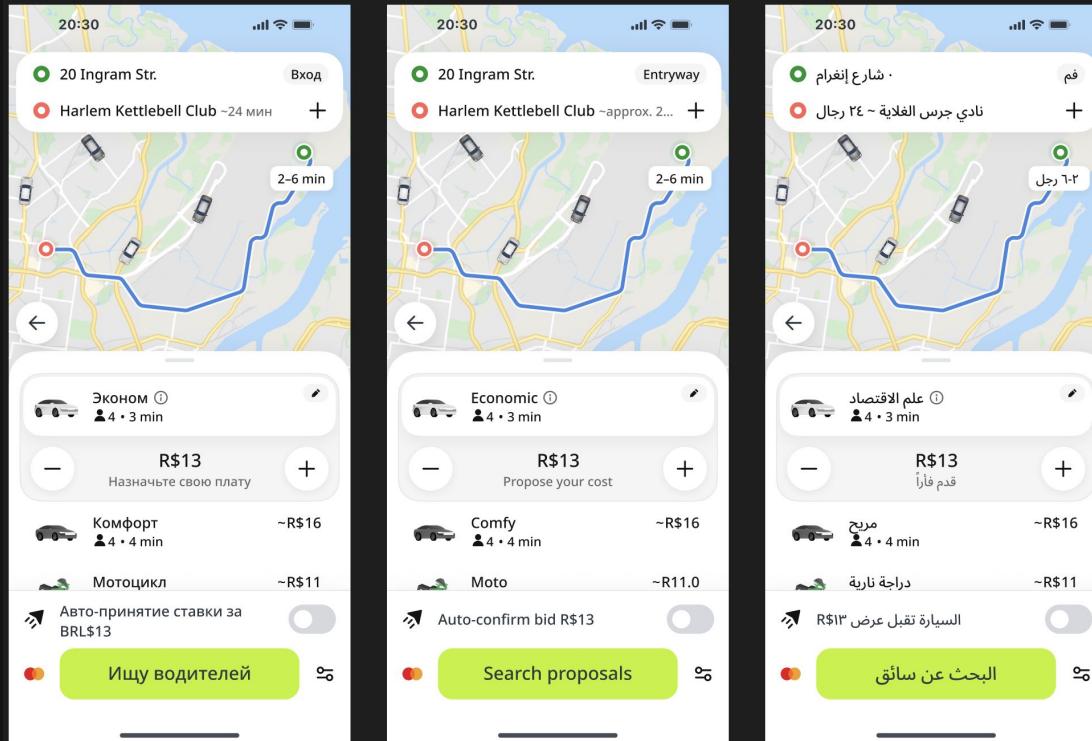
# Как заходить в хату новый репозиторий (SKILL.md)

- **/init-project**: сгенерить Claude.md для бэкенд репо
- **/repo-scout**: провести анализ целевого репозитория
- **/test-cases**: создать список тестов
- Начать репо API тестов с создания:
  - /init-project:** Claude.md
  - /init-agent:** agent.md
  - /init-skill:** SKILL.md
- **/api-tests:** генерить API тесты

\*man in the loop



# Визуальный и лингвистический анализ



# Переходим к практике

Клонируйте репозиторий:

<https://github.com/vlad-ryzhkov/AI-QA-workshop-feb19>

Убедитесь, что есть доступ к AI-ассистенту

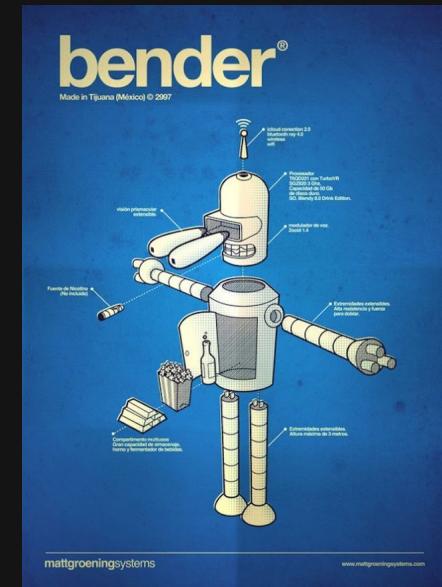
Основной стек демонстрации: IntelliJ IDEA + Claude Code CLI

Альтернативы (см. README):

- IDE: Cursor, VS Code, OpenCode
- AI: GitHub Copilot, Gemini, любой чат с ИИ

**Минималистичный стек:**  
.claude/settings.json (!)

- 1) kotlin-lsp Plugin
- 2) claude-plugins-official
- 3) context7 MCP
- 4) sequential-thinking MCP



# Идеи для самостоятельной практики

- MCP Playwright + Jira для запуска анализа web сайта и создание баг репортов в Jira (приключение на 30-60 минут)
- Агент/SKILL для код-ревью
- Упаковать ваш фреймворк автотестов:
  - SKILL
  - Antipatterns
  - References / examples

