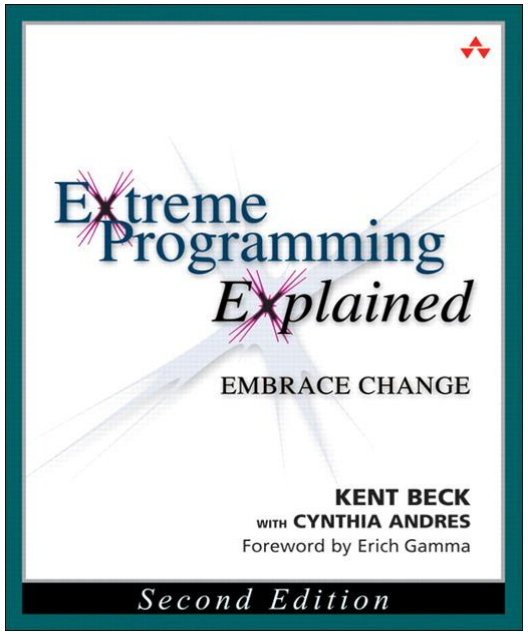
Экстремальное программирование (XP)

Дорожинский
Митина
Якимов
Черкасов
Кхай

Extreme Programming

XP отличается от других гибких методологий тем, что **применимо только в области разработки программного обеспечения**

Оно не может быть использовано в другом бизнесе или повседневной жизни, как scrum, kanban или lean



Extreme Programming

Цель методики XP — справиться с постоянно меняющимися требованиями к программному продукту и повысить качество разработки. Поэтому XP **хорошо подходит для сложных и неопределенных проектов**

Методология XP строится вокруг четырех процессов: кодирования, тестирования, дизайна и слушания. Кроме того, экстремальное программирование имеет ценности: простоту, коммуникацию, обратную связь, смелость и уважение

Вся команда

Все участники проекта с применением XP работают как одна команда

В нее обязательно входит представитель заказчика, лучше, если это будет реальный конечный пользователь продукта, разбирающийся в бизнесе

Игра в планирование

Планирование в XP проводят в два этапа — планирование релиза и планирование итераций

Частые релизы версий

В XP версии выпускаются часто, но с небольшим функционалом

Пользовательские тесты

Заказчик сам определяет автоматизированные приемочные тесты, чтобы проверить работоспособность очередной функции продукта

Команда пишет эти тесты и использует их для тестирования готового кода.

Коллективное владение кодом

В XP любой разработчик может править любой кусок кода, т.к. код не закреплен за своим автором

Кодом владеет вся команда

Непрерывная интеграция кода

Это значит, что новые части кода сразу же встраиваются в систему — команды XP заливают новый билд каждые несколько часов и чаще.

Стандарты кодирования

Когда кодом владеют все, важно принять единые стандарты оформления, чтобы код выглядел так, как будто он написан одним профессионалом

Можно выработать свои стандарты или принять готовые

Метафора системы

Метафора системы — это ее сравнение с чем-то знакомым, чтобы сформировать у команды общее видение

Обычно метафору системы продумывает тот, кто разрабатывает архитектуру и представляет систему целиком

Устойчивый темп

XP команды работают на максимуме продуктивности, сохраняя устойчивый темп

При этом экстремальное программирование негативно относится к переработкам и пропагандирует 40-часовую рабочую неделю

Разработка, основанная на тестировании

В XP тесты пишутся самими программистами, причем ДО написания кода, который нужно протестировать

При таком подходе каждый кусок функционала будет покрыт тестами на 100%

Парное программирование

Из двух вариантов решения проблемы выбирается лучший, код оптимизируется сразу же, ошибки отлавливаются еще до их совершения

В итоге имеем чистый код, в котором хорошо разбираются сразу двое разработчиков

Простой дизайн

Простой дизайн в XP означает делать только то, что нужно сейчас, не пытаясь угадать будущую функциональность

Простой дизайн и непрерывный рефакторинг дают синергетический эффект — когда код простой, его легко оптимизировать

Рефакторинг

-это процесс постоянного улучшения дизайна системы, чтобы привести его в соответствие новым требованиям.

Рефакторинг включает удаление дублей кода, повышение связности и снижение сопряжения

XP предполагает постоянные рефакторинги, поэтому дизайн кода всегда остается простым.

Принципы XP

- Простота
- Коммуникация
- Обратная связь
- Смелость
- Уважение

Алгоритм внедрения XP

- Тестирование
- Проектирование
- Планирование
- Менеджмент
- Разработка

Преимущества XP

- заказчик получает именно тот продукт, который ему нужен, даже если в начале разработки сам точно не представляет его конечный вид
- команда быстро вносит изменения в код и добавляет новую функциональность за счет простого дизайна кода, частого планирования и релизов
- код всегда работает за счет постоянного тестирования и непрерывной интеграции
- команда легко поддерживает код, т.к. он написан по единому стандарту и постоянно рефакторится

Недостатки XP

- успех проекта зависит от вовлеченности заказчика, которой не так просто добиться
- трудно предугадать затраты времени на проект, т.к. в начале никто не знает полного списка требований
- успех XP сильно зависит от уровня программистов, методология работает только с senior специалистами
- менеджмент негативно относится к парному программированию, не понимая, почему он должен оплачивать двух программистов вместо одного
- регулярные встречи с программистами дорого обходятся заказчикам

Выводы

- **Экстремальное программирование — гибкая методология**, в центре которой качественный работоспособный код с простой архитектурой.
  Ее предназначение — снизить уровень неопределенности в проектах и по-настоящему гибко реагировать на изменения требований к продукту.
- Никто не обязывает внедрять XP по принципу «все или ничего». В конце концов, гибкие методологии должны быть гибкими и в плане применения — подстраиваться под нужды конкретной команды и проекта.