



Extreme Programming (XP) – Обзор методологии и практических инструментов

Extreme Programming (XP): Обзор методологии

Практические инструменты и их влияние на разработку

Что такое Extreme Programming?

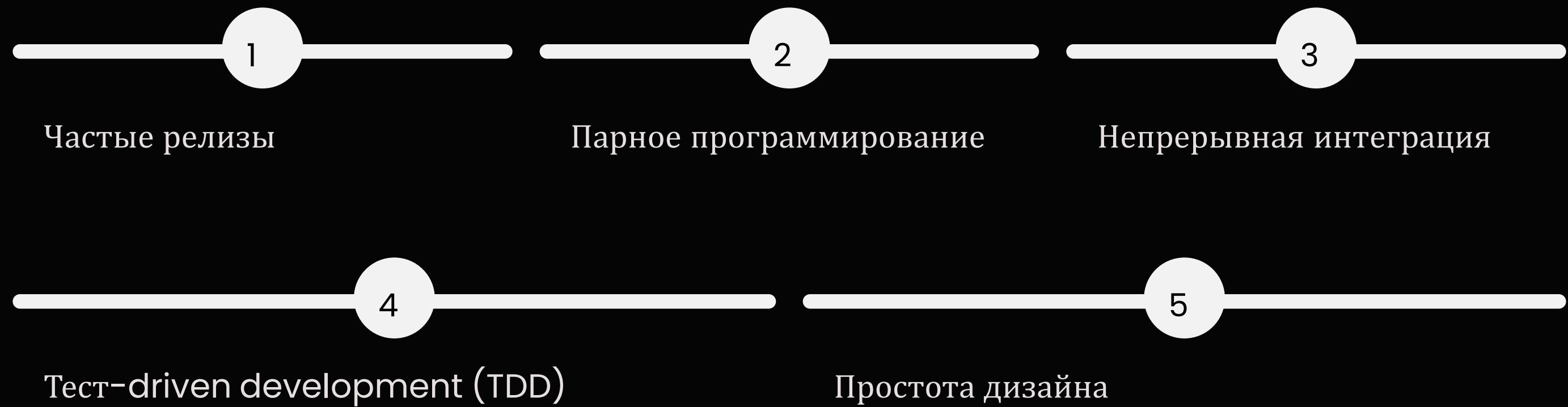
XP – гибкая методология разработки ПО

Фокус на:

- Качестве кода
- Скорости доставки
- Тесном взаимодействии с заказчиком

Основатель: Кент Бек (1990-е)

Основные принципы XP



Частые релизы

- Короткие итерации (1–2 недели)
- Быстрая доставка функциональности
- Постоянная обратная связь от заказчика

Пример: Выпуск MVP каждые 10 дней



Парное программирование



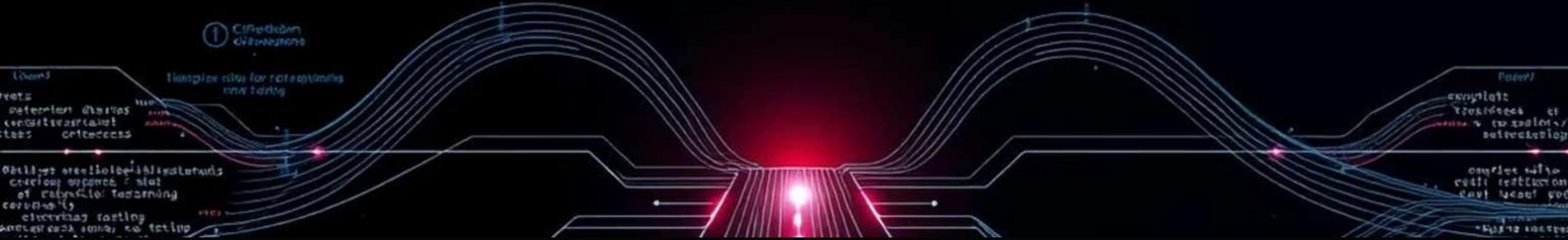
Два разработчика за одним компьютером



Роли: "Драйвер" и "Наблюдатель"

Преимущества:

- Меньше ошибок
- Обмен знаниями
- Высокое качество кода



Непрерывная интеграция

- Регулярное объединение кода
- Автоматическое тестирование
- Выявление ошибок на ранних стадиях

Инструменты: Jenkins, GitLab CI, Travis CI

Test-driven development (TDD)



Пишем тесты перед кодом

Цикл: Тест → Код → Рефакторинг

Преимущества:

- Надежный код
- Упрощение рефакторинга
- Высокое покрытие тестами



Простота дизайна

"Делай проще, чем кажется возможным"

- Избегай избыточной функциональности
- Постоянный рефакторинг

Пример: Реализация только текущих требований

Преимущества ХР

Высокое качество кода

Быстрая адаптация к
изменениям

Улучшение коммуникации в
команде

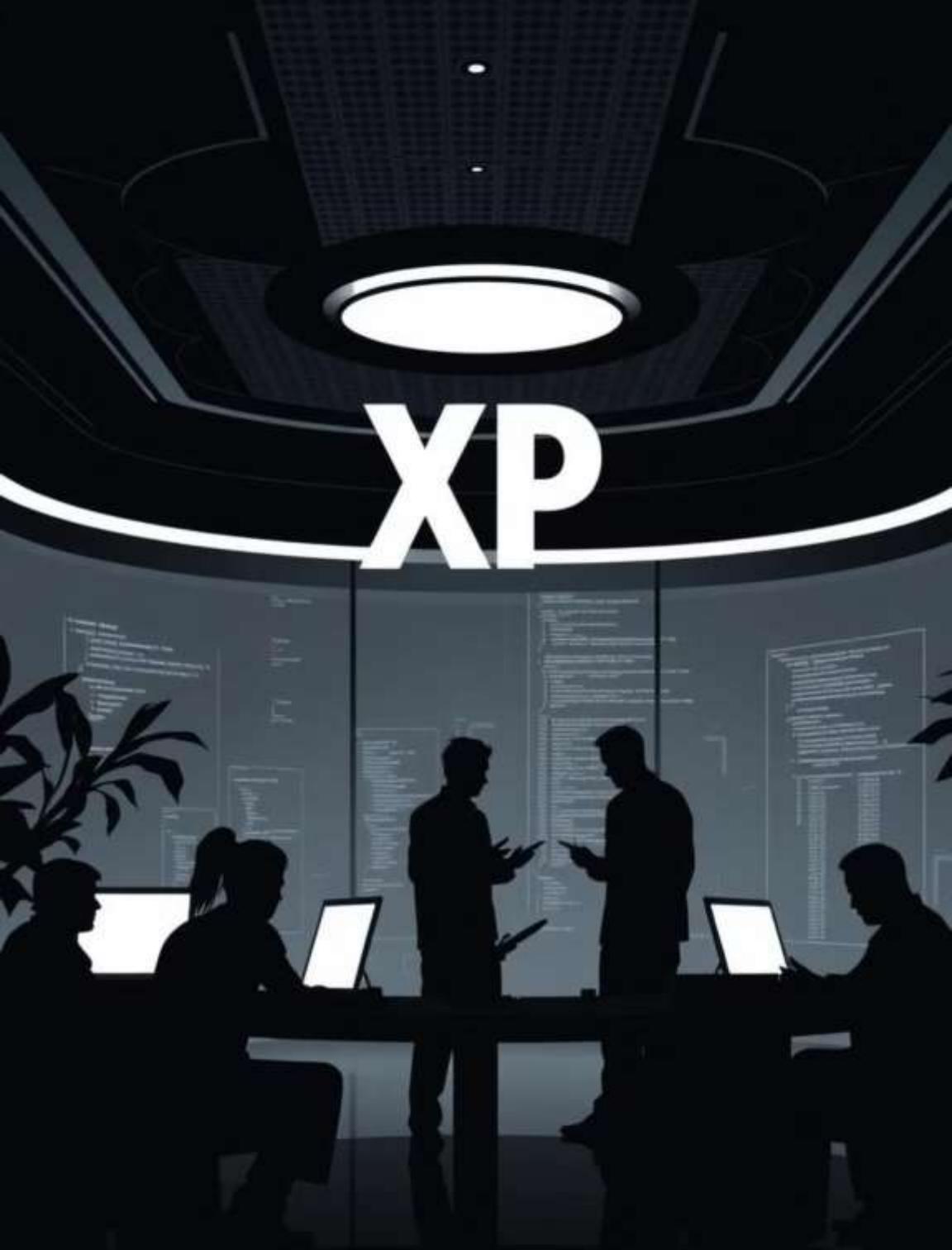
Доверие заказчика

Снижение рисков

XP и командное взаимодействие

- Ежедневные стендалы
- Совместное владение кодом
- Тесное сотрудничество с заказчиком

Пример: Регулярные демо для клиента



Сравнение XP с другими методологиями

Методология	Итерации	Фокус	Командная работа
XP	Короткие	Качество кода, клиент	Парное программирование
Scrum	2–4 недели	Управление процессом	Сprintы, роли
Kanban	Непрерывные	Поток задач	Визуализация

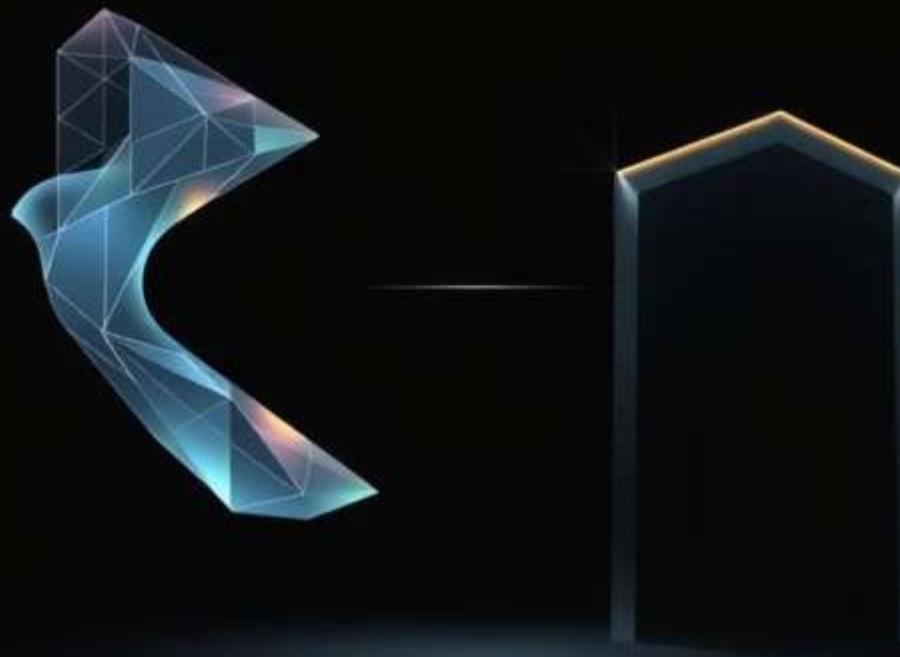
Когда использовать XP?

Подходит для:

- Проектов с частыми изменениями
- Небольших команд (5–10 человек)
- Высоких требований к качеству

Не подходит для:

- Жестко фиксированных требований
- Больших распределенных команд



Практические инструменты ХР



CI/CD:

Jenkins, GitLab CI



Тестирование:

JUnit, pytest



Управление задачами:

Jira, Trello



Код:

Git, GitHub, Bitbucket



Рефакторинг:

IDE (IntelliJ, VS Code)

Кейсы успеха XP

Chrysler C3 Project
(1990-е):

зарплатная система

Результат:

Высокое качество, быстрая
доставка

Современные примеры:

- Стартапы
- Финтех и EdTech проекты



Заключение

XP – мощная методология для гибкой разработки, имеющая ключевые преимущества:

Качество кода

Быстрая адаптация

Сильная командная работа