

Технологии тестирования

и что нам мешает



Прошел год



Что сделали и как

- Продумали больше **100** вопросов для исследования



Что сделали и как

- Продумали больше **100** вопросов для исследования
- Заручились поддержкой сообществ



Что сделали и как

- Продумали больше **100** вопросов для исследования
- Заручились поддержкой сообществ
- Собрали больше **~300** анонимных ответов



Что сделали и как

- Продумали больше **100** вопросов для исследования
- Заручились поддержкой сообществ
- Собрали больше **~300** анонимных ответов

Вопросы про инструменты, процессы,
виды тестирования и про ВАС



Что узнали

- Как мы тестируем



Что узнали

- Как мы тестируем
- Как мы хотели бы тестировать



Что узнали

- Как мы тестируем
- Как мы хотели бы тестировать
- Что нам мешает

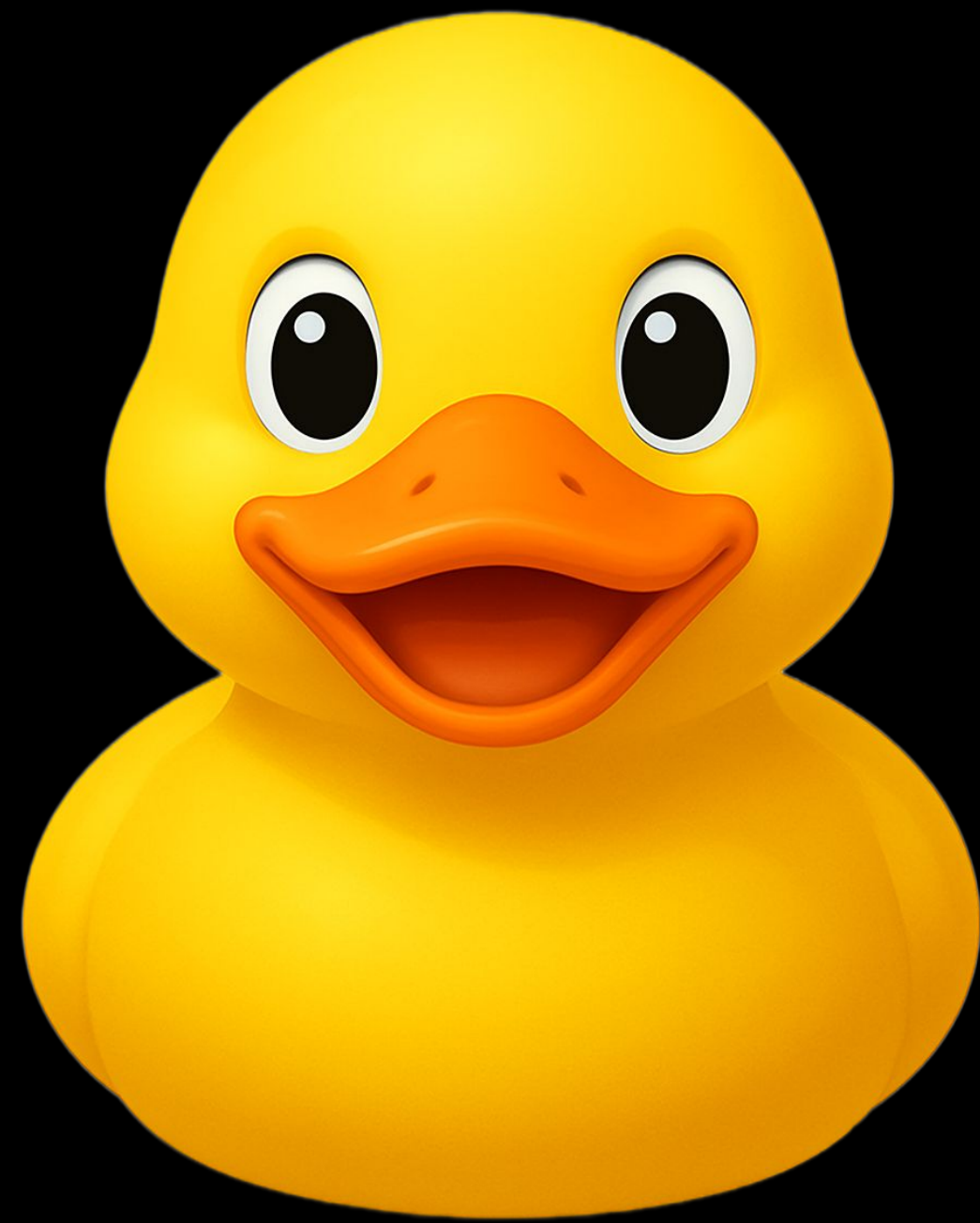


**А какие изменения
замечаете вы?**

Респонденты



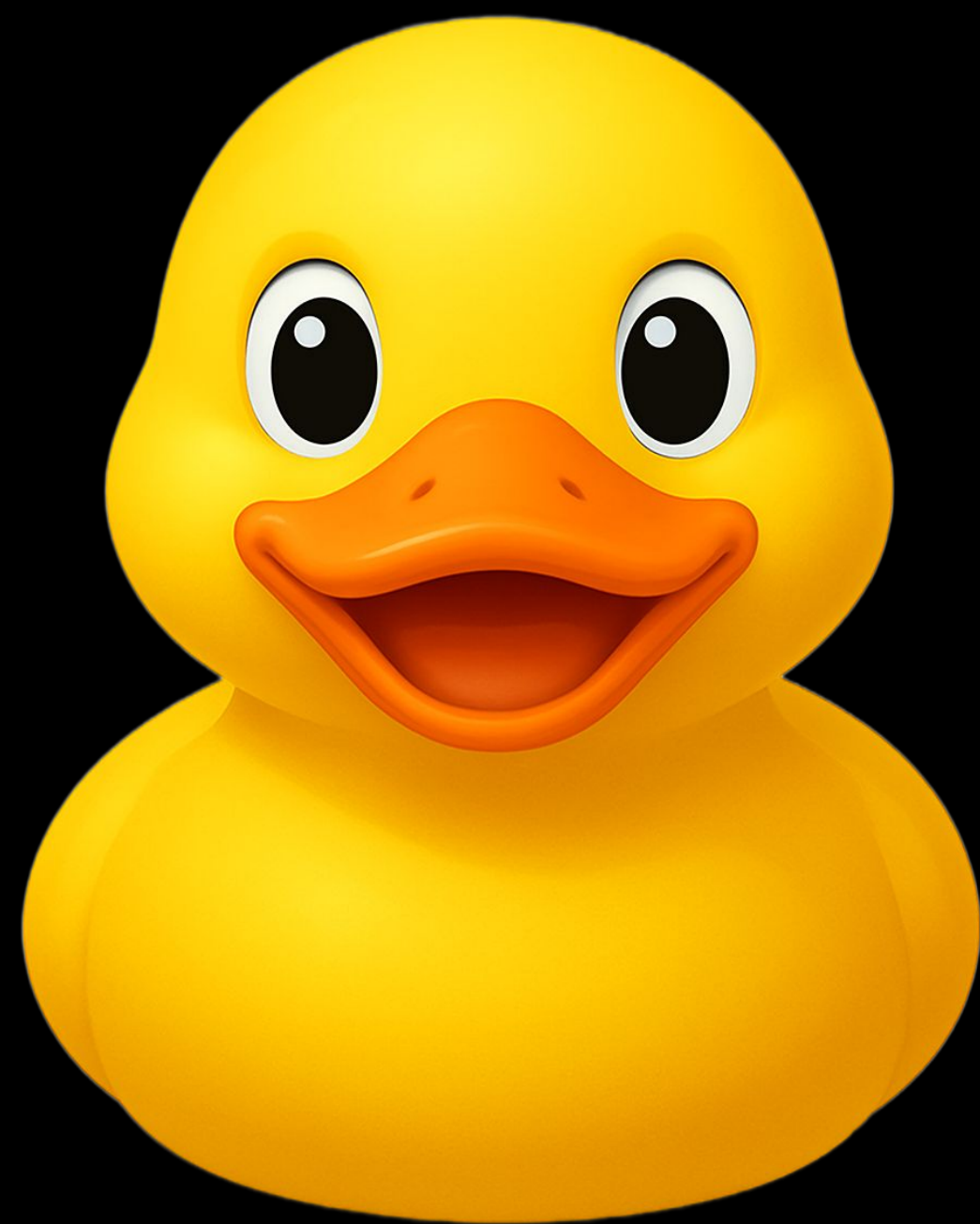
Опыт в тестировании



1-3 года

20%

Опыт в тестировании



1-3 года

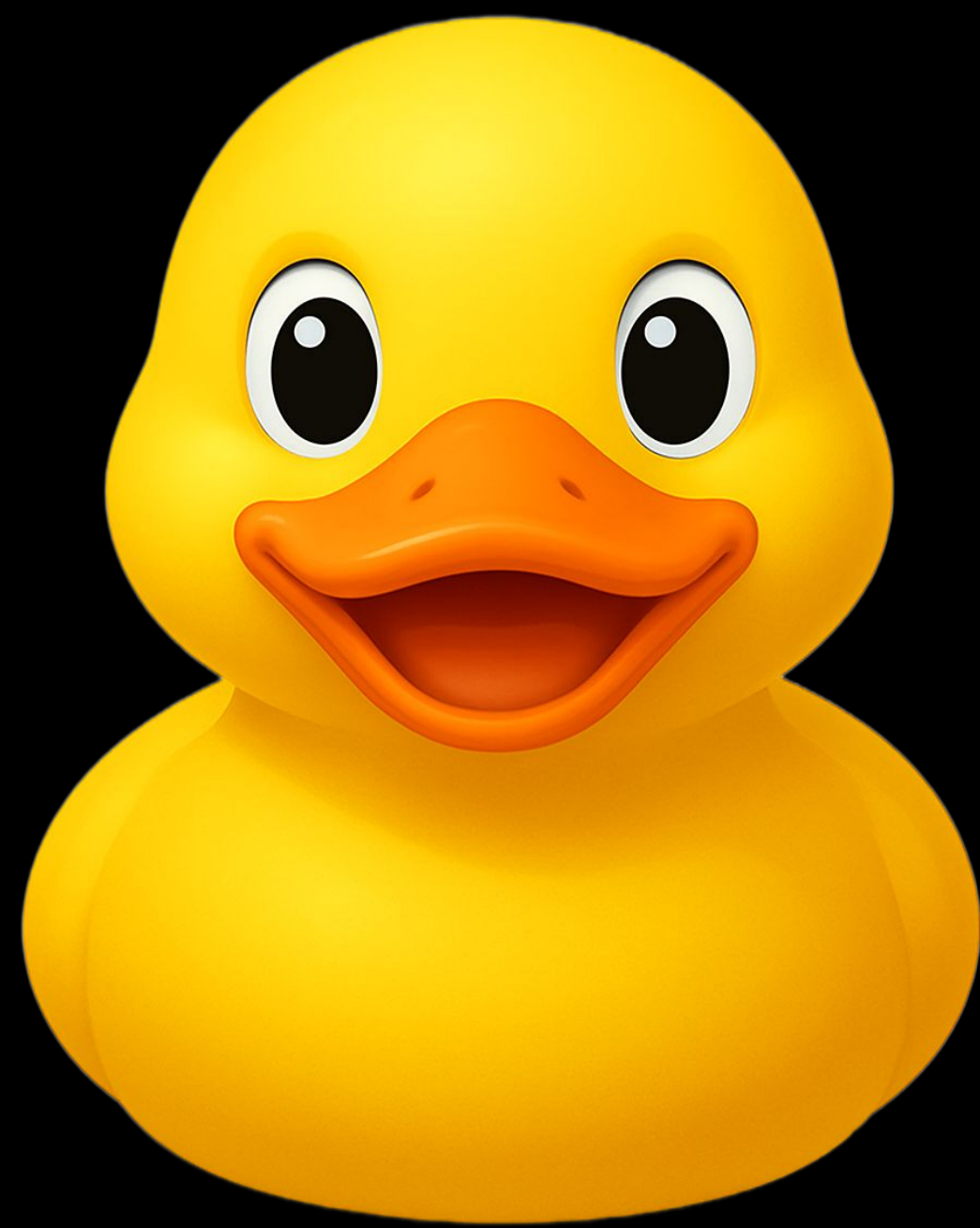
20%



4-6 лет

40%

Опыт в тестировании



1-3 года
20%



4-6 лет
40%

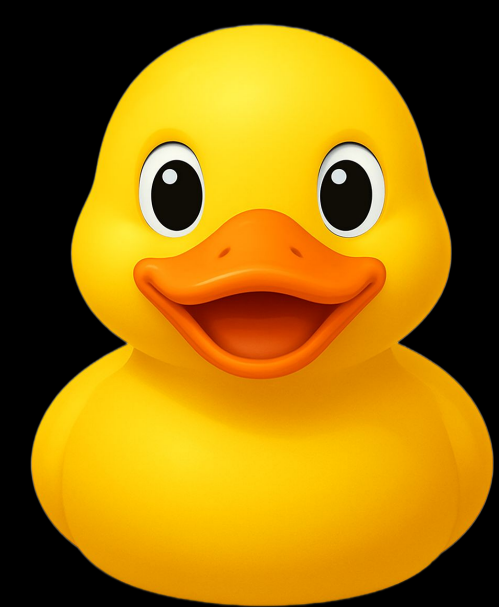


7-10 лет
19%



>10 лет
18%

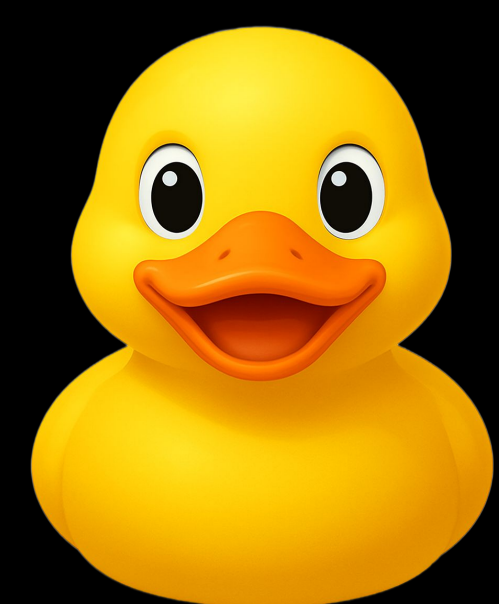
Уровень опыта



Junior

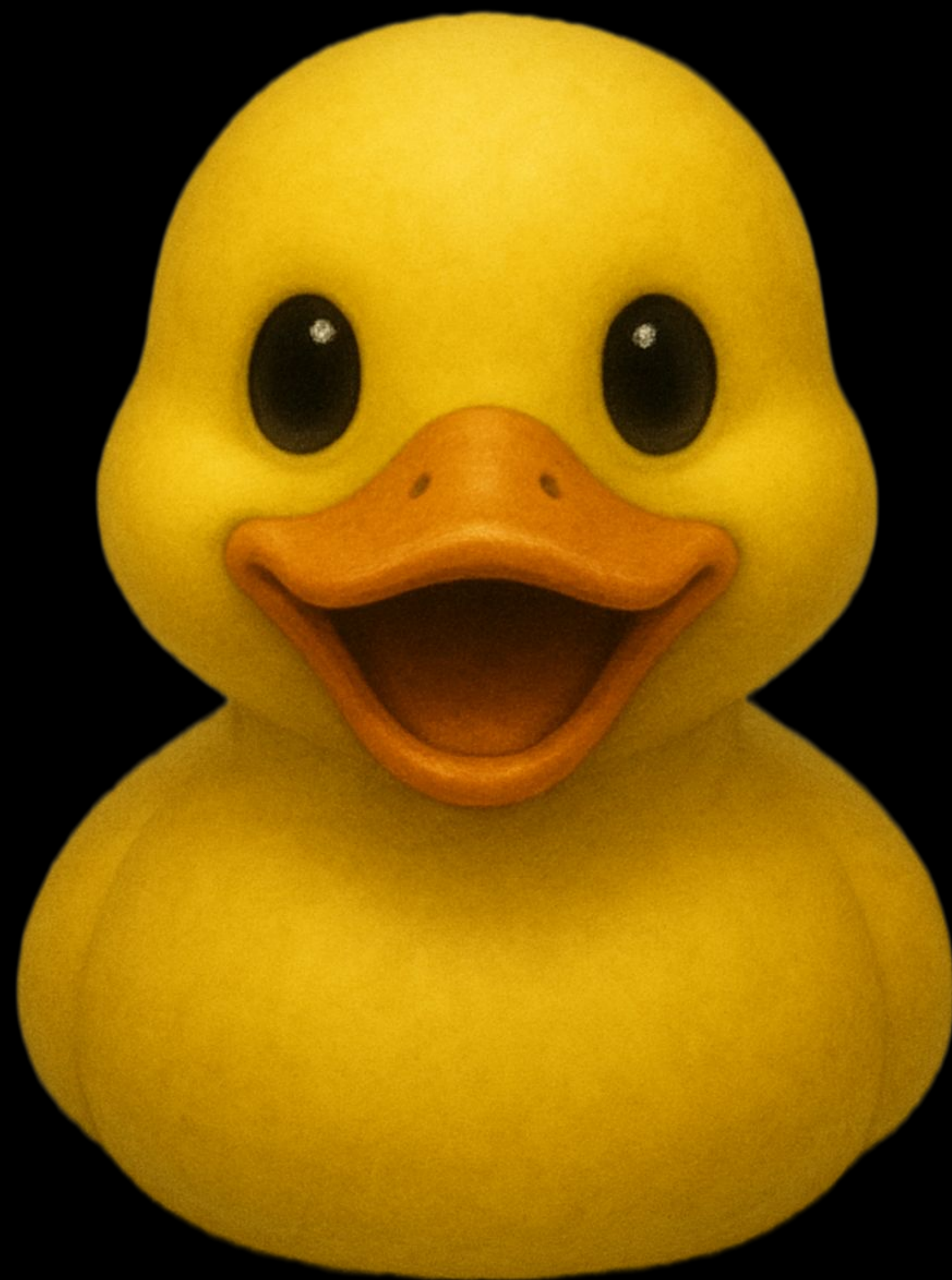
3%

Уровень опыта



Junior

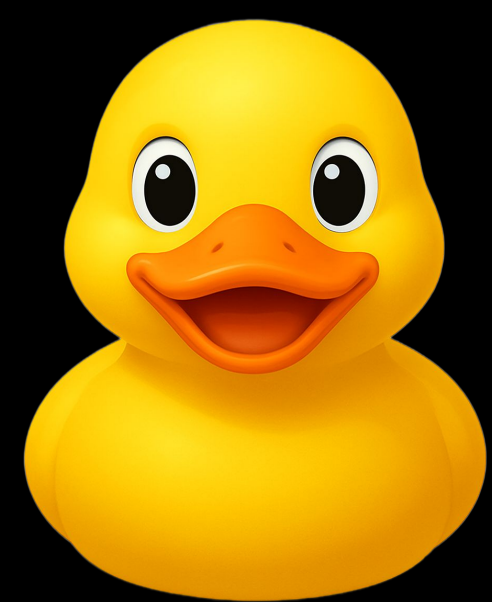
3%



Middle

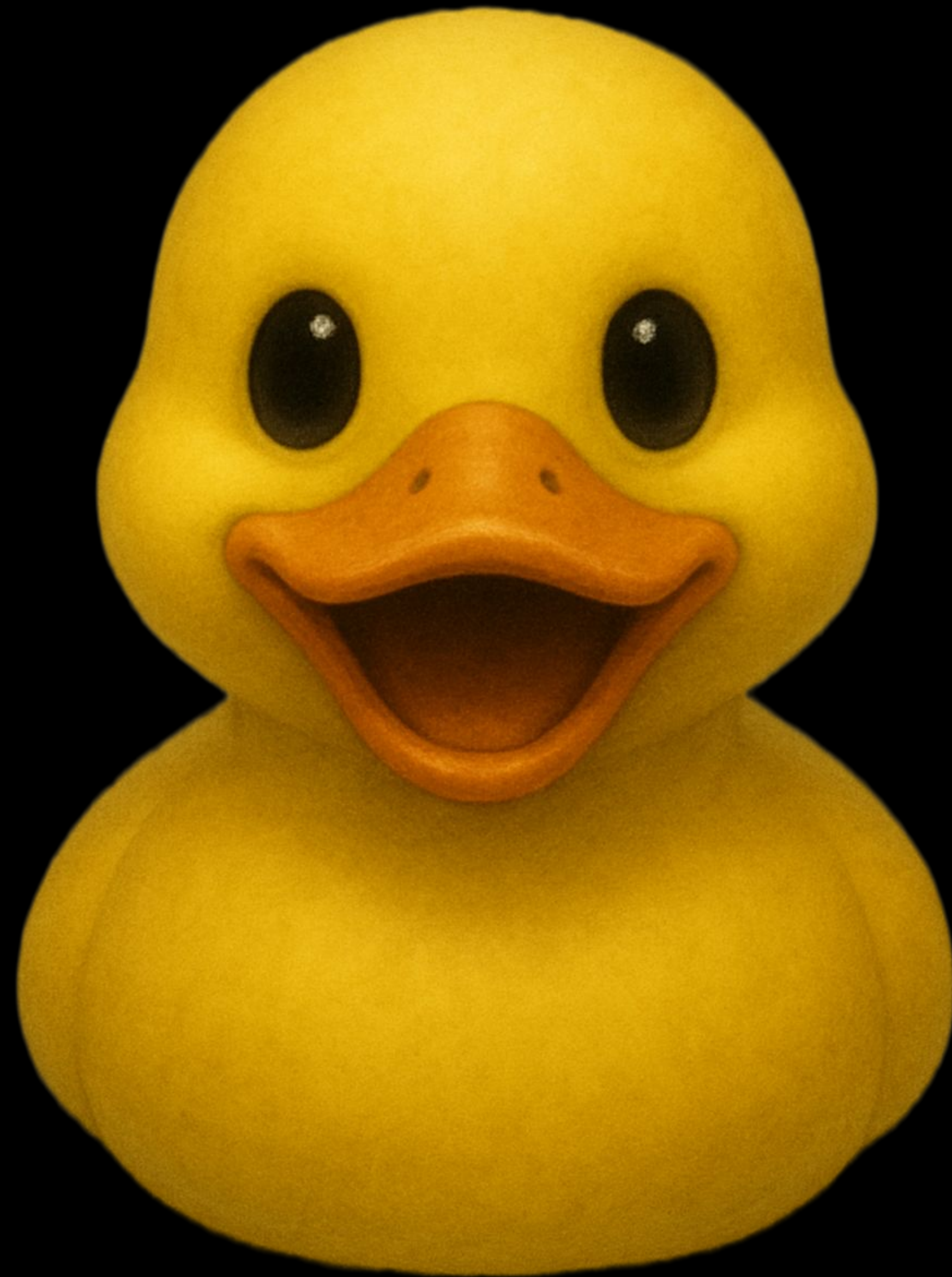
43%

Уровень опыта



Junior

3%



Middle

43%



Senior

26%



Lead

18%



Manager

5%

Возраст

32

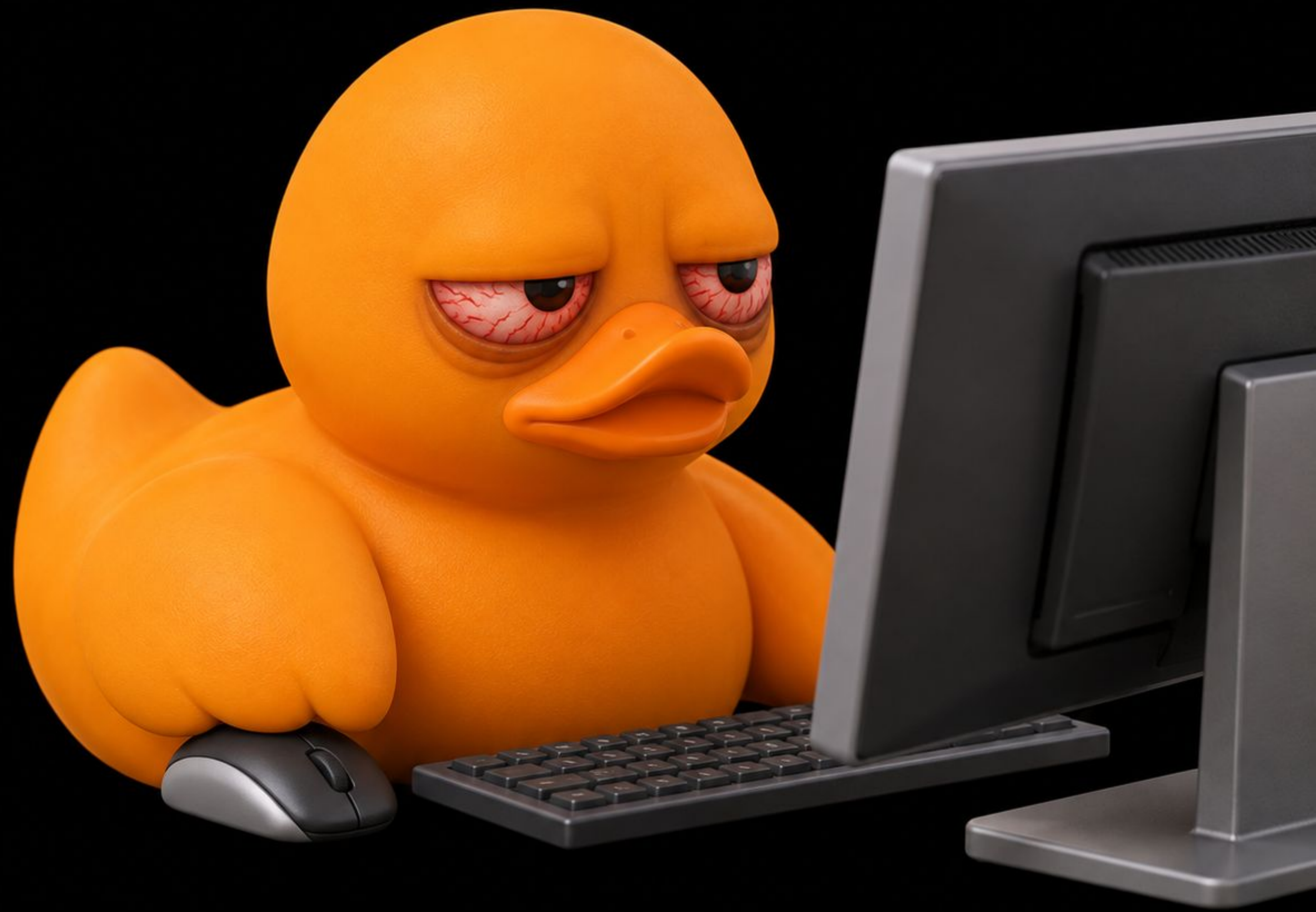


Возраст

32

34

Год идет за два



**Что и как мы
тестируем?**



Виды тестирования за год

Функциональное 94%

Автоматизированное 77%

Графического интерфейса 67% **Документации** 60%

Виды тестирования за год

Функциональное 94%

Автоматизированное 77%

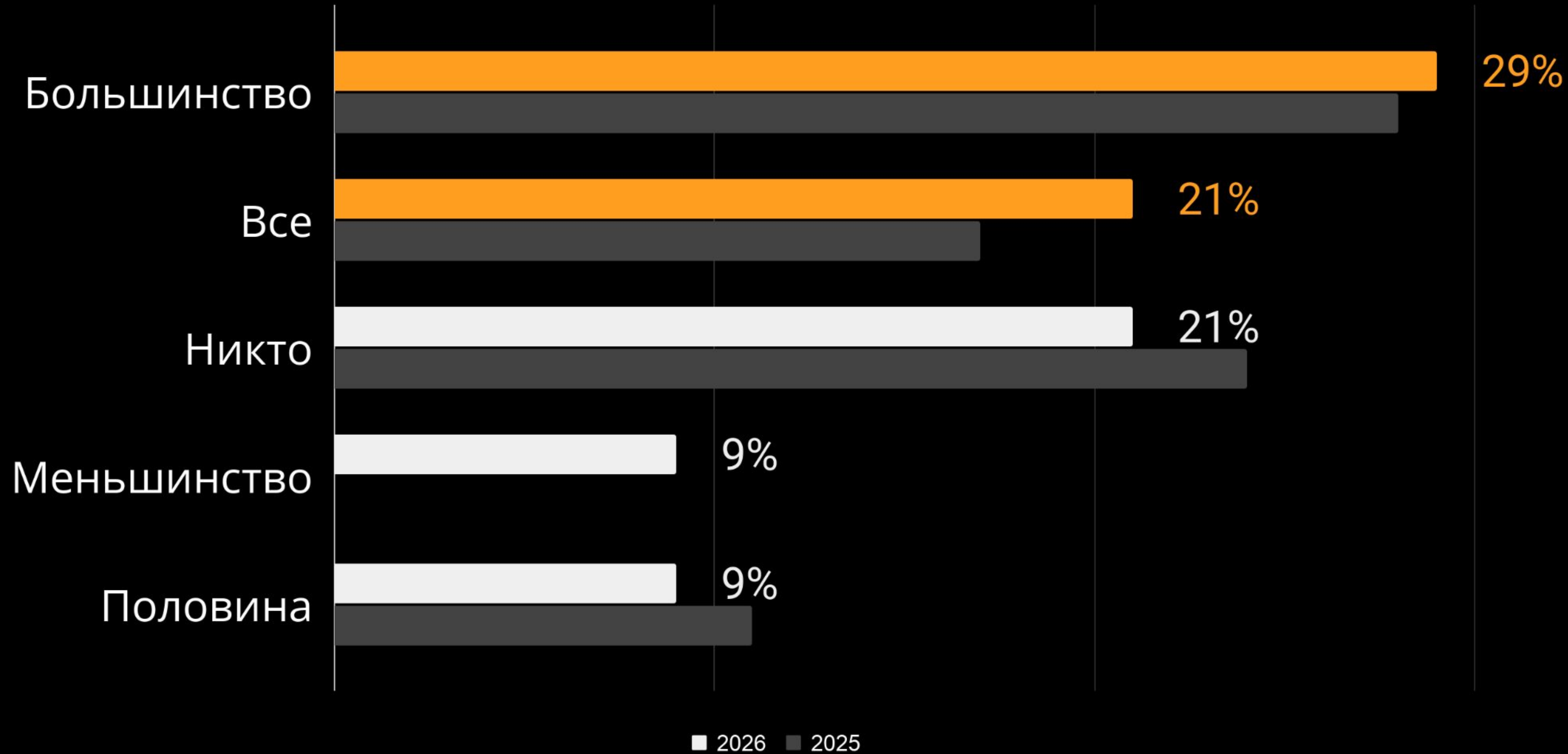
Графического интерфейса 67% **Документации** 60%

Удобства использования 47% **Производительности** 33%

Установки 31% **Локализации** 26% **Совместимости** 23% **Контрактное** 20%

Безопасности 15%

Тестируют только вручную



Ручное



Автоматизация

Платформы и типы продуктов

Платформы и типы продуктов

85%

API, Backend,
Web

Платформы и типы продуктов

85%

API, Backend,
Web

59% Базы данных

Платформы и типы продуктов

85%

API, Backend,
Web

59% Базы данных

39% Mobile

Платформы и типы продуктов

85%

API, Backend,
Web

59% Базы данных

39% Mobile

28% Desktop

Платформы и типы продуктов

85%

API, Backend,
Web

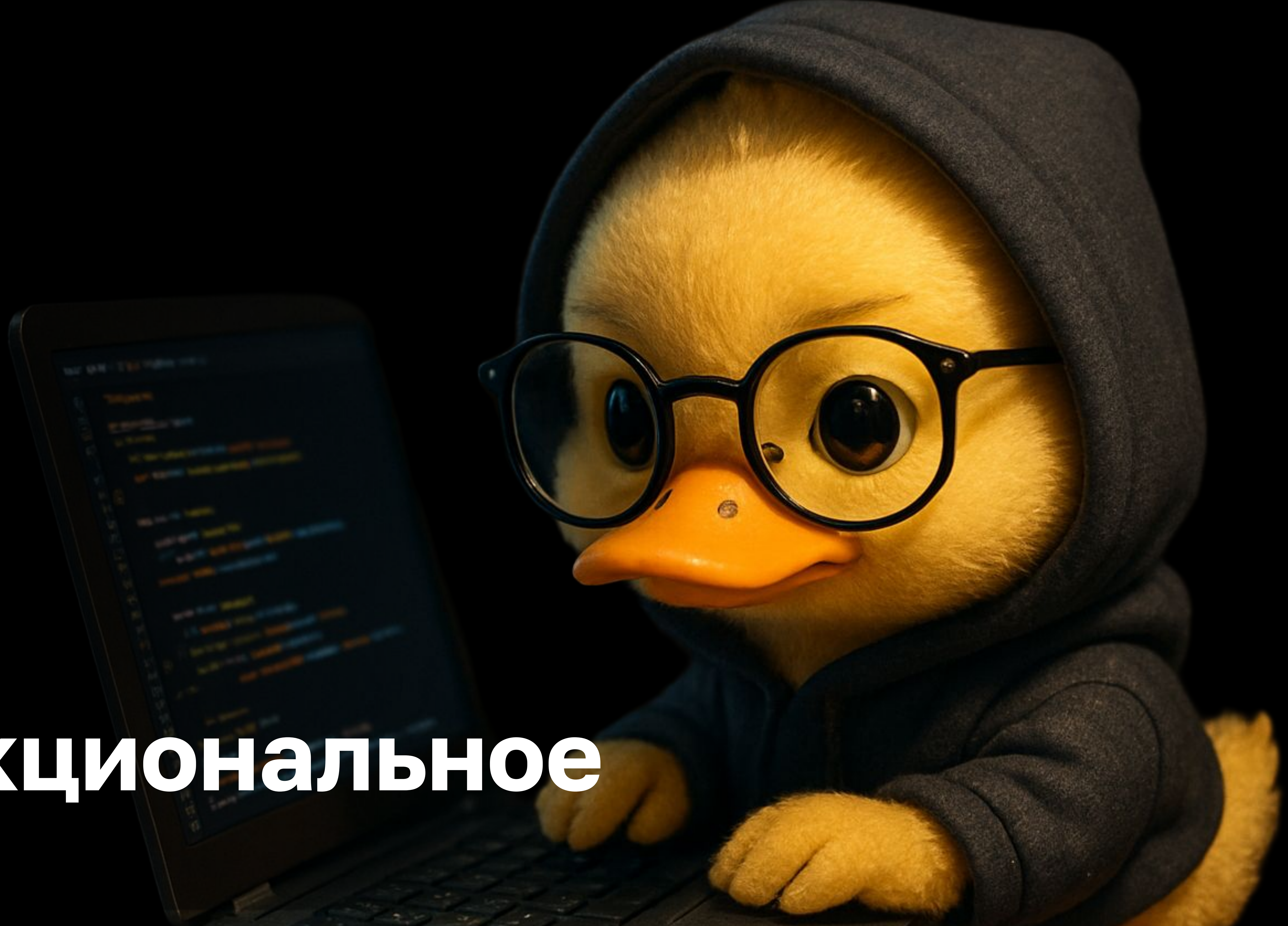
59% Базы данных

39% Mobile

28% Desktop

18% Приложения с AI-модулями

Функциональное



Инструменты для API

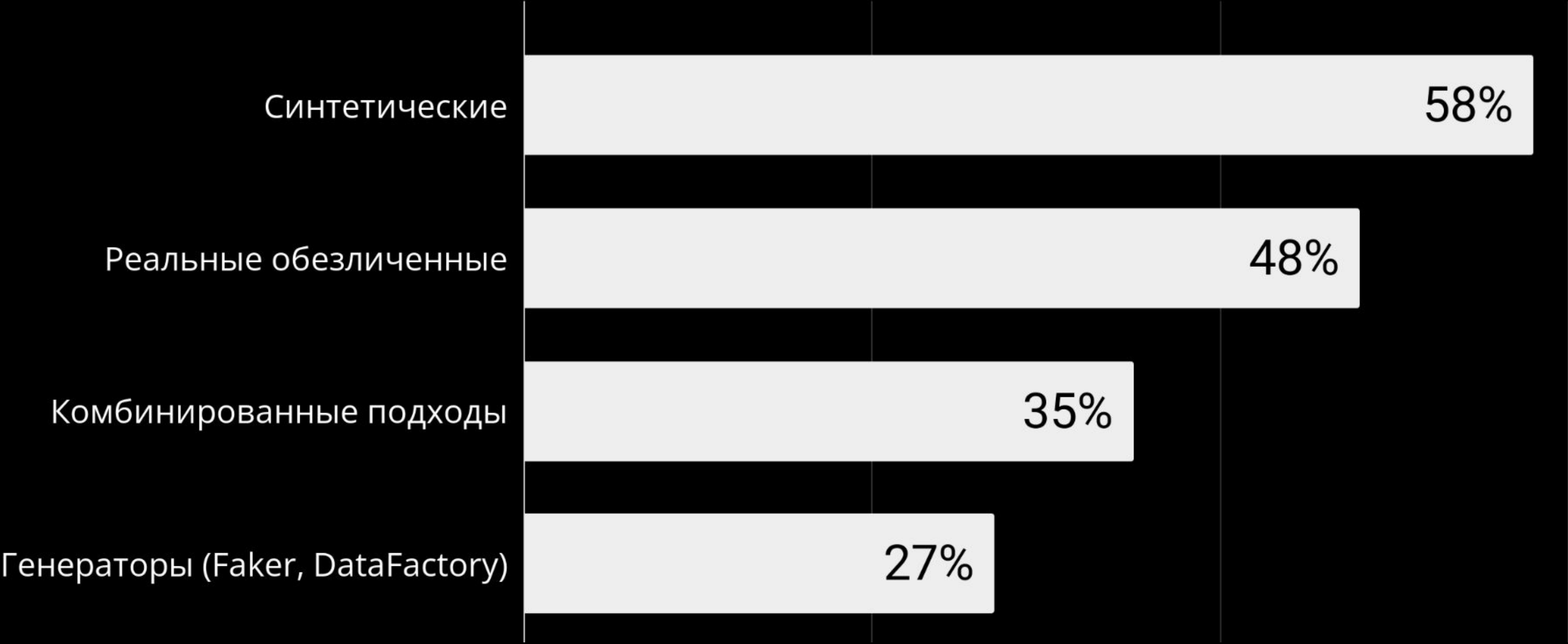
DevTools 60% **Postman** 57% **curl** 44%

Самописные библиотеки для API 24% **Playwright API** 21%

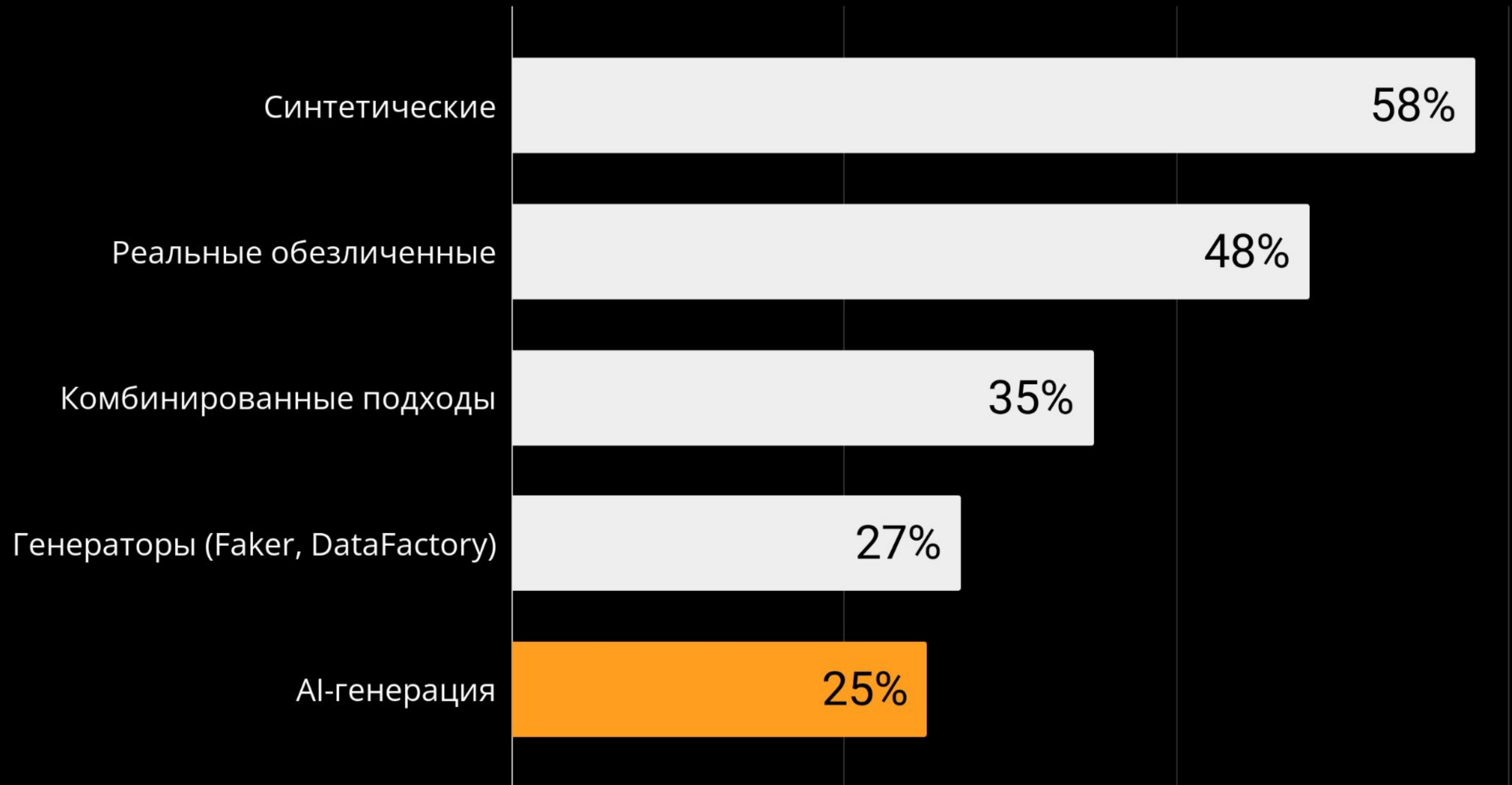
PyTest + requests 20% **Charles** 19% **Insomnia** 19% **HTTP Client в IDE** 16%

RestAssured 15%

Тестовые данные



Тестовые данные



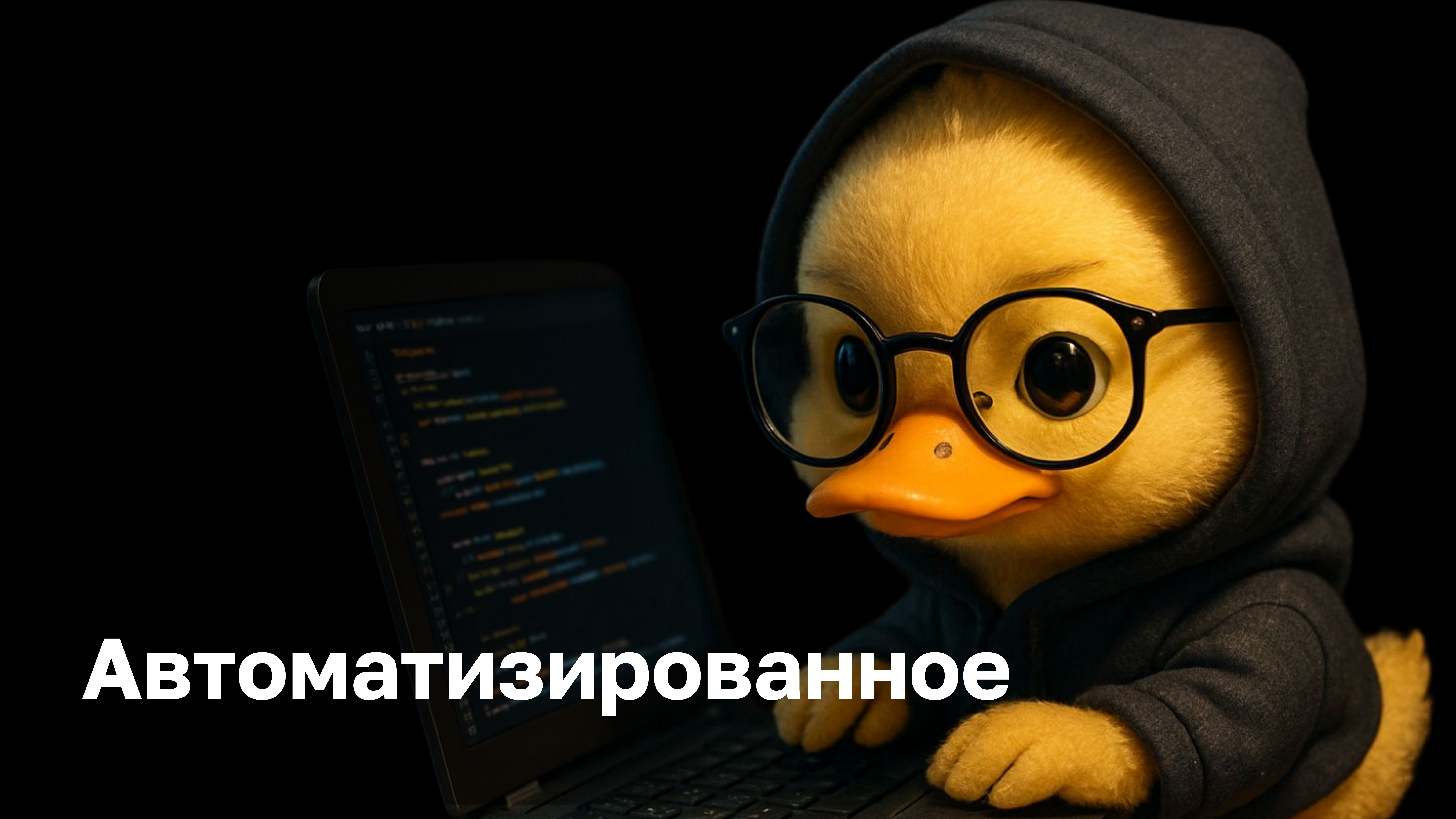
Инструменты 2026

Инструменты тестировщика 2026



Юлия Атлыгина



A yellow duck wearing a grey hoodie and glasses is sitting at a laptop. The laptop screen displays lines of code in a dark theme. The duck is looking at the screen with a focused expression.

Автоматизированное

На чем пишем?

30%



Java™

На чем пишем?

30%



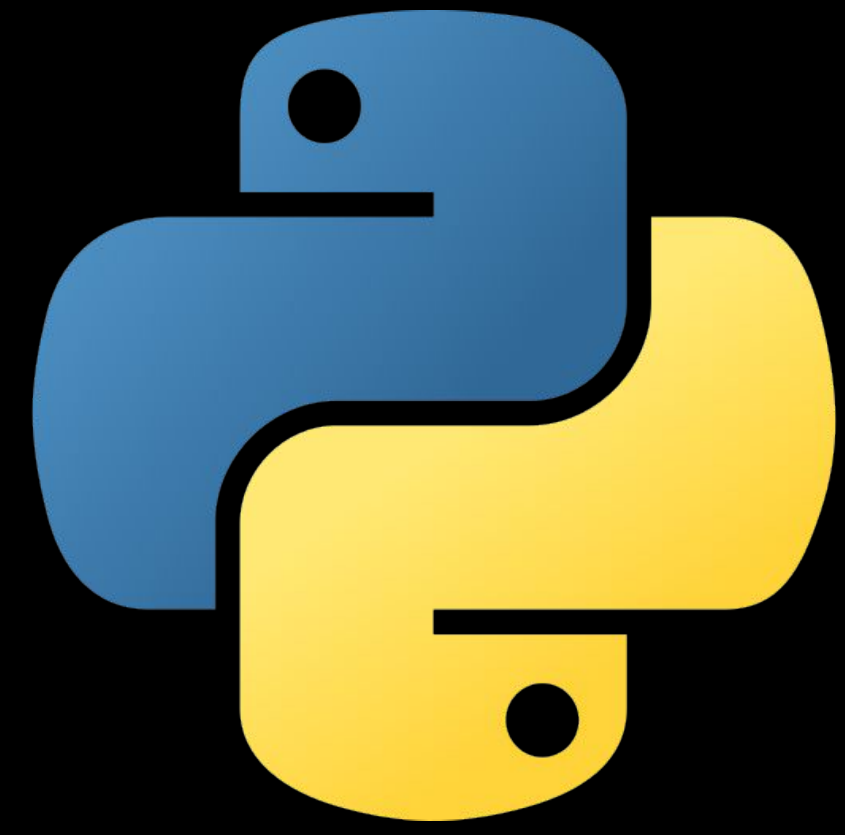
Java™

20%



На чем пишем?

30%



Java™

20%

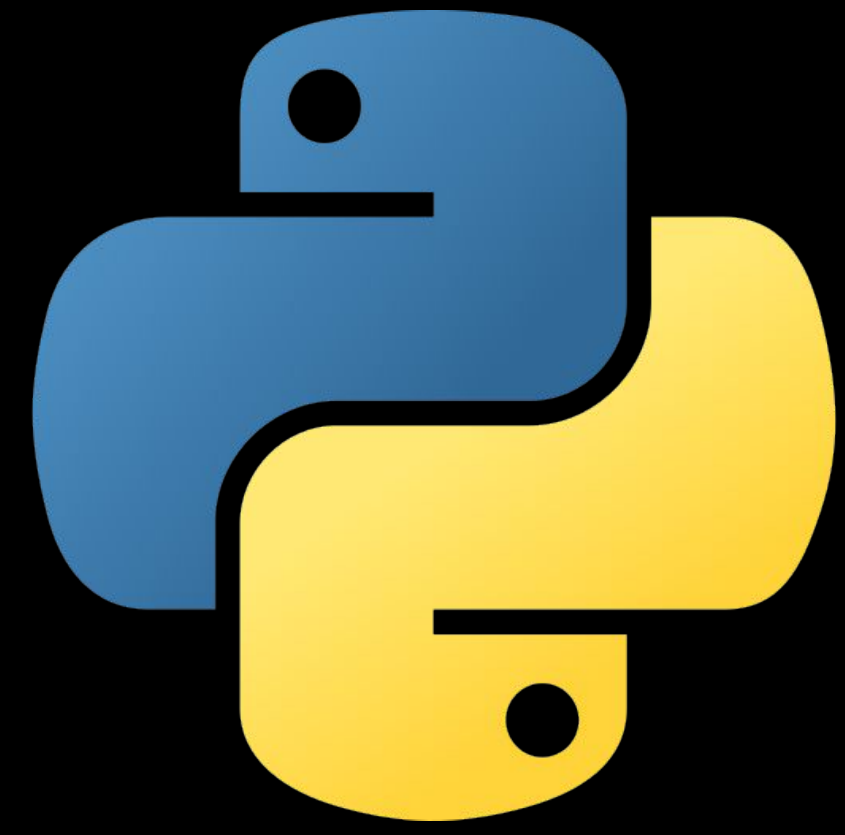


7%



На чем пишем?

30%



Java™

20%



7%



GO

18%

Считает, что язык не важен, для них код пишет AI

Фреймворки

34%
Playwright

30%
Selenium

26%
Pytest

21%
JUnit

< 10%
TestNG,
Cucumber,
Appium,
MSTest

CI-платформа

GitLab CI 55% **Jenkins** 32% **TeamCity** 10%

Azure DevOps Pipelines 8% **Bitbucket Pipelines** 6% **Своя CI-платформа** 5%

GitHub Actions 3%



Инструменты
не изменились

Что и как тестируем

Виды тестирования, объекты тестирования, тестовые данные и документация

TECHRADAR TESTING 2026

● без изменений ▲ рост к 2025 ▼ снижение к 2025 ◆ нет данных 2025
○ пунктир и серый текст — данные только со скрытых слайдов

Виды тестирования

● 1	Функциональное	94%
◆ 2	Автоматизированное	77%
▲ 3	Графического интерфейса	▲ 67%
● 4	Документации	60%
● 5	Удобства использования	47%
● 6	Производительности	33%
● 7	Установки	31%
● 8	Локализации	26%
● 9	Совместимости	23%
◆ 10	Контрактное	20%
● 11	Безопасности	15%
○ 12	Соответствия	13%
○ 13	Доступности	12%
○ 14	Отказоустойчивости	12%
◆ 15	Хаос-тестирование	12%

Тестовые данные

◆ 23	Синтетические	58%
◆ 24	Реальные обезличенные	48%
◆ 25	Комбинированные подходы	35%
◆ 26	Генераторы (Faker, DataFactory)	27%
◆ 27	AI-генерация	25%

Что тестируем

▲ 16	API / Backend	▲ 85%
● 17	Web	85%
◆ 18	Базы данных	59%
▲ 19	Mobile	▲ 39%
▲ 20	Desktop	▲ 28%
○ 21	CLI-инструменты	▲ 23%
◆ 22	Приложения с AI-модулями	18%

Тестовая документация

◆ 28	Тест-кейсы	87%
◆ 29	Чек-листы	77%
◆ 30	Тест-планы	59%
◆ 31	Тест-отчёты	36%
◆ 32	Стратегии тестирования	17%
◆ 33	Матрица трассировки	15%
◆ 34	Test Cases as Code	14%
◆ 35	Майнд-карты	12%



Источник: опрос сообщества TechRadar Testing 2026 (~250 ответов). % — доля респондентов; кольцо определяется долей.

Стек автоматизации

Языки, фреймворки, IDE и инструменты мобильного тестирования

TECHRADAR TESTING 2026

● без изменений ▲ рост к 2025 ▼ снижение к 2025 ◆ нет данных 2025
○ пунктир и серый текст — данные только со скрытых слайдов

Языки

▼ 1	Python	▼ 32%
● 2	Java	29%
▲ 3	JavaScript / TypeScript	▲ 19%
◆ 4	«Код пишет AI»	17%
● 5	Kotlin	7%
▲ 6	C#	▲ 7%
● 7	Go	4%

IDE и редакторы

◆ 16	JetBrains IDE	63%
◆ 17	Visual Studio Code	43%
◆ 18	Cursor	12%
◆ 19	Sublime Text	5%
◆ 20	Vim / Neovim	3%

Фреймворки

● 8	Playwright	34%
▲ 9	Selenium	▲ 30%
● 10	PyTest	26%
● 11	JUnit	21%
● 12	TestNG	8%
● 13	Cucumber	8%
● 14	Appium	6%
◆ 15	MSTest	3%

Мобильное тестирование

◆ 21	Эмуляторы / симуляторы	21%
◆ 22	Appium (mobile)	9%
◆ 23	Proxyman	8%
◆ 24	Playwright (mobile web)	5%
◆ 25	BrowserStack	4%
◆ 26	Kaspresso	3%
◆ 27	Espresso	3%
◆ 28	XCTest	3%



Источник: опрос сообщества TechRadar Testing 2026 (~250 ответов). % — доля респондентов; кольцо определяется долей.

API, данные и нагрузка

Инструменты для API и баз данных, нагрузочное тестирование: инструменты и практики

TECHRADAR TESTING 2026

● без изменений ▲ рост к 2025 ▼ снижение к 2025 ◆ нет данных 2025
○ пунктир и серый текст — данные только со скрытых слайдов

API

▼ 1	Browser DevTools	▼ 60%
▼ 2	Postman	▼ 57%
◆ 3	curl	44%
◆ 4	Самописные библиотеки	24%
◆ 5	Playwright API	21%
◆ 6	PyTest + requests	20%
▼ 7	Charles	▼ 19%
◆ 8	Insomnia	19%
◆ 9	HTTP Client в IDE	16%
◆ 10	RestAssured	15%
◆ 11	SoapUI	9%
◆ 12	Fiddler	6%
◆ 13	Bruno	6%

Нагрузка: инструменты

◆ 19	Apache JMeter	14%
◆ 20	Самописный инструмент	10%
◆ 21	k6	9%
◆ 22	Gatling	5%
◆ 23	Locust	4%
◆ 24	Яндекс.Танк	2%

Базы данных

◆ 14	DBeaver	62%
◆ 15	pgAdmin	23%
◆ 16	Консольные клиенты (psql, mysql...)	20%
◆ 17	DataGrip / встроенные в IDE	15%
◆ 18	Custom SQL-скрипты	12%

Нагрузка: практики и метрики

● 25	Нагрузка по мере необходимости	35%
● 26	Доля ошибочных запросов	24%
▼ 27	Время отклика и перцентили	▼ 24%
▲ 28	Регулярная нагрузка (каждый спринт)	▲ 22%
▼ 29	Пропускная способность	▼ 22%
▼ 30	Потребление памяти	▼ 20%
▼ 31	Потребление CPU/GPU	▼ 19%
● 32	Метрики БД и хранилищ	10%
◆ 33	Метрики сети	10%



Источник: опрос сообщества TechRadar Testing 2026 (~250 ответов). % — доля респондентов; кольцо определяется долей.

Инфраструктура и процессы

CI/CD, управление тестами, логи и observability, модель разработки

TECHRADAR TESTING 2026

● без изменений ▲ рост к 2025 ▼ снижение к 2025 ◆ нет данных 2025
○ пунктир и серый текст — данные только со скрытых слайдов

CI/CD

▼ 1	GitLab CI	▼ 55%
▲ 2	Jenkins	▲ 32%
◆ 3	Feature flags	23%
● 4	TeamCity	10%
▲ 5	Azure DevOps Pipelines	▲ 8%
● 6	Bitbucket Pipelines	6%
● 7	Своя CI-платформа	5%
▼ 8	GitHub Actions	▼ 3%

Логи и observability

◆ 15	Prometheus + Grafana	34%
◆ 16	Elastic Stack (ELK)	33%
◆ 17	Grafana Loki	31%
◆ 18	Sentry	18%
◆ 19	Grafana Cloud	10%
◆ 20	OpenSearch	9%
◆ 21	Elastic APM	9%
◆ 22	OpenTelemetry	6%
◆ 23	Graylog	5%
◆ 24	VictoriaLogs	3%
◆ 25	Dynatrace	2%

Управление тестами

◆ 9	Allure TestOps	40%
◆ 10	Jira	33%
◆ 11	Jira: Zephyr	20%
◆ 12	Test IT	15%
◆ 13	Собственная TMS	13%
◆ 14	Azure DevOps	7%

Модель разработки

◆ 26	Scrum	43%
◆ 27	Kanban	26%
◆ 28	Гибрид Agile + Waterfall	18%
◆ 29	Scrumban	11%
◆ 30	SAFe / LeSS	8%
◆ 31	Waterfall	2%



Источник: опрос сообщества TechRadar Testing 2026 (~250 ответов). % — доля респондентов; кольцо определяется долей.

ИИ в тестировании

50% тестировщиков пользуются ИИ каждый день — но пока как помощником, а не исполнителем

● без изменений ▲ рост к 2025 ▼ снижение к 2025 ◆ нет данных 2025
○ пунктир и серый текст — данные только со скрытых слайдов

Модели и ассистенты

▼ 1	ChatGPT	▼ 38%
◆ 2	DeepSeek	32%
▲ 3	Claude	▲ 31%
▲ 4	Qwen	▲ 27%
▲ 5	Self-hosted LLM	▲ 17%
◆ 6	Cursor	12%
◆ 7	Корпоративные AI-инструменты	7%

Данные и знания

◆ 13	AI-генерация тестовых данных	25%
◆ 14	Готовые RAG / AI-поиск	25%
◆ 15	Свой RAG для проекта	9%
◆ 16	RAG для личного пользования	6%



Роль ИИ в работе QA

▼ 8	Справка и поиск	▼ 44%
▲ 9	Помогает, решение за QA	▲ 28%
◆ 10	Анализ требований и рисков	12%
▲ 11	Генерирует и запускает тесты	▲ 10%
◆ 12	Тестирует самостоятельно	3%

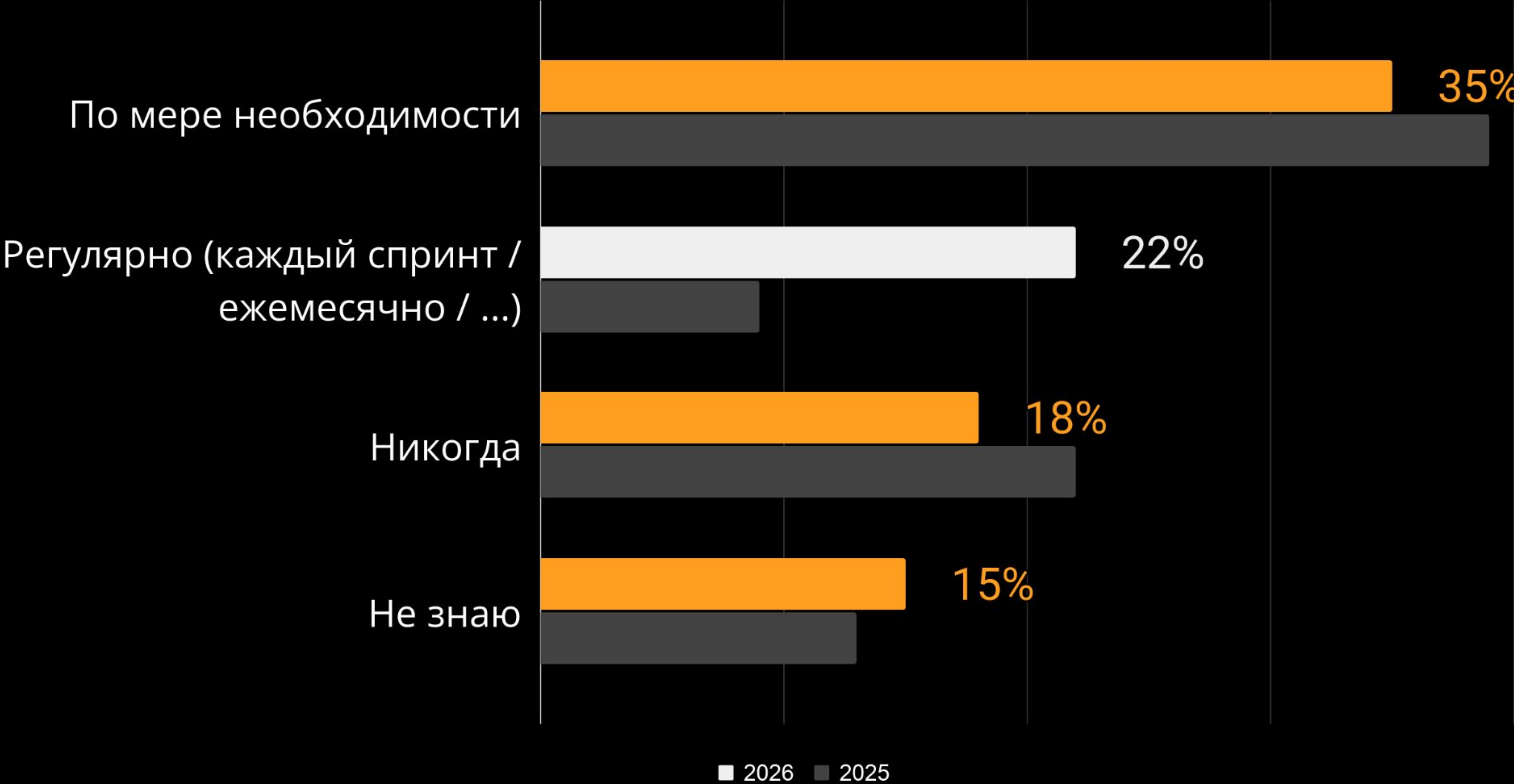
Тестирование AI/LLM

◆ 17	LLM-as-Judge	14%
◆ 18	Внутренние инструменты	11%
◆ 19	Answer relevance	9%
◆ 20	Human evaluation	8%
◆ 21	Свои benchmark-датасеты	8%
◆ 22	Custom-скрипты	6%
◆ 23	Semantic similarity	5%
◆ 24	Faithfulness	5%
◆ 25	Hallucination rate	5%
◆ 26	DeepEval	3%
◆ 27	Prompt injection / jailbreak	2%

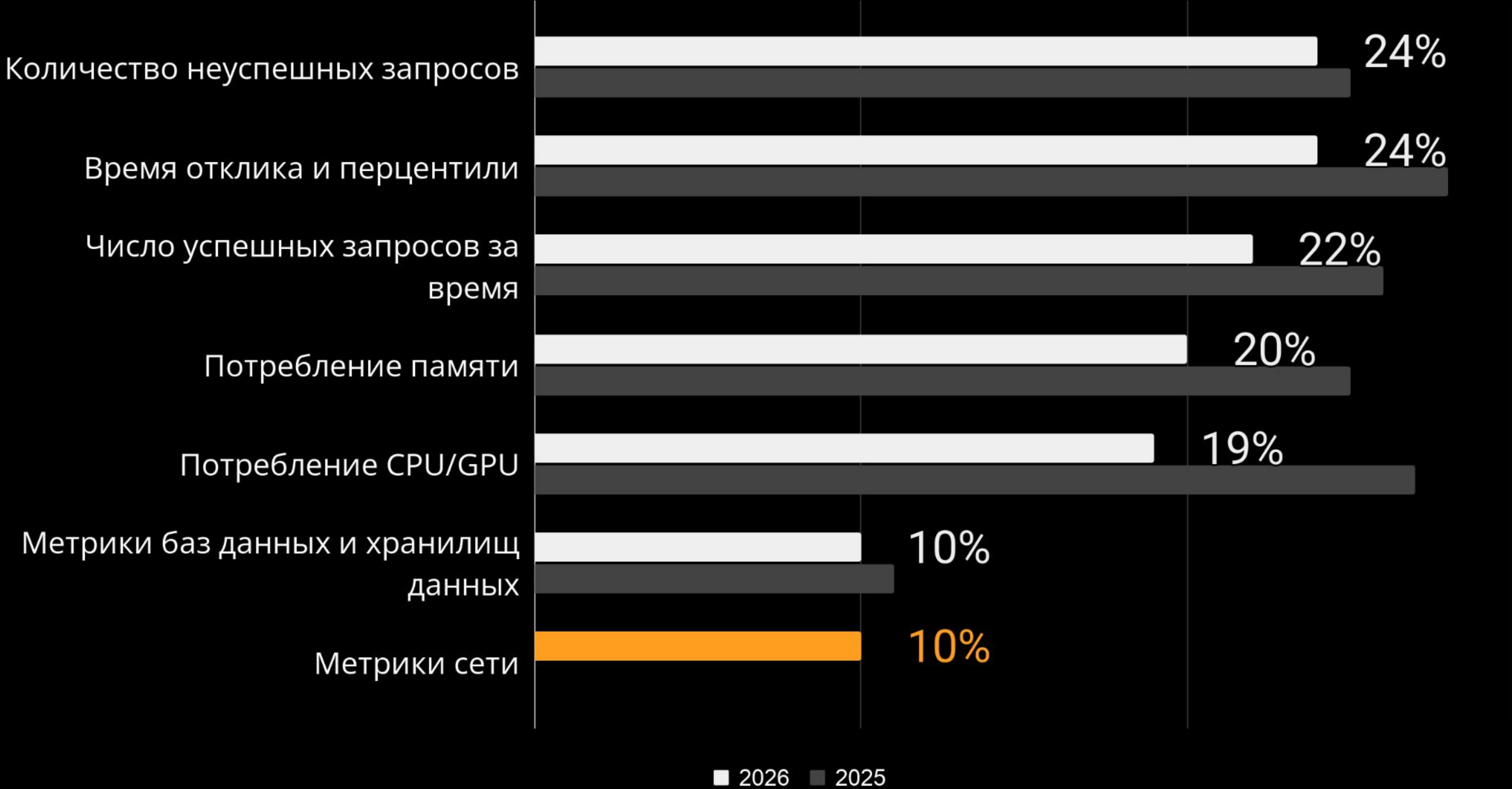
Нагрузочное



Как часто загружаем?



На что смотрим в нагрузке?



**Железо и сеть более
не проблема**

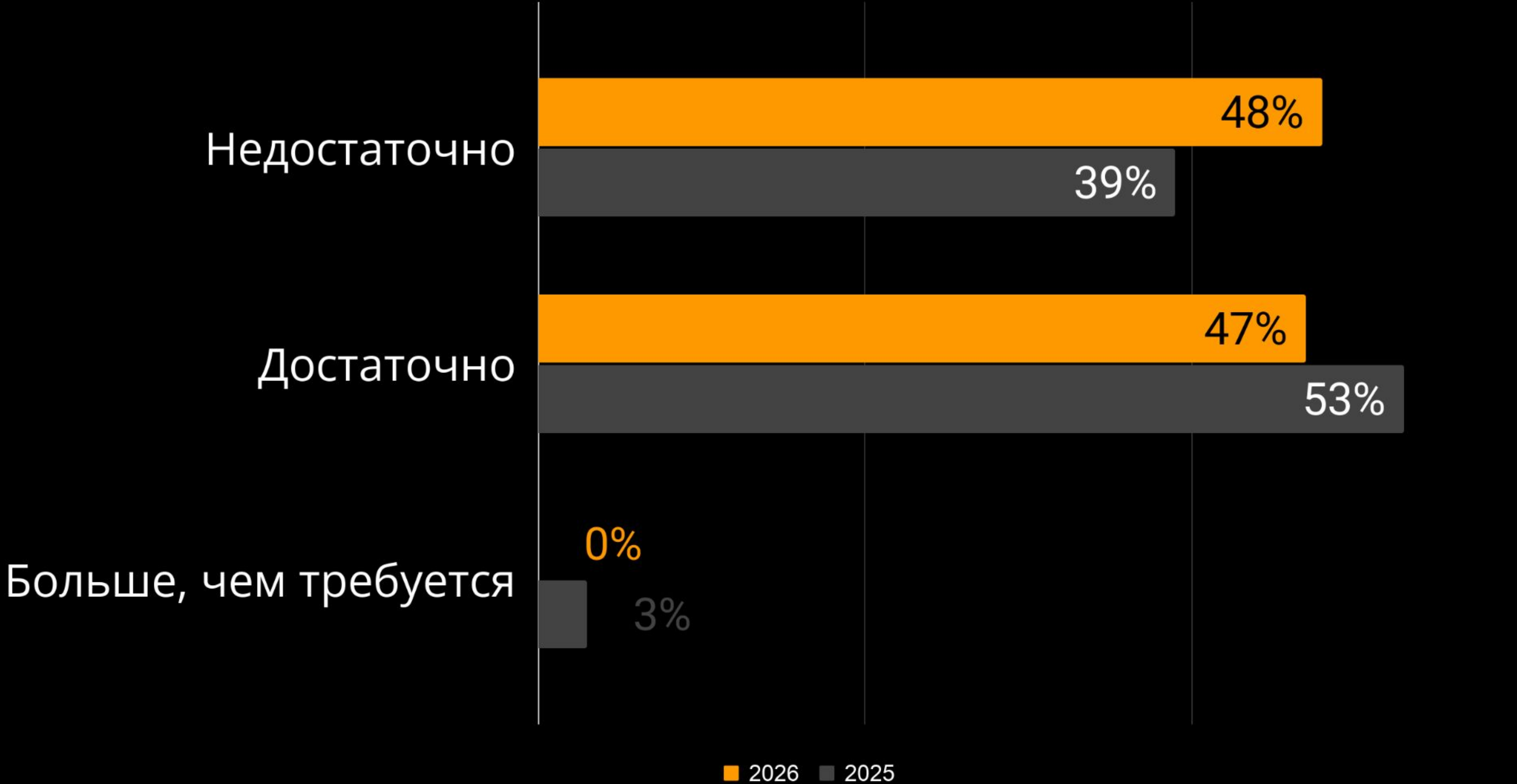
Люди



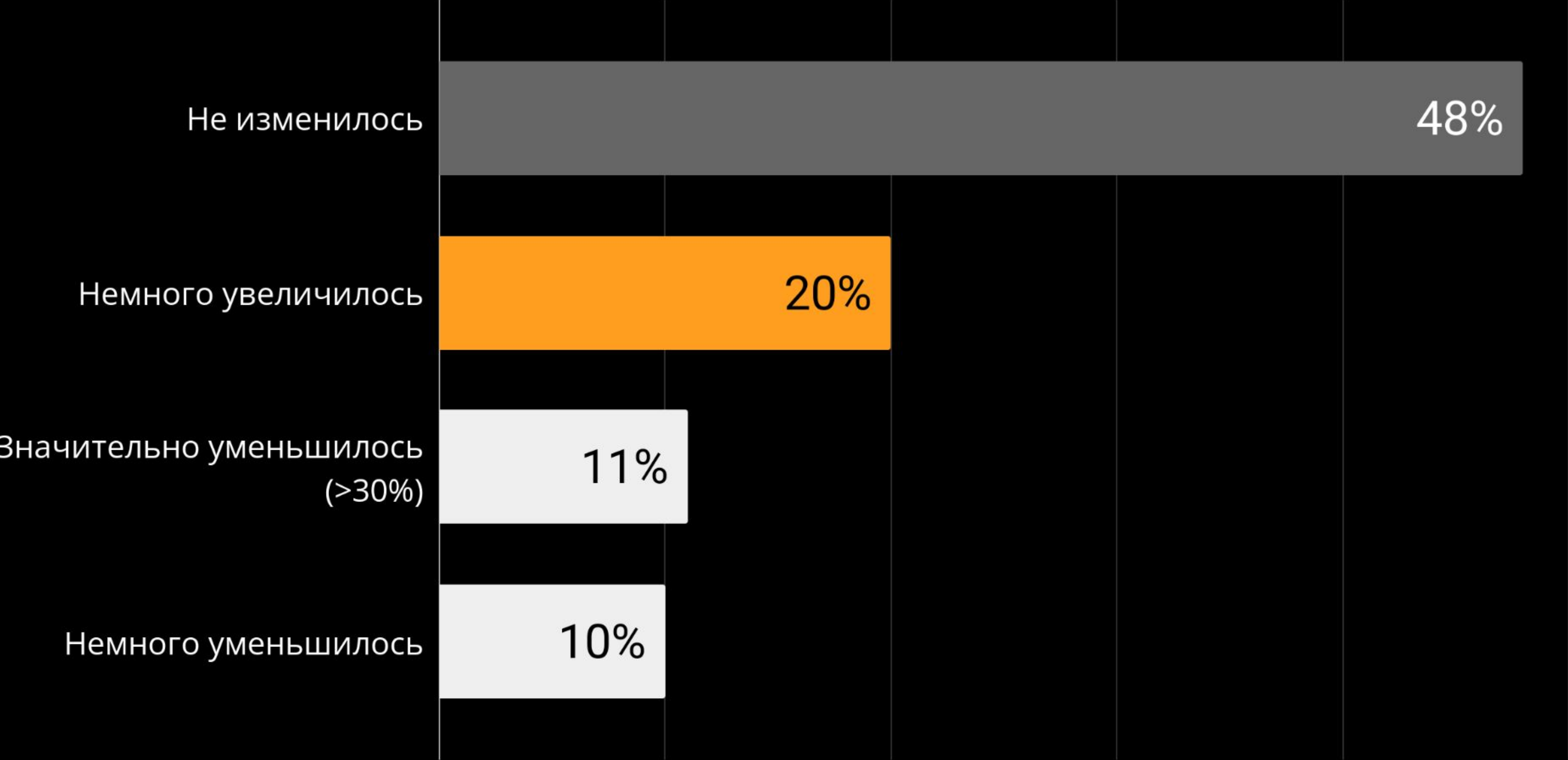
48%

Считает, что QA инженеров в команде недостаточно

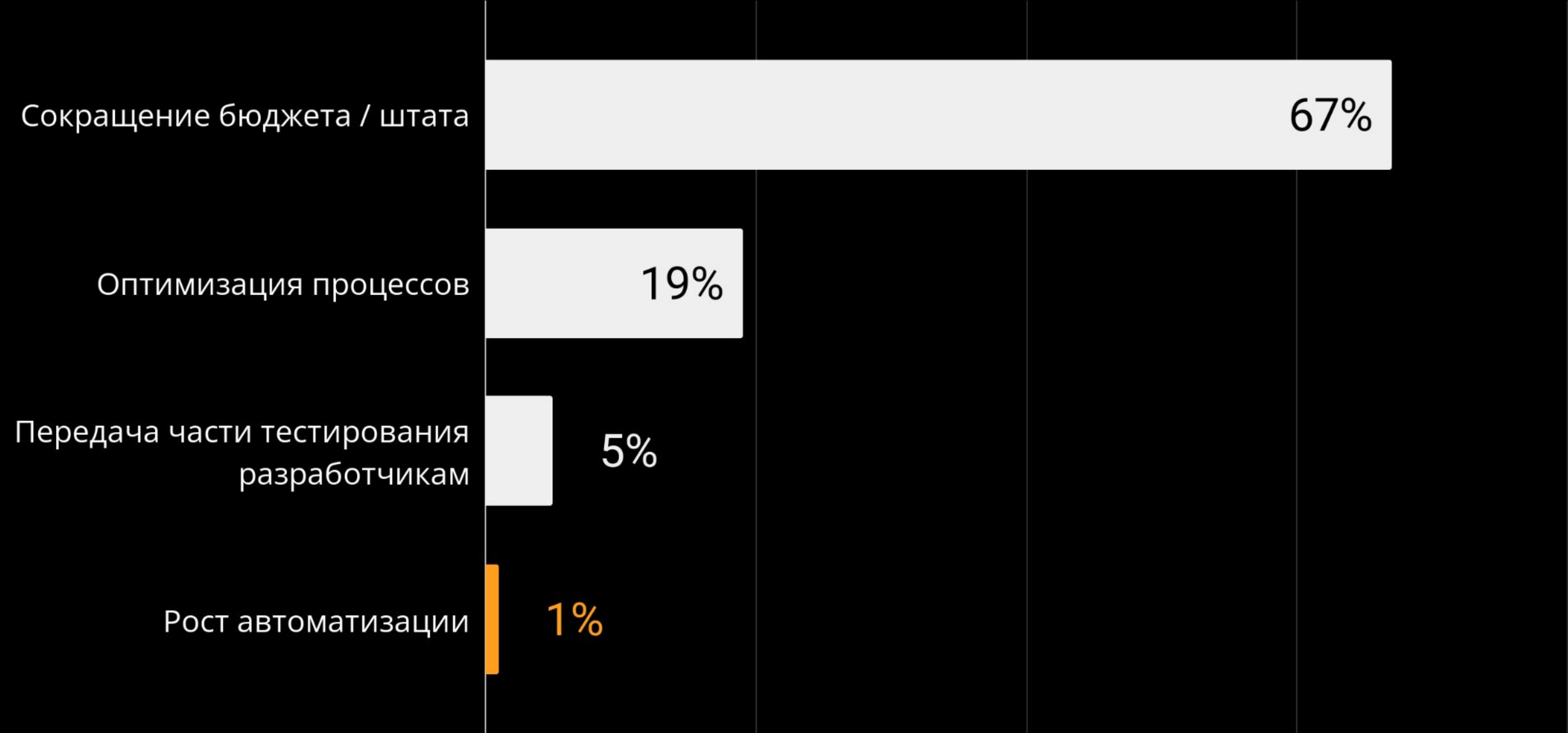
Количество QA-инженеров



Изменение количества QA



Почему уменьшилась QA-команда



Развитие



Как вы оцениваете **свою** эффективность?

69%

влияние на качество
продукта и снижение
рисков

53%

обратная связь от
команды / заказчика

51%

покрытие тестами

40%

найденные дефекты

28%

не формализовано,
субъективно

18%

через KPI / OKR от
руководства

Как вы **определяете эффективность** QA

81%

отсутствие
критических проблем
у пользователей

75%

улучшение качества
продукта

57%

снижение рисков
релиза

45%

скорость и
эффективность
процессов тестирования

35%

автоматизация
рутины

10%

количество
найденных дефектов

35%

Развиваются в сторону T-shaped

А помимо тестирования?

- 15% респондентов умеют в DevOps
- 36% знают контейнеризацию
- 13% имеют скиллы системного и бизнес анализа
- 13% занимаются backend-разработкой, 7% frontend
- 35% имеют навыки управления

Софт-скиллы

Умение выявлять риски 85%

Критическое мышление 82%

Коммуникация с командой и стейкхолдерами 80%

Софт-скиллы

Умение выявлять риски 85%

Критическое мышление 82%

Коммуникация с командой и стейкхолдерами 80%

Системное мышление 77% **Работа в команде** 71%

Адаптивность и способность быстро учиться 68%

Переговоры и аргументация решений 52% **Тайм-менеджмент** 50%

Эмпатия и понимание пользователей 34% **Презентация результатов и работа с обратной связью** 28%

Управление конфликтами 28%

Культура качества

	Сейчас
Качество — ответственность всей команды	61%
Фокус на пользователе и бизнес-ценности	45%
QA подключается на ранних этапах	36%
Постоянное улучшение процессов	35%
QA отвечает за стратегию качества и риски	27%

Культура качества

	Сейчас	Хотели бы
Качество — ответственность всей команды	61%	↑ 74%
Фокус на пользователе и бизнес-ценности	45%	↓ 36%
QA подключается на ранних этапах	36%	↑ 62%
Постоянное улучшение процессов	35%	↑ 53%
QA отвечает за стратегию качества и риски	27%	30%



AI подскажет путь

50%

Тестировщиков используют ИИ каждый день

Роль ИИ в тестировании

	Сейчас
Только справка и поиск	44%
Помогает, принимает решение QA	28%
Генерирует и запускает тесты сам	10%

Роль ИИ в тестировании

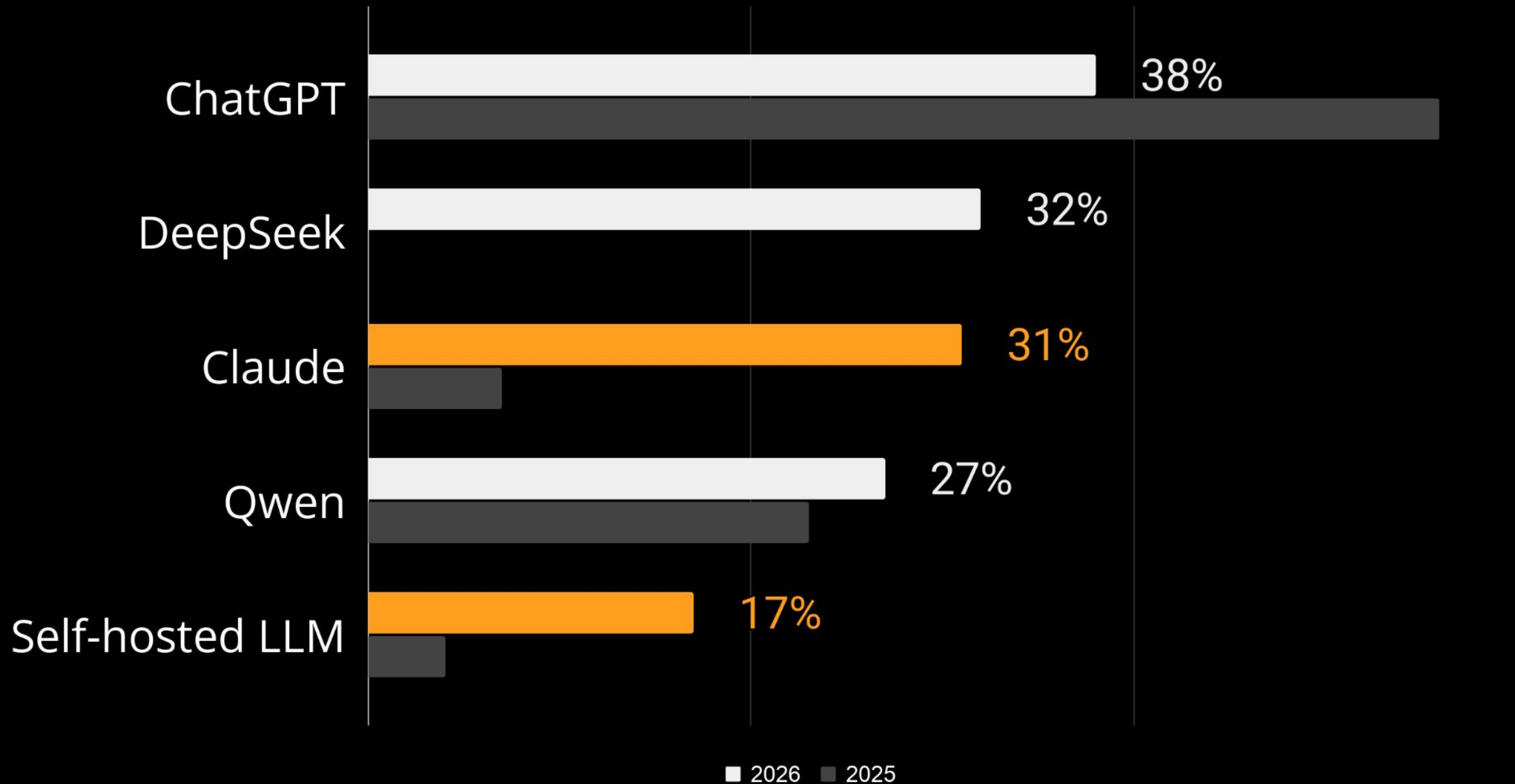
	Сейчас	Хотели бы
Только справка и поиск	44%	↓ 16%
Помогает, принимает решение QA	28%	↑ 40%
Генерирует и запускает тесты сам	10%	↑ 30%

ИИ - пока не исполнитель

С ЭТИМ СТОЛКНУЛИСЬ НЕ ТОЛЬКО МЫ

	Мы	Мир
Регулярно пользуются AI	81%	85%
Осторожны с результатом AI	94%	46%
AI работает как исполнитель	10%	15%
Не хватает навыков работы с AI	41%	50%

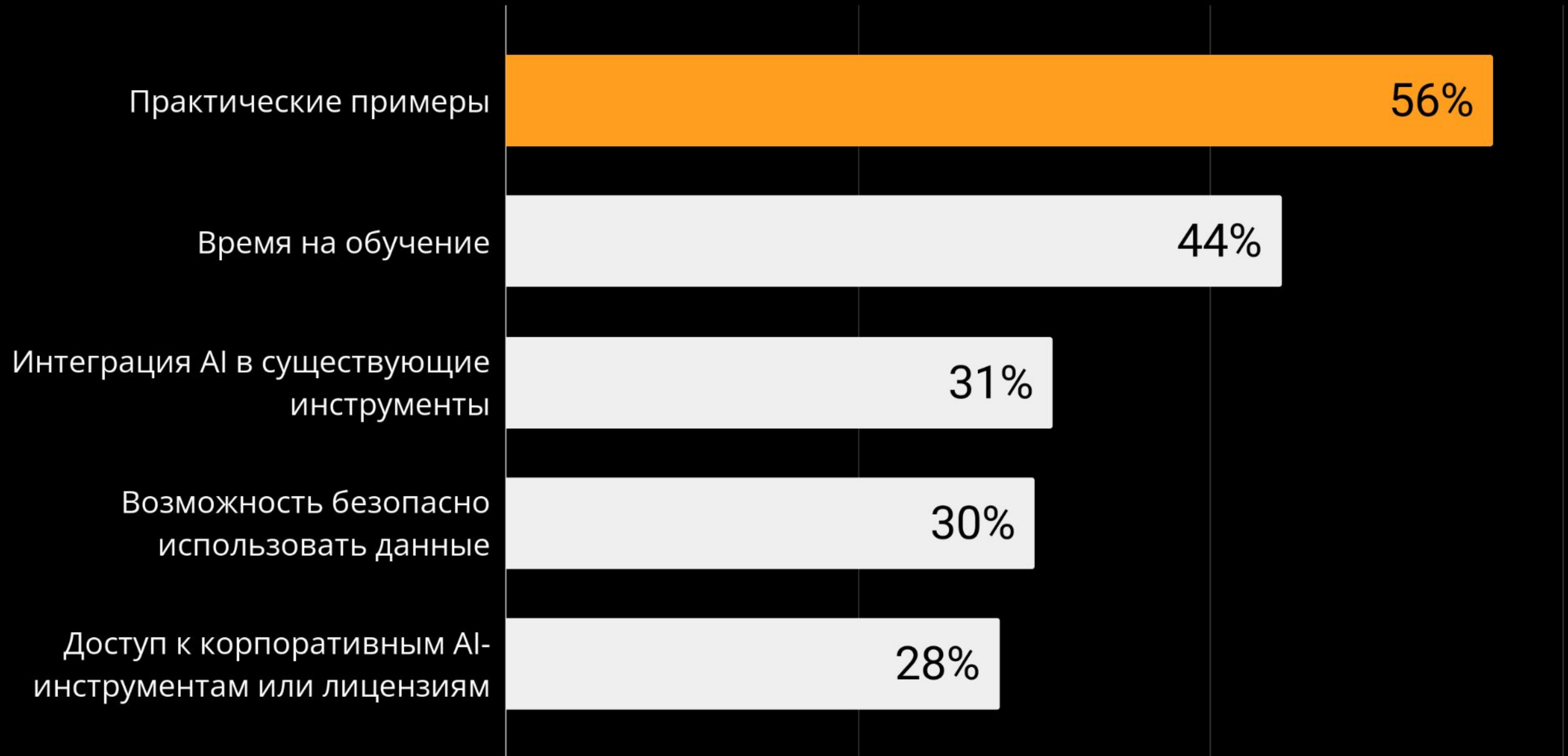
AI-инструменты



Что мешает внедрять ИИ



Что бы вам помогло внедрить ИИ



Кто платит за ИИ



AI-секция на конференции

AI-агенты для Auto QA:
от хаотичных запросов
к системной работе
с навыками



Никита Осипов Т-Банк

Как собрать своего
QAI-агента:
тестирование задач и
root-cause анализ



Александр Волков cloud.ru

Token FinOps: как
инженеры перестают
быть расходной статьёй



Евгений Гребенчиков Сбер

LangFuse для LLM-
агентов: как съесть
кактус тестирования,
не уколовшись



Артём Самойлов Сбер

UI-тесты с AI:
сравниваем подходы и
считаем токены



Алексей Тюрин

Инструкция к агенту:
что прячется за
каждой кнопкой



Константин Волков Veai



Зарплатный каньон

Желаемая зарплата

Уровень (грейд)	Желаемая зарплата												
	0-100 тыс	100-200 тыс	200-300 тыс	300-400 тыс	400-500 тыс	500-600 тыс	600-700 тыс	700-1000 тыс	1-1,5 млн	1,5-2 млн	2-3 млн	3-5 млн	>5 млн
Junior		2.6				0.4							
Middle	0.4	10.3	25.8	7.3	2.1	0.4							
Senior			9.9	9.0	5.2	2.6			0.4				
Lead / Principal / Staff		0.9	2.1	6.9	5.2	2.1		0.9	0.4				0.9
Manager			0.4	1.3	0.9	0.9		0.4					0.4

Желаемая зарплата

Опыт работы в IT	0-100 тыс	100-200 тыс	200-300 тыс	300-400 тыс	400-500 тыс	500-600 тыс	600-700 тыс	700-1000 тыс	1-1,5 млн	1,5-2 млн	2-3 млн	3-5 млн	>5 млн
Менее 1 года			0.4										
1–3 года		6.3	8.8	1.3	0.4	0.4							
4–6 лет	0.4	4.6	17.6	9.2	4.2	1.3							
7–10 лет		2.1	6.3	3.8	2.5	2.1			0.4				
Более 10 лет		1.7	5.4	9.6	5.9	2.5		1.3	0.4				1.3

Карьера



38%



Не видят понятных возможностей развития



У тестировщика нет цели,
есть только путь

Радар тестирования 2026: главное

Ключевые техники, инструменты, платформы и фреймворки по данным опроса (~250 ответов)

TECHRADAR TESTING 2026

● без изменений ▲ рост к 2025 ▼ снижение к 2025 ◆ нет данных 2025
○ пунктир и серый текст — данные только со скрытых слайдов

Техники и подходы

● 1	Функциональное тестирование	94%
◆ 2	Автоматизированное тестирование	77%
▲ 3	Тестирование GUI	▲ 67%
◆ 4	Синтетические тестовые данные	58%
● 5	ИИ как помощник (справка, поиск)	44%
● 6	Тестирование производительности	33%
◆ 7	AI-генерация тестовых данных	25%
▲ 8	Регулярное нагрузочное	▲ 22%
◆ 9	Контрактное тестирование	20%
● 10	Тестирование безопасности	15%
◆ 11	LLM-as-Judge	14%
◆ 12	Хаос-тестирование	12%
● 13	ИИ генерирует и запускает тесты	10%

Платформы и инфраструктура

▼ 26	GitLab CI	▼ 55%
◆ 27	Prometheus + Grafana	34%
◆ 28	Elastic Stack (ELK)	33%
▲ 29	Jenkins	▲ 32%
◆ 30	Эмуляторы / симуляторы	21%
▲ 31	Self-hosted LLM	▲ 17%
● 32	TeamCity	10%
▲ 33	Azure DevOps Pipelines	▲ 8%
◆ 34	OpenTelemetry	6%
▼ 35	GitHub Actions	▼ 3%



Инструменты

◆ 14	DBeaver	62%
▼ 15	Browser DevTools	▼ 60%
▼ 16	Postman	▼ 57%
◆ 17	curl	44%
◆ 18	Allure TestOps	40%
▼ 19	ChatGPT	▼ 38%
◆ 20	DeepSeek	32%
▲ 21	Claude	▲ 31%
▲ 22	Qwen	▲ 27%
◆ 23	Apache JMeter	14%
◆ 24	Cursor	12%
◆ 25	k6	9%

Языки и фреймворки

● 36	Playwright	34%
▼ 37	Python	▼ 32%
▲ 38	Selenium	▲ 30%
● 39	Java	29%
● 40	PyTest	26%
● 41	JUnit	21%
▲ 42	JavaScript / TypeScript	▲ 19%
◆ 43	«Язык не важен — код пишет AI»	18%
● 44	TestNG	8%
● 45	Cucumber	8%
● 46	Kotlin	7%
● 47	Appium	6%
● 48	Go	4%

**Что нам мешает быть
лучшим QA?**



Неправления, которые не смогли внедрить

AI 43%

Автотесты 31%

Нагрузочное тестирование 21%

Генерация тестовых данных 20%

А почему не получилось?

- 41% Не хватает знаний или опыта
- 20% Нет бюджета
- 17% Нет поддержки руководства
- 14% Нет инфраструктуры

65%

Не хватает времени на новые направления



Рулетка кейсов: тестирование провалено

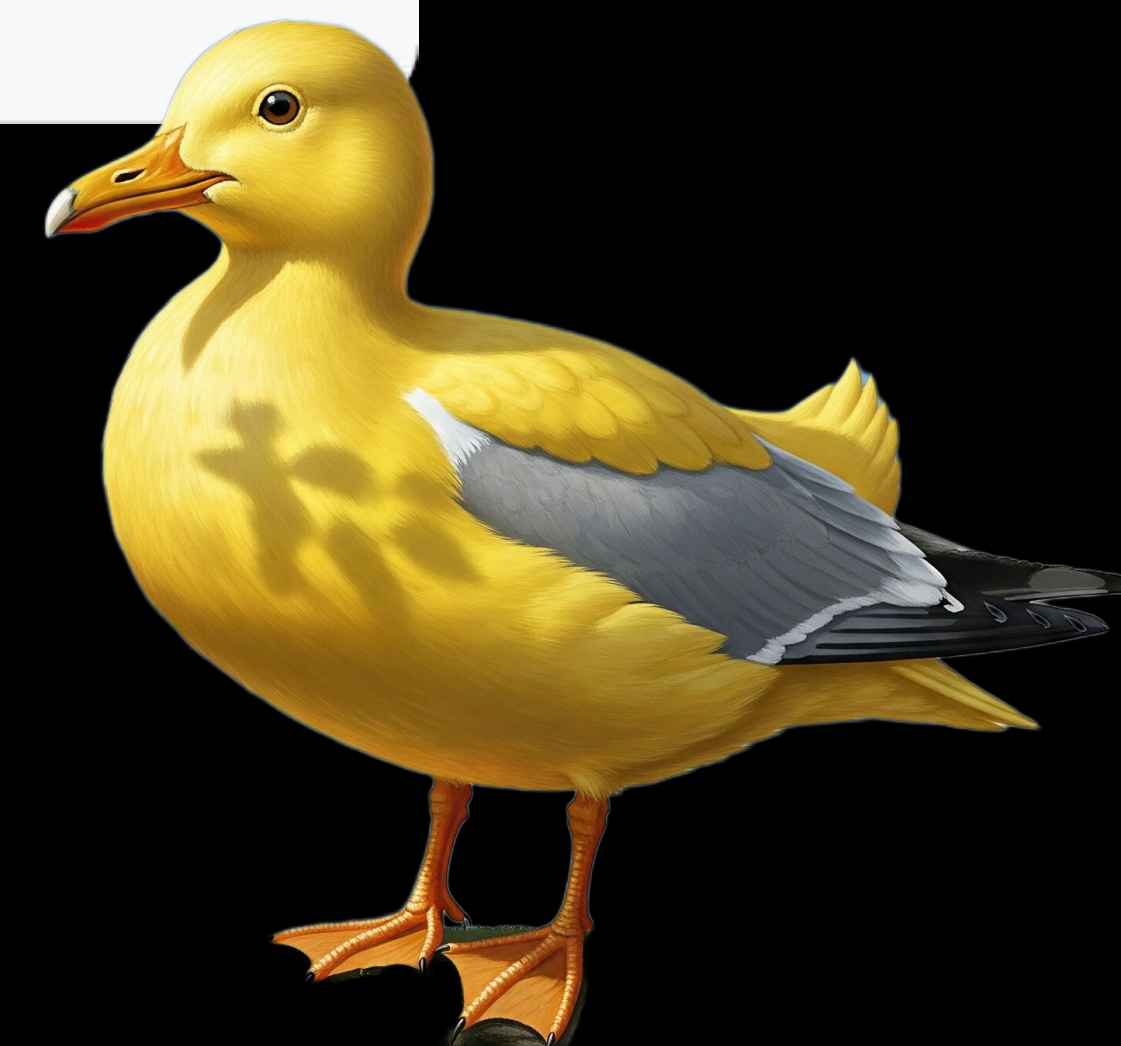
-  Дмитрий Тучс Dodo Engineering
-  Михаил Новотарский Сбер
-  Эмилия Куцарева VK

ЧУВС: что у вас сломалось?

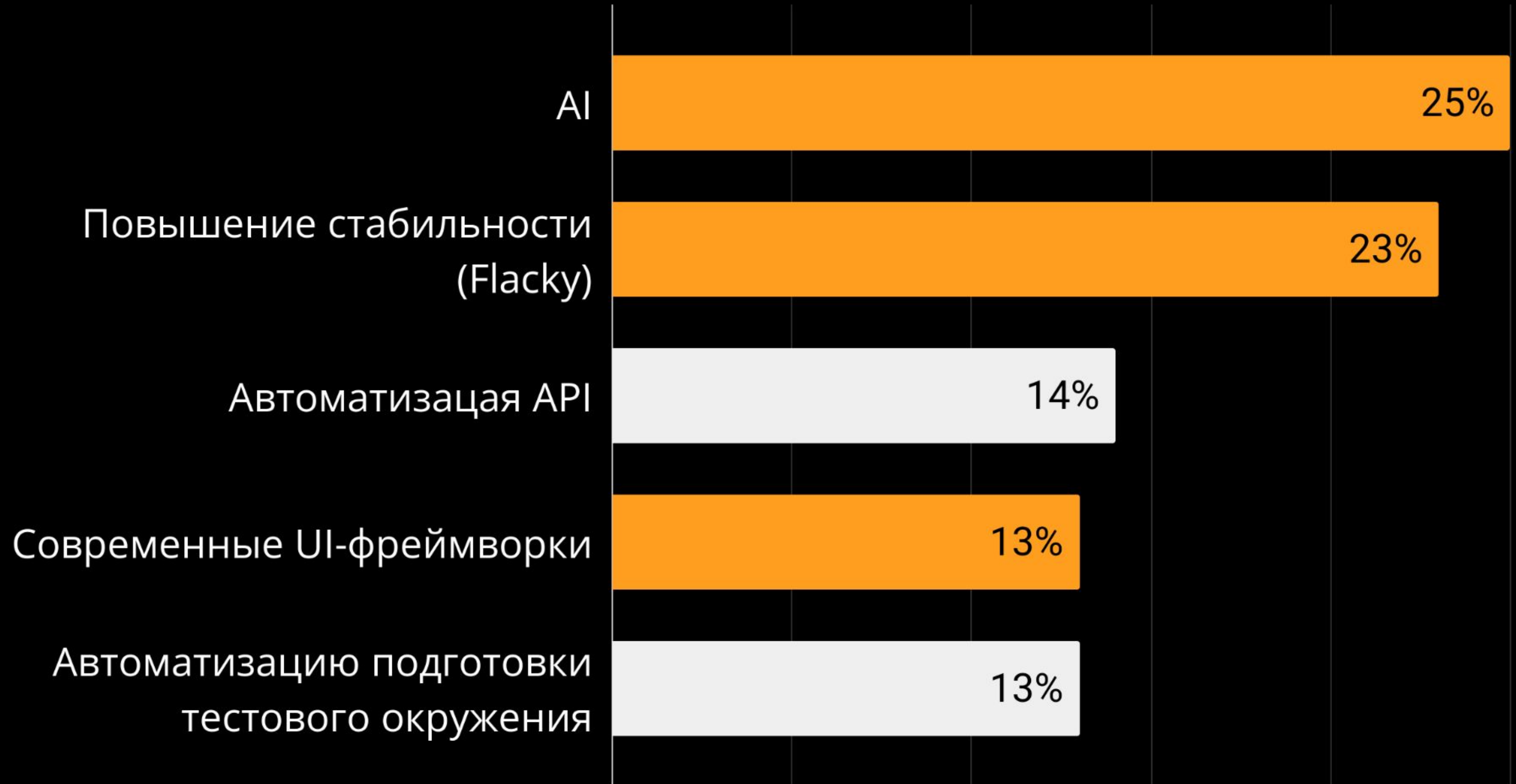
-  Дмитрий Тучс Dodo Engineering
-  Дарья Манухина Wildberries & Russ
-  Мария Палагина Т-Банк

Fail Talks

-  Михаил Гамаюнов VK



Что из автоматизации не удалось внедрить?



Почему?

Не хватает знаний или опыта 27%

Нет поддержки руководства 10% **Нет бюджета** 8%

Не хватает специалистов 8% **Нет инфраструктуры** 7%

Высокая стоимость внедрения 6%

Почему?

Не хватает времени 48%

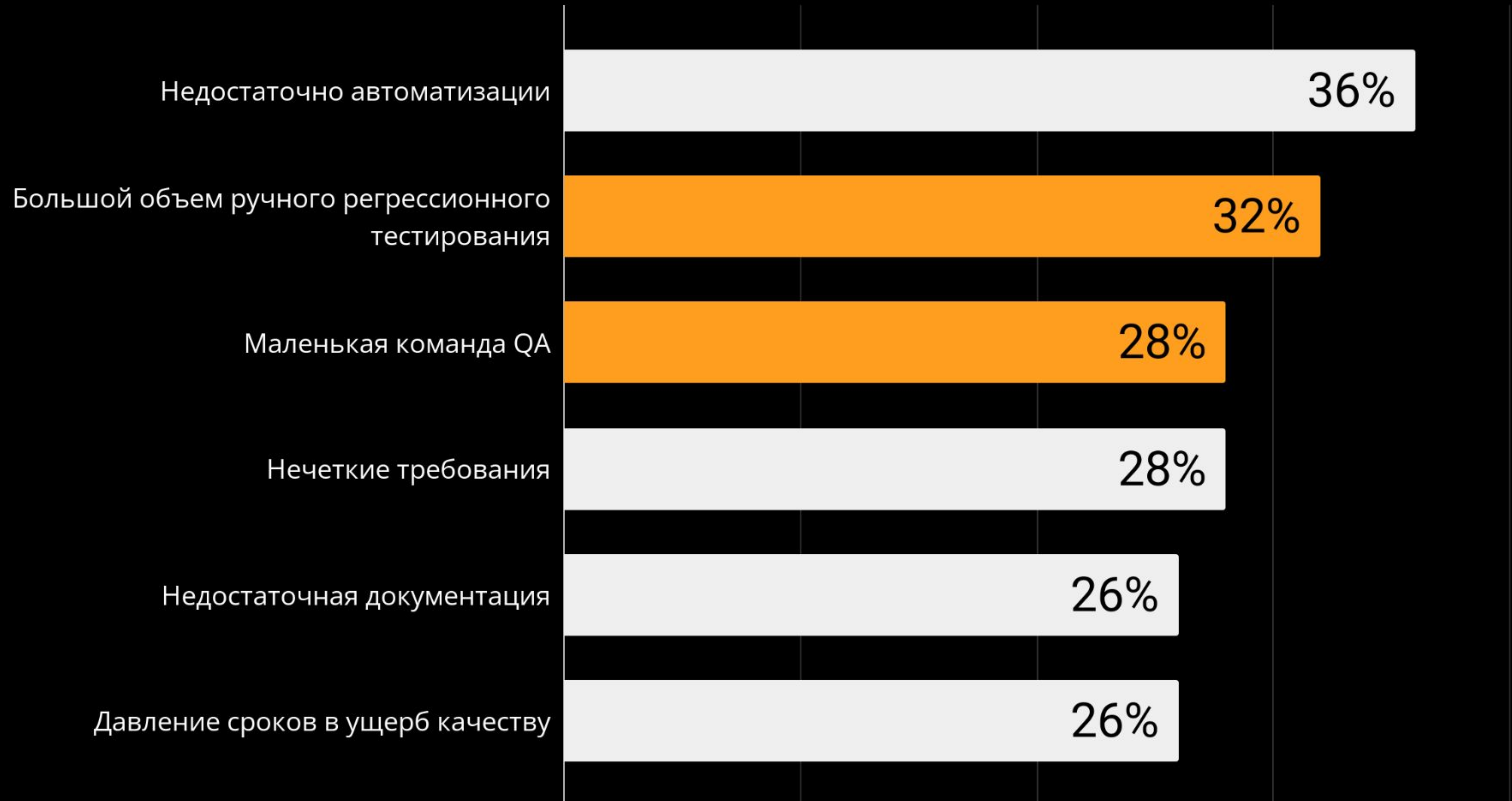
Не хватает знаний или опыта 27%

Нет поддержки руководства 10% **Нет бюджета** 8%

Не хватает специалистов 8% **Нет инфраструктуры** 7%

Высокая стоимость внедрения 6%

Что не устраивает в текущем процессе?



18%

Считают, что AI инструменты внедряются непродуманно

Что мешает это исправить?

- 35% Частые изменения требований и приоритетов
- 33% Давление бизнеса на скорость выпуска
- 24% Недостаточная вовлеченность QA на ранних этапах
- 24% Недостаточная автоматизация
- 23% Низкое качество документации
- 22% Текущие процессы устраивают, существенных проблем нет

Что мешает профессиональному развитию?



Уровень

Management

Lead / Staff

Engineer

Вход + AI

Сдвиг модели компетенций

Management: цели и контекст

Глубина

T-shape: своя экспертиза

Product engineer

Отвечает за результат: от задачи до прода

Продукт

UX

Архитектура

Код + AI

Качество

Метрики

AI-агенты берут рутину: шаблонный код, тесты, доки

Понять

Построить

Запустить

SA

Design

Dev

QA

DevOps

Основные боли

Технологии
меняются
слишком
быстро

Не хватает
времени
разбираться

Нужны
практические
примеры

Что делать?

Учиться на практике

Читать статьи, смотреть обучающие видео

ИИ невозможно оценить заранее
Рoadmap придётся построить для себя самим

Но, честно говоря, мы не знаем...

Мы еще много вам не рассказали!

12:00

Встречаемся в 4 зале

26 октября

**Продолжим обсуждение на
онлайн-митапе**



Продолжаем наблюдение