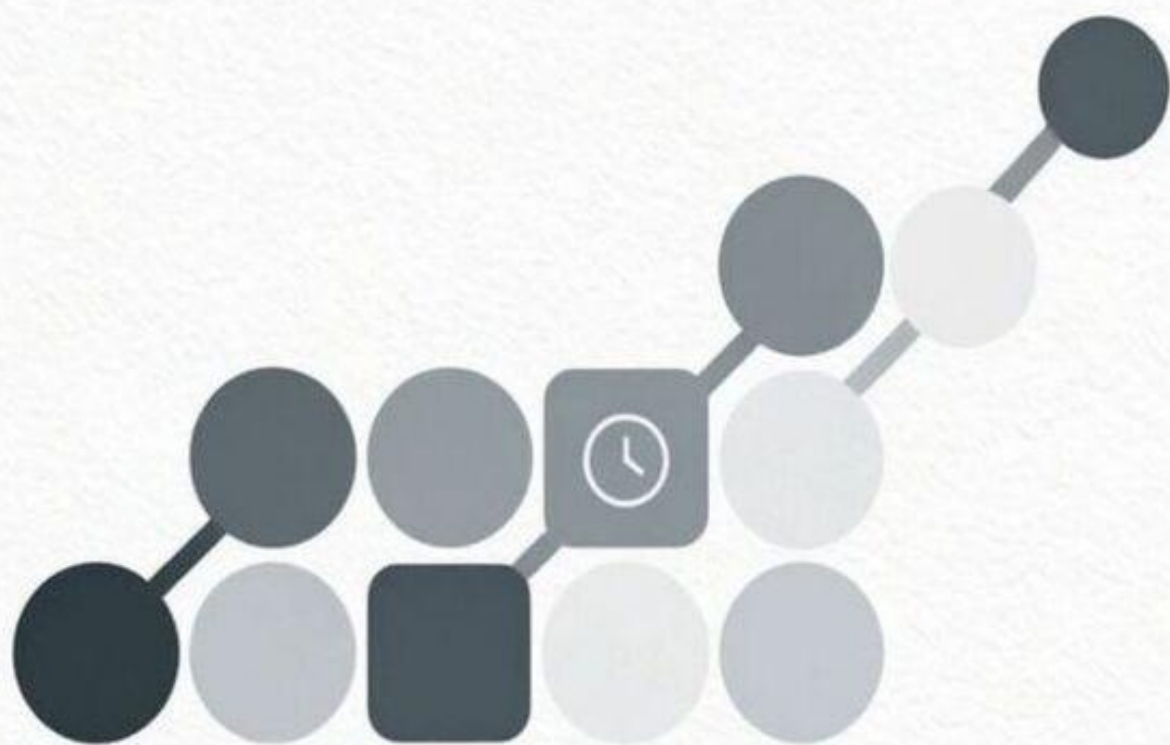


12+

ДМИТРИЙ БОЛЕСОВ



**РАБОТА КОМАНДЫ
В СТИЛЕ AGILE**

Дмитрий Болесов

Работа команды в стиле Agile

«Издательские решения»

Болесов Д.

Работа команды в стиле Agile / Д. Болесов — «Издательские решения»,

Практическое руководство по Agile для команд из разных сфер. Без сложной теории — только работающие инструменты: планирование, синхронизация, метрики, масштабирование. С упражнениями и примерами для B2B- и B2C-проектов.

© Болесов Д.

© Издательские решения

Содержание

Введение	6
Глава 1. Ценности и принципы Agile: что стоит за «гибкостью»	7
Четыре ценности	8
Двенадцать принципов простым языком	9
Заблуждения	10
Глава 2. Команда в Agile: роли, размер, компетенции	11
Кросс-функциональность	12
Размер команды	13
Роли	14
Самоорганизация	15
Глава 3. Цели и ценность: как команда понимает, что делает нужное	16
От задач к ценности	17
Product Backlog и приоритизация	18
Конец ознакомительного фрагмента.	19

Работа команды в стиле Agile

Дмитрий Болесов

© Дмитрий Болесов, 2026

ISBN 978-5-0071-1310-6

Создано в интеллектуальной издательской системе Ridero

Введение

Agile — слово, которое за последние годы превратилось из узкого термина из мира IT в нечто гораздо большее. Его произносят на совещаниях, пишут в вакансиях, добавляют в слоганы консалтинговых компаний. И часто за этим словом стоит не совсем то, что подразумевали его создатели.

Эта книга не о фреймворках. Не о Scrum-досках, не о Story Points и не о том, как проводить Daily Stand-up так, чтобы никому не было скучно. Хотя всё это тоже будет — но как инструмент, а не как цель. Главная тема книги — команда. Живые люди, которые работают вместе, создают что-то вместе, ошибаются вместе и учатся вместе.

Agile родился из простого наблюдения: Detailed plans, составленные на месяцы вперёд, почти всегда оказываются неверными к моменту их реализации. Мир меняется быстрее, чем мы успеваем его описать. И единственный способ справиться с этой неопределённостью — не строить жёсткие планы, а научиться быстро реагировать. Но быстро реагировать может не отдельный человек, а команда, в которой есть доверие, прозрачность и общее понимание цели.

Книга написана для тех, кто хочет понять, как Agile работает на практике — не в теории методического пособия, а в реальной жизни команд. Менеджеры, тимлиды, специалисты, предприниматели — всем, кому интересно, как организовать работу так, чтобы результат появлялся быстрее, а люди при этом не выгорали. Вам не нужен IT-бэкграунд: примеры взяты из разных сфер — от разработки приложений до маркетинга и образования.

Структура книги построена от общего к частному. Сначала — ценности и принципы, потому что без них любые ритуалы превращаются в пустую формальность. Затем — команда: кто в ней, как она устроена, что делает её эффективной. Дальше — практика: планирование, ежедневная работа, обратная связь, улучшения. И наконец — масштабирование, метрики и внедрение.

Читайте последовательно или используйте отдельные главы как справочник. Каждая глава заканчивается практическим блоком — упражнением или чек-листом, который можно применить сразу, не дочитав книгу до конца.

Глава 1. Ценности и принципы Agile: что стоит за «гибкостью»

В феврале 2001 года семнадцать человек собрались на горнолыжном курорте в штате Юта. Они представляли разные школы и подходы к разработке программного обеспечения, но их объединяло одно: все они были недовольны тем, как обычно работали проекты — с тяжёлыми планами, длинными циклами и результатом, который к моменту сдачи уже никому не был нужен.

Они написали короткий текст — Манифест Agile-разработки. В нём всего четыре ценности и двенадцать принципов. Никаких методологий, никаких схем процессов. Просто — что важнее чего.

Четыре ценности

Первая ценность: **люди и взаимодействия важнее процессов и инструментов**. Процессы нужны — без них хаос. Но когда процесс становится важнее человека, который по нему работает, что-то пошло не так. Команда, которая знает, зачем работает и доверяет друг другу, найдёт способ доставить результат даже с несовершенным процессом. Идеальный процесс в команде, где нет доверия, не спасёт.

Вторая: **работающий продукт важнее исчерпывающей документации**. Документация нужна, но она не самоцель. Сто страниц описания, которое не соответствует реальности, хуже, чем одна страница, которая актуальна. Работающий продукт — это то, что можно показать пользователю и получить обратную связь.

Третья: **сотрудничество с заказчиком важнее согласования условий контракта**. Это не значит, что контракты не нужны. Это значит, что отношения с заказчиком строятся не на защите пунктов договора, а на совместной работе над тем, чтобы результат был полезен. Если заказчик вовлечён в процесс, он получает то, что ему нужно, даже если первоначальные требования уже изменились.

Четвёртая: **готовность к изменениям важнее следования плану**. План — это гипотеза. Изменения — это реальность. Команда, которая умеет менять направление без боли и виноватых, сильнее команды, которая идеально выполняет устаревший план.

В каждой паре справа — не отрицание. Процессы важны. Документация важна. Контракты важны. Планы важны. Но когда приходится выбирать — выбирают то, что слева.

Двенадцать принципов простым языком

Принципы Agile конкретнее ценностей и дают ориентиры для повседневной работы. Вот их суть.

Самый важный — удовлетворение клиента за счёт ранней и непрерывной поставки ценности. Не через год, а через две недели. Не «всё сразу», а маленькими полезными кусками.

Изменения приветствуются даже на поздних этапах. Они не «поломают план» — они помогут сделать лучше. Agile-процесс harnesses change для преимущества клиента.

Работа должна выполняться небольшими итерациями — недели, не месяцы. Чем короче цикл, тем меньше риск, что команда уйдёт не туда.

Бизнес-люди и разработчики работают вместе ежедневно. Не «заказчик бросил ТЗ и ушёл» — а постоянный контакт.

Проект строится вокруг мотивированных людей. Дайте им среду и поддержку — и доверьте работу. Микроменеджмент убивает мотивацию.

Самый эффективный способ общения — личный разговор. Не email, не таск-трекер, не протокол совещания. Разговор.

Работающий продукт — главный показатель прогресса. Не часы, не задачи, не отчёты. То, что работает.

Agile-процессы поддерживают устойчивый темп. Спринты не марафон — это бег на среднюю дистанцию, который можно повторять снова и снова. Без выгорания.

Постоянное внимание к техническому совершенству и качеству улучшает гибкость. Не «сделаем быстро и криво, потом поправим» — а «сделаем хорошо, чтобы потом было легко менять».

Искусство максимизации незавершённой работы. Чем меньше задач «в воздухе», тем быстрее они доходят до конца. Незавершённая работа — это скрытый долг.

Самоорганизующиеся команды создают лучшие результаты. Когда люди сами решают, как делать работу, они делают её лучше.

Команда регулярно рефлексит и корректирует курс. Что хорошо — продолжать. Что плохо — менять. Без этого нет Agile.

Как ценности влияют на поведение

Ценности — не декорация. Они определяют, как команда принимает решения в ситуации выбора. Когда продукт-оунер говорит: «Нужно добавить фичу, которой не было в плане», — реакция зависит от ценностей. Команда без Agile-мышления скажет: «Это не в плане, мы не будем». Команда с Agile-мышлением скажет: «Хорошо, давайте посмотрим, что убрать из плана, чтобы это влезло, и обсудим приоритет».

Доверие — фундамент. Без него люди не будут говорить о проблемах, не будут предлагать идеи, не будут брать ответственность. Прозрачность — инструмент доверия. Когда все видят, что происходит, у кого какие задачи, что в работе, что заблокировано — меньше места для слухов и подозрений.

Готовность к изменениям — не слабость, а сила. Это не значит «меняем всё по первому требованию». Это значит «мы умеем оценивать, что изменение принесёт, и принимать осознанное решение».

Заблуждения

«Agile — это хаос». Нет. Agile — это дисциплина, но другая. Вместо жёсткого плана — короткие циклы с проверкой. Вместо «сделаем всё сразу» — «сделаем самое важное, проверим, повторим».

«При Agile нет планов». Есть. Но план — это не договор, а ориентир. Он меняется, когда меняется реальность.

«Agile — это только для IT». Нет. Agile-подход работает везде, где есть неопределённость и нужна обратная связь. Маркетинговые кампании, образовательные программы, запуск продукта на рынок — везде, где нельзя заранее знать, что сработает.

«Agile — это Scrum». Нет. Scrum — один из фреймворков. Есть ещё Kanban, Extreme Programming, Lean и другие. Agile — это ценности, а фреймворки — способы их применения.

Глава 2. Команда в Agile: роли, размер, компетенции

Agile начинается не с досок и спринтов. Он начинается с команды. Правильной команды.

Кросс-функциональность

В традиционном подходе команда часто устроена по принципу конвейера: аналитики передают ТЗ разработчикам, разработчики — тестировщикам, тестировщики — деплойщикам. Каждый делает свой кусок и передаёт дальше. Проблема — на стыках. Там, где одна роль передаёт работу другой, возникают задержки, недопонимание и потеря информации.

Кросс-функциональная команда — это команда, в которой есть все компетенции, необходимые для создания готового результата. Не «разработчики, которые иногда спрашивают дизайнера», а команда, в которой дизайнер, разработчик и тестировщик работают вместе над одной задачей от начала до конца.

Зачем это нужно? Чтобы минимизировать передачи. Каждая передача работы от одного человека к другому — это потеря контекста, время на объяснение и риск, что что-то поймут неправильно. Когда команда делает работу от начала до конца вместе, контекст не теряется.

«Узкий» специалист — человек, который умеет делать только одну вещь — это риск для команды. Если он заболел, уволился или просто занят — работа останавливается. В кросс-функциональной команде люди могут подстраховывать друг друга. Не значит, что каждый должен уметь всё. Значит, что у команды в целом достаточно компетенций, а отдельные люди могут заменять друг друга в смежных областях.

Размер команды

Знаменитое правило «двух пицц» Джеффа Безоса: команда должна быть такого размера, чтобы её можно было накормить двумя пиццами. На практике это 5—9 человек. Меньше пяти — трудно набрать все нужные компетенции. Больше девяти — коммуникация становится сложной, решения принимаются медленнее, появляются подгруппы и «коалиции».

Это не случайность. Исследования показывают, что коммуникационная сложность растёт быстрее, чем размер команды. Команда из 4 человек имеет 6 каналов общения. Команда из 10 — уже 45. При 20 человеках — 190. Каждый дополнительный человек добавляет не линейную, а квадратичную нагрузку на коммуникацию.

Если проект требует больше людей, ответ — не одна большая команда, а несколько команд, которые координируются между собой. Об этом — глава 9.

Роли

В Scrum роли определены чётко. Product Owner — человек, который отвечает за то, что команда делает. Он управляет бэклогом, приоритизирует задачи, принимает готовый результат. Это не «начальник», а представитель интересов клиента внутри команды. Хороший Product Owner — мост между бизнесом и командой: он знает, зачем каждая задача, и может объяснить её ценность.

Scrum Master — фасилитатор и защитник процесса. Он не раздаёт задачи и не контролирует выполнение. Он следит за тем, чтобы команда работала по правилам Scrum, помогает устранять препятствия и развивает самоорганизацию. Хороший Scrum Master — тот, кто со временем делает себя ненужным: команда учится самоорганизовываться и ему всё меньше нужно вмешиваться.

Developers — все, кто делает продукт. Не обязательно программисты. В маркетинговой команде это копирайтер, дизайнер, аналитик. В образовательной — методист, преподаватель, редактор. Название роли — «Developers» — условность; суть в том, что это люди, создающие результат.

В Kanban ролей меньше. Нет обязательных Product Owner и Scrum Master. Команда сама решает, как организовать работу. Это даёт гибкость, но требует больше зрелости: без чётких ролей легче скатиться в неструктурированную работу.

Самоорганизация

Самоорганизация — не «каждый делает, что хочет». Это команда, которая сама решает, как достичь цели. Менеджер не раздаёт задачи — он ставит цель. Команда сама определяет, кто что делает, как делает и в каком порядке.

Это требует трёх вещей: ясной цели, компетенции принимать решения и ответственности за результат. Без цели — хаос. Без компетенции — решения будут некачественные. Без ответственности — безразличие.

Самоорганизация не приходит по приказу. Она растёт, когда люди видят, что их решения имеют значение, что их не накажут за ошибку, что результат зависит от них. Менеджер, который хочет самоорганизацию, должен начать с делегирования — постепенно, с малого, но последовательно.

Практический блок: чек-лист «Готова ли ваша команда к Agile»

Ответьте на каждый пункт «да» или «нет».

- В команде есть все компетенции, необходимые для создания готового результата.
- Команда состоит из 5—9 человек.
- Каждый член команды понимает цель проекта и может объяснить её своими словами.
- Команда может принимать решения о том, как делать работу, без согласования с руководством на каждом шагу.
- В команде есть человек, который отвечает за приоритеты (что делать, а что — нет).
- Команда регулярно показывает результат тем, кто будет им пользоваться.
- Ошибки обсуждаются открыто, без поиска виноватых.
- Команда может изменить план, если обстоятельства изменились.

Если «да» на шесть и более — команда готова. Если меньше — стоит начать с работы над ценностями и доверием, прежде чем вводить ритуалы.

Глава 3. Цели и ценность: как команда понимает, что делает нужное

Самая частая проблема Agile-команд — не в процессе, а в фокусе. Команда делает задачи, закрывает тикеты, проходит спринты — но результат не приносит пользы. Причина: фокус на активности, а не на ценности.

От задач к ценности

Традиционная задача звучит так: «Сделать форму обратной связи на странице контактов». Это описание действия. Но зачем? Какую проблему это решает? Что изменится, когда это будет сделано?

Задача, ориентированная на ценность, звучит иначе: «Клиент, который хочет связаться с компанией, может быстро отправить сообщение и получить ответ». Разница — в фокусе. Первая задача про кнопку. Вторая — про человека и его потребность.

Метрики результата, а не активности: сколько пользователей воспользовались формой, сколько получили ответ, сколько остались довольны. Не «сколько задач закрыла команда», а «какую проблему решила».

Product Backlog и приоритизация

Product Backlog — упорядоченный список всего, что может понадобиться сделать. Это не список задач на спринт. Это список идей, требований, улучшений, исправлений — всего, что может принести ценность.

Ключевое слово — «упорядоченный». В бэклоге важна не полнота, а порядок. Самое важное — наверху. Менее важное — ниже. Команда всегда берёт сверху, и если бэклог упорядочен правильно, команда всегда работает над самым важным.

Как определять «важное»? Несколько методов.

MoSCoW: делит всё на четыре категории — Must (обязательно), Should (важно), Could (желательно), Won't (не сейчас). Простой и быстрый метод, хорош для первого прохода.

ICE: каждая задача оценивается по трём параметрам — Impact (влияние), Confidence (уверенность), Ease (лёгкость реализации). Баллы перемножаются, и задачи ранжируются. Подходит для экспериментов и гипотез.

Конец ознакомительного фрагмента.

Текст предоставлен ООО «Литрес».

Прочитайте эту книгу целиком, [купив полную легальную версию](#) на Литрес.

Безопасно оплатить книгу можно банковской картой Visa, MasterCard, Maestro, со счета мобильного телефона, с платежного терминала, в салоне МТС или Связной, через PayPal, WebMoney, Яндекс.Деньги, QIWI Кошелек, бонусными картами или другим удобным Вам способом.