

Коллектив авторов
Agile. Практическое руководство



Текст предоставлен издательством
«Agile: практическое руководство»: Олимп–Бизнес; Москва; 2018
ISBN 978-5-9693-0403-1, 978-1-62825-418-1

Аннотация

«Публикуемые Институтом управления проектами (Project Management Institute, Inc., сокращенно PMI) стандарты и руководства, к числу которых принадлежит и данный документ, разработаны согласно процессу разработки стандартов на основе добровольного участия и общего консенсуса. В ходе такого процесса объединяются усилия волонтеров и/или сводятся воедино замечания и мнения лиц, заинтересованных в предмете, которому посвящено данное издание. Хотя PMI администрирует этот процесс и устанавливает правила, гарантирующие непредвзятость при достижении консенсуса, PMI не занимается написанием документа, а также независимым тестированием, оценкой и проверкой точности или полноты материала, содержащегося в издаваемых PMI стандартах и руководствах. Подобным же образом, PMI не занимается проверкой обоснованности мнений, высказанных в этих документах...»

В формате PDF A4 сохранен издательский дизайн.

Agile: практическое руководство

Подана заявка на библиографическую запись в Библиотеке Конгресса США.

Опубликовано:

Project Management Institute, Inc.

14 Campus Boulevard

Newtown Square, Pennsylvania 19073-3299 США

Телефон: +1 610-356-4600

Факс: +1 610-356-4647

Эл. почта: customercare@pmi.org

Интернет: www.pmi.org/

Материалы Project Management Institute, Inc. охраняются авторским правом в соответствии с законом США об интеллектуальной собственности, который признан в большинстве стран.

Для переиздания или воспроизведения материалов PMI вам необходимо получить наше разрешение. Для получения более подробной информации посетите www.pmi.org/permissions.

Для размещения торгового заказа или получения информации о расценках обратитесь в Independent Publishers Group:

Independent Publishers Group

Order Department

814 North Franklin Street

Chicago, IL 60610 США

Телефон: +1 800-888-4741

Факс: +1 312- 337-5985

Эл. почта: orders@ipgbook.com (только для заказов)

По всем остальным вопросам обращайтесь в PMI Book Service Center.

PMI Book Service Center

P.O. Box 932683, Atlanta, GA 31193-2683 США

Телефон: 1-866-276-4764 (в США или Канаде) или +1-770-280-4129 (по всему миру)

Факс: +1-770-280-4113

Эл. почта: info@bookorders.pmi.org

Напечатано в Соединенных Штатах Америки. Запрещается воспроизведение или передача в любой форме или любыми средствами, электронными, ручными, путем фотокопирования,

записи или с помощью любой системы хранения и извлечения информации любой части данного издания без предварительного разрешения издателя.

PMI, логотип PMI, PMBOK, OPM3, PMP, CAPM, PgMP, PfMP, PMI-RMP, PMI-SP, PMI-ACP, PMI-PBA, PROJECT MANAGEMENT JOURNAL, PM NETWORK, PMI TODAY, PULSE OF THE PROFESSION и девиз MAKING PROJECT MANAGEMENT INDISPENSABLE FOR BUSINESS RESULTS. являются товарными знаками Project Management Institute, Inc. Для получения полного списка товарных знаков PMI обратитесь в юридический отдел PMI. Все остальные товарные марки, знаки обслуживания, торговые наименования, торговое оформление, названия продуктов и логотипы, появляющиеся в данном документе, являются собственностью их соответствующих владельцев. Любые права, не переданные в явной форме в настоящем документе, принадлежат владельцу авторского права.

SAFe является зарегистрированной маркой компании Scaled Agile, Inc.

Agile Alliance и логотип Agile Alliance являются торговыми знаками Agile Alliance.

Финансирование издания настоящего Практического руководства и его подготовка осуществлялись совместно с Agile Alliance®.

Agile Alliance® не занимается вопросами утверждения методологий и сертификатов agile.

Все права защищены. Воспроизведение всей книги или ее части в любом виде воспрещается без письменного разрешения издателя.

© 2017 Project Management Institute, Inc. Все права защищены.

© Перевод на русский язык, издание, оформление издательство «Олимп – Бизнес», 2018

Уведомление

Публикуемые Институтом управления проектами (Project Management Institute, Inc., сокращенно PMI) стандарты и руководства, к числу которых принадлежит и данный документ, разработаны согласно процессу разработки стандартов на основе добровольного участия и общего консенсуса. В ходе такого процесса объединяются усилия волонтеров и/или сводятся воедино замечания и мнения лиц, заинтересованных в предмете, которому посвящено данное издание. Хотя PMI администрирует этот процесс и устанавливает правила, гарантирующие непредвзятость при достижении консенсуса, PMI не занимается написанием документа, а также независимым тестированием, оценкой и проверкой точности или полноты материала, содержащегося в издаваемых PMI стандартах и руководствах. Подобным же образом, PMI не занимается проверкой обоснованности мнений, высказанных в этих документах.

PMI не несет ответственность за какие-либо травмы, повреждения, нанесенные собственности, или какие-либо другие убытки, будь то реальные, косвенные или компенсаторные, произошедшие непосредственно или косвенно вследствие издания, применения или использования данного документа. PMI не несет ответственность и не предоставляет гарантию, прямую или предполагаемую, относительно точности или полноты любого материала, содержащегося в данном документе, а также не несет ответственность и не предоставляет гарантию того, что содержащаяся в данном документе информация отвечает каким-либо вашим целям или нуждам. PMI не предоставляет гарантию относительно качества каких-либо продуктов или услуг отдельного производителя или продавца, проистекающего из использования данного стандарта или руководства.

Издавая и распространяя данный документ, PMI не оказывает профессиональные или иные услуги какому-либо лицу или организации или от имени какого-либо лица или организации; также PMI не выполняет обязательства какого-либо лица или организации по отношению к какой-либо третьей стороне. При использовании данного документа использующее его лицо должно самостоятельно определять действия, необходимые в конкретных обстоятельствах,

полагаясь при этом исключительно на свое суждение или, при необходимости, на совет компетентного профессионала. Информация относительно темы, освещаемой данным документом, или относящиеся этой теме стандарты могут быть получены из других источников, к которым пользователь может при необходимости обратиться, чтобы получить дополнительную информацию, не содержащуюся в данном документе.

PMI не имеет полномочий и не берет на себя обязательства по контролю за соответствием существующих практик содержанию данного документа или приведению этих практик в соответствие с данным документом. PMI не занимается сертификацией, проведением контрольных испытаний или инспекций в отношении продуктов, проектов или конструкций на предмет безопасности их эксплуатации или безопасности для здоровья потребителей. Любой сертификат или иное утверждение соответствия какой-либо информации относительно безопасности эксплуатации или безопасности для здоровья, содержащейся в данном документе, не могут быть приписаны PMI; в таком случае ответственность лежит всецело на лице, выдавшем сертификат или высказавшем такое утверждение.

Предисловие

Институт управления проектами (Project Management Institute, PMI) и Agile Alliance® создали настоящее Практическое руководство с целью достичь более глубокого понимания подходов agile в своих сообществах. Назначение настоящего практического руководства состоит в том, чтобы наделить команды проектов инструментами, ситуационными принципами и пониманием существующих методов и подходов agile, которые позволяют добиться лучших результатов.

Команды проектов применяют подходы agile не только в области разработки программных продуктов, но и в самых разных отраслях. Обе организации осознают, что в результате более широкого охвата возникла необходимость в создании общего языка, отказе от предвзятого мышления и готовности гибко подходить к тому, как продукты и поставляемые результаты выводятся на рынок. Кроме этого, обе организации исходят из того, что существует множество способов обеспечить успешную поставку. Имеется широкий набор инструментов, методов и фреймворков, и у команд есть выбор подходов и практик, отвечающих особенностям их проектов и организационных культур, которые они могут использовать для достижения желаемого конечного результата.

Члены основного комитета разработчиков *Практического руководства agile* имеют разный практический опыт и применяют различные подходы. Некоторые из членов комитета работают консультантами, а другие – непосредственно в организациях. Но все они уже многие годы ведут работу в направлении agile.

1. Введение

Предлагаем вашему вниманию «*Agile: практическое руководство*» (*Agile Practice Guide*)! Настоящее Руководство является результатом совместной работы Project Management Institute (PMI) и Agile Alliance®. Среди участников основной команды разработчиков текста настоящего Руководства были добровольцы из обеих организаций, опирающиеся на экспертные знания данной предметной области, полученные от широкого круга действующих практиков и руководителей с самыми различными практическим опытом, представлениями и культурой.

Настоящее Руководство содержит практические указания, предназначенные для использования руководителями проектов и членами команд в процессе перехода к подходу agile при планировании и исполнении проектов. Хотя наша основная команда разработчиков текста понимает, что существует устойчивая поддержка в пользу предиктивных подходов, и, с другой стороны, стремление переходить к образу мышления, ценностям и принципам agile, настоящее Практическое руководство освещает практический подход к agility (гибкости)

проектов. Настоящее Руководство служит ключом к пониманию пути перехода от предиктивного подхода к подходу agile. В принципе, эти два подхода предусматривают аналогичные виды деятельности, например, планирование, которые, хотя и осуществляются по-разному, но обязательно выполняются в обеих средах.

Наша основная команда разработчиков текста применяла на практике образ мышления agile при разработке этого первого издания Руководства. По мере изменения технологий и культуры в Руководство будут вноситься изменения и дополнения, отражающие текущие подходы.

Наша основная команда при подготовке настоящего Руководства решила использовать относительно неофициальный и свободный стиль изложения по сравнению с принятым типичным стилем стандартов PMI. Руководство включает новые элементы, например, полезные советы, боковые вставки и практические примеры, для улучшения наглядности изложения ключевых положений и концепций. Наша команда с помощью этих изменений стремилась сделать настоящее Руководство более удобным для чтения и использования. Рассмотрение agile в Руководстве выходит за рамки отрасли разработки компьютерного программного обеспечения, поскольку они нашли применение в средах, не связанных с разработкой ПО. Agile в разной мере применяется в промышленном производстве, образовании, здравоохранении и других отраслях, поэтому их применение вне области разработки программного обеспечения вошло в содержание настоящего Практического руководства.

ОБУЧЕНИЕ НА ОСНОВЕ AGILE

Сфера образования является основным и продуктивным подспорьем для расширения практики agile за пределы области разработки программного обеспечения. Преподаватели в школах и ВУЗах по всему миру стали применять agile с целью создания культуры, способствующей обучению. Методики agile позволяют сосредоточить усилия на приоритизации задач. Личное взаимодействие, осмысленное изучение, самоорганизующиеся команды и инкрементное и/или итеративное обучение, где используются возможности воображения, – это все принципы agile, которые могут изменить способ мышления в классе и продвинуть вперед цели образования (Briggs, 2014).*

*Briggs, Sara. «Обучение на основе agile. Что это такое и как оно может изменить образование?» *Opencolleges.edu.au* 22 февраля 2014 г., взято с вебсайта <http://www.opencolleges.edu.au/informed/features/agile-based-learning-what-is-it-and-how-can-it-change-education/>.

Так зачем же нужно «*Agile: практическое руководство*» и почему именно сейчас? Команды проектов используют методики и подходы agile в различных формах уже несколько десятилетий. В Agile-манифесте [1]1 были представлены фундаментальные ценности и принципы agile, когда эта методика уже набрала популярность (см. раздел 2.1). Сегодня руководители и команды проектов оказываются в среде, которая дезорганизуется под воздействием экспоненциальных технологических достижений и спроса со стороны заказчика на более срочную поставку ценности. Методики и подходы agile позволяют эффективно управлять прорывными технологиями. Кроме того, первый принцип agile ставит удовлетворение потребностей заказчика на первое место среди приоритетов и является ключевым при поставке продуктов и услуг, которыми заказчик полностью доволен (см. раздел 2.1). В условиях широкого распространения социальных сетей всегда доступны быстрые и прозрачные циклы обратной связи с заказчиками. Соответственно, чтобы сохранять конкурентоспособность и релевантность, организации больше не могут замыкаться внутри себя, а должны основное внимание сосредоточить вовне, на степени удовлетворенности заказчика качеством обслуживания.

ПРОРЫВНЫЕ ТЕХНОЛОГИИ

Прорывные технологии получают особое развитие в результате перехода к

облачным вычислениям. Компании по всему миру эффективно используют эту модель для быстрого и дешевого доступа к вычислительным ресурсам и получения выхода на традиционные рынки. Облачные вычисления требуют предварительной оплаты в меньшем размере и оплачиваются в течение определенного срока за обслуживание по подписке на условиях модели «оплата за время использования» или «оплата за объем использования». Обновленные приложения, инфраструктура и платформы выпускаются в облако в итеративном или инкрементном порядке, следуя нога в ногу с развитием технологий и постоянно меняющимся спросом клиентов.

Прорывные технологии быстро меняют условия игры за счет сокращения препятствий для входа в нее. Более зрелые организации все больше склонны к излишнему усложнению и потенциальному замедлению в инновациях и отстают в поставке новых решений своим заказчикам. Этим организациям приходится конкурировать с организациями меньшего размера и стартапами, которые в состоянии быстро создавать продукты, удовлетворяющие потребности заказчиков. Скорость изменений будет и дальше заставлять большие организации принимать на вооружение образ мышления agile, чтобы оставаться конкурентоспособными и удерживать долю рынка, которой они владеют.

«Agile: практическое руководство» сфокусировано на управлении проектами и рассматривает выбор жизненного цикла проекта, внедрение agile и организационные соображения для проектов agile. Управление организационными изменениями (ОСИ) является совершенно необходимым для реализации или трансформации практик, но поскольку ОСИ – это отдельная дисциплина, она не входит в предмет рассмотрения настоящего Практического руководства. Тем, кому нужны наставления по ОСИ, рекомендуем ознакомиться с документом «Управление изменениями в организациях – практическое руководство» (*Managing Change in Organizations – A Practice Guide*) [2]. Дополнительные вопросы, которые как входят, так и не входят в предмет рассмотрения настоящего Практического руководства, приведены в таблице 1–1.

Таблица 1–1. Вопросы, входящие и не входящие в предмет рассмотрения

Входящие в предмет рассмотрения вопросы	Не входящие в предмет рассмотрения вопросы
Осуществление подходов agile на уровне проекта или команды	Внедрение agile в масштабах всей организации или создание использующих agile программ
Описание наиболее распространенных подходов agile в виде перечисленных отраслевых обзоров	Описание нишевых подходов, специфичных для компании методов или методов неполных жизненных циклов
Требующие рассмотрения факторы, подходящие для выбора и/или практики agile	Рекомендация или одобрение конкретного подхода/практики
Картирование agile в процессах и областях знаний, описанных в <i>Руководстве РМВОК®</i>	Изменение или дополнение процессов и областей знаний, описанных в <i>Руководстве РМВОК®</i>
Обсуждение вопросов использования подходов agile вне области разработки ПО	Устранение влияния отрасли разработки ПО на подходы agile. (Обращаем внимание, что разработка ПО включена в предмет рассмотрения настоящего руководства несмотря на то, что использование agile получает распространение во многих других отраслях)
Указания, методы и подходы для рассмотрения в процессе внедрения agile в проекты или организации	Предписывающие пошаговые инструкции о порядке внедрения agile в проекты или организации
Определения общепринятых терминов	Новые термины и/или определения

Настоящее Руководство предназначено для команд проектов, которые оказываются в сложной ситуации между предиктивным подходом и подходом agile и пытаются решать вопросы с учетом необходимости быстрой инновации и сложности проекта, а также

стремятся к совершенствованию команды. Настоящее Руководство содержит полезные указания для успешного осуществления проектов, которые обеспечивают бизнес-ценность для удовлетворения ожиданий и потребностей заказчика.

Настоящее Руководство состоит из следующих частей:

Раздел 2. Введение в agile. Этот раздел включает в себя описание образа мышления Agile-манифеста, его ценностей и принципов. В нем также идет речь о концепциях, поддающихся определению и характеризующихся высокой неопределенностью работ, а также о соотношении подходов бережливого производства, методов «канбан» и agile.

Раздел 3. Выбор жизненного цикла. В этом разделе дано описание различных жизненных циклов, которые описаны в настоящем Руководстве. Он также содержит описание фильтров приемлемости, указания по адаптации и распространенные сочетания различных подходов.

Раздел 4. Реализация Agile. Создание среды agile. В этом разделе речь идет о важнейших факторах, которые необходимо учитывать при создании среды agile, к примеру, таких как обслуживающее лидерство и состав команды.

Раздел 5. Реализация Agile. Поставка в среде agile. В этом разделе представлена информация о том, как организовать команду, и об общих приемах, которые команда может использовать для поставки ценности на регулярной основе. Здесь приводятся примеры эмпирических измерений, используемых в работе команд и в отчетности о ходе работ.

Раздел 6. Организационные соображения для гибкости проекта. В этом разделе исследуются организационные факторы, которые оказывают влияние на применение подходов agile, например, культура, готовность, бизнес-практики и роль ОУП.

Раздел 7. Призыв к действию. «Призыв к действию» предлагает направлять предложения и замечания для непрерывного совершенствования настоящего Практического руководства. Дополнения, приложения, ссылки, список использованной литературы и глоссарий содержат дополнительную полезную информацию и определения.

◆ **Дополнения.** Содержат обязательную информацию, которая является слишком объемной для ее включения в основной текст Руководства.

◆ **Приложения.** Содержат необязательную информацию, которая дополняет основное содержание Руководства.

◆ **Ссылки.** Показывают, где можно найти стандарты и другие публикации, цитаты из которых приводятся в Руководстве.

◆ **Список использованной литературы.** Содержит дополнительные публикации к каждому разделу, в которых можно найти подробную информацию по темам, освещаемым в Руководстве.

◆ **Глоссарий.** Содержит перечень терминов, которые используются в Руководстве, и их определения.

2. Введение в agile

2.1 Поддающиеся определению работы в сравнении с работами с высокой неопределенностью

Характер работ по проекту меняется в диапазоне от поддающихся определению работ до работ с высокой неопределенностью. Для проектов с поддающимися определению работами характерны четкие процедуры, которые на практике доказали свою успешность при осуществлении аналогичных проектов в прошлом. Производство автомобиля, электроприбора или строительство дома после завершения этапа проектирования являются примерами поддающихся определению работ. Связанные с такими работами область производства и процессы, как правило, совершенно ясны, и неопределенность исполнения и риска обычно низкая.

Новый проект, разрешение проблем и работы, которые ранее никогда не производились, требуют предварительного исследования. В совместной работе и разрешении проблем для выработки решения требуется участие экспертов по предметным областям. В качестве людей, которым приходится иметь дело с работами с высокой неопределенностью, можно назвать, например, инженеров программных систем, разработчиков, врачей, учителей, юристов и многих технических специалистов, которые решают возникающие проблемы. Поскольку многие поддающиеся определению работы автоматизированы, командам проектов поручают больше работ по проектам, связанным с высокой степенью неопределенности, где требуется использовать методы, описанные в настоящем Руководстве. Проекты с высокой неопределенностью характеризуются высокими темпами изменений, сложностью и уровнем риска. В случае применения традиционных предиктивных подходов, которые предназначены для предварительного определения практически всех требований и осуществления управления изменениями с помощью процесса запросов на изменения, указанные особенности могут привести к возникновению проблем. Вместо этого были созданы подходы Agile для выяснения реализуемости требований в рамках коротких циклов и быстрой адаптации по результатам оценок и обратной связи.

2.2 Agile-манифест и образ мышления agile

Ведущие эксперты отрасли разработки ПО оформили движение agile еще в 2001 г., когда был опубликован Agile-манифест разработки программного обеспечения (Manifesto for Agile Software Development) (см. рис. 2–1).

Мы постоянно открываем для себя более совершенные методы разработки программного обеспечения, занимаясь непосредственно разработкой и помогая в этом другим. Благодаря проделанной работе мы смогли осознать, что:

Люди и взаимодействие важнее процессов и инструментов

Работающий продукт важнее исчерпывающей документации

Сотрудничество с заказчиком важнее согласования условий договора

Готовность к изменениям важнее следования первоначальному плану

Таким образом, не отрицая важности того, что справа, мы все же больше ценим то, что слева.

© 2001, авторы Agile-манифеста

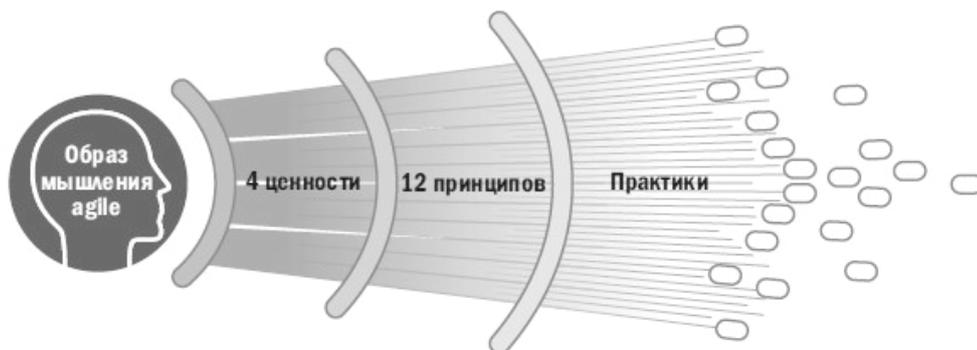
Рис. 2–1. Четыре ценности Agile-манифеста

Двенадцать уточняющих принципов, которые вытекают из этих ценностей, представлены на рис. 2–2.

1. Наивысшим приоритетом является удовлетворение потребностей заказчика за счет ранней и непрерывной поставки ценного программного обеспечения.
2. Изменение требований приветствуется, даже на поздних стадиях разработки. Процессы agile используют изменения в целях создания конкурентных преимуществ для заказчика.
3. Работающий программный продукт следует выпускать как можно чаще, с периодичностью от пары недель до пары месяцев, отдавая предпочтение более коротким срокам.
4. На протяжении всего проекта разработчики и представители бизнеса должны ежедневно работать вместе.
5. Над проектом должны работать мотивированные люди. Чтобы работа была сделана, создайте условия, обеспечьте поддержку и полностью доверьтесь им.
6. Личное общение является наиболее эффективным и результативным способом обмена информацией как с самой командой, так и внутри команды.
7. Работающий продукт — основной показатель хода исполнения.
8. Процессы agile способствуют устойчивой разработке. Спонсоры, разработчики и пользователи должны иметь возможность поддерживать постоянный ритм неопределенно долго.
9. Постоянное внимание к техническому совершенству и качеству проектирования повышает гибкость.
10. Простота — искусство максимизации объема работы, которую не нужно выполнять — крайне необходима.
11. Самые лучшие архитектуры, требования и проектные решения рождаются у самоорганизующихся команд.
12. Команда должна систематически анализировать возможные способы улучшения результативности и соответственно корректировать стиль своей работы.

Рис. 2–2. Двенадцать принципов, вытекающих из Agile-манифеста

Хотя эти принципы впервые возникли в отрасли разработки ПО, за прошедшее время они нашли применение во многих других отраслях. Это сочетание образа мышления, ценностей и принципов и составляет подход agile. Различные варианты подходов agile, которые применяются в настоящее время, имеют общие корни с образом мышления, ценностями и принципами agile. Эта взаимосвязь показана на рис. 2–3.



Подход agile — это образ мышления, определяемый ценностями, направляемый принципами и реализуемый в нескольких различных практиках. Agile-практик выбирает практики, исходя из своих потребностей.

Рис. 2–3. Взаимосвязь между ценностями, принципами и общепринятыми практиками Agile-манифеста

Как следует из рис. 2–3, созданная на основе идей Ахмеда Сидки (Ahmed Sidky) модель формулирует agile как образ мышления, определяемый на основе ценностей Agile-манифеста, основанный на принципах Agile-манифеста и получивший применение в разнообразных практиках. Стоит заметить, что, хотя понятие agile стало входить в общее употребление после публикации Agile-манифеста, те подходы и методы, которыми пользуются команды проектов сегодня, применялись за много лет, а в некоторых случаях – даже десятилетий до его публикации.

Подходы agile и методы agile являются обобщающими понятиями, которые описывают различные фреймворки и методы. На рис. 2–4 понятие agile показано в контексте и наглядно представлено как общий термин, относящийся к любому типу подхода, метода, фреймворка, методике или практики, в которых применяются ценности и принципы Agile-манифеста. На рис. 2–4 agile и метод «канбан» также представлены как подсистемы, формирующие суть бережливого подхода. Это объясняется тем, что они являются получившими собственные названия примерами бережливого мышления, где применяются общие концепции бережливого подхода, такие как: «основное внимание на ценности», «партии малого размера» и «устранение потерь».

Является ли agile подходом, методом, практикой, методикой или фреймворком? В зависимости от конкретной ситуации может использоваться любое из этих понятий. В настоящем Руководстве используется понятие «подход», за исключением случаев, когда использование другого понятия очевидно является более правильным.

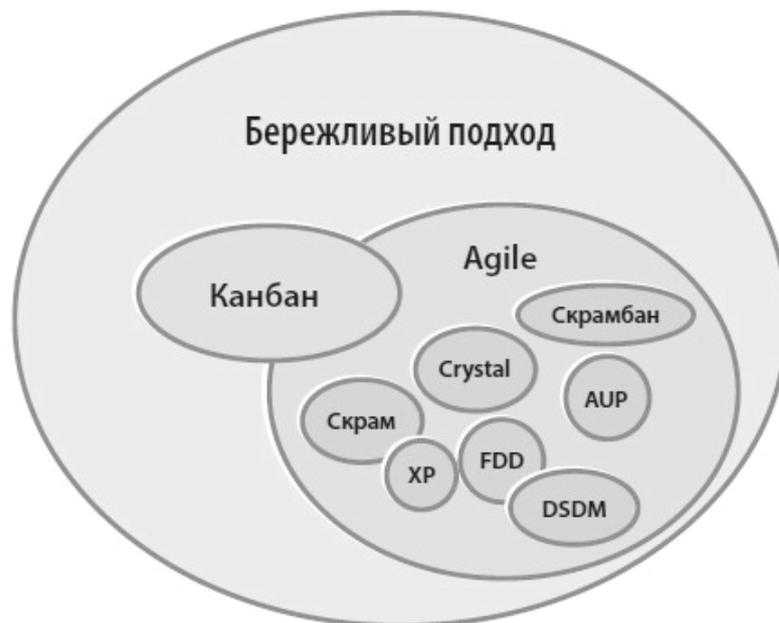


Рис. 2–4. Agile – объединяющее понятие для многих подходов

В общем, существует две стратегии применения ценностей и принципов agile. Первая состоит в принятии формального подхода agile, который разработан специально и практически подтвержден для достижения желаемых результатов. Затем необходимо уделить время, чтобы изучить и понять подходы agile, прежде чем изменять или адаптировать их. Преждевременная или необдуманная адаптация может свести к минимуму результаты применения этого подхода и, соответственно, ограничить выгоды от них. (См. в приложении X2 о соображениях по адаптации).

Вторая стратегия состоит во внесении изменений в практики проекта таким образом, чтобы это соответствовало контексту проекта, в целях обеспечения прогресса по главной ценности или принципу. Следует использовать временные рамки для создания свойств или использовать специальные методы для их итеративного уточнения. Следует рассмотреть возможность разделения одного большого проекта на несколько релизов, если это поможет в осуществлении проекта в конкретном контексте. Необходимо внести изменения, которые помогут в успешной реализации проекта; при этом не требуется, чтобы эти изменения были частью формальных практик организации. Конечная цель состоит не в том, чтобы быть agile, только лишь ради этого, а в том, чтобы обеспечить непрерывную поставку ценности заказчикам и добиться лучших конечных результатов для бизнеса.

2.3 Бережливый подход и метод «канбан»

Взаимосвязь между бережливым мышлением, подходом agile и методом «канбан» можно представить себе, например, если рассматривать agile и метод «канбан» как производные бережливого мышления. Другими словами, бережливое мышление включает более широкий набор характеристик, имеющий общие свойства с подходом agile и методом «канбан». Эти общие свойства весьма схожи и сосредоточены на поставке ценности, уважении к людям, сокращении потерь, обеспечении прозрачности, адаптации к изменениям и непрерывном совершенствовании. В некоторых случаях команда проекта может счесть полезным объединить вместе различные методы: все то, что может принести пользу организации или команде, следует сделать, независимо от его природы. Цель состоит в том, чтобы получить наилучший конечный результат, независимо от применяемого подхода. Метод «канбан» разработан на основе оригинальной системы бережливого производства и используется в частности в сфере умственного труда. Он появился в середине 2000-х гг. в качестве альтернативы методам agile, которые в то время преобладали. Метод «канбан» является менее директивным в сравнении с некоторыми подходами agile, и менее «дезорганизирующим», поскольку является оригинальным подходом, основанным на принципе «начинай прямо там, где находишься». Командам проектов сравнительно просто начать применять метод «канбан» и затем перейти к более прогрессивным подходам agile, если они сочтут это необходимым или целесообразным. Более подробно метод «канбан» описан в приложении А3, содержащем обзор фреймворков подхода agile и бережливого подхода.

ПРАКТИЧЕСКИЙ ПРИМЕР

Ведутся сейчас и, вероятно, еще долго будут идти споры о методе «канбан» и о том, куда его отнести – к бережливому подходу или движению agile. Он был задуман в рамках и в связи с бережливым производством, но широко применяется в средах agile.

2.4 Неопределенность, риск и выбор жизненного цикла

Некоторые проекты отличаются значительной неопределенностью при определении требований проекта и того, как исполнить эти требования, используя имеющиеся знания и технологии. Эти неопределенности могут стать причиной увеличения темпов изменений и усложнения проекта. Эти свойства наглядно представлены на рис. 2–5. По мере усиления неопределенности проекта происходит также увеличение риска возникновения необходимости доработок и применения другого подхода. Для снижения уровня этих рисков команды выбирают жизненные циклы, которые позволяют им заниматься проектами с высокой степенью неопределенности путем исполнения небольших инкрементов работы.

При использовании небольших инкрементов команды получают возможность проверять результаты своей работы и вносить изменения в то, что предстоит сделать. Когда команды производят поставки небольшими инкрементами, улучшается их способность понимать подлинные требования заказчиков, и делать это в более короткие сроки и точнее, чем в случае работы по неизменным письменным спецификациям.

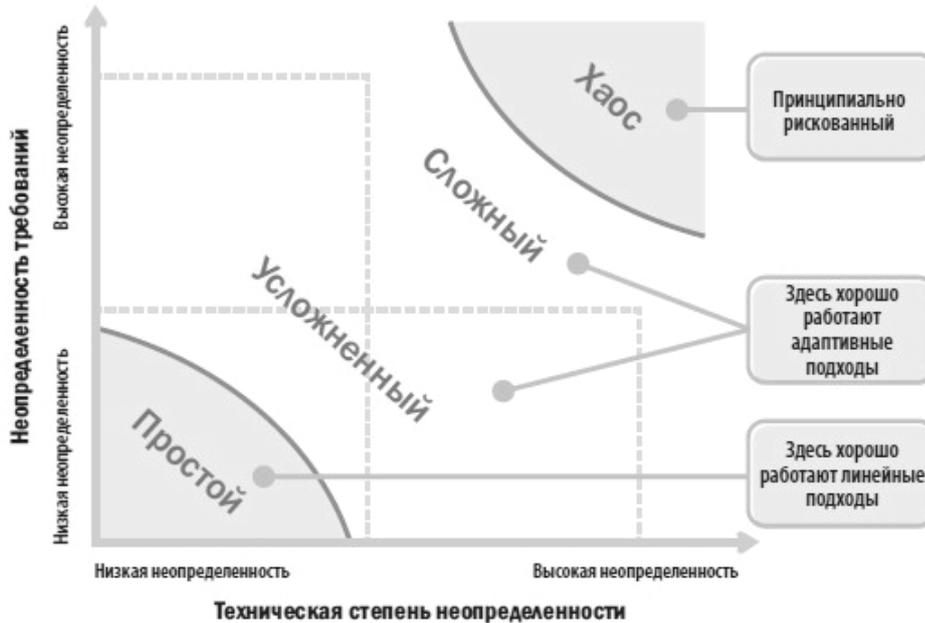


Рис. 2–5. модель неопределенности и сложности на основе модели сложности Стейси (Ralph D. Stacey)

Команда может без особых затруднений планировать проект и управлять им, имея четкие и стабильные требования, а также ясные и легко решаемые технические задачи. Однако, по мере нарастания неопределенности в проекте, вероятность необходимости внесения изменений, бесполезной работы и доработок также возрастает, что влечет убытки и потерю времени.

Некоторые команды используют развитые жизненные циклы проектов, где используются как итеративные, так и инкрементные подходы. Многие команды обнаружили, что когда они изучают требования итеративно и осуществляют поставки чаще и по частям (инкрементно), им становится легче адаптироваться к изменениям. Такие итеративные и инкрементные подходы позволяют сократить объемы потерь и доработок, поскольку команда получает обратную связь. В этих подходах используются:

- ◆ очень короткие циклы обратной связи,
- ◆ частая адаптация процесса,
- ◆ пересмотр приоритетов,
- ◆ регулярное обновление планов,
- ◆ частые поставки.

ПОЛЕЗНЫЙ СОВЕТ

Что означают определения проектов «простой», «усложненный» и «сложный»? Возьмем большие проекты, например, проект строительства Большого бостонского тоннеля. На первый взгляд, этот проект выглядит довольно очевидным: просто переместить автомагистраль с эстакады под землю. Был заключен генеральный договор о требованиях (см. ось Y на рис. 2–5). Степень неопределенности в отношении порядка исполнения проекта была невысокой, пока не приступили к его осуществлению. И, как это часто бывает с большими проектами, на всем протяжении осуществления этого проекта

постоянно возникали неприятные сюрпризы.

Когда команда ведет работу над проектом, причем возможности поставки промежуточных результатов или создания прототипов практически отсутствуют, она, скорее всего, будет использовать для управления проектом предиктивный жизненный цикл. Команда может вносить изменения с учетом новых обстоятельств, но не сможет использовать подходы agile для управления итеративно возникающими новыми требованиями или осуществлять инкрементные поставки с целью обратной связи.

Проект строительства Большого тоннеля, безусловно, не был простым. Однако при осуществлении многих проектов, начало которых лежит в нижней левой части Модели сложности Стейси, реальная возможность перейти к другим моделям практически отсутствует. Необходимо оценить проект с точки зрения как требований, так и средств поставки, чтобы определить наилучший подход при выборе жизненного цикла для данного проекта.

Эти итеративные, инкрементные agile-подходы хорошо работают в проектах, которые связаны с использованием новых или инновационных инструментов, методов, материалов или областей применения. (См. раздел 3, посвященный вопросам выбора жизненного цикла). Они также хорошо работают в проектах, которые:

- ◆ требуют проведения НИОКР,
- ◆ имеют высокие темпы изменений,
- ◆ имеют неясные или неизвестные требования, неопределенность или риск,
- ◆ имеют конечную цель, которую сложно описать.

Создав небольшой инкремент и проведя затем его испытания и исследование, команда может изучить неопределенность с незначительными затратами и в короткий срок, снизить уровень риска и добиться поставки максимальной бизнес-ценности. Центральными вопросами в отношении неопределенности могут быть: пригодность продукта и требования (создается ли именно тот продукт, который нужен?); техническая возможность и исполнение (возможно ли создать этот продукт таким образом?) или процесс и кадры (эффективный ли это способ работы команды?). Все три указанные характеристики (спецификация продукта, производственные возможности и пригодность процесса) обычно включают элементы высокой неопределенности.

Однако применение итеративных и инкрементных подходов имеет известные ограничения. В случаях, когда высока степень неопределенности, как технической, так и связанной с требованиями (верхняя правая сторона на рис. 2–5), проект выходит за пределы «сложного» и становится «хаотическим». Чтобы иметь уверенность в возможности осуществления проекта, необходимо установить одну из переменных неопределенности.

3. Выбор жизненного цикла

Проекты могут иметь много форм, и существуют разнообразные способы их осуществления. Командам проектов нужно знать характеристики и имеющиеся варианты, из которых можно выбрать тот подход, который с наибольшей вероятностью обеспечит успех в данной ситуации.

В настоящем Руководстве речь идет о четырех типах жизненных циклов, которые можно определить следующим образом:

- ◆ **Предиктивный жизненный цикл.** Относительно традиционный подход, который предусматривает осуществление основной части планирования до начала работы по проекту с его последующим исполнением за одинарный проход и последовательным процессом.
- ◆ **Итеративный жизненный цикл.** Подход, позволяющий использовать обратную связь с целью доработки и уточнения незавершенной работы.

- ◆ **Инкрементный жизненный цикл.** Подход, дающий конечные поставляемые результаты, которые заказчик может немедленно использовать.
- ◆ **Жизненный цикл agile.** Подходы, которые являются итеративными и, в то же время, инкрементными и предназначены для уточнения элементов работы и частой поставки.

ЧТО МОЖНО НАЗВАТЬ ПОДХОДАМИ, НЕ ЯВЛЯЮЩИМИСЯ AGILE?

Единого универсального термина, который описывает не являющиеся agile подходы, не существует. Изначально в Практическом руководстве, чтобы подчеркнуть акцент на предварительное планирование с последующим исполнением, использовалось понятие подход на основе плана. Некоторые специалисты для описания этого жизненного цикла предпочитают использовать понятия *водопад* или *последовательный*. В конце концов мы остановили выбор на понятии предиктивный, поскольку оно используется в *Руководстве к своду знаний по управлению проектом (Руководство PMBOK®)* [3] и в *Дополнении к Руководству PMBOK® по разработке программного обеспечения – пятое издание* [4].

В практике многих организаций нет таких крайних позиций, и они просто занимают какое-то среднее положение. Это естественно, но нам все же необходимо определить, как называть крайние позиции этого диапазона понятий. Если подход agile находится на одном конце диапазона, то предиктивный подход – на другом.

3.1 Характеристики жизненных циклов проектов

В таблице 3–1 обобщены характеристики четырех категорий жизненных циклов, о которых идет речь в настоящем Руководстве.

Таблица 3–1. Характеристики четырех категорий жизненных циклов

Характеристики				
Подход	Требования	Операции	Поставка	Цель
Предиктивный	Фиксированные	Выполняются однократно за весь проект	Разовая поставка	Управление стоимостью
Итеративный	Динамичные	Повторяются до полного уточнения	Разовая поставка	Правильность решения
Инкрементный	Динамичные	Производятся однократно для каждого инкремента	Частые поставки частями меньшего размера	Скорость
Agile	Динамичные	Повторяются до полного уточнения	Частые поставки небольшими частями	Ценность для заказчика за счет частых поставок и обратной связи

Важно отметить, что все проекты имеют эти характеристики – ни в одном проекте нельзя полностью избавиться от соображений о требованиях, поставке, изменениях и целях. Какой жизненный цикл лучше всего подходит для использования в данном проекте, определяют внутренне присущие ему характеристики.

Еще один путь к пониманию различий жизненных циклов проектов состоит в использовании континуума в диапазоне от предиктивных циклов на одном конце до циклов agile на другом, где те или иные итеративные или инкрементные циклы занимают среднее положение между ними.

На рис. X3-1 в приложении X3 к *Руководству PMBOK® – Шестое издание* данный континуум представлен линией. Данное представление подчеркивает смещение характеристик из одного конца в другой. Еще одним способом визуализации данного континуума является двумерный квадрат, который показан на рис. 3–1.



Рис. 3–1. Континуум жизненных циклов проектов

Ни один жизненный цикл не может идеально подходить для всех проектов. Напротив, каждому проекту соответствует определенная точка континуума, которая обеспечивает оптимальный баланс характеристик для его контекста. А именно:

- ◆ **Предиктивные жизненные циклы.** Используются преимущества того, что известно и прошло проверку практикой. Такое снижение уровня неопределенности и сложности позволяет командам разбить работу по следующим в определенном порядке прогнозируемым группам работ.
- ◆ **Итеративные жизненные циклы.** Позволяют использовать обратную связь в отношении частично завершенной или незавершенной работы с целью ее доработки и уточнения.
- ◆ **Инкрементные жизненные циклы.** Дают конечные поставляемые результаты, которые заказчик может немедленно использовать.
- ◆ **Жизненные циклы agile.** Используют преимущества как итеративных, так и инкрементных характеристик. При использовании командой подходов agile продукт производится итерациями в виде создания готовых поставляемых результатов. Команда получает обратную связь уже на раннем этапе и обеспечивает заказчику наглядность, уверенность и контроль в отношении продукта. Поскольку команда может выпустить продукт раньше, проект может обеспечить окупаемость инвестиций в более короткие сроки, так как команда в первую очередь производит имеющую наибольшую ценность работу.

ПЛАНИРОВАНИЕ ОСУЩЕСТВЛЯЕТСЯ ПРИ ЛЮБЫХ ОБСТОЯТЕЛЬСТВАХ

Когда речь идет о жизненных циклах, следует всегда помнить, что в каждом из них присутствует элемент планирования. Разница между жизненными циклами состоит не в том, осуществляется ли планирование в принципе, а в том, в каком объеме и на каком этапе проекта это происходит.

На предиктивном конце континуума работы производятся по плану. Планирование в максимально возможном объеме производится заблаговременно. Требования выясняются настолько подробно, насколько это возможно. Команда

оценивает, когда она будет в состоянии поставить каждый из поставляемых результатов и выполняет закупочную деятельность в полном объеме.

В случае с применением итеративных подходов планирование выпуска прототипов и проведения испытаний также производится, однако полученные результаты предназначены для уточнения первоначальных планов. Анализ незавершенной работы на ранних этапах помогает получить информацию для использования при производстве последующих работ по проекту.

С другой стороны, в инкрементных инициативах планируется последовательная поставка промежуточных результатов проекта в целом. Команды заранее могут планировать несколько последовательных поставок или только по одной поставке за один раз. Такие поставки обеспечивают информацию для использования при производстве последующих работ по проекту.

Планирование в проектах agile также осуществляется. Основным отличием является то, что команда планирует и пересматривает планы по мере поступления новой информации, получаемой по результатам периодических поставок. Любой проект требует планирования, независимо от типа его жизненного цикла.

3.1.1 Характеристики предиктивных жизненных циклов

Предиктивные жизненные циклы предполагают использование преимуществ от высокой определенности твердо установленных требований, стабильного состава команды и низкого уровня риска. Как следствие, операции проекта во многих случаях исполняются в определенной последовательности, как показано на рис. 3–2.

Чтобы иметь возможность использовать этот подход, команде необходимо иметь подробные планы, которые определяют, какие производятся поставки и как. Такие проекты завершаются успешно, когда для других потенциальных изменений (например, изменений в требованиях; члены команды проекта вносят изменения в поставки) установлены ограничения. В предиктивном проекте руководитель стремится свести объем изменений к минимуму.

Когда команда определяет подробные требования и разрабатывает планы в самом начале проекта, ограничения можно сформулировать. В последующем команда может использовать эти ограничения для управления рисками и затратами. В процессе постепенной реализации подробного плана команда осуществляет мониторинг и контроль изменений, которые могут оказать влияние на содержание, расписание или бюджет проекта.

В предиктивном проекте поставка бизнес-ценности производится лишь в конце проекта из-за сосредоточенности на эффективности подразделений и на установленной последовательности производства работы. Если в ходе предиктивного проекта возникают изменения или расхождения с требованиями, или техническое решение перестает быть очевидным, осуществление проекта будет связано с непредвиденными затратами.

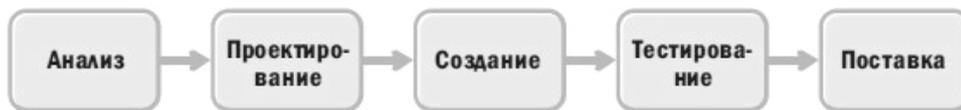


Рис. 3–2. Предиктивный жизненный цикл

3.1.2 Характеристики итеративных жизненных циклов

Итеративные жизненные циклы позволяют улучшать продукт или результат посредством последовательного создания прототипов или подтверждения концепции. Результатом каждого нового прототипа становятся новая информация от обратной связи с заинтересованными сторонами и углубление понимания командой предметной области.

Затем команда внедряет полученные новые знания, повторяя одну или несколько операций в проекте в ходе следующего цикла. Команды могут ограничить временные рамки для данной итерации несколькими неделями, собрать важную информацию и после этого произвести доработку операции на основе этих новых знаний. Таким образом, итерации помогают выявить и снизить уровень неопределенности в проекте.

Преимущества от использования итеративного жизненного цикла проект получает в тех случаях, когда он имеет высокую сложность, в нем часто происходят изменения или его содержание зависит от различий в точках зрения заинтересованных сторон на желаемый конечный продукт. Итеративные жизненные циклы могут занимать больше времени, поскольку их назначение состоит в получении знаний, а не в сокращении сроков поставки. На рис. 3–3 показаны некоторые элементы итеративного жизненного цикла проекта для единичной поставки продукта.



Рис. 3–3. Итеративный жизненный цикл

Приходилось ли вам участвовать когда-либо в проекте, когда, по вашим ощущениям, требования изменялись ежедневно, и вам казалось: «Мы узнаем требования, когда поставим прототип, который получит одобрение предприятия»? Если «да», то это был проект, в котором могло бы помочь использование подходов agile. Прототип служит стимулом обратной связи и помогает лучше понять требования, которые могут быть исполнены в каждом поставляемом результате.

3.1.3 Характеристики инкрементных жизненных циклов

Оптимизация некоторых проектов осуществляется с целью сокращения сроков поставки. Многие предприятия и инициативы не имеют возможности дожидаться, когда работы будут завершены полностью, и в таких случаях заказчики желают предварительно получить составную часть общего решения. Частую поставку поставляемых результатов меньшего объема называют «инкрементный жизненный цикл» (см. рис. 3–4).



Рис. 3–4. Жизненный цикл с поставкой инкрементов разного объема

ПОЛЕЗНЫЙ СОВЕТ

У вас нет уверенности в том, как новая бизнес-услуга может работать на практике? Разработайте обоснование концепции, предусматривающее критерии оценки, для анализа желаемых конечных результатов. Используйте итеративные

подходы, когда есть основания ожидать, что требования будут изменяться по результатам обратной связи с заказчиком.

Инкрементные жизненные циклы оптимизируют работу для поставки ценности спонсору или заказчику часто, а не однократно, в виде конечного продукта. Команды планируют поставляемые результаты до начала работы и приступают к работе по созданию первого поставляемого результата как можно раньше. Поставка ценности в некоторых проектах agile происходит в течение нескольких дней с момента его инициации. В других проектах для этого требуется больше времени – от 1 до нескольких недель.

По ходу исполнения проекта команда может отклоняться от изначальных представлений. Команда может управлять отклонениями, так как она осуществляет поставку в более короткие сроки. Не так важна степень изменений и вариаций, как поставка клиентам ценности до окончания проекта.

Поставка заказчику отдельного свойства или законченной части работ является примером инкрементного подхода.

Например, у строителей может возникнуть потребность показать заказчику завершенное помещение или этаж в здании до продолжения работ в остальной его части. В этом случае они могут полностью выполнить работы на этаже с установкой недвижимого оборудования, завершением отделочных и прочих работ, которые должны быть произведены на данном этаже для приемки, прежде чем переходить к работам на следующем этаже. Заказчик может осмотреть и принять стиль, цвет и другие элементы, что дает возможность внести какие-то изменения до того, как будут произведены дальнейшие затраты рабочего времени и денежных средств. Это сокращает объемы возможной доработки и/или степень неудовлетворенности заказчика.

Завершенность и поставка являются субъективными понятиями. Команде может потребоваться обратная связь в отношении прототипа, после чего она может принять решение о поставке минимально приемлемого продукта (minimum viable product, MVP) части заказчиков. Обратная связь с заказчиками помогает команде узнать, что из свойств конечного готового продукта ей необходимо обеспечить в последующих поставках.

Ключевое отличие agile-команды заключается в том, что она поставляет бизнес-ценность часто. При условии постепенного расширения набора свойств продукта и круга его потребителей, можно сказать, что его поставка осуществляется инкрементно.

3.1.4 Характеристики жизненных циклов agile

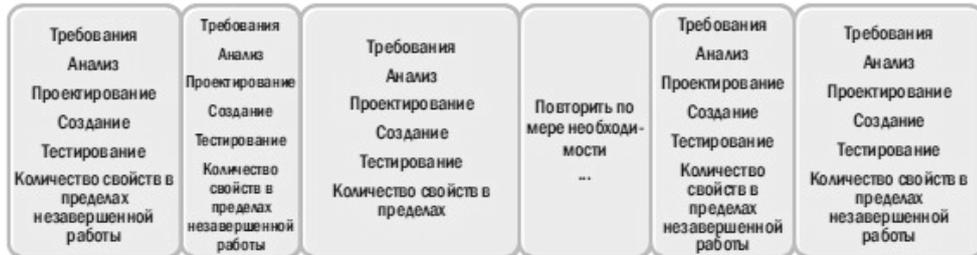
В среде agile команда ожидает, что в требованиях будут происходить изменения. Итеративные и инкрементные подходы обеспечивают обратную связь, которая позволяет лучше планировать следующую часть проекта. Однако в проектах agile инкрементная поставка выявляет скрытые или неправильно понятые требования. На рис. 3–5 наглядно представлены два возможных способа осуществления инкрементной поставки так, чтобы проект был согласован с потребностями заказчика и при необходимости мог быть адаптирован.

Итерационный agile



ПРИМЕЧАНИЕ. Все временные рамки имеют одинаковый размер. Результатом каждой временной рамки является тестирование свойств.

Потоковый agile



ПРИМЕЧАНИЕ. В потоке для завершения каждого свойства требуется разное время.

Рис. 3–5. Итерационный и потоковый жизненные циклы agile

В случае итерационного жизненного цикла agile команда работает в рамках итераций (временные рамки имеют одинаковую длительность) с целью поставки завершенных свойств. Команда работает над наиболее важным свойством для его завершения силами всей команды. Затем команда приступает к работе над следующим по важности свойством и полностью завершает его. Команда может принять решение вести работу по нескольким свойствам сразу, но она не занимается всей работой для данной итерации одновременно (т. е. не занимается всеми требованиями по результатам всех анализов и т. д.).

В потоковом agile команда берет для работы свойства из бэклога в зависимости от своей ресурсной возможности начать работу, а не по основанному на итерациях расписанию. Команда определяет для себя поток работ с помощью столбцов доски задач и управляет незавершенными работами в каждом столбце. Для завершения работы по каждому свойству может требоваться разное время. Команды сохраняют небольшой объем незавершенных работ, чтобы было легче определить проблемы на раннем этапе и сократить объем доработок при возникновении необходимости внести изменения. Поскольку для определения точек планирования и контроля уже не используются итерации, команда и заинтересованные стороны со стороны бизнеса определяют наиболее целесообразное расписание для планирования, анализа продукта и ретроспективного анализа.

К жизненным циклам agile относятся те, в которых соблюдаются принципы Agile-манифеста. В частности, досрочная и непрерывная поставка имеющих ценность продуктов обеспечивает рост удовлетворенности заказчика. Кроме того, поставляемый инкрементно результат, который является функциональным и обладает ценностью, является главным показателем прогресса. В жизненных циклах agile в целях адаптации к условиям высоких темпов изменений и более частой поставки ценности проекта сочетаются итеративные и инкрементные подходы.

3.1.5 Фильтры приемлемости agile

Имеются различные модели оценки для определения вероятной пригодности или пробелов в использовании подходов agile. Эти модели служат для оценки факторов проекта и организации, связанных с внедрением и пригодностью, и затем дают оценочные баллы, показывающие области согласованности или потенциальных рисков. В приложении X3

приводится сводная информация о получивших распространение моделях оценки, которые можно использовать в качестве фильтров приемлемости agile.

3.1.6 Характеристики гибридных жизненных циклов

Нет необходимости применять единый подход на протяжении всего проекта. В проектах для достижения конкретных целей часто сочетаются элементы разных жизненных циклов. Сочетание предиктивного, итеративного, инкрементного подходов и подхода agile является гибридным подходом.

На рис. 3–6 представлены базовые, чистые подходы к определенным типам проектов, которые в сочетании образуют гибридную модель. В ранних процессах используется жизненный цикл разработки agile, после которого следует предиктивная фаза развертывания. Этот подход может быть использован в условиях неопределенности, сложности и риска на этапе разработки проекта, который выигрывает от использования подхода agile; затем следует определенная, повторяемая фаза развертывания, условия которой предполагают применение предиктивного способа, возможно, другой командой. Примером такого подхода является разработка нового высокотехнологичного продукта, после которой следует этап его внедрения и обучения тысяч пользователей.

ПРИМЕР ПРОЕКТА С ГИБРИДНЫМ ЖИЗНЕННЫМ ЦИКЛОМ

Фармацевтической компании нужно было пройти длительную процедуру лицензирования в Управлении по санитарному надзору за качеством пищевых продуктов и медикаментов США (FDA), и она связала ее с окончанием процесса разработки, в результате чего полный жизненный цикл ее проекта выглядел, как показано на рис. 3–6. Хотя команды проекта проводили испытания препарата в режиме agile, им необходимо было представить его внешней группе для проведения процедуры лицензирования в FDA. Консультант помог включить этап процедуры лицензирования в FDA в процесс разработки agile с целью формирования более упорядоченного гибридного подхода.

В кратком изложении дело обстоит так: поскольку лицензирование в FDA требуется завершить в конце процесса разработки или повторять в случае любых изменений (причем даже самых незначительных), то эта процедура проводится только в самом конце в рамках отдельной фазы. Интеграция с использованием итеративного процесса оказалась безуспешной. Однако консультант создал несколько полезных кратких руководств относительно начала работы и протоколов испытаний, которые позволили сократить время процедуры лицензирования в FDA.



Рис. 3–6. Разработка agile с последующим предиктивным развертыванием

3.1.7 Комбинация подхода agile и предиктивного подхода

Еще одним вариантом может быть комбинирование подходов agile и предиктивных подходов на всем протяжении жизненного цикла.



Рис. 3–7. Комбинация подхода agile и предиктивного подхода с их одновременным использованием

На рис. 3–7 в одном и том же проекте используется комбинация предиктивного подхода и подхода agile. Допустим, команда осуществляет переход к agile инкрементным путем и использует некоторые подходы, например, короткие итерации, ежедневные летучки и ретроспективный анализ, однако в то же время другие аспекты проекта, такие как предварительная оценка, распределение работ и отслеживание хода работ осуществляются на основе предиктивных подходов.

Одновременное использование предиктивного подхода и подхода agile является распространенным сценарием. С точки зрения толкования называть такой подход «agile» было бы неверно, поскольку совершенно ясно, что он не включает в себя образ мышления, ценности и принципы agile в полном объеме. Однако было бы также неправильно называть его «предиктивным», так как это гибридный подход.

3.1.8 Преимущественно предиктивный подход с компонентами agile

На рис. 3–8 представлены небольшие элементы agile внутри преимущественно предиктивного проекта. В этом случае часть проекта, отличающаяся неопределенностью, сложностью или возможностью расползания содержания, исполняется на основе подхода agile, а управление его остальной частью – с использованием предиктивных подходов. В качестве примера такого подхода можно было бы привести проектную организацию, которая занимается строительством объекта с новым компонентом.



Рис. 3–8. Преимущественно предиктивный подход с компонентами agile

Хотя в основном проект может быть самым обычным и предсказуемым, подобно многим другим проектам по сооружению объектов, которые данная организация выполняла раньше, в данном проекте используется новый кровельный материал. Подрядчик может планировать для начала несколько пробных монтажных работ небольшого объема на земле с целью определить наилучший способ монтажа и заблаговременно выявить проблемы, пока есть время, чтобы найти их решение и усовершенствовать процессы инкрементно путем экспериментирования и адаптации.

3.1.9 Преимущественно подход agile с предиктивным компонентом

На рис. 3–9 представлен преимущественно подход agile с предиктивным компонентом. Этот подход может использоваться, когда определенный элемент не может быть изменен путем переговоров или исполнен с использованием подхода agile. В качестве примеров можно привести интеграцию внешнего компонента, разработанного другим производителем-поставщиком, который не может или не желает сотрудничать на основе совместной работы или инкрементным путем. После поставки компонента требуется разовая интеграция.

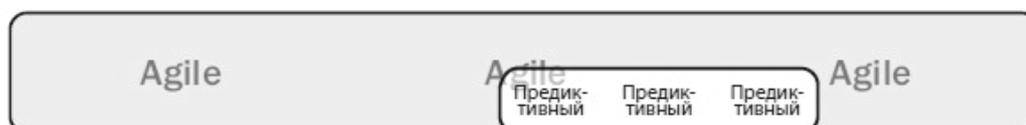


Рис. 3–9. Преимущественно подход agile с предиктивным компонентом

3.1.10 Гибридные жизненные циклы как способ обеспечения целевой пригодности

Команда проекта может создать гибридный жизненный цикл с учетом имеющихся в проекте рисков. Например, проект строительства кампуса может предусматривать реконструкцию и возведение нескольких зданий. При инкрементном подходе ресурсы были бы распределены так, чтобы завершение работ по некоторым зданиям произошло раньше, чем по другим, чтобы ускорить окупаемость инвестиций. Вероятно, то, что каждая отдельная поставка для данного конкретного здания выиграет в случае использования предиктивного жизненного цикла, достаточно хорошо известно.

Цель управления проектом состоит в создании бизнес-ценности наилучшим возможным способом при данных конкретных обстоятельствах. Не имеет значения, является ли этот способ agile или предиктивным. Вопрос, на который нужно ответить, состоит в следующем: «Как мы можем добиться наибольшего успеха?»

Требуется ли обратная связь в процессе создания командой ценности? Если «да», задачу поможет решить инкрементный подход. Есть ли необходимость в управлении рисками в ходе исследования идей? Если «да», задачу лучше решать с помощью итерационного подхода или подхода agile.

В случаях, когда организация не может поставить промежуточную ценность, она, вероятно, не сможет использовать подходы agile. И в этом нет ничего плохого: цель состоит не в том, чтобы использовать подходы agile сами по себе. Главное – выбрать жизненный цикл или комбинацию жизненных циклов, которые подходят для проекта, учитывают риски и культуру.

Суть подхода agile состоит в обеспечении частых поставок с учетом пожеланий заказчика. Такая поставка обеспечивает команде обратную связь. Команда использует обратную связь при разработке и пересмотре планов следующей части работы.

Государственное ведомство занималось проектом разработки приложения кредитного страхования. Задача этого многолетнего проекта состояла в замене устаревающей системы страхования новой, имеющей более эффективный пользовательский интерфейс и элементы системной интеграции. Основная часть проекта осуществлялась с использованием подхода agile на основе непрерывного поступления предложений и замечаний от бизнеса.

Расчеты ставок страховой премии были представлены Организацией экономического сотрудничества и развития (ОЭСР) в спецификации объемом 200 страниц. Было дано совершенно четкое разъяснение шагов, практически исключающее возможность ошибочного понимания (или подтверждения промежуточного результата бизнесом), программирование которых произвела отдельная команда, работавшая над шагами расчета самостоятельно. Две команды совместно работали над необходимыми для расчета входными переменными и тем, как потреблять и выводить на дисплей выходные значения, но в остальном занимавшаяся расчетами команда работала преимущественно в предиктивном порядке.

После завершения части работы, которую выполняла занимавшаяся расчетами команда, выходные данные расчета ставок премии были показаны на экранах и в отчетах. Затем бизнес-пользователи по каналам обратной связи сообщили свое мнение о внешнем виде и использовании этой информации. Обе команды работали параллельно, но у них практически не возникало необходимости во взаимодействии. То, что они физически находились рядом друг с другом, делало задачу отслеживания хода разработки проще, но, по большому счету, это были два отдельных подпроекта.

3.1.11 Гибридные жизненные циклы как переходная стратегия

Многие команды не в состоянии за один день переключиться на способы ведения работы на принципах agile. Людям, которые привыкли к предиктивной среде и успешно в ней работали раньше, методы agile кажутся чем-то совершенно иным. Чем больше организация и чем больше в ней подвижных частей, тем больше требуется времени для перехода. По этой причине имеет смысл планировать постепенный переход.

Постепенный переход связан с добавлением итеративных по характеру методов для улучшения обмена знаниями и согласованности между командами и заинтересованными сторонами. В дальнейшем можно подумать о включении инкрементных по характеру методов с целью ускорения поставки ценности и окупаемости инвестиций для спонсоров. Такое сочетание различных подходов считается гибридным подходом.

Испытайте эти новые методы на менее рискованном проекте с уровнем неопределенности от среднего до низкого. Потом, когда организация получит хороший результат при использовании гибридного подхода, попробуйте применить его в более сложных проектах, которые требуют включения большего числа этих методов. Это способ провести постепенный переход с использованием гибридной методики с учетом обстановки в организации и конкретных рисков, а также готовности команды принять и практически использовать эти изменения.

3.2 Сочетание подходов agile

Agile-команды редко ограничивают свою практику использованием только одного подхода agile. Контекст каждого проекта имеет свои особенности, например, различия в сочетании навыков и практического опыта членов команды, разный состав компонентов разрабатываемого продукта, а также ограничения, связанные с возрастом, масштабом, важностью, сложностью и нормативно-правовым регулированием в той среде, где ведется работа.

Фреймворки agile не адаптируются специально для данной команды. Команде может потребоваться адаптировать практики, чтобы обеспечить поставку ценности на регулярной основе. Команды часто применяют на практике собственную особую комбинацию методов agile, даже когда используют какой-то конкретный фреймворк в качестве отправной точки.

КОМБИНИРОВАНИЕ ПОДХОДОВ

В качестве примера адаптации фреймворков agile можно привести одну из наиболее распространенных комбинаций, которая включает согласованное использование скрам-фреймворка, метода «канбан» и элементов метода экстремального программирования (XP). Скрам дает представление об использовании бэклога продукта, владельце продукта, скрам-мастере и кроссфункциональной команде разработки, включая планирование спринта, ежедневный скрам, анализ спринта и ретроспективные сессии спринта. Доска «канбан» помогает команде еще больше повысить свою результативность благодаря визуализации потока работы, обеспечивая хорошую наглядность препятствий и позволяя управлять потоком с помощью регулирования работы в рамках процесса. Кроме того, практики проектирования на основе XP, такие как использование карточек историй (story cards), непрерывная интеграция, рефакторинг, автоматизированное тестирование и разработка на основе тестов еще больше повышают результативность работы agile-команды. Подводя итог, можно сказать, что комбинирование практик из этих разнообразных источников дает синергетический результат с более высокими показателями исполнения, чем у каждого компонента в отдельности.

3.3 Факторы проекта, влияющие на адаптацию

В некоторых случаях особые свойства проекта требуют адаптировать подход, чтобы добиться его лучшей пригодности. В таблице 3–2 представлены некоторые факторы проекта и варианты адаптации для рассмотрения.

Таблица 3–2. Варианты адаптации для улучшения пригодности

Фактор проекта	Варианты адаптации
Состояние спроса: устойчивый или эпизодический	Многие команды приходят к выводу, что использование каденции (в форме регулярных временных рамок) помогает производить демонстрации, ретроспективный анализ и браться за новую работу. Помимо этого, некоторым командам требуется больше гибкости при приеме дополнительных объемов работы. Команды могут использовать потоковый agile с использованием каденции, чтобы получить максимальную выгоду от обоих вариантов.
Темп совершенствования процесса, необходимый с учетом уровня опыта команды	Чаще проводите ретроспективный анализ и выбирайте улучшения.
Поток работы часто прерывается из-за различных задержек или препятствий	Попробуйте наглядно представлять работу с помощью досок «канбан» и экспериментирования с ограничениями для различных областей рабочего процесса с целью улучшения потока.
Низкое качество инкрементов продукта	Попробуйте использовать различные практики разработки на основе тестов. Эта предотвращающая ошибки дисциплина практически исключает возможность того, что дефекты останутся незамеченными.
Требуется больше одной команды для создания продукта	Чтобы расширить масштаб с одной до нескольких agile-команд с минимальными переоями в работе, необходимо сначала приобрести знания об управлении использующими agile программами или о формальных фреймворках для масштабирования. После этого разработайте подход, который будет соответствовать контексту проекта.
Члены команды проекта не имеют опыта в применении подходов agile	Попробуйте начать работу с обучения членов команды основам образа мышления и принципов agile. Если команда принимает решение использовать конкретный подход, например скрам или канбан, организуйте семинар по этому подходу так, чтобы члены команды могли освоить его использование.

Дополнительные указания о факторах, которые влияют на адаптацию, смотрите в приложении X2 о свойствах, оказывающих влияние на адаптацию.

4. Реализация agile. Создание среды agile

4.1 Начните с образа мышления agile

Для управления проектом с использованием подхода agile необходимо, чтобы команда проекта приняла образ мышления agile. Ответы на следующие вопросы помогут выработать стратегию реализации.

- ◆ Как команда проекта может действовать в режиме agile?
- ◆ Что команда может поставить в короткие сроки и получить по каналам обратной связи в интересах следующего цикла поставки?

- ◆ Как команда проекта может действовать с обеспечением прозрачности?
- ◆ Какую работу можно исключить, чтобы сосредоточить усилия на высокоприоритетных элементах?
- ◆ Какие преимущества дает подход обслуживающего лидерства для достижения целей команды?

4.2 Обслуживающее лидерство усиливает команду

Подходы agile выделяют обслуживающее лидерство как способ усиления команды. Обслуживающее лидерство – это практика, состоящая в служении команде путем фокусирования на понимании ее нужд и поиске средств для их удовлетворения, а также развития команды для достижения максимальной эффективности и результативности ее работы.

Роль обслуживающего лидера состоит в создании для команды благоприятных условий в целях освоения и понимания среды agile. Обслуживающие лидеры применяют на практике и распространяют вокруг среду agile. Обслуживающие лидеры подходят к работе по проекту следующим образом:

- ◆ **Цель.** Работа с командой для определения «почему» или «для чего» таким образом, чтобы члены команды могли сосредоточить усилия и объединиться вокруг цели проекта. Оптимизация команды в полном составе происходит на уровне проекта, а не на индивидуальном уровне.
- ◆ **Люди.** После того, как цель установлена, добивайтесь, чтобы команда создала среду, где каждый может достичь успеха. Попросите каждого члена команды вносить вклад в рамках всей работы по проекту.
- ◆ **Процесс.** Не стройте планов исполнения «совершенного» процесса agile, а стремитесь к достижению результатов. Когда кроссфункциональная команда часто поставяет готовую ценность и оценивает продукт и процесс, она является командой agile. Не имеет значения, как команда называет свой процесс.

Следующие характеристики обслуживающего лидерства позволяют руководителям проекта стать более agile и создать условия для успеха команды:

- ◆ развитие навыков самоанализа,
- ◆ умение слушать,
- ◆ обслуживание членов команды,
- ◆ содействие росту людей,
- ◆ коучинг, а не контроль,
- ◆ обеспечение безопасности труда, а также уважения и доверия,
- ◆ содействие развитию энергичности и умения думать о других.

Обслуживающее лидерство используется не только в agile. Но приобретя практические навыки его использования, обслуживающие лидеры, как правило, могут увидеть, насколько хорошо обслуживающее лидерство интегрируется в образ мышления и ценности agile.

Если руководитель выработал в себе навыки обслуживающего лидерства или содействия другим, вероятность стать agile у него будет выше. В результате обслуживающие лидеры могут помочь своим командам объединить усилия для поставки ценности в более короткие сроки.

Успешные agile-команды принимают направленный на рост образ мышления, когда люди верят, что они могут приобрести новые навыки. Когда команда и обслуживающие лидеры верят, что все они могут учиться, каждый из них становится более способным.

4.2.1 Обязанности обслуживающего лидера

Обслуживающие лидеры строят отношения с целью установления коммуникации и координации внутри команды и с организацией в целом. Эти отношения помогают лидерам управлять отношениями с организацией для поддержки команды. Поддержка такого рода помогает устранить препятствия и способствует оптимизации командой своих процессов. Поскольку обслуживающие лидеры понимают практику agile и применяют конкретный подход к agile, они могут содействовать в удовлетворении потребностей команды.

4.2.1.1 Обслуживающие лидеры выполняют роль фасилитатора

Когда руководитель проекта действует как обслуживающий лидер, главным становится не «управление координацией», а «способствование совместной работе». Фасилитатор способствует наилучшему способу мышления и работе. Фасилитатор поощряет участие команды в конечном результате ее работы, понимание этого результата и совместную ответственность за него. Фасилитатор помогает команде находить приемлемые решения.

Обслуживающий лидер способствует сотрудничеству и обмену мнениями внутри команды и между ее членами. Например, обслуживающий лидер помогает выявлять и сообщать об узких местах внутри команд и между ними. Затем команды находят пути устранения таких узких мест.

Кроме этого, фасилитатор способствует сотрудничеству путем проведения интерактивных совещаний, неформальных бесед и обмена знаниями. Обслуживающий лидер делает это, принимая на себя функции непредвзятого наведения мостов и коучинга, а не принятия решений, за исполнение которых должны отвечать другие.

4.2.1.2 Обслуживающие лидеры устраняют организационные препятствия

Первая определенная в Agile-манифесте ценность состоит в том, что люди и взаимодействие важнее процессов и инструментов. У обслуживающего лидера нет важнее обязанности, чем тщательный анализ процессов, которые затрудняют гибкость (agility) работы команды или организации и их оптимизацию. Например, если какое-то подразделение требует излишнюю документацию, то роль обслуживающего лидера может состоять в том, чтобы провести работу вместе с этим подразделением с целью анализа требуемой документации, оказать помощь в выработке общего понимания того, как поставляемые результаты agile удовлетворяют эти требования, и оценить объем требуемой документации с тем, чтобы команды тратили больше времени на поставку ценного продукта, а не на отбирающую силы подготовку излишней документации.

Обслуживающий лидер должен также анализировать другие процессы, которые занимают много времени, порождают узкие места и затрудняют гибкость (agility) команды или организации. Примеры процессов или подразделений, которые могут требовать внимания, включают финансы, советы по контролю изменений или аудиты. Обслуживающий лидер может объединять усилия и вести работу с другими людьми, чтобы заставить их заняться анализом своих процессов с целью оказать поддержку командам и лидерам, работающим в среде agile. Например, какой смысл команде поставлять рабочий продукт каждые 2 недели только для того, чтобы этот продукт попал в очередь, или исполнять процесс, релиз которого может занять 6 или более недель из-за длительности процесса релиза? Слишком много организаций имеют такие процессы с «узкими местами», которые не дают командам быстро поставлять имеющие ценность продукты или услуги. Обслуживающий лидер имеет возможности изменить или устранить такие организационные препятствия с целью поддержки своих команд.

4.2.1.3 Обслуживающие лидеры создают условия для вклада других людей в работу

В agile своим рабочим процессом и своим продуктом работы управляет команда. Такое самоуправление и самоорганизация распространяются на всех, кто обслуживает и поддерживает организацию и проект. Работа обслуживающих лидеров состоит в удовлетворении потребностей команды, проектов и организации. Обслуживающий лидер может вести работу с необходимыми обслуживающими средствами для обеспечения команды рабочим пространством, работу с руководством для предоставления команде возможности сосредоточиться только на одном проекте или работу с владельцем продукта для разработки историй вместе с командой. Некоторые обслуживающие лидеры ведут работу с аудиторами, чтобы уточнить процессы, требуемые в тех или иных средах нормативно-правового регулирования, а другие – с финансовым подразделением, чтобы обеспечить переход организации к инкрементному бюджетированию.

Обслуживающий лидер основное внимание уделяет созданию условий для команды, чтобы она могла выполнить свою работу наилучшим образом. Обслуживающий лидер оказывает влияние на проекты и побуждает организацию мыслить иначе.

НАВЫКИ МЕЖЛИЧНОСТНЫХ ОТНОШЕНИЙ В СРАВНЕНИИ С ТЕХНИЧЕСКИМИ НАВЫКАМИ

Помимо значения обслуживающего лидерства, члены команды подчеркивают большое значение навыков межличностных отношений и эмоционального интеллекта, а не только технических навыков. Каждый член команды делает все для демонстрации большей степени инициативности, добросовестности, эмоционального интеллекта, честности, готовности к сотрудничеству, скромности, а также коммуникации различными способами так, чтобы команда могла хорошо работать как единое целое.

Эти навыки необходимы команде, чтобы быть в состоянии надлежащим образом реагировать на изменения в управлении проектом и на технические изменения продукта. Если каждый член команды может адаптироваться к работе и другим членам команды, достижение успеха командой в целом становится более вероятным.

4.2.1.4 Обдумайте следующие обязанности обслуживающего лидера

Обслуживающий лидер может занимать самые разные должности, но главное – это то, что он делают. Приведем несколько примеров обязанностей, которые может выполнять обслуживающий лидер:

- ◆ разъясняет всем заинтересованным сторонам, зачем нужен agile и как стать agile; объясняет выгоды бизнес-ценности, основанной на приоритизации, более высокой ответственности и производительности от наделенных возможностями команд и повышении качества в результате часто проводимых обзоров и т. п.;

- ◆ поддерживает членов команды путем наставничества, поощрения и оказания помощи; выступает в поддержку обучения и карьерного продвижения членов команды. Парадоксальное высказывание «мы ведем команды вперед, пребывая в их тени» говорит о роли лидера в профессиональном развитии членов своей команды. Благодаря поддержке, поощрению и профессиональному развитию члены команды приобретают уверенность в своих силах, берут на себя дополнительные задачи и вносят вклад на более высоких уровнях в своих организациях. Главная роль обслуживающего лидера состоит в воспитании и развитии членов команды в ходе исполнения ими своих текущих ролей и в последующем, даже если это может привести к их уходу из команды;

- ◆ помогает команде в решении вопросов технического управления проектом, например, в проведении количественного анализа риска. В некоторых случаях члены команды могут не обладать знаниями или опытом для исполнения некоторых ролей или функций. Обслуживающие лидеры, которые могут иметь больше опыта практического применения или теоретических знаний тех или иных методов, могут оказать команде поддержку путем проведения обучения или взяв на себя такую деятельность;

♦ отмечает успех команды, а также оказывает поддержку и является связующим звеном в деятельности по развитию отношений с внешними группами. Добивается увеличения признания и доброй воли для расширенного сотрудничества.

4.2.2 Роль руководителя проекта в среде agile

Роль руководителя проекта в проекте agile в определенном смысле остается неизвестной, поскольку во многих фреймворках и подходах agile роль руководителя проекта не рассматривается. Некоторые agile-практики полагают, что роль руководителя проекта в них не требуется, так как самоорганизующиеся команды принимают на себя прежние обязанности руководителя проекта. Однако стоящие на прагматических позициях agile-практики и организации понимают, что руководители проекта во многих ситуациях могут дополнительно принести значительную ценность. Основным отличием является то, что их роли и обязанности выглядят несколько иначе.

ПОЛЕЗНЫЙ СОВЕТ

Ценность руководителя проекта определяется не занимаемой им должностью, а способностью сделать всех остальных лучше.

4.2.3 Применение руководителями проекта методов обслуживающего лидерства

Согласно определению в *Руководстве PMBOK®* – Шестое издание, руководитель проекта – это «лицо, назначенное исполняющей организацией руководить командой и отвечающее за достижение целей проекта».

Многие руководители проектов привыкли находиться в центре координации проекта, отслеживать и представлять всей остальной организации статус команды. Такой подход не вызывал возражений, пока проекты распались на разрозненные функции.

Однако проект с высоким уровнем изменений настолько сложен, что один человек управлять им не в состоянии. В таких случаях кроссфункциональные команды координируют свою работу и сотрудничают с представителем бизнеса (владельцем продукта) самостоятельно.

При работе в проекте agile руководители проектов переходят от роли центра к роли обслуживания команды и руководства. В среде agile руководители проектов становятся обслуживающими лидерами, делая основной упор в своей работе на коучинг для сотрудников, которым требуется помощь, а также на развитии взаимопомощи в команде и согласовании потребностей с заинтересованной стороной. В качестве обслуживающего лидера руководитель проекта поощряет распределение ответственности среди членов команды, передавая ее тем, кто обладает необходимыми знаниями для выполнения работы.

4.3 Состав команды

Главная установка как ценностей, так и принципов Agile-манифеста состоит в утверждении важности людей и взаимодействия. Подход agile оптимизирует поток ценности, обращая основное внимание на быструю поставку определенного свойства заказчику, а не на то, как «используются» люди.

ПОЛЕЗНЫЙ СОВЕТ

Над проектом должны работать мотивированные люди. Создайте для них рабочую среду, помогайте в удовлетворении их потребностей и доверяйте им в выполнении работы.

Когда команда думает, как оптимизировать поток ценности, очевидными становятся следующие преимущества:

- ♦ вероятность сотрудничества между людьми становится выше;
- ♦ команды быстрее завершают создающие ценность работы;
- ♦ уменьшается количество потерянного времени в командах, поскольку они не решают одновременно много задач и им не приходится воссоздавать контекст работы.

4.3.1 Agile-команды

Главной задачей agile-команды является быстрая разработка продукта для получения обратной связи. На практике численный состав наиболее эффективных команд варьируется в пределах от трех до девяти человек. Лучше всего, когда члены agile-команды располагаются все в одном месте. Все рабочее время членов команды полностью посвящено работе команды. Agile стимулирует формирование самоуправляющихся команд, когда члены команды сами решают, кто будет выполнять работу в пределах содержания, определенного на следующий период. Agile-команды преуспевают в условиях обслуживающего лидерства. Лидеры поддерживают подход команд к их работе.

Кроссфункциональные agile-команды часто производят функциональный продукт инкрементно. Это объясняется тем, что члены таких команд несут коллективную ответственность за работу и только все вместе обладают необходимыми навыками для поставки конечного результата.

Несмотря на особенности применяемого общего подхода agile, чем больше команда ограничивает объемы незавершенной работы, тем выше вероятность, что ее члены смогут сотрудничать для ускорения работы, определенной на доске задач. Члены успешных agile-команд стремятся к сотрудничеству разными способами (например, в парах, методами роения или моббинга), чтобы не попасть в ловушку «мини-водопадов», вместо совместной работы. «Мини-водопады» – это когда команда занимается всеми требованиями в данный период, затем пытается выполнить все проектирование и после этого переходит к созданию всего сразу. По этому сценарию в какой-то момент при создании или тестировании по завершении создания команда может прийти к пониманию того, что она использовала допущения, которые более недействительны. Это значит, что в этом случае команда потеряла время, когда занималась всеми требованиями сразу. Напротив, когда члены команды совместно производят небольшое количество свойств, определенных на доске задач, они приобретают знания по ходу работы и поставок небольших готовых свойств.

Структуры команд проекта, которые улучшают сотрудничество внутри команд и между ними, дают преимущества проектам agile. В таблице 4–1 показано, как сотрудничающие члены команды резко повышают производительность и способствуют инновационному решению проблем.

Таблица 4–1. Свойства успешных agile-команд

Характеристика	Цель
Выделенные люди	<ul style="list-style-type: none"> • Концентрация фокуса и увеличение производительности • Небольшая команда, меньше десяти человек
Члены кроссфункциональной команды	<ul style="list-style-type: none"> • Частая разработка и поставка • Поставка готовой ценности самостоятельной командой • Интеграция всех рабочих операций для поставки готовой работы • Получение информации по каналам обратной связи как от членов команды, так и от других людей, например от владельца продукта
Совместное размещение или способность найти решение любых проблем по размещению членов команды	<ul style="list-style-type: none"> • Улучшение коммуникаций • Улучшение динамичности команды • Обмен знаниями • Сокращение затрат на обучение • Способность полностью посвятить себя работе друг с другом
Смешанный состав команды, включающий широких и узких специалистов	<ul style="list-style-type: none"> • Узкие специалисты обеспечивают глубокие знания, а специалисты широкого профиля — гибкость при распределении заданий • Команда начинает с узкой специализации и часто становится командой специалистов расширенного профиля, имеющих основную специальность и, сверх этого, широкий опыт применения нескольких навыков
Стабильная рабочая среда	<ul style="list-style-type: none"> • Зависимость друг от друга в процессе поставки • Предварительно согласованный подход к работе • Упрощение расчета затрат команды (скорость потребления ресурсов) • Сохранение и расширение интеллектуального капитала

4.3.2 Роли agile

В agile имеются три общепринятые роли:

- ◆ члены кроссфункциональной команды,
- ◆ владелец продукта,
- ◆ фасилитатор команды.

Эти роли в команде описаны в таблице 4–2.

Таблица 4–2. Роли в agile-команде

Роль	Описание
Член кроссфункциональной команды	<p>В состав кроссфункциональных команд входят лица, обладающие всеми навыками, необходимыми для создания работающего продукта. В области разработки ПО в состав кроссфункциональных команд обычно входят проектировщики, разработчики, тестировщики и любые другие необходимые роли. Кроссфункциональные команды разработчиков включают в себя профессионалов, поставляющих продукт с возможностью релиза с регулярной каденцией. Создание кроссфункциональной команды имеет важнейшее значение, поскольку она не зависит от внешних условий и может поставлять готовую работу в кратчайшие, насколько это практически возможно, сроки с более высоким качеством.</p>
Владелец продукта	<p>Владелец продукта отвечает за определение направления работы над продуктом. Владельцы продукта определяют важность работы на основе ее бизнес-ценности. Владельцы продукта ведут работу со своими командами ежедневно, обеспечивая обратную связь о продукте и определяя направление по следующему элементу функциональности, который требуется разработать/поставить. Это означает, что работа имеет небольшой объем, часто настолько небольшой, что ее можно описать в одной учетной карточке.</p> <p>Владелец продукта ведет работу с заинтересованными сторонами, заказчиками и командами, чтобы определить направление работы над продуктом. Обычно владельцы продукта имеют прошлый опыт работы в бизнесе и могут использовать глубокие экспертные знания в предметных областях при принятии решений. В некоторых случаях владелец продукта обращается за помощью к людям, обладающим глубокими экспертными знаниями в специальных областях, например к архитекторам, или в работе с заказчиками, например к менеджерам по продукции. Владельцам продукта необходимо пройти обучение по вопросам организации и управления потоком работы в команде.</p> <p>В agile владельцы продукта с участием команды создают бэклог. Бэклог помогает командам понять, как обеспечить поставку максимальной ценности без создания потерь.</p> <p>Важнейшим фактором успеха для agile-команд является уверенное владение продуктом. Если не заниматься вплотную вопросами обеспечения максимальной ценности для заказчика, команда agile может создать свойства, которые заказчик не оценит, то есть не представляющие достаточной ценности для заказчика, и таким образом бесполезно потратить рабочее время.</p>
Фасилитатор команды	<p>Третья по важности роль, которую можно обычно найти в agile-команд, принадлежит фасилитатору команды или обслуживающему лидеру. Эту роль выполняет лицо, которое можно называть руководителем проекта, скрам-мастером, лидером команды проекта, коучем команды или фасилитатором команды.</p> <p>Всем agile-командам требуется обслуживающее лидерство для организации ее работы. Людям требуется время для приобретения навыков обслуживающего лидерства, позволяющих осуществлять фасилитацию, коучинг и устранение препятствий.</p> <p>В начале многие организации приглашают внешнего agile-коуча, чтобы он помог, когда внутренний потенциал коучинга в команде еще не развит в полной мере.</p> <p>Внешние коучи имеют преимущество прежнего опыта, но у них есть и недостаток, который состоит в недостаточно прочных отношениях с организацией заказчика. С другой стороны, у внутренних коучей есть прочные отношения со своей организацией, но может отсутствовать широта опыта, которая бы сделала их работу высокоэффективной.</p>

4.3.3 Специалисты широкого профиля

Agile-команды являются кроссфункциональными, но люди, приступая к работе, часто не являются таковыми. С другой стороны, многие успешные agile-команды состоят из специалистов широкого профиля или людей с «Т-образной» квалификацией.

Это означает, что члены команды, помимо основной специальности, имеют большой опыт применения самых разных навыков, а не какую-то одну специализацию. Члены agile-команд, в силу необходимости интенсивного сотрудничества и самоорганизации, стремятся развивать такие качества, чтобы можно было быстро выполнить работу совместными усилиями, что предполагает повседневную взаимопомощь.

Отдача одного отдельного человека не является существенной. Сосредоточение на отдаче одного человека может даже причинить вред, если это становится узким местом для остальной команды. Цель команды состоит в оптимизации поставки готовой работы с получением обратной связи.

Если заказчик хочет получить великолепные результаты, например, быструю поставку свойства с отличным качеством, структура команды не может быть определена просто на основе ролей специалистов в стремлении максимально повысить эффективность использования ресурсов. Цель команды – добиться эффективности потока, оптимизировать отдачу всей команды. Небольшие размеры партий способствуют работе команды как единого целого. Цель работы владельца продукта – сделать так, чтобы команда занималась работой, имеющей наибольшую ценность.

«ЛЮДИ С I-ОБРАЗНОЙ И T-ОБРАЗНОЙ КВАЛИФИКАЦИЕЙ»

Некоторые люди имеют глубокую специализацию в какой-то одной области и редко вносят вклад в работу в других областях. Такие люди в сообществах agile известны как «I-образные», поскольку, подобно букве «I», они обладают глубиной, но особой широтой не отличаются. Наоборот, для «T-образных» в дополнение к экспертному знанию одной области характерны владение вспомогательными, хоть и менее развитыми, навыками в смежных областях и хорошие навыки совместной работы. К примеру, «T-образным» считается человек, который может проводить тестирование некоторых свойств продукта и, кроме того, разрабатывать другие его свойства.

«T-образный» человек имеет определенную признанную специализацию и основную роль, но, помимо этого, обладает навыками, разносторонностью и расположенностью к совместной работе для оказания помощи другим людям по мере необходимости. Такое сотрудничество сокращает количество передач работы между членами команды и ограничения, связанные с тем, что работу может выполнить только один человек.

4.3.4 Структуры команды

Во многих отраслях команды приняли принципы и практики agile. Они организуют людей на принципах кроссфункциональных команд с целью итеративной разработки рабочих продуктов.

ПРАКТИЧЕСКИЙ ПРИМЕР

Члены основной команды, сформированной для подготовки текста настоящего Практического руководства, имели разный предшествующий опыт работы, когда некоторые из них представляли PMI, другие – Agile Alliance. Чтобы выполнить работу, они самоорганизовались и работали инкрементно. PMI собрал группу экспертов по предметным областям для инспектирования работы, что позволило команде использовать результаты обратной связи и улучшать продукт по мере его разработки. Однако основная команда не была образцом типичной команды agile, так как рабочее время ее членов не было полностью (на 100 %) посвящено этому проекту.

В некоторых организациях смогли создать кроссфункциональные команды с совместным размещением, в других ситуация была иная. Вместо того, чтобы размещать всех членов команды в одном месте, некоторые организации используют «распределенные» или «рассеянные» команды. В случае с распределенными командами кроссфункциональные команды находятся в разных местах. Члены рассеянных команд могут работать в совершенно разных местах, как в офисе, так и на дому. И хотя такие формы организации работы не являются идеальными из-за увеличения расходов на связь, они все же могут быть вполне целесообразными.

В одном крупном финансовом учреждении США осуществлялась программа с участием нескольких команд, одни члены которых находились на Восточном побережье США, а другие – на территории Индии. Когда эта команда впервые приступила к работе, она представляла собой одну большую рассеянную команду (UX, аналитики, разработчики и тестировщики), члены которой вели разработку в порядке «следуя за солнцем»², когда рабочее время разных членов команды накладывалось друг на друга, что позволяло передавать работу из рук в руки на ходу. Члены команды проводили совместные ежедневные летучки с использованием веб-камер, чтобы охватить всех членов команды. Основные роли (аналитики, владельцы продукта, UX-дизайнеры и лидеры разработки) в США выходили на

связь раньше, чтобы ответить на все вопросы, которые возникали у их коллег в Индии, и помочь в устранении препятствий.

Когда объем продукта начал расти и стали поступать дополнительные средства финансирования, члены команды решили разделить на пять команд меньшего размера. С этой целью они решили образовать распределенные команды с совместным размещением в разных местах. Было принято решение создать кроссфункциональные команды с совместным размещением в каждом из этих мест в составе разработчиков и тестировщиков.

У них также был основной состав аналитиков, находившихся в двух местах в США, которые вели работу со своим менеджером продукта и владельцами продукта в США, а затем – с каждой из команд соответственно. Хотя у них действовала определенная структура, – они проводили анализ продукта в рамках всей программы, – большая часть их прочей деятельности осуществлялась на уровне команды, исходя из того, что было наиболее полезно для команды, чтобы создать для ее членов возможность самоорганизации.

4.3.5 Выделенные члены команды

Что происходит, когда члены команды уделяют менее 100 % рабочего времени работе в команде? Хотя такая ситуация далека от идеала, к сожалению, в некоторых случаях ее невозможно избежать.

Основная проблема в ситуации, когда сотрудник уделяет работе в команде от 25 % до 50 % своего времени, состоит в том, что он работает в многозадачном режиме и вынужден переключаться от задачи к задаче. Многозадачный режим работы снижает производительность команды и отрицательно влияет на ее способность уверенно прогнозировать поставку.

ПОЛЕЗНЫЙ СОВЕТ

Многозадачность замедляет ход работы всей команды, поскольку члены команды теряют время на переход от одной рабочей задачи к другой и/или на ожидание окончания работы. Когда люди посвящают 100 % своего рабочего времени команде, она демонстрирует наибольшую скорость выпуска продукции.

При переходе от одной задачи к другой производительность труда людей снижается на 20–40 %. Снижение производительности труда растет экспоненциально по мере увеличения количества задач.

Когда человек выполняет два разных проекта, его вовлеченность в каждый проект не составляет 50 %. В действительности, из-за дополнительных потерь от переключения между задачами его отдача в каждом проекте составит от 20 % до 40 %.

При работе людей в многозадачном режиме выше вероятность возникновения ошибок. Переключение между задачами потребляет рабочую память, и люди, работающие в многозадачном режиме, скорее забывают контекст своей работы.

Когда все члены команды на 100 % задействованы в одном проекте, у них есть возможность для постоянного сотрудничества в составе одной команды, что делает работу каждого из них более результативной.

ПРАКТИЧЕСКИЙ СОВЕТ

Поскольку основные члены команды, разрабатывающие настоящее Руководство, не могут на 100 % посвятить себя работе в составе команды, производительность у них значительно ниже, чем она могла бы быть, если бы у них была возможность находиться в одном месте и уделять проекту все свое внимание в полном объеме. Однако, несмотря на то, что организация совместной работы экономически оправдана, даже если работа ведется в условиях рассеянной команды или лишь с частичной занятостью работников, совместное расположение и сосредоточение сил в полном объеме не представляются возможными. Поэтому команда отнесла распределенный режим своей работы к потенциальному риску. Команда отслеживает и осуществляет мониторинг хода своей работы с помощью инструментов совместной работы и регулирует распределение заданий на основе учета индивидуальной нагрузки каждого члена команды.

Дополнительные советы о работе команд в средах agile, в частности о процессах в области знаний об управлении ресурсами проекта, см. в таблице А1-2 о разделении по группам процессов управления проектом и областям знаний.

ПОЛЕЗНЫЙ СОВЕТ

Не во всех командах есть все нужные роли. Например, некоторым командам нужна поддержка администраторов баз данных или аналитиков-исследователей. Когда в команде есть временно включенные в их состав специалисты, важно обеспечить наличие у всех одинаковых ожиданий. Уделяет ли этот специалист все 100 % времени работе в команде, и на какой срок он включен в состав команды? Согласуйте ожидания со всеми (специалистом и командой) с целью уточнить уровень его участия в работе, чтобы команда могла обеспечить поставку. Назначения с неполной занятостью создают риски для проекта.

4.3.6 Рабочее пространство команды

Команде требуется рабочее пространство, в котором ее члены могут работать совместно, чтобы осознавать себя единой командой и сотрудничать друг с другом. Некоторые agile-команды работают вместе в одном помещении. Другие команды имеют выделенное для них рабочее пространство для проведения летучек и обсуждений, а для самостоятельной работы используются отгороженные секции или отдельные офисы.

Хотя компании стремятся к созданию открытого, способствующего совместной работе рабочего пространства, организациям необходимо также позаботиться о создании тихих пространств для работников, которым требуется время для обдумывания и работы без помех. По этой причине компании проектируют офисное пространство со сбалансированным сочетанием общих или социальных мест с изолированными или личными пространствами, где люди могут работать без помех (иногда называемых «пещерами и парками»).

Если команда географически распределена, она решает, какой объем ее рабочего пространства будет виртуальным, а какой – физическим. Такие технологии, как обмен документами, проведение видеоконференций и другие формы виртуального взаимодействия, позволяют людям вести работу удаленно друг от друга.

Географически распределенным командам необходимо иметь виртуальное рабочее пространство. Кроме этого, следует подумать о том, как собирать членов команды через регулярные промежутки времени для личного контакта с целью создания атмосферы доверия и обучения совместной работе.

В числе заслуживающих внимания методов управления коммуникациями в рассеянных командах можно назвать «аквариумные окна» (fishbowl windows) и «удаленную работу в парах» (remote pairing):

- ◆ Создайте «аквариумное окно» путем образования постоянных каналов для видеоконференций между различными местами, в которых рассеяна команда. Люди включают канал в начале и закрывают его по окончании рабочего дня. Благодаря этому люди могут в любое время видеть и спонтанно общаться друг с другом, сокращая тем самым задержки в совместной работе, которые иначе неизбежно возникают в случае географического распределения команды.

- ◆ Организуйте «удаленную работу в парах», используя инструменты виртуальных конференций для общего доступа к экранам, включая каналы голосовой и видеосвязи. При создании условий работы с учетом разницы в часовых поясах такая организация работы может на практике быть не менее эффективной, чем работа в личном контакте.

ПОЛЕЗНЫЙ СОВЕТ

Формируйте команды, собирая вместе людей с различными навыками из разных функциональных подразделений. Обучайте руководителей и лидеров образу мышления agile и задействуйте их на ранних этапах преобразования в agile.

4.3.7 Преодоление внутриорганизационной изоляции

При формировании agile-команд лучше всего начинать с формирования закладывающей основу отношений атмосферы доверия и безопасной рабочей среды, в которой все члены команды будут иметь одинаковое право голоса и смогут высказать свое мнение, которое будет обязательно принято во внимание. Это, наряду с созданием образа мышления agile, является базовым фактором успеха, а все другие вызовы и риски могут быть снижены.

Во многих случаях в организациях с внутренней изоляцией при формировании кроссфункциональных agile-команд возникают препятствия. Члены команд, которые должны сформировать кроссфункциональные команды, обычно находятся в подчинении у разных руководителей и имеют разные метрики, с помощью которых руководители оценивают эффективность их работы. Руководителям необходимо сосредоточить внимание на эффективности потока (и показателях работы команды в целом), а не на эффективности использования ресурсов.

Для преодоления факторов внутриорганизационной изоляции ведите работу с разными руководителями этих членов команды и добейтесь, чтобы они выделили необходимых людей для работы в составе кроссфункциональной команды. Это не только создает синергетический эффект, но и позволяет организации увидеть, как рациональное использование кадровых ресурсов оптимизирует проект и создаваемый продукт.

Дополнительную информацию о командах см. в приложении X2 о свойствах, оказывающих влияние на адаптацию.

ПОЛЕЗНЫЙ СОВЕТ

В качестве лидера проекта agile в первую очередь обратите внимание на то, как можно сформировать команду, которая будет кроссфункциональной и состоять из членов, имеющих возможность на 100 % посвятить себя работе в команде. Даже если вначале будет возможность добиться только того, чтобы основные члены команды, такие как разработчики и тестировщики, каждый день вместе работали и общались, это станет шагом в верном направлении к формированию среды agile.

5. Реализация agile. Поставка в среде agile

5.1 Устав проекта и устав команды

Каждому проекту нужен устав проекта, чтобы команда проекта знала, для чего создается проект, в каком направлении нужно двигаться команде и в чем состоит цель проекта. Вместе с тем, одного устава проекта команде может быть недостаточно. Agile-командам требуются нормативы и понимание того, как вести совместную работу. В таком случае, команде может понадобиться устав команды.

Процесс создания устава помогает команде узнать, как вести совместную работу и сплотиться вокруг проекта.

Для проекта agile команде как минимум необходимо иметь общее представление о проекте или его цели, а также четкий набор рабочих соглашений. Устав проекта agile должен содержать ответы на следующие вопросы:

- ◆ Почему мы занимаемся этим проектом? Это общее видение проекта.
- ◆ Кто и как получит выгоды от проекта? Это может входить в общее видение проекта и/или в описание цели проекта.
- ◆ Что означает статус «сделано» для данного проекта? Это критерии релиза проекта.
- ◆ Как мы намерены работать совместно? Это объяснение предполагаемого потока работы.

Обслуживающий лидер может способствовать осуществлению процесса создания устава. Команда может сплотиться в ходе совместной работы, и подготовка устава проекта – отличная возможность, чтобы приступить к работе. Кроме того, совместная работа может быть необходима членам команды, чтобы понять, как они будут вместе работать дальше.

Командам не нужен формальный процесс подготовки устава при условии, что они знают, как нужно работать вместе. Некоторые команды получают выгоды от процесса подготовки устава команды. Ниже приводятся несколько соображений, которые могут использоваться членами команды при работе над уставом в качестве основы для их «социального договора»:

- ◆ ценности команды, такие как равномерное распределение работы и обязательные рабочие часы;
- ◆ соглашения о работе, например, что значит статус «подготовлено», при котором команда может принять работу; что значит статус «сделано», при котором команда может последовательно судить о завершенности; соблюдение установленных сроков или использование пределов незавершенных работ;
- ◆ основные правила, такие как регламент выступления одного человека на совещании;
- ◆ групповые нормы, такие как отношение команды ко времени проведения совещаний.

Вместе с членами команды обслуживающий лидер может принять решение рассмотреть другие правила поведения.

Следует помнить, что «социальный договор» команды (т. е. ее устав) определяет порядок взаимодействия ее членов. Цель устава команды состоит в формировании среды agile, в которой члены команды могут вести работу в составе команды с полной отдачей как единое целое.

5.2 Общепринятые практики agile

В разделах с 5.2.1 по 5.2.8 рассмотрены некоторые наиболее распространенные практики проектов agile.

5.2.1 Ретроспективы

Наиболее важной практикой является ретроспектива, так как она позволяет команде получить знания о своем процессе, усовершенствовать и адаптировать его.

Ретроспектива помогает команде учитывать опыт предыдущей работы над продуктом и своего процесса в прошлом. Один из принципов Agile-манифеста сформулирован следующим образом: «Команда должна систематически анализировать возможные способы улучшения эффективности и соответственно корректировать стиль своей работы».

Многие команды используют итерации (особенно продолжительностью 2 недели), поскольку после окончания каждой итерации следует демонстрация и ретроспектива. Однако команде не требуются итерации для ретроспективы. Команда всегда может решить провести ретроспективу в любой из следующих ключевых моментов:

- ◆ когда команда завершает релиз или поставляется что-либо – это не обязательно должен быть какой-то значительный инкремент, это может быть любой релиз, даже самый незначительный;
- ◆ по прошествии срока более нескольких недель после предшествующей ретроспективы;
- ◆ когда есть основания полагать, что в работе команды произошла заминка и поток завершённой работы в команде прекратился;
- ◆ когда в работе команды наступает какое-то контрольное событие.

Команды выигрывают в результате выделения достаточного времени для извлечения уроков как с помощью промежуточной ретроспективы, так и ретроспективы по окончании проекта. Командам необходимо извлекать уроки из продукта своей работы и своего рабочего процесса. Например, у некоторых команд возникают проблемы на этапе завершения работы. Когда в плане команды предусмотрено достаточное время, она может организовать проведение ретроспективы для сбора данных, обработать эти данные и решить, что потом можно попытаться сделать в качестве эксперимента.

Прежде всего, ретроспектива нужна команде не для поиска виновных, а для извлечения уроков из предыдущей работы и внесения небольших улучшений.

Ретроспектива предназначена для рассмотрения качественных (восприятие людей) и количественных (измерения) данных с последующим использованием этих данных для выяснения основных причин, разработки контрмер и планов действий. Команда проекта может принять состоящий из многих пунктов перечень мер для устранения препятствий.

Следует подумать об ограничении количества пунктов в перечне мер с учетом потенциала команды относительно решения вопросов по совершенствованию в ходе предстоящих итерации или рабочего периода. Попытка осуществить сразу слишком много улучшений, не добившись в итоге завершения ни одного из них, – это значительно хуже, чем включить в план меньше пунктов, но успешно довести до завершения их все. Затем, при наличии времени, команда может перейти к следующей возможности для улучшения, содержащейся в перечне. При выборе командой, какими улучшениями заниматься, следует решить, как будут измеряться конечные результаты. Потом, в следующем периоде времени, следует произвести оценку результатов, чтобы подтвердить успех или неудачу по каждому улучшению.

Члены команды под руководством фасилитатора из их числа определяют степень важности каждого пункта об улучшении, включенного в перечень. После того, как пункты об улучшении будут расположены в порядке их важности, команда выбирает пункт с соответствующим номером для работы в ходе следующей итерации (или добавляет работу в поток, если она ведется на основе потока).

5.2.2 Подготовка бэклога

Бэклог – это упорядоченный перечень всей работы, представленный в форме историй, для команды. Нет необходимости создавать все истории для всего проекта в целом до начала работы: требуется лишь такой объем, которого будет достаточно, чтобы создать общую картину первого релиза и далее определить достаточное количество рабочих задач для следующей итерации.

Владельцы продуктов (или команда владельцев продукта, в которую входит менеджер продукта и все соответствующие владельцы продукта для конкретных областей продукта) могут создать дорожную карту продукта, чтобы показать предполагаемую последовательность поставляемых результатов с течением времени. Владелец продукта пересматривает дорожную карту с учетом того, что производит команда. (Примеры дорожных карт см. в приложении X3 об инструментах фильтров приемлемости agile).

5.2.3 Уточнение бэклога

В итерационном agile владелец продукта во многих случаях проводит одно или несколько совещаний с командой посередине текущей итерации для подготовки некоторых историй для предстоящей итерации. Цель этих совещаний состоит в отработке достаточного числа историй, чтобы команда понимала, в чем они состоят и как соотносятся их объем друг с другом.

Общего мнения о том, какое время должен занимать процесс уточнения, не существует. Это непрерывный процесс, включающий следующие составные части:

- ◆ Уточнение «точно вовремя» для потокового agile. Команда берет для обсуждения следующую карточку из столбца, где указаны подготовленные рабочие задачи.

- ◆ Во многих командах, работающих в итерационном agile, проводятся обсуждения, ограниченные временными рамками продолжительностью 1 час, где-то посередине двухнедельной итерации. (Команда выбирает длительность итерации, которая обеспечивает им достаточную частоту обратной связи.)

- ◆ Многократные обсуждения уточнений в командах, работающих в итерационном agile. Команды могут прибегать к ним в случаях, когда члены команды не знакомы с продуктом, областью продукта или проблемной областью.

ПОЛЕЗНЫЙ СОВЕТ

Стоит подумать об использовании картирования воздействия, чтобы понять, как части продукта складываются в одно целое. При обычных обстоятельствах эту работу возглавляет владелец продукта. Обслуживающий лидер может организовывать проведение необходимых собраний при обслуживании проекта.

Совещания по уточнению бэклога позволяют владельцу продукта представить команде соображения по историям, а команде – узнать о потенциальных вызовах или проблемах, которые есть в данных историях. Если у владельца продукта нет уверенности относительно зависимостей, то он может предложить команде выполнить исследовательскую задачу по данному свойству, чтобы понять риски.

У владельца продукта есть много способов проведения подготовки бэклога и совещаний по его уточнению, в том числе:

- ◆ предложить команде работать тройками с участием разработчика, тестировщика, бизнес-аналитика/владельца продукта при обсуждении и написании истории;
- ◆ представить команде полную концепцию истории – команда обсуждает и уточняет ее с разбивкой на несколько историй, если требуется;
- ◆ вести работу с командой, чтобы найти различные способы для изучения и написания историй совместными усилиями, стремясь сделать все истории достаточно небольшими, чтобы команда могла создать устойчивый поток завершенной работы – стоит подумать, как добиться способности завершать историю не реже одного раза в день.

Команды часто ставят цель тратить не более 1 часа в неделю на уточнение историй для следующей части работы. Команды стремятся максимально увеличить время на выполнение работы за счет сокращения времени на планирование. Если команде требуется тратить более 1 часа в неделю на уточнение историй, то, вероятно, владелец продукта слишком много внимания уделяет подготовительной работе, или команде не хватает каких-то важнейших навыков, необходимых для оценки и уточнения работы.

5.2.4 Ежедневные летучки

Команды используют летучки, чтобы коротко договориться друг с другом, выявить возникающие проблемы и добиться бесперебойного потока работы в команде.

Ограничьте временные рамки для летучки 15 минутами. Команда «проходит» доску «канбан» или доску рабочих задач тем или иным способом, и кто-либо из членов команды выступает фасилитатором на летучке.

В итерационном agile каждый из участников «по кругу» отвечает на следующие вопросы.

- ◆ Что я сделал(-а) после последней летучки?
- ◆ Что я планирую закончить в промежутке между этой и следующей летучкой?
- ◆ Какие у меня есть препятствия (или риски, проблемы)?

Ответы на подобные вопросы позволяют команде осуществлять самоорганизацию и добиваться взаимной ответственности друг перед другом за исполнение порученной в предыдущий день работы и на протяжении всей итерации.

Потоковый agile предусматривает иной подход к проведению летучек, когда основное внимание обращается на производительность команды. Команда рассматривает доску рабочих задач справа налево. Нужно ответить на следующие вопросы:

- ◆ Что нам требуется сделать, чтобы продвинуться вперед в исполнении этой части работы?
- ◆ Делает ли кто-нибудь какую-то работу, которая не указана на доске?
- ◆ Что нам нужно завершить как команде в целом?
- ◆ Существуют ли какие-то узкие места, мешающие потоку работы?

Один из антишаблонов проведения летучек состоит в том, что летучка превращается в совещание о ходе выполнения работ. Для команд, которые раньше обычно работали в

предиктивных средах, может быть характерно стремление к этому антишаблону, поскольку такие команды привыкли обсуждать именно ход исполнения работ.

Другой антишаблон, который обычно встречается при проведении летучек, проявляется в том, что команда начинает искать решение проблем по мере их выявления. Летучка нужна для установления факта существования проблем, а не поиска их решений. Следует внести указанные проблемы в перечень требующих решения вопросов и созвать другое совещание, которое может состояться даже сразу после летучки, для поиска решения проблемы.

Команды проводят свои собственные летучки. Если летучки проводятся правильно, то они могут принести много пользы при условии, что характер работы команды требует интенсивной совместной работы. Необходимо осознанное решение о том, когда команда нуждается в проведении летучек или может их эффективно проводить.

ПОЛЕЗНЫЙ СОВЕТ

Следует предложить вести летучку любому члену команды, исключая руководителя проекта или лидера, с целью не допустить ее превращения в совещание о ходе выполнения работ, а добиться, чтобы она стала временем для самоорганизации команды и принятия обязательств ее членами друг перед другом.

5.2.5 Демонстрации / обзоры

По мере завершения работы над свойствами, как правило в форме пользовательских историй, команда периодически проводит демонстрации рабочего продукта. Владелец продукта смотрит демонстрацию и принимает или отклоняет истории.

В итерационном agile команда демонстрирует всю завершённую работу по окончании итерации. В потоковом agile команда демонстрирует завершённую работу, когда для этого наступает время, как правило, после того, как формируется комплект с достаточным количеством взаимосвязанных свойств. Командам, включая владельца продукта, требуется обратная связь для принятия решения о том, когда именно требуется получить отзывы о продукте по каналам обратной связи.

В качестве общей рекомендации советуем проводить демонстрацию того, что команда может показать как рабочий продукт, не реже одного раза каждые две недели. Для большинства команд такой периодичности демонстраций будет достаточно, чтобы члены команды получали сведения по обратной связи с частотой, исключающей возможность движения в неправильном направлении. Это также будет достаточной частотой, чтобы команды могли обеспечить приемлемую чистоту разработки продукта для создания завершённого продукта настолько часто, насколько они хотят или насколько это необходимо.

Основопологающим фактором, который делает проект проектом agile, является частота поставки рабочего продукта. Команда, которая не производит демонстрации или релизы, не может извлекать уроки достаточно часто и, скорей всего, не перейдет к методам agile. Команде может потребоваться дополнительный коучинг для того, чтобы она смогла осуществлять частые поставки.

5.2.6 Планирование для итерационного agile

У всех команд разные ресурсные возможности. У каждого владельца продукта типичный размер истории разный. Команды принимают решения в отношении объема своих историй с таким расчетом, чтобы брать на себя обязательства в объеме лишь такого числа историй, какой команда может завершить в течение одной итерации в соответствии со своей ресурсной возможностью.

Владелец продукта понимает, что в периоды отсутствия людей на работе (например, во время праздников, отпусков или других событий, которые не позволяют людям принять участие в работе на следующем этапе), ресурсная возможность команды снижается. Команда не сможет произвести такой же объем работы, который она завершила в предшествующий

период времени. Когда команда имеет пониженную ресурсную возможность, она планирует работу только в тех объемах, которые соответствуют ее имеющимся силам.

Команды дают оценку объема работы, который они смогут освоить, что является определенным показателем их ресурсной возможности (см. примеры в разделе 4.10). Команды не в состоянии с точностью до 100 % прогнозировать, что они будут в состоянии поставить, поскольку не могут учесть непредвиденные обстоятельства. Когда владельцы продукта готовят истории меньшего размера, а команды видят прогресс в форме конечного продукта, они узнают, что им по силам сделать при планировании на будущее.

Agile-команды осуществляют планирование не одновременно одним куском. Вместо этого постоянно возобновляемый цикл работы agile-команды включает планирование небольшой части работы, осуществление поставки, извлечение уроков и новое планирование небольшого объема.

ПОЛЕЗНЫЙ СОВЕТ

Следует привлечь внимание команды к неправильному подходу и помочь команде найти пути к улучшению ее летучек.

5.2.7 Практики исполнения, помогающие командам поставлять ценность

Если команда не уделяет внимания качеству, то вскоре она не сможет быстро выпускать что-либо.

Следующие технические практики, многие из которых пришли из сферы экстремального программирования, могут помочь команде осуществлять поставку с максимальной для нее скоростью.

♦ **Непрерывная интеграция.** Следует производить частую интеграцию работы в целое, независимо от продукта, и затем повторять тестирование, чтобы убедиться, что продукт в целом продолжает работать, как положено.

♦ **Тестирование на всех уровнях.** Следует проводить тестирование на уровне системы, чтобы получить информацию полного цикла, и испытания составных частей для создаваемых блоков. В промежутке следует изучить вопрос, есть ли необходимость производить интеграционные испытания и когда это лучше сделать. Команды находят полезным «дымовое» тестирование, чтобы убедиться в пригодности рабочего продукта. Команды установили, что принятие решений о том, когда и какие именно проводить регрессивное тестирование, помогает им обеспечивать качество продукта при хорошем исполнении сборки. Agile-команды имеют сильную склонность к проведению тестирования в автоматическом режиме, чтобы наращивать и поддерживать темпы поставки.

♦ **Разработка на основе приемочных тестов (ATDD).** При ATDD команда собирается вместе в полном составе и обсуждает критерии приемки для определенного рабочего продукта. После этого команда разрабатывает тесты, которые позволяют ей написать программу и автоматизированные тесты в объеме, достаточном для соответствия принятым критериям. В не связанных с разработкой ПО проектах следует подумать, как проводить тестирование имеющих ценность кусков работы по мере их завершения командой.

♦ **Разработка на основе тестов (TDD) и разработка на основе поведения (BDD).** Написание программ автоматизированных тестирований до написания/создания продукта помогает людям в проектировании и проверке отсутствия в продукте ошибок. В не связанных с разработкой ПО проектах следует подумать, как направлять работу команды по проектированию на основе тестирования. В проектах по созданию аппаратных и технических средств для проведения промежуточных тестирований часто используются технологии симуляции.

♦ **Исследовательские задачи (исследования и эксперименты в пределах временных рамок).** Исследовательские задачи полезны при извлечении уроков и могут использоваться для оценки, определения критериев приемки и понимания потока действий пользователя на

всем протяжении создания продукта. Исследовательские задачи оказываются полезными тогда, когда команде требуется изучить какой-то имеющий критическое значение технический или функциональный элемент.

5.2.8 Как итерации и инкременты помогают в поставке рабочего продукта

Итерации помогают команде в создании каденции для поставки и получения многих видов результатов обратной связи. Команды производят имеющие ценность инкременты для поставки и обратной связи. Первой частью этой поставки является демонстрация. Команды получают обратную связь о том, как продукт выглядит и работает, с помощью демонстрации. Члены команды проводят ретроспективы, чтобы понять, как они могут инспектировать и адаптировать свои процессы для успешной работы в дальнейшем.

Демонстрации или обзоры являются необходимой составляющей потока работ в проектах agile. Команде следует составлять расписание демонстраций с учетом каденции поставки.

5.3 Поиск и решение проблем проекта agile

Подходы agile возникают в связи с необходимостью поиска решений проблем, вытекающих из высоких темпов изменений в проектах, уровня их неопределенности и сложности. В силу своего происхождения они включают в себя разнообразные инструменты и методы, предназначенные для решения вопросов, которые приобрели характер проблем в среде предиктивных подходов. См. таблицу 5–1.

ПОЛЕЗНЫЙ СОВЕТ

Командам следует проводить демонстрации часто, чтобы получать обратную связь и наглядно представлять ход исполнения. Привлекайте ОУП и другие заинтересованные стороны к просмотру демонстраций, чтобы лица, принимающие решения о портфеле проектов, могли видеть фактический ход исполнения.

Таблица 5–1. Болевые точки подхода agile и возможности решения проблем

Болевые точки	Возможности решения проблем
Нечеткое определение цели или миссии для команды	Разработка устава на основе подхода agile для определения цели: видение, миссия и проверка миссии
Нечеткое определение рабочих соглашений команды	Разработка устава на основе подхода agile для согласования: ценности, принципы и рабочие соглашения
Нечетко определенный контекст работы команды	Разработка устава на основе подхода agile для определения контекста: границы, выделенные команде ресурсы и перспективный анализ
Нечетко определенные требования	Помогите спонсорам и заинтересованным сторонам выработать видение продукта. Попробуйте создать дорожную карту продукта, используя спецификацию на примерах, картирование пользовательских историй и картирование воздействия. Устройте вместе с командой и владельцем продукта обсуждение для прояснения ожиданий и ценности требования. Последовательно разложите дорожную карту на бэклог из более мелких конкретных требований.
Низкий уровень пользовательского восприятия продукта	Практики проектирования пользовательского восприятия, используемые командой разработчиков, предусматривают частое привлечение пользователей уже на ранних этапах.
Неточная оценка	Уменьшение размера историй, разбивая большие истории на несколько мелких. Использование относительной оценки с привлечением команды в полном составе. Следует подумать о гибком моделировании или исследовательских задачах, чтобы понять суть истории.
Нечеткое распределение рабочих заданий или ход исполнения работ	Помощь команде в понимании, что она самостоятельно организует свою работу. Следует подумать об использовании досок «канбан», чтобы наглядно представить поток работы. Следует подумать о рассмотрении доски на ежедневных летучках, чтобы видеть состояние дел с каждой работой.
В работе команды постоянно возникают препятствия	Обслуживающий лидер может помочь устранить препятствия. Если команда не знает, какие у нее есть варианты, подумайте об использовании коуча. В некоторых случаях команде необходимо использовать эскалацию историй, которые команда или обслуживающий лидер были не в состоянии устранить.
Задержки/опережения в исполнении работы по причине недостаточной проработки рабочих задач бэклога	Владелец продукта и команда проводят совместные семинары для отработки историй. Следует создать критерии готовности для историй. Следует подумать о возможности разбивки историй с целью использования историй меньшего объема.
Дефекты	Использование технических практик, соответствующих среде проекта. Среди возможных вариантов есть следующие: работа в парах, коллективное владение продуктом, комплексное тестирование (подход на основе приемочных тестов и автоматизированное тестирование) и хорошо продуманные критерии выполнения.
Работа не завершена	Команда определяет для историй критерии выполнения, включая критерии приемки. Следует использовать также критерии релизов для проектов.
Технический долг (низкое качество кода)	Рефакторинг, гибкое моделирование, комплексное тестирование, автоматизированный анализ качества кода, критерии выполнения.

Таблица 5–1. Болевые точки подхода agile и возможности решения проблем (прод.)

Болевые точки	Возможности решения проблем
Чрезмерная сложность продукта	Стимулирование команд, связанных и не связанных с разработкой ПО проектов, всегда задавать себе вопрос: «Какое самое простое решение позволит решить задачу?», и применение принципа agile: «Простота — искусство максимизации объема работы, которую не нужно выполнять». Эти приемы помогают снизить уровень сложности.
Медленное совершенствование работы или его полное отсутствие в процессе команды	Выбор для рассмотрения не более трех рабочих задач для улучшения на каждой ретроспективе. Предложите обслуживающему лидеру помочь команде выяснить, как интегрировать эти рабочие задачи.
Доработка по причине чрезмерного объема предварительной работы	Для извлечения уроков следует подумать об использовании исследовательских задач вместо значительных объемов предварительной работы. Кроме этого, следует измерять незавершенный объем работ уже в начале проекта и смотреть, какие у команды есть варианты для поставки ценности, а не проектных решений. Следует сократить сроки итераций и создать хорошо продуманные критерии выполнения.
Фальстарты, работа впустую	Владелец продукта должен стать неотъемлемой частью команды.
Нерациональный порядок расположения рабочих задач бэклога продукта	Расположите элементы в порядке возрастания ценности, включая стоимость задержки задачи, поделенную на ее длительность (CO3), и другие модели оценки ценности.
Поток работы с периодами спешки/простоев	Планирование работы только в объемах, не превышающих ресурсные возможности команды. Следует предложить людям прекратить работу в многозадачном режиме и полностью посвятить время только одной команде. Предложите команде работать в парах, в режиме роения или моббинга, чтобы равномерно распределить производственные силы по всей команде.
Невыполнимые требования заинтересованной стороны	Использование обслуживающего лидерства для работы с такой заинтересованной стороной (и, возможно, с владельцем продукта).
Внезапные или непредвиденные задержки	Команда должна чаще отслеживать ход работ, используя доски «канбан», чтобы видеть поток работы и пределы незавершенной работы, чтобы понимать воздействие требований на команду и продукт. Следует следить также за ситуаций с препятствиями и их устранением, используя доску препятствий.
Изолированные команды вместо кроссфункциональных команд	Предложение людям, задействованным в проекте, самостоятельно образовать кроссфункциональные команды. Следует использовать навыки обслуживающего лидерства, чтобы помочь руководителям понять, почему agile требует создания кроссфункциональных команд.

5.4 Измерения в проектах agile

Переход к agile требует применения иных измерений. Использование agile означает поиск новых показателей, которые имеют значение для команды и руководства. Эти метрики имеют важное значение, поскольку сфокусированы на поставке ценности заказчику.

Одной из проблем отчетности о ходе работ является способность команды прогнозировать завершение или использовать трехцветное представление статуса для описания проекта. Например, лидеры проекта определяют, что проект выполнен «на 90 %». В этот момент команда пытается осуществить интеграцию отдельных частей в продукт. Команда обнаруживает недостающие требования или непредвиденные обстоятельства, или выясняет, что ход интеграции продукта идет не так, как предполагалось.

Проект выполнен не полностью, и отчет с трехцветным представлением статуса не отражает истинного состояния. Слишком часто команда проекта приходит к заключению, что ей потребуется еще столько же времени на завершение остальной части проекта. Есть слишком много проектов, в которых команда приходит к выводу, что из-за вновь выявленных проблем часть завершенных работ по проекту составляет всего 10 % общего объема.

Проблема предиктивных измерений состоит в том, что эти измерения во многих случаях не отражают истинного состояния. Часто случается, что статус проекта показан зеленым цветом вплоть до 1 месяца перед датой релиза, что иногда называют проектом, подобным

арбузу (снаружи – зеленый, внутри – красный). Часто случается так, что цвет представления статуса проекта становится красным внезапно, без каких-либо признаков, из-за отсутствия эмпирических данных о проекте вплоть до 1 месяца перед датой релиза.

Метрики проектов agile содержат значимую информацию, которая дает исторические сведения по отслеживанию, поскольку поставка ценности (готовой работы) в проектах agile осуществляется на регулярной основе. Команды проекта могут использовать такие данные для улучшения прогноза и принятия решений.

Суррогатные измерения, такие как процент исполнения, менее полезны в сравнении с эмпирическими показателями, такими как завершенные свойства. Дополнительную информацию об измерении ценности смотрите в разделе 4.10. Agile помогает командам видеть проблемы и сложные вопросы, что позволяет ей диагностировать и решать их.

Кроме сбора количественных измерений команда может рассмотреть возможность сбора качественных измерений. В центре некоторых из таких качественных измерений находятся выбранные командой практики, и они служат для оценки того, насколько хорошо команда использует эти практики, например, степень удовлетворенности бизнеса поставленными свойствами, моральное состояние команды и тому подобные свойства, за которыми команде нужно следить с использованием качественных показателей.

5.4.1 Результаты измерений работы agile-команд

В agile предпочтительно использовать эмпирические и основанные на ценности показатели, а не предиктивные измерения. В agile измеряется то, что команда поставляет, а не то, что она предполагает поставить.

Команда, которая привыкла иметь базовые планы проекта и оценки освоенного объема и возврата инвестиций (Return On Investment, ROI), может оказаться в затруднительном положении при работе над проектом, если она не управляет им по базовому плану. Agile основана на работающих продуктах, имеющих ценность, которую можно продемонстрировать заказчикам.

Базовые планы часто являются артефактом попытки прогнозирования. В agile команда ограничивает свою оценку периодом продолжительностью не более нескольких следующих недель. В agile в условиях низкой степени вариативности в работе команды и при условии работы ее членов не в режиме многозадачности ресурсная возможность команды может стать стабильной. Это позволяет лучше прогнозировать результаты на период следующих двух недель.

После завершения командой работы в итерации или потоке она может выполнить перепланирование. Agile не создает возможности выполнять больший объем работы. Однако, существует подтверждение того, что чем меньше часть работы, тем выше вероятность, что люди смогут выполнить ее поставку.

Разработка продукта программного обеспечения, как и другая умственная работа, состоит в обучении – обучении в процессе поставки ценности. Разработка аппаратных средств и техническая разработка подобны в части проектирования проекта. Обучение происходит при экспериментировании, поставке небольших инкрементов и получении обратной связи о том, что было выполнено на данный момент. Многие проекты по разработке продукта также включают обучение.

Спонсоры обычно хотят знать, когда проект будет завершен. Команда может прогнозировать, сколько времени еще потребуется для завершения проекта после того, как определит достоверную скорость (среднее количество историй или относительных единиц на итерацию) или средний срок цикла.

Например, если команда в среднем выполняет 50 относительных единиц историй за одну итерацию, и команда считает, что ей остается выполнить еще 500 единиц, то, по оценке команды, ей надо пройти еще около 10 итераций. По мере уточнения владельцем продукта остающихся историй, а командой – своих оценок, оценка продолжительности проекта может увеличиваться или сокращаться, но команда в состоянии дать некоторую оценку.

Если команда оценивает время цикла в среднем как три дня на каждую историю, и ей остается пройти еще 30 историй, команде потребуется на исполнение работы еще 90 рабочих дней, т. е. примерно от 4 до 5 месяцев.

Следует отразить вариативность оценки с помощью диаграммы в стиле «урагана» или какого-либо другого способа измерения вариативности, который будет понятен спонсорам.

Поскольку обучение является значительной частью проекта, команде требуется найти баланс между неопределенностью и обеспечением ценности для заказчиков. Команда планирует следующую небольшую часть проекта. С целью управления неопределенностью проекта команда сообщает эмпирические данные и пересматривает планирование последующих небольших инкрементов.

В некоторых основанных на итерации проектах, чтобы наглядно представить, где проект превышает установленные сроки, используются диаграммы сгорания. На рис. 5–1 показан пример диаграммы сгорания, где команда планировала осуществить поставку 37 относительных единиц. Относительные единицы определяют темпы соответствующей работы, риск и сложность требования или истории. Многие agile-команды используют относительные единицы для оценки трудозатрат. Пунктирная линия сгорания представляет план. На рис. 5–1 по результатам на третий день команда может видеть, что у нее появился риск для данной поставки.



Рис. 5–1. Диаграмма сгорания остатка относительных единиц

В некоторых проектах предпочитают использовать диаграммы выгорания. Те же данные, которые использованы на рис. 5–1, представлены на рис. 5–2 в форме диаграммы выгорания.



Рис. 5–2. Диаграмма выгорания для наглядного представления завершенных относительных единиц

На диаграммах выгорания представляется завершенная работа. Две диаграммы на рис. 5–1 и 5–2 основаны на одних и тех же данных, но представляют их двумя разными способами. Команды могут сами решать, какое представление своих данных лучше использовать.

Когда команда видит, какую работу она еще не сделала в ходе итерации, ее моральный дух может упасть, и существует вероятность, что она начнет спешно завершать работы, перестав соблюдать критерии приемки. Однако у команды может быть ряд достаточных соображений не завершать работу так, как предполагалось. Диаграммы сгорания показывают последствия работы команды в многозадачном режиме, использования историй чрезмерно большого объема или отсутствия членов команды на рабочем месте.

Диаграммы выгорания показывают изменения содержания в течение итерации, что особенно характерно для команд, которые только начинают работать в agile. Диаграммы выгорания позволяют командам видеть уже сделанное ими, что помогает им перейти к следующему элементу работы.

При использовании как диаграмм выгорания, так и диаграмм сгорания команды видят то, что они уже завершили в рамках своих итерационных процессов. В конце итерации они могут произвести оценку следующего объема работ (количество историй или относительных единиц) на основе того, какой объем работ они завершили в данной итерации. Это позволяет владельцу продукта вместе с командой пересмотреть план в отношении того, что команда с высокой степенью вероятности сможет поставить в следующей итерации.

Скорость, сумма объемов относительных единиц для фактически завершенных свойств в данной итерации позволяет команде планировать следующий объем с большей степенью точности, исходя из ее производительности в прошлом.

Команды, работающие по потоковому agile, используют разные измерения: время ожидания (общее время, которое требуется для поставки какого-то элемента, измеряемое от времени его включения в доску заданий до момента его завершения), время цикла (время, необходимое на обработку элемента), и время отклика (время, которое элемент находится в состоянии ожидания до начала работы). Команды измеряют время цикла, чтобы увидеть узкие места и задержки, которые не обязательно находятся внутри команды.

ПОЛЕЗНЫЙ СОВЕТ

Команды могут определить, что может потребоваться от четырех до восьми итераций для достижения устойчивой скорости работы. Команде нужна обратная связь

по итогам каждой итерации, чтобы понимать, как она работает и как можно улучшить свою работу.

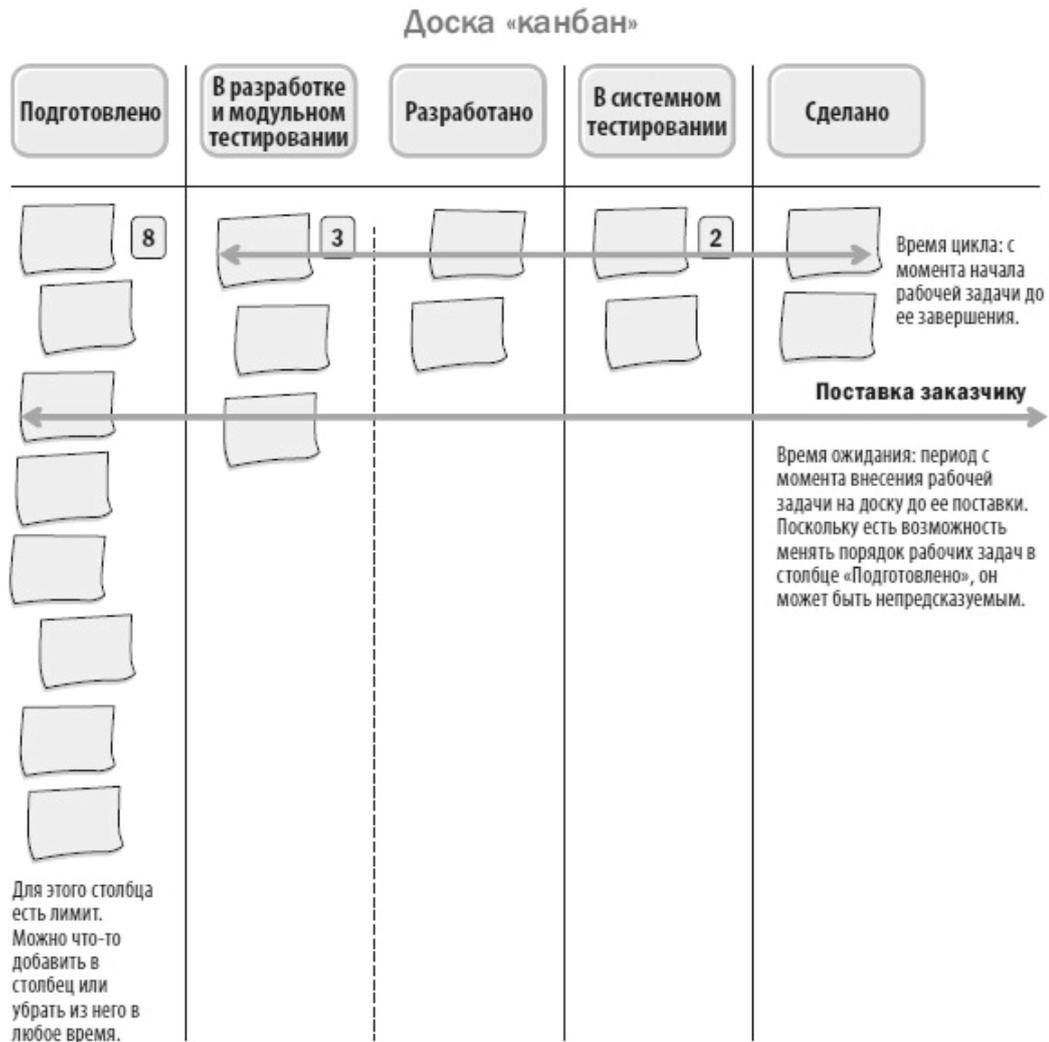


Рис. 5–3. Пример доски «канбан»

Знать время ожидания полезно, чтобы понять время цикла между первым подходом к определенному свойству и началом отрезка времени, которое требуется для релиза этого свойства заказчику. Лимиты незавершенной работы (work in progress, WIP) вверху каждого столбца, показанные здесь в квадратиках, позволяют команде видеть, как продвигать работу по доске. После того, как команда исчерпала свои лимиты незавершенной работы, она не может перемещать работу слева в следующий столбец справа. Вместо этого, команда выполняет работу из наиболее заполненного столбца справа и ставит вопрос: «Что мы делаем как команда, чтобы переместить эту работу в следующий столбец?»

Каждое свойство является уникальным, поэтому время его цикла также уникально. Однако владелец продукта может заметить, что меньшие свойства имеют более короткое время цикла. Владелец продукта хочет видеть выпускаемый объем, поэтому он создает меньшие свойства или работает с командой, чтобы сделать это.

Диаграммы сгорания и выгорания (измерения ресурсных возможностей) и время ожидания, а также время цикла (измерения прогнозируемости) полезны для моментальных измерений. Они помогают команде понять, какой объем работы у них остается и сможет ли команда выполнить его в срок.

Измерение в относительных единицах – это не то же самое, что измерение завершённых историй или свойств. Некоторые команды пытаются измерять относительные единицы без завершения реального свойства или истории. При измерении командами в относительных единицах они измеряют только ресурсные возможности, а не завершённую работу, что нарушает принцип: «работающий продукт – основной показатель хода исполнения» (так же, как и любой другой продукт, если это не программное обеспечение).

Каждая команда имеет свою собственную ресурсную возможность. При использовании командой относительных единиц следует иметь в виду, что количество относительных единиц работы, которые может завершить команда в заданный период времени, является уникальным для данной команды.

ПОЛЕЗНЫЙ СОВЕТ

Когда работа команды зависит от внешних людей или групп, следует измерять время цикла, чтобы определить, сколько команде требуется времени, чтобы завершить работу. Измерьте время ожидания, чтобы увидеть внешние зависимости после завершения работы командой. Команды могут также измерить время реакции, время перемещения подготовленных задач в первый столбец, чтобы увидеть, сколько им требуется времени (в среднем) для реакции на новые запросы.

Когда команды применяют собственные единицы измерения, они могут лучше определять и оценивать, а также поставлять свою работу. Негативной стороной относительной оценки является отсутствие возможности сравнивать работу команд или наращивать скорость работы разных команд в совокупности.

Команда может измерять завершённую работу диаграммой сгорания/выгорания работ по свойству или диаграммой выгорания бэклога продукта. Эти диаграммы наглядно представляют тенденции завершения работ с течением времени, как показано на рис. 5–4.



Рис. 5–4. Диаграмма свойств

Диаграммы сгорания/выгорания свойств могут показывать, что в ходе выполнения проекта количество требований растёт. Линия завершённых свойств показывает, что команда завершает свойства в равномерном темпе. Линия общего количества свойств показывает, как изменяется общее количество свойств с течением времени. Линия сгорания остающихся свойств показывает, что темп завершения свойств меняется. Изменение линии сгорания свойств происходит каждый раз при добавлении новых свойств в проект.

В agile освоенный объем определяется объемом завершенных свойств, как показано на рис. 5–5. Диаграмма выгорания бэклога продукта показывает объем завершенной работы в сравнении с предполагаемым общим объемом работы через промежутки времени между контрольными событиями или итерациями.

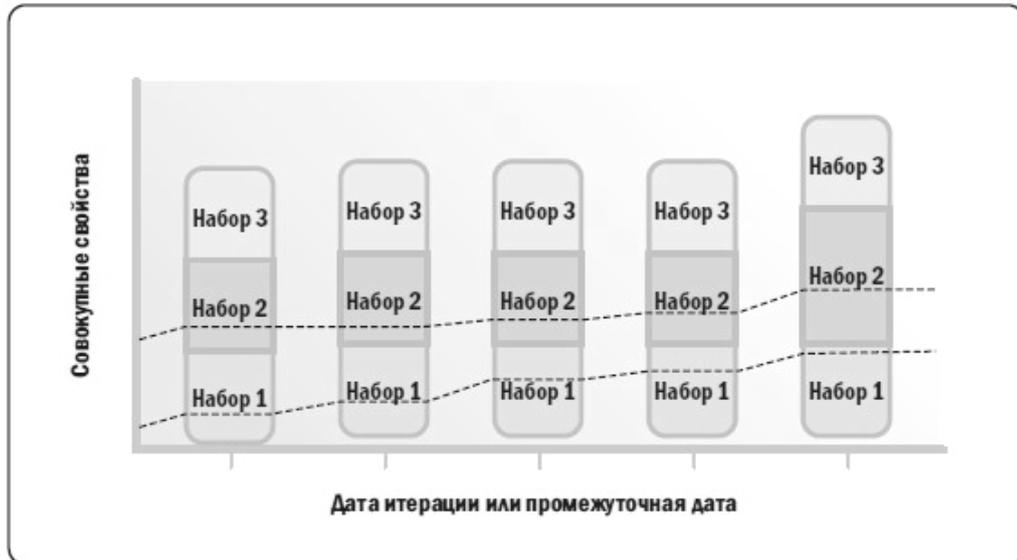


Рис. 5–5. Диаграмма выгорания бэклога продукта

Команда может одновременно завершить только одну историю. Чтобы завершить более объемное свойство, которое включает несколько историй, команде требуется завершить остающиеся истории, и она может закончить работу с данным свойством в полном объеме только по истечении еще нескольких периодов времени. Команда может показать созданную ею ценность с помощью диаграммы выгорания бэклога продукта, как показано на рис. 5–5.

Если команде требуется измерить освоенный объем, она в качестве образца может рассмотреть возможность использования диаграммы выгорания, которая представлена на рис. 5–6. Обратите внимание, что ось Y слева представляет количество относительных единиц как содержание, а ось Y справа показывает расходы по проекту.

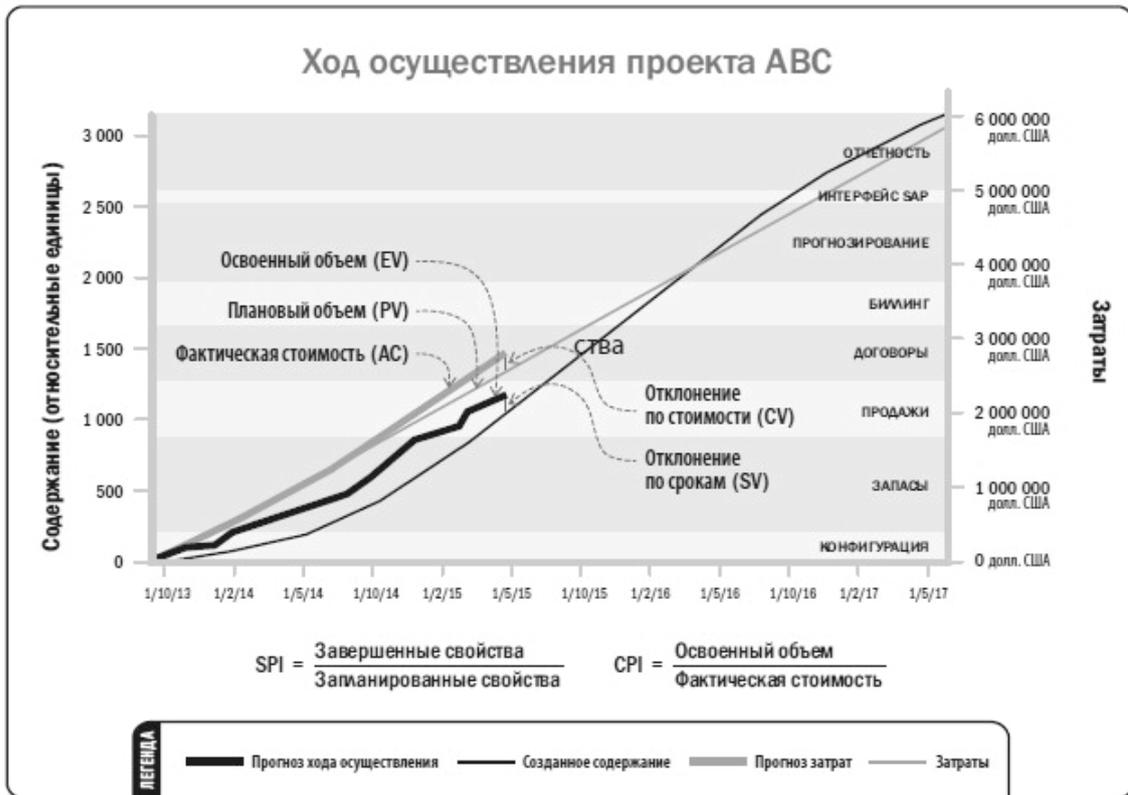


Рис. 5–6. Освоенный объем в контексте agile

Обычно используемые метрики управления освоенным объемом (earned value management, EVM), как, например, индекс выполнения сроков (schedule performance index, SPI) и индекс выполнения стоимости (cost performance index, CPI), можно без труда представить в терминах agile. Например, если команда планировала завершить 30 относительных единиц в одной итерации, но фактически завершила лишь 25, то SPI составляет 25/30 или 0,83 (т. е. темп работы команды составляет лишь 83 % от планового). Таким же образом, CPI – это освоенный объем (т. е. ценность завершенных свойств) на текущую дату, поделенный на фактическую стоимость на текущую дату или, как показано на рис. 5–6, 2,2 млн долл. США / 2,8 млн долл. США = 0,79. Это означает, что в сравнении с планом из каждого доллара по плану получено лишь 79 центов (но, безусловно, этот расчет основан на предположении, что прогноз остается верным).

На приведенной на рис. 5–7 диаграмме суммарного потока показана работа в процессе исполнения по всей доске задач. Если команда имеет много историй, ожидающих тестирования, то на диаграмме ширина зоны тестирования увеличится. Достаточно одного взгляда, чтобы увидеть, где происходит аккумуляция работы.

У команд возникают трудности в случае аккумуляции работы: команда получает незавершенную работу вместо завершенной. Когда команда имеет большой объем незавершенной работы, происходит запаздывание поставки целого свойства. Чем больше команде требуется времени, чтобы произвести поставку, тем большее у нее нагрузка с учетом увеличения объема свойств, которые ей необходимо завершить в течение того же периода времени.

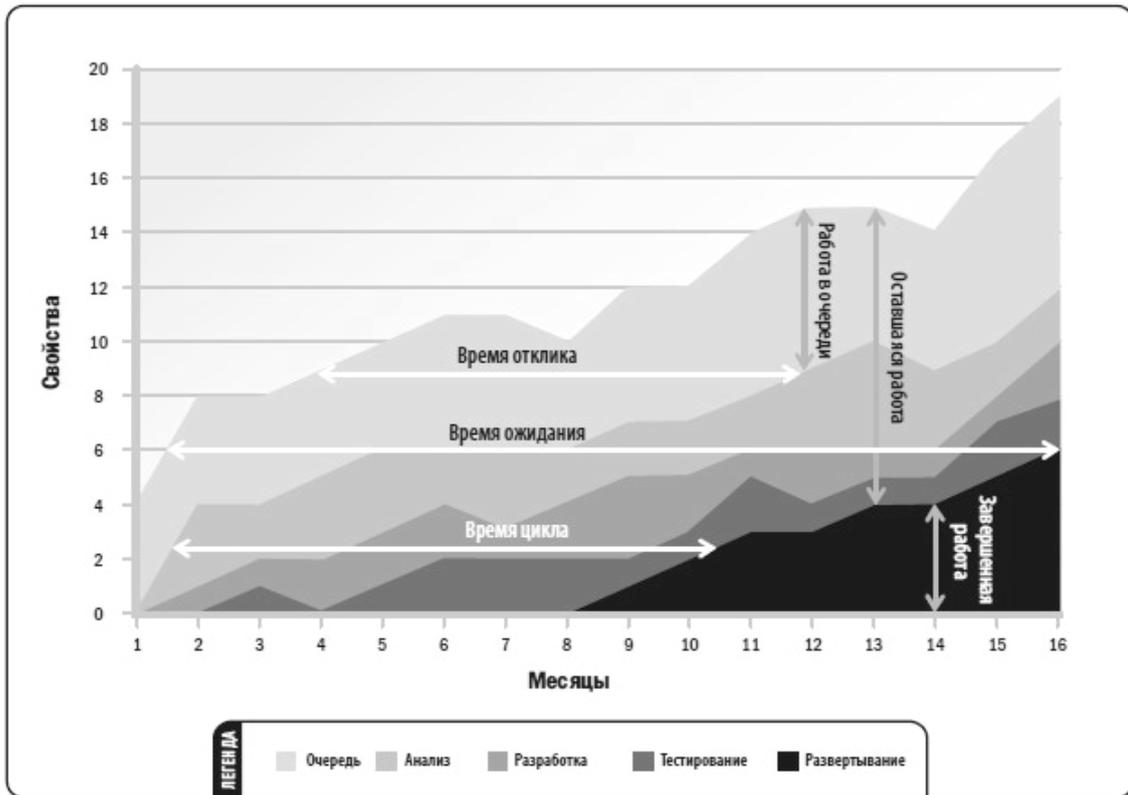


Рис. 5–7. Диаграмма суммарного потока завершенных свойств

Адаптируйте этот суммарный поток для доски рабочих задач проекта.

6. Организационные соображения для гибкости проекта

Каждый проект существует в некотором организационном контексте. Культуры, структуры и политики могут влиять как на управление проектом, так и на его результат. Эти факторы могут создавать проблемы лидерам проектов.

Хотя у лидеров может не быть возможности для изменения организационных факторов по своему усмотрению, ожидается, что они должны быть способны вести работу профессионально в условиях этих факторов.

В этом разделе исследуется, как организация, а при определенных обстоятельствах и контекст проекта, влияют на проекты. Лидеры могут изучить имеющиеся у них варианты внедрения изменений для повышения успешности проекта.

Гибкость проекта становится более результативной и устойчивой по мере приспособления организации к ее поддержке.

6.1 Управление организационными изменениями

Управление организационными изменениями включает в себя навыки и методы, необходимые для оказания влияния на изменения, которые помогают повысить уровень гибкости.

В публикации PMI «Управление изменениями в организациях: практическое руководство» (*Managing Change in Organizations: A Practice Guide*) [2] описан комплексный и целостный подход, обеспечивающий успешное осуществление значимых изменений. Содержащиеся там рекомендации включают в себя следующее:

- ◆ модели для описания динамики изменений,

- ◆ фреймворк для реализации изменений,
- ◆ применение практик управления изменениями на уровне проекта, программы и портфеля.

В разделах 6.1.1 и 6.1.2 рассматриваются соображения по управлению изменениями, относящимися непосредственно к контексту agile.

На рис. 6–1 показано отношение между этими двумя темами.

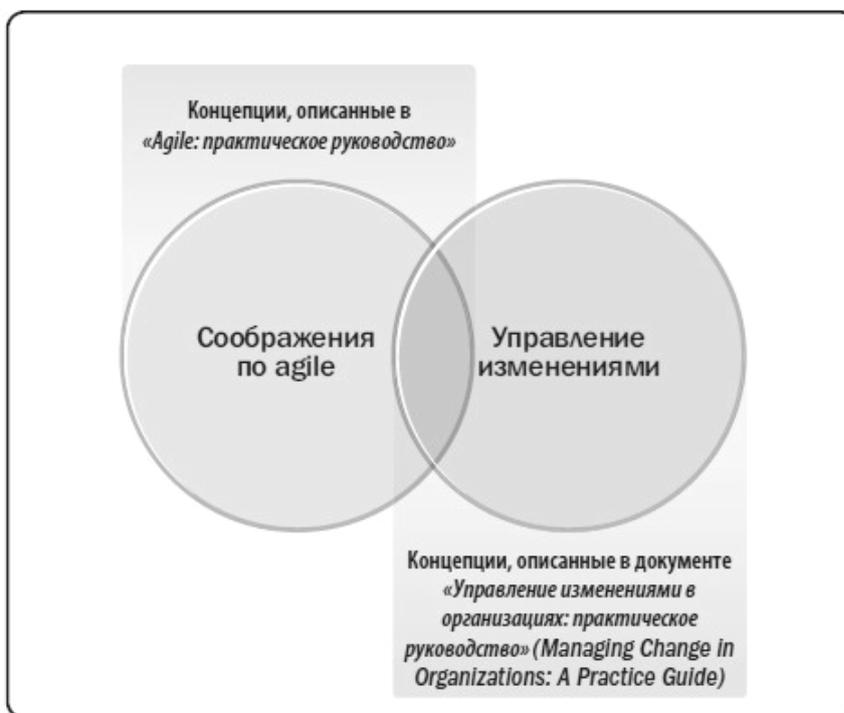


Рис. 6–1. Отношение между управлением изменениями и подходами agile

6.1.1 Источники управления изменениями

Все проекты решают задачу осуществления тех или иных изменений. Однако есть два ключевых фактора, которые дополнительно мотивируют к использованию практик управления изменениями в контексте agile:

◆ **Изменения, связанные с ускоренной поставкой.** Основная задача подходов agile состоит в ускоренной и частой поставке выходов проекта. Однако принимающая организация может оказаться не готовой в полной мере инкорпорировать эти выходы в ускоренном темпе. Ускоренная поставка проверяет способность организации принять ее. Успешного определения и поставки свойств проекта недостаточно. Если организация оказывает сопротивление выходу проекта, то происходит задержка предусмотренной окупаемости инвестиций. В среде agile принятие и согласование заказчиком выходов проекта становится даже еще более важным обстоятельством.

◆ **Изменения, связанные с подходами agile.** Организации, которые только приступают к использованию подходов agile, также сталкиваются с высокой степенью изменений. Более высокие уровни сотрудничества могут требовать более частой передачи работ между командами, подразделениями и поставщиками. Декомпозиция работы на итеративные прототипы связана с доработками, которые могут восприниматься отрицательно. Лидерам следует рассмотреть методы управления изменениями, чтобы найти способы устранения барьеров для перехода к подходам agile.

6.1.2 Готовность к изменениям

Организациям, только начинающим применять подходы agile, следует понимать относительную совместимость этих методов с их текущими подходами. Некоторые

организации будут иметь характеристики, которые облегчат поддержку принципов agile при сотрудничестве между подразделениями, непрерывном обучении и поступательном развитии внутренних процессов. Примеры таких способствующих изменениям характеристик включают в себя следующие:

- ◆ готовность высшего руководства к осуществлению изменений;
- ◆ готовность организации изменить то, как она относится к сотрудникам, рассматривает и оценивает их;
- ◆ функции централизации и децентрализации управления проектом, программой и портфелем;
- ◆ фокус на краткосрочном бюджетировании и метриках в противоположность долгосрочным целям;
- ◆ зрелость и возможности управления талантами.

И наоборот, существуют другие организационные и институциональные характеристики, которые могут стать препятствием на пути к осуществлению изменений, связанных с организационной гибкостью. Примеры таких характеристик включают в себя:

- ◆ Работа декомпозируется по изолированным подразделениям организации, что создает зависимости, препятствующие ускоренной поставке, вместо формирования кроссфункциональных команд с руководством из центров компетенции.
- ◆ Стратегии закупок определяются на основе краткосрочных стратегий ценообразования, а не на основе долгосрочных компетенций.
- ◆ Лидеры получают вознаграждения за локальную эффективность, а не за поставку на всем протяжении проекта или оптимизацию в целом в рамках организации.
- ◆ Сотрудники являются узкими специалистами с ограниченным набором инструментов или поощрений для разностороннего развития своих навыков, вместо формирования «Т-образных» специалистов.
- ◆ Децентрализованные портфели вовлекают сотрудников в слишком большое число проектов одновременно, вместо создания им условий для концентрации усилий на одном проекте в определенный период времени.

Мера готовности организации к пересмотру и изменению этих практик определяет, насколько быстро и результативно подходы agile могут быть приняты в организации. Однако, в ответ на указанные препятствия для гибкости, лидеры проекта могут попробовать применить разные подходы для ускорения культурной совместимости в целях:

- ◆ наглядного и активного спонсорства высших руководителей;
- ◆ внедрения практик управления изменениями, включающих в себя коммуникации и коучинг;
- ◆ неуклонного расширения внедрения практик agile от проекта к проекту;
- ◆ инкрементного введения практик agile в работу команды;
- ◆ осуществления руководства на основе личного примера использования методов и практик agile, где это возможно.

6.2 Культура организации

Культура организации является ее ДНК, то есть сутью ее идентичности. Культура неизбежно оказывает влияние на использование подходов agile. Культура организации представляет собой спектр от крайне предиктивных планов до бережливых стартапов, в которых все является экспериментальным. Хотя подходы agile прекрасно подходят для культуры бережливых стартапов, высокопредиктивная организация может поощрять

вероятностные измерения, небольшие эксперименты и обучение с целью продвижения к гибкости.

«Культура есть стратегия на завтрак», – сказал Питер Друкер (Peter Drucker).

Это утверждение подчеркивает важность приверженности и страсти людей к своему делу. Не имеет значения, какую стратегию или план вы со своей командой осуществляете, ее успех определяется людьми, реализующими план. Если люди, от которых зависит осуществление стратегии, не относятся к изменениям с должным энтузиазмом или, хуже того, вообще не переживают за свою работу и организацию, то вероятность осуществления изменения будет очень невысокой.

6.2.1 Создание безопасной среды

Культуру организации изменить трудно, но наиболее важная культурная норма в организации, желающей испытать новую технологию или метод, состоит в обеспечении безопасной рабочей среды.

Только в атмосфере безопасности труда, добросовестности и прозрачности члены команды и лидеры действительно имеют возможность реально размышлять над своими успехами, чтобы добиться продвижения своих проектов или применения уроков, извлеченных из неудачных проектов так, чтобы не совершать те же ошибки.

6.2.2 Оценка культуры

В каждом проекте возникают сложности из-за конфликтующих стремлений. Как команда может быстро двигаться вперед без ухудшения качества? Как команда может сохранить гибкость, но при этом выполнить задание строго в установленный срок? И, самое главное, как команда удовлетворяет и соблюдает требования заказчика?

Лидеры проектов могут чувствовать, что их работа заключается в удовлетворении всех ожиданий всех заинтересованных сторон, однако, когда им приходится делать выбор, зачастую существует приоритет, зависящий от культуры и требований деловой среды организации. Например, в проекте в области мобильной связи большее значение имеют соображения скорости, в то время как в государственной программе больше внимания может уделяться соображениям широкой употребимости и стабильности.

Чтобы ориентироваться среди этих разноплановых факторов влияния, лидерам проектов не следует жалеть времени на оценку того, на что чаще всего делается упор в конкретной организации. На рис. 6–2 представлено, как может выглядеть такая оценка. На этом примере лидер проекта инициирует обсуждение с заинтересованными сторонами, членами команды и вышестоящим руководством вопроса о приоритетах организации. Затем положения этих приоритетов отмечаются на скользящей шкале в промежутке между крайними точками. После этого результаты используются для поиска методов agile, которые лучше всего подходят для этих приоритетов.

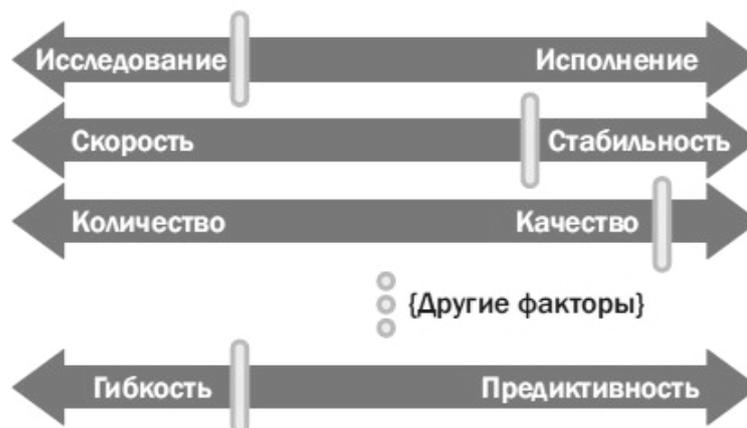


Рис. 6–2. Пример оценки культуры организации

Существует несколько моделей для оценки таких факторов, однако, не так важно, какая именно модель или какой метод используются. Лидерам проектов гораздо важнее приложить усилия для понимания сил, которые формируют контекст этих факторов. Понимание требований организации и отрасли, которые организации необходимо выполнить, позволят выбрать правильные обсуждения, надлежащие компромиссы и, в особенности, подходящие методы.

6.3 Закупки и договоры

Как было сказано выше в настоящем Практическом руководстве, одной из ценностей Agile-манифеста является следующая: «сотрудничество с заказчиком важнее согласования условий договора». Многие проекты потерпели неудачу из-за разлада в отношениях между заказчиком и поставщиком. Уровень риска по проекту выше, когда участвующие в договоре стороны занимают позицию с точки зрения «победитель против проигравшего». Сутью подхода сотрудничества является стремление установить отношения «совместный риск – общая выгода», когда выигрывают все участвующие стороны. В числе методов заключения договоров, которые могут формально закрепить данные отношения, можно назвать следующие:

♦ **Многоуровневая структура.** Вместо формального оформления договорных отношений в полном объеме в одном единственном документе стороны проекта могут обеспечить большую степень гибкости, закрепив различные аспекты отношений в отдельных документах. В рамочном соглашении могут быть закреплены, главным образом, не подлежащие пересмотру положения (например, гарантийные обязательства, порядок урегулирования споров). В то же время все стороны переносят другие положения, связанные с возможными изменениями (например, тарифы на услуги, описания продукта) в приложение с описанием услуг. В договоре рамочное соглашение об оказании услуг может ссылаться на эти положения. И наконец, сильно изменяющиеся положения, такие как содержание, расписание, бюджет, могут быть официально оформлены в упрощенном описании работ. Выделение более изменчивых элементов договора в единый документ упрощает процедуру внесения изменений и повышает уровень гибкости.

♦ **Особое внимание поставляемой ценности.** Часто отношения с поставщиками определяются фиксированными контрольными событиями или «воротами фазы», в фокусе которых находятся промежуточные артефакты, а не полный поставляемый результат инкрементной бизнес-ценности. Во многих случаях эти способы контроля ограничивают использование обратной связи для совершенствования продукта. Вместо этого контрольные события и условия оплаты могут определяться на основе ценностно-ориентированных поставляемых результатов с целью повышения гибкости проекта.

♦ **Инкременты с фиксированной ценой.** Вместо того, чтобы «втискивать» все содержание проекта и бюджет в рамки одного соглашения, в проекте содержание может быть разделено на поставляемые результаты небольшого размера с фиксированной ценой, такие как пользовательские истории. Заказчику это обеспечивает больше контроля за расходованием денег. Что касается поставщика, это ограничивает финансовый риск от принятия чрезмерных обязательств в отношении отдельного свойства или поставляемого результата.

ПОЛЕЗНЫЙ СОВЕТ

Культура против структуры

Некоторые люди настаивают на том, что до начала любых культурных перемен необходимо сначала установить новые организационные структуры. Другие утверждают обратное: такие новые организационные структуры являются лишь не имеющими существенного значения изменениями, пока в культуре коллектива не произойдет содержательный сдвиг. В реальной жизни первое не может существовать без второго.

Лидерам проектов, желающим достичь гибкости, следует учитывать текущее и будущее положение с обоими этими аспектами в их организации.

♦ **«Время и материалы» без превышения.** Заказчики несут нежелательные риски в связи с традиционным подходом «время и материалы». Одним из возможных вариантов решения этой проблемы является ограничение общего бюджета фиксированной суммой. Это позволит заказчику включать в проект новые идеи и инновации, которые изначально планом не предусматривались. Когда заказчики пожелают включить новые идеи, им придется сохранить определенный объем, заменяя изначально предусмотренные работы новыми. За работой нужно внимательно следить по мере приближения выделенных на нее часов к установленному ограничению. Кроме этого, максимальный бюджет может предусматривать дополнительное время на возможные потери.

♦ **«Время и материалы» с дифференциацией.** Другим возможным вариантом является подход разделенного финансового риска. В agile критерии качества входят в понятие того, что считается «выполненным». В силу этого поставщик может получить вознаграждение в виде повышенного тарифа оплаты рабочего времени в случае, когда поставка произведена раньше предусмотренного договором крайнего срока. И наоборот, тариф оплаты рабочего времени поставщика может быть снижен за просрочку поставки.

♦ **Опция досрочной отмены.** Когда работающий в agile поставщик поставил достаточную ценность при выполнении содержания проекта лишь наполовину, заказчик не должен нести обязательств по оплате остающейся половины, если она ему больше не нужна. Вместо этого договор может предусматривать для заказчика возможность выкупить остающуюся часть договора с оплатой неустойки за отмену. Заказчик ограничивает бюджетные риски, а поставщик получает дополнительный доход за услуги, которые больше не требуются.

♦ **Опция динамического содержания.** В договорах с фиксированным бюджетом поставщик может предложить заказчику вариант изменения содержания проекта в определенные моменты в ходе исполнения проекта. Заказчик может изменять свойства при условии соответствия ресурсным возможностям. Кроме того, заказчик может использовать возможности инновации при одновременном ограничении риска превышения обязательств у поставщика.

♦ **Дополнение команды.** Пожалуй, создающим наибольшие возможности для сотрудничества подходом заключения договоров является включение услуг поставщика непосредственно в структуру организации заказчика. Финансирование команд, а не определенного содержания, сохраняет заказчику стратегическую свободу выбора того, какие работы следует произвести в действительности.

♦ **Предпочтение поставщиков с полным комплексом услуг.** В целях диверсификации рисков заказчики могут придерживаться стратегии работы с несколькими поставщиками. Однако появляется искушение передавать работу по договорам так, чтобы каждый поставщик выполнял только какую-то одну задачу, в результате чего возникает сложная схема зависимостей, еще до того, как появятся какие-то полезные услуги или продукты. Вместо этого упор надо делать на взаимодействие, которое обеспечивает поставку полной ценности (как в идее с завершенными независимыми наборами свойств).

Возможно заключение договоров на основе agile. Agile создается на основе синергии сотрудничества и доверия. Поставщик может помочь этому, поставляя ценность рано и часто. Заказчик, со своей стороны, может помочь, своевременно предоставляя обратную связь.

6.4 Бизнес-практики

Готовность и способность создавать по мере необходимости новые компетенции внутри организации являются признаком гибкости организации. Это не обязательно должны быть какие-то судьбоносные изменения, и они могут не слишком нарушать работу в организации, центром внимания которой является сама гибкость и ее результаты. Прозрачность и открытое сотрудничество, безусловно, являются ключевыми факторами.

В тех случаях, когда кроссфункциональные команды поставляют ценность, команды и отдельные специалисты могут сталкиваться с проблемами необходимости поддержки различных подразделений в организации.

По мере поставки командой ценности на регулярной основе у финансовых подразделений может быть возможность по-разному извлекать выгоду из продукта. Если у команды есть договоры с другими организациями, отвечающим за закупки, отделам может понадобиться внести в эти договоры изменения, чтобы помочь другим организациям поставлять ценность часто и синхронизировать поставки с командой.

После того как команды начинают работать согласованно на основе сотрудничества, они предъявляют требования к политикам внутреннего управления. Кадровые службы могут заметить, что индивидуальные поощрения становятся менее целесообразными, а руководители могут сталкиваться с трудностями при оценке исполнения самоорганизующихся работников. В каждом случае это возможности для анализа того, в какой мере существующие практики поддерживают способы работы agile.

По мере продвижения организаций к большей гибкости будут существовать очевидные потребности в изменении дополнительными бизнес-подразделениями способов, с помощью которых они взаимодействуют друг с другом и исполняют свои обязанности. Изменения, которые принесли выгоды в других областях деятельности организации, должны теперь использоваться в полной мере, чтобы реализовать результативность работы всей организации в целом.

6.5 Координация работы и зависимостей нескольких команд (масштабирование)

Во многих проектах возникают зависимости, даже тогда, когда управление ими не осуществляется в рамках программы. По этой причине необходимо достичь понимания того, как agile работает в контексте существующего управления программами и портфелями.

6.5.1 Фреймворки

Указания по наиболее распространенным методам agile, таким как скрам и экстремальное программирование, сосредоточены на работе единственной, небольшой, обычно с совместным расположением кроссфункциональной команды. Хотя это очень полезно для работ, когда требуется только одна команда, этих указаний может оказаться недостаточно для проектов, требующих совместной работы нескольких команд agile в рамках одной программы или портфеля.

Ряд фреймворков (таких как масштабированный agile-фреймворк, крупномасштабный скрам и упорядоченный agile), а также подходы (например, скрам скрамов) появились для использования именно в таких обстоятельствах. Более подробную информацию по этим вопросам можно найти в приложении А3.

6.5.2 Соображения

Существует несколько способов масштабирования работ. Команде может понадобиться масштабировать работу нескольких проектов agile в одной программе agile. Как вариант, организация может разработать структуру, которая поддерживает подходы agile в рамках портфеля в целом.

Например, целесообразно начать с малого и как можно быстрее узнать, что хорошо работает в контексте данной организации. Команды могут добиться успешных результатов даже тогда, когда еще не все полностью преобразовано в подход agile.

Независимо от используемого подхода, критическим фактором успеха является здоровая команда agile. Если использование подхода agile для одной команды не приносит успеха, не пытайтесь распространять его дальше, а займитесь поиском мер для устранения препятствий, которые мешают командам вести работы на принципах agile.

Цель крупномасштабных проектов agile состоит в координации работы разных команд для поставки ценности заказчиком. Для этого существует несколько способов. Команды могут использовать формальный фреймворк или применить мышление agile, чтобы привести в соответствие существующие практики управления программами.

6.6 Agile и офис управления проектами (ОУП)

ОУП существует для курирования создания бизнес-ценности в масштабах всей организации. Он может решать эту задачу путем оказания помощи командам проектов в достижении стоящих перед ними целей. В некоторых случаях ОУП обучает команды (или организует обучение), а также осуществляет поддержку проектов. В некоторых случаях ОУП консультирует руководство об относительной бизнес-ценности конкретного проекта или группы проектов.

Поскольку agile вызывает с течением времени культурные изменения, организации может потребоваться осуществить необходимые изменения, включая изменения в ОУП. Например, руководители принимают решения о том, какие проекты и когда финансировать, а команды решают, что им нужно для обучения или консультаций.

6.6.1 ОУП в agile является ценностно-ориентированным

Каждый проект должен поставить нужную ценность нужной аудитории в нужное время. Задача ОУП состоит в создании необходимых условий для достижения этой цели. Основанный на agile подход ОУП базируется на образе мышления сотрудничества с заказчиком и присутствует во всех программах ОУП. Во многих случаях это означает, что ОУП работает так, как если бы он был консультантом, адаптирующим свою работу для удовлетворения особых нужд, заявленных данным проектом. Некоторым проектам могут понадобиться инструменты и шаблоны, а другие могут получить выгоды от стратегического коучинга. ОУП должен стремиться к поставке всего необходимого и следить за нуждами заказчиков, чтобы иметь уверенность в том, что он знает их нужды и способен адаптировать свою работу для их удовлетворения. Главное место в таком интрапренерском подходе занимают действия ОУП, которые считаются наиболее ценными для поддерживаемых им проектов.

6.6.2 ОУП в agile является ориентированным на вовлечение

В целях ускорения прогресса в соответствии с основанной на ценности концепции у ОУП может появиться желание сделать обязательными определенные решения или подходы, например, обязать всех делать что-то одинаково, чтобы добиваться быстрых побед. Однако более продуманная перспектива включает в себя стремление к вовлечению сотрудников. Это достигается за счет привлечения только тех, кто заинтересован в использовании услуг ОУП. Чем шире участие в практиках ОУП, тем легче эти практики становятся привычными. Если ОУП поставляет своим клиентам ценность, то вероятность, что клиенты будут обращаться за его услугами и принимать его практики, будет выше.

6.6.3 ОУП в agile является многопрофильным

Чтобы обеспечить поддержку особых потребностей конкретного проекта, ОУП необходимо обладать компетенцией в нескольких областях, помимо собственно управления проектом, так как разные проекты требуют особых возможностей. Например, в одном проекте может быть необходимо разработать организационную структуру для решения кадровых проблем, а в другом могут потребоваться методы управления организационными

изменениями для привлечения заинтересованных сторон или уникальные бизнес-модели для поддержки целей заказчика.

В некоторых организациях офисы управления проектами трансформируются в центры компетенций agile, которые оказывают, среди прочих, следующие услуги:

♦ **Разработка и внедрение стандартов.** Готовят шаблоны для пользовательских историй, тестовые задания, диаграммы суммарного потока и тому подобное. Предлагают инструменты agile и обучают вспомогательные группы по тематике итеративной разработки.

♦ **Развитие персонала с помощью обучения и наставничества.** Осуществляют координацию проведения курсов обучения по agile, коучинг и наставничество с целью помочь людям перейти к образу мышления agile и развить их навыки. Поощряют и обеспечивают участие людей в локальных мероприятиях agile.

♦ **Управление несколькими проектами.** Координируют работы нескольких команд agile через коммуникацию между проектами. Рассматривают обмен информацией по таким вопросам, как ход исполнения проектов, проблемы и результаты ретроспективы, а также эксперименты по совершенствованию. Помогают в организации крупных релизов для заказчиков на уровне программы и направлений инвестирования на уровне портфеля, используя соответствующий фреймворк.

♦ **Создание условий для организационного обучения.** Собирают профили скоростей проектов, а также получают, хранят и индексируют результаты ретроспективы.

♦ **Управление заинтересованными сторонами.** Обеспечивают владельцу продукта обучение и консультирование по вопросам приемочного тестирования, оценки и осуществления обратной связи о системах. Отстаивают важность экспертов по предметным областям (subject matter experts, SMEs) для проектов.

♦ **Осуществление набора, отбора и оценки лидеров команд.** Разрабатывают указания по проведению собеседований с agile-практиками.

♦ **Исполнение специальных задач для проектов.** Обучают и предоставляют фасилитаторов ретроспективы, заключают соглашения со специалистами по устранению проблем в проектах agile, а также предоставляют наставников и коучей.

6.7 Организационная структура

Структура организации оказывает сильное влияние на ее способность учитывать новую информацию или меняющиеся потребности рынка. Ниже приводятся ключевые характеристики:

♦ **География.** Географически распределенные и рассредоточенные проектные организации могут столкнуться с несколькими вызовами, создающими помехи в их работе по любому проекту. Лидеры проектов и региональные менеджеры могут иметь альтернативные или даже конкурирующие цели. Кроме того, культурные различия, языковые барьеры и снижение уровня наглядности могут снизить производительность. К счастью, использование подходов agile может способствовать большему сотрудничеству и доверию в сравнении с тем, что было бы без их использования. Лидеры проектов в таких условиях должны поощрять диалог на уровне команды и высшего руководства с целью адаптации методов для данного контекста и управления ожиданиями в отношении трудозатрат, необходимых для этого.

♦ **Функциональные структуры.** Диапазон структур некоторых организаций различается от в высшей степени проектно-ориентированных до матричных и до в высшей степени функционально-ориентированных. Проекты в высшей степени функционально-ориентированных структурах, могут испытывать общее противодействие сотрудничеству между разными подразделениями организации.

♦ **Размер поставляемого результата проекта.** Уменьшение размера поставляемого результата проекта стимулирует более частую передачу работы между подразделениями и, таким образом, более частые итерации и ускорение потока ценности через организацию в целом.

♦ **Распределение людей по проектам.** Другой подход: выделить все рабочее время одного человека от каждого подразделения на участие в проекте с наивысшим приоритетом в течение определенного срока.

♦ **Организации с высокой степенью закупочной деятельности.** Некоторые организации решают осуществлять проекты главным образом через поставщиков. Хотя цели проекта могут быть ясными, за свою финансовую жизнеспособность поставщики отвечают сами. Кроме того, после исполнения поставщиками своих обязательств они забирают с собой связанные с проектом знания при выходе из проекта. Это ограничивает внутренние компетенции в организации, необходимые для устойчивой гибкости и скорости. Методы agile, такие как ретроспективы и деятельность по результатам проекта по областям возможного совершенствования, когда поставщик все еще связан с организацией, могут помочь смягчить последствия утраты знаний о продукте.

6.8 Поступательное развитие организации

При принятии мер по отдельной проблемной области или реализации нового гибридного подхода или подхода agile рекомендуется производить работы инкрементно. Обычная практика – относиться к процессу изменений как к проекту agile с его собственным бэклогом изменений, который может быть принят и приоритизирован командой на основе предполагаемой ценности или других соображений. К каждому из изменений можно относиться как к эксперименту, который проходит недолгое тестирование для установления приемлемости в состоянии «как есть» или необходимости дополнительной доработки/рассмотрения.

Для отслеживания прогресса используйте доски «канбан», показывающие новые и уже используемые подходы как «выполненные», проходящие испытания как «незавершенные» и ожидающие введения как «намеченные для исполнения». На рис. 6–3 представлен исходный вид доски с ранжированным бэклогом. На рис. 6–4 приведен пример того, как может выглядеть доска по мере выполнения работ.

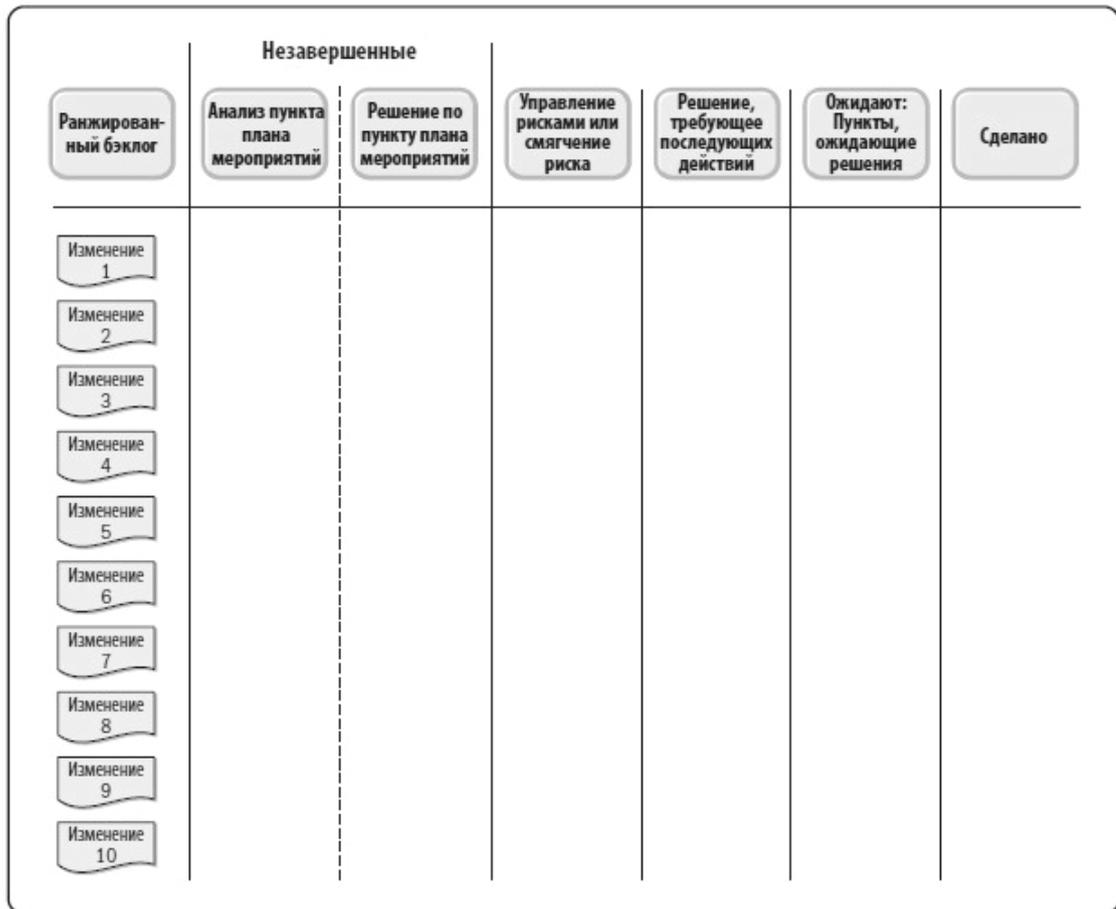


Рис. 6–3. Исходный вид ранжированного бэклога для изменений

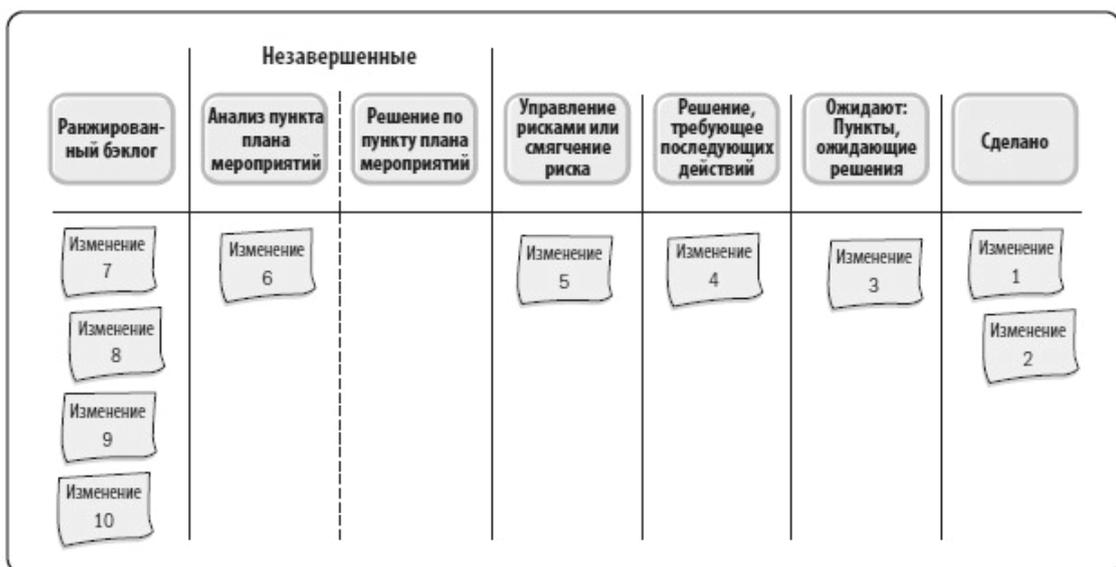


Рис. 6–4. Использование бэклогов и досок «канбан» для организации и отслеживания работ по изменениям

Использование этих инструментов для организации и управления осуществлением изменений обеспечивает наглядность прогресса, а также моделирует подходы, находящиеся в процессе реализации. Развертывание изменений прозрачным и привлекательным образом повышает вероятность успеха при их реализации.

7. Призыв к действию

Принятие agile и его подходов для управления проектами значительно расширилось за период после первой публикации Agile-манифеста в 2001 г. Принятие и желание вести работу на основе образа мышления agile больше не ограничено организациями определенного размера или теми из них, которые специализируются на информационных технологиях. Этот образ мышления получил всеобщее применение, а подходы успешно используются во многих средах.

Сегодня стремление «быть agile» получило более широкое распространение, чем когда-либо. Споры о наилучшем пути к гибкости продолжают поддерживать поступательное развитие дискуссии и инноваций. Неизменной остается одна истина: инспекции, адаптация и прозрачность имеют решающее значение для успешной поставки ценности.

Возможно, данное практическое руководство не содержит в полном объеме все то, что вы ожидали в нем найти. Наша основная команда понимает, что вы можете быть не согласны с какими-то элементами или подходами, которые мы решили представить в нем, причем самым решительным образом. Мы настоятельно призываем вас продолжить беседу и внести свой вклад в улучшение следующей итерации разработки данного Практического руководства. Отправляйтесь в путь: изучайте, экспериментируйте, получайте обратную связь и снова экспериментируйте. А затем помогите нам в ретроспективе: предложите свое мнение об этих рекомендациях и внесите свой вклад в будущие издания данного Практического руководства. В конечном счете, инспекция без адаптации – бесполезная работа.

И наконец, мы призываем вас принять участие в работе с более широкими сообществами специалистов по управлению проектами и agile, чтобы углубить разговоры по этой тематике. Ищите представителей из PMI и Agile Alliance® на конференциях и собраниях и привлекайте их к дискуссии. Используйте социальные сети и излагайте в блогах свои соображения.

Вы можете обеспечить обратную связь и вступить в разговор о содержании этого Практического руководства в блоге под названием «Agile на практике» («Agile in Practice») по ссылке <https://www.projectmanagement.com/blogs/347350/Agile-in-Practice>.

Приложение А1. Картирование руководства PMBOK®

В таблице А1-1 показано соотношение групп процессов управления проектом с областями знаний, которые определены в *Руководстве PMBOK®* – Шестое издание.

В настоящем приложении описано, как при гибридном подходе и подходе agile рассматриваются свойства, описанные в областях знаний *Руководства PMBOK®* (см. таблицу А1-2). В нем, наряду с некоторыми указаниями, позволяющими повысить вероятность успеха, разъясняется, что остается неизменным, а что может выполняться иначе.

Таблица А1-1. Картирование групп процессов управления проектом и областей знаний

Области знаний	Группы процессов управления проектом				
	Группа процессов инициации	Группа процессов планирования	Группа процессов и исполнения	Группа процессов мониторинга и контроля	Группа процессов закрытия
4. Управление интеграцией проекта	4.1 Разработка устава проекта	4.2 Разработка плана управления проектом	4.3 Руководство и управление работами проекта 4.4 Управление знаниями проекта	4.5 Мониторинг и контроль работ проекта 4.6 Интегрированный контроль изменений	4.7 Закрытие проекта или фазы
5. Управление содержанием проекта		5.1 Планирование управления содержанием 5.2 Сбор требований 5.3 Определение содержания 5.4 Создание ИСР		5.5 Подтверждение содержания 5.6 Контроль содержания	
6. Управление расписанием проекта		6.1 Планирование управления расписанием 6.2 Определение операций 6.3 Определение последовательности операций 6.4 Оценка длительности операций 6.5 Разработка расписания		6.6 Контроль расписания	
7. Управление стоимостью проекта		7.1 Планирование управления стоимостью 7.2 Оценка стоимости 7.3 Определение бюджета		7.4 Контроль стоимости	
8. Управление качеством проекта		8.1 Планирование управления качеством	8.2 Управление качеством	8.3 Контроль качества	
9. Управление ресурсами проекта		9.1 Планирование управления ресурсами 9.2 Оценка ресурсов операций	9.3 Приобретение ресурсов 9.4 Развитие команды проекта 9.5 Управление командой проекта	9.6 Контроль ресурсов	
10. Управление коммуникациями проекта		10.1 Планирование управления коммуникациями	10.2 Управление коммуникациями	10.3 Мониторинг коммуникаций	
11. Управление рисками проекта		11.1 Планирование управления рисками 11.2 Идентификация рисков 11.3 Качественный анализ рисков 11.4 Количественный анализ рисков 11.5 Планирование реагирования на риски	11.6 Осуществление реагирования на риски	11.7 Мониторинг рисков	
12. Управление закупками проекта		12.1 Планирование управления закупками	12.2 Проведение закупок	12.3 Контроль закупок	
13. Управление заинтересованными сторонами проекта	13.1 Идентификация заинтересованных сторон	13.2 Планирование вовлечения заинтересованных сторон	13.3 Управление вовлечением заинтересованных сторон	13.4 Мониторинг вовлечения заинтересованных сторон	

Таблица A1-2. Применение подходов agile в областях знаний *Руководства PMBOK®*

Руководство РМВОК® Область знаний	Применение в процессе работы agile
<p>Раздел 4 Управление интеграцией проекта</p>	<p>Итеративный и гибкий подходы способствуют вовлечению членов команды как локальных экспертов в управление интеграцией. Порядок интеграции планов и компонентов определяют члены команды.</p> <p>Ожидания руководителя проекта, как отмечено в разделах <i>Ключевые концепции управления интеграцией Руководства РМВОК®</i>, в адаптивной среде не изменяются, но контроль над подробным планированием продукта и его поставкой делегируется команде. Руководитель проекта сосредотачивает основное внимание на создании общей среды принятия решений, а также обеспечивает способность команды реагировать на изменения. Данное сотрудничество может быть еще больше усилено, если члены команды обладают широкой базой навыков, а не узкой специализацией.</p>
<p>Раздел 5 Управление содержанием проекта</p>	<p>В проектах с постоянно развивающимися требованиями, с высоким уровнем риска или большой степенью неопределенности во многих случаях на начальной стадии проекта его содержание остается неясным или развивается по мере осуществления. На ранней стадии проекта при использовании гибких методов на определение и согласование содержания целенаправленно выделяется меньше времени, чем на организацию процесса непрерывного раскрытия и уточнения требований. Во многих средах с вновь возникающими требованиями становится понятно, что между реальными бизнес-требованиями и бизнес-требованиями, которые были изначально заявлены, существует определенный разрыв. В этой связи при использовании гибких методов с целью уточнения требований целенаправленно создаются и анализируются прототипы и версии. В результате определение и уточнение содержания происходит на всем протяжении проекта. При применении гибких подходов требования формируют бэклог.</p>

Таблица А1-2. Применение подходов agile в областях знаний Руководства РМВОК®(прод.)

Руководство РМВОК® Область знаний	Применение в процессе работы agile
<p>Раздел 6 Управление расписанием проекта</p>	<p>В адаптивных подходах по мере необходимости используются короткие циклы для выполнения работ, рассмотрения результатов и адаптации. Данные циклы обеспечивают быстрое получение откликов об этих подходах и соответствие требованиям поставляемых результатов, и в целом проявляются как итеративное и основанное на спросе расписание по требованию, как описано в разделе <i>Ключевые тенденции и вновь появляющиеся практики главы Управление расписанием проекта в Руководстве РМВОК®</i>.</p> <p>В больших организациях небольшие проекты могут сочетаться с масштабными инициативами, требующими разработки долгосрочных дорожных карт для управления развитием этих программ с использованием масштабирующих факторов (например, численный состав команды, географическое распределение, соблюдение регуляторных требований, организационная и техническая сложность). При решении задачи полного жизненного цикла поставки крупных, охватывающих все предприятие систем может потребоваться внедрение ряда методов, использующих предиктивный подход, адаптивный подход или гибрид этих двух подходов. У организации может возникнуть необходимость комбинировать практики из нескольких базовых методов или внедрять метод, в котором это уже реализовано, а также принять несколько принципов и практик более традиционной методике.</p> <p>Роль руководителя проекта в связи с управлением проектами, в которых используется предиктивный жизненный цикл разработки, или управлением проектами в адаптивных средах не изменится. Однако для достижения успеха в использовании адаптивных подходов руководителю проекта необходимо хорошо знать инструменты и методы чтобы понимать, как следует применять их результативно.</p>
<p>Раздел 7 Управление стоимостью проекта</p>	<p>Проекты, характеризуемые большой степенью неопределенности, или те из них, где содержание не определено полностью, могут не получить преимуществ от детальных расчетов стоимости по причине частых изменений. Вместо этого можно использовать методы облегченной оценки для получения быстрого высокоуровневого прогноза трудовых затрат для проекта, который в последующем можно легко корректировать с учетом происходящих изменений. Подробные оценки откладываются на периоды краткосрочного планирования по принципу «точно в срок».</p> <p>В случаях, когда для проектов с высокой степенью изменчивости также устанавливается строгий бюджет, их содержание и расписание требуют более частой корректировки, чтобы не выйти за пределы установленных ограничений стоимости.</p>

Таблица А1-2. Применение подходов agile в областях знаний *Руководства РМВОК®* (прод.)

Руководство PMBOK® Область знаний	Применение в процессе работы agile
<p>Раздел 8 Управление качеством проекта</p>	<p>В целях правильного осуществления изменений гибкие методы предполагают неоднократное повторение шагов контроля и анализа качества, предусмотренных в процедурах проекта на всем протяжении проекта, а не по мере приближения его завершения.</p> <p>Проверка результативности процессов контроля качества проводится с помощью периодического ретроспективного анализа на регулярной основе. Данные проверки призваны выявить первопричину проблем и предложить новые подходы к совершенствованию качества для практических испытаний. Последующий ретроспективный анализ оценивает процессы практических испытаний, чтобы определить, работают ли они и следует ли применять их в дальнейшем в неизменном виде, с изменениями или отказаться от них.</p> <p>С целью создания благоприятных условий для периодических, инкрементных поставок при применении гибких методов основное внимание обращают на небольшие пакеты работ, включающих максимально возможное количество элементов поставляемых результатов проекта. Системы небольших пакетов направлены на выявление несоответствий и проблем качества на ранних стадиях жизненного цикла проекта, когда общая стоимость изменений ниже.</p>
<p>Раздел 9 Управление ресурсами проекта</p>	<p>Проекты с высокой степенью вариативности получают преимущество от использования организационных структур команды, которые позволяют максимально сосредоточить усилия и обеспечить совместную работу, например таких, как самоорганизующиеся команды включающие в себя специалистов широкого профиля.</p> <p>Коллективная работа призвана существенно повысить производительность и создать благоприятные условия для инновационного подхода к разрешению проблем. Основанные на принципах коллективной работы команды могут, помимо прочих преимуществ, создать условия для ускорения интеграции определенных рабочих операций, улучшения коммуникации, расширения обмена знаниями и обеспечения гибкости в распределении рабочих заданий.</p> <p>Несмотря на то, что выгоды коллективной работы также свойственны другим средам проекта, основанные на принципах коллективной работы команды часто являются важнейшим условием успеха проектов, характеризующихся высокой степенью вариативности и быстро текущими изменениями, так как для постановки задач и принятия решений в централизованном порядке в проектах такого типа имеется меньше времени.</p> <p>Планирование материальных и человеческих ресурсов значительно менее предсказуемо в проектах с высокой степенью вариативности. В подобных средах соглашения для быстрой поставки, а также бережливые методы являются критичными для контроля стоимости и соблюдения расписания.</p>

Таблица А1-2. Применение подходов agile в областях знаний Руководства PMBOK® (прод.)

Руководство РМВОК® Область знаний	Применение в процессе работы agile
<p>Раздел 10 Управление коммуникациями проекта</p>	<p>Среды проекта, подверженные различным факторам неопределенности и изменчивости, имеют внутренне присущую им потребность чаще и более оперативно информировать о непрерывно развивающихся и формирующихся деталях. Это мотивирует оптимизацию доступа к информации члена команды, установление частых точек проверки команды и совместное расположение членов команды, насколько это возможно.</p> <p>В дополнение к этому, открытая публикация артефактов проекта, а также проведение регулярных обзоров заинтересованными сторонами призваны содействовать коммуникациям с руководством и заинтересованными сторонами.</p>
<p>Раздел 11 Управление рисками проекта</p>	<p>Среды с высокой вариативностью по определению отличаются более высоким уровнем неопределенности и риска. С учетом этого обстоятельства, в управлении проектами с использованием адаптивных подходов применяются метод частого рассмотрения инкрементных продуктов работы, а также кросс-функциональные команды проекта для ускорения процесса обмена знаниями и обеспечения понимания рисков и управления ими. Риски рассматриваются всякий раз при выборе содержания каждой итерации; анализ, идентификация рисков и управление ими осуществляются также в ходе каждой итерации.</p> <p>Кроме этого, учет требований ведется на основе непрерывно уточняемого документа, который обновляется регулярно, а приоритизация работ может изменяться по мере прогресса проекта на основе более полного понимания текущей подверженности рискам.</p>

Таблица А1-2. Применение подходов agile в областях знаний *Руководства РМВОК®*(прод.)

Руководство РМВОК® Область знаний	Применение в процессе работы agile
<p>Раздел 12 Управление закупками проекта</p>	<p>В гибких средах некоторые поставщики могут использоваться для расширения состава команды. Отношения в рамках совместного ведения работы могут привести к использованию модели закупки с совместным риском, когда покупатель и продавец разделяют риски и выгоды, связанные с проектом.</p> <p>Крупные проекты могут использовать адаптивный подход для некоторых поставляемых результатов и более стабильный подход для других частей. В таких случаях может использоваться генеральное соглашение, например генеральный договор об оказании услуг (Master Services Agreement, MSA), для определения объема работ в целом, а адаптируемые работы выносятся в приложение или дополнение к договору. Это позволяет вносить изменения в адаптируемое содержание без последствий для договора в целом.</p>
<p>Раздел 13 Управление заинтересованными сторонами проекта</p>	<p>Проекты с большим объемом изменений требуют активного вовлечения и участия заинтересованных сторон проекта. Чтобы облегчить своевременную, продуктивную дискуссию и принятие решений адаптивные команды вовлекают в работу заинтересованные стороны напрямую, не создавая ненужную бюрократию и лишние уровни управления. Часто обмен информацией между клиентом, пользователем и разработчиком происходит в рамках динамического процесса совместной работы, который позволяет добиться большей степени вовлеченности и удовлетворенности заинтересованных сторон. Регулярное взаимодействие с сообществом заинтересованных сторон на всем протяжении проекта снижает уровень риска, формирует доверие и способствует внесению изменений на более ранних стадиях цикла проекта, сокращая затраты и повышая вероятность успешного завершения проекта.</p> <p>Применение гибких методов способствует созданию условий доступности информации в целях ускорения обмена ею внутри. Цель приглашения любых заинтересованных сторон на совещания и консультации или публикация артефактов проекта в открытом доступе состоит в выявлении в кратчайшие сроки любых отклонений от норм, зависимостей или других проблем, связанных с изменением проекта.</p>

Приложение А2. Картирование Agile-манифеста

В этом приложении описывается, как элементы Agile-манифеста представлены в «*Agile: практическое руководство*».

Таблица A2-1. Ценности Agile-манифеста, представленные в «*Agile: практическое руководство*»

Ценность	Обзор « <i>Agile: практическое руководство</i> » по разделам и названиям
Люди и взаимодействие важнее процессов и инструментов	4.2 Обслуживающее лидерство усиливает команду 4.3 Состав команды 5.1 Устав проекта и устав команды 5.2.4 Ежедневные летучки 6.2 Культура организации
Работающий продукт важнее исчерпывающей документации	5.2.2 Подготовка бэклога 5.2.3 Уточнение бэклога 5.2.5 Демонстрации/обзоры 5.2.7 Практики исполнения, помогающие командам поставлять ценность
Сотрудничество с заказчиком важнее согласования условий договора	4.3 Состав команды 5.4 Измерения в проектах agile 6.2 Культура организации 6.3 Закупки и договоры 6.7 Организационная структура
Готовность к изменениям важнее следования первоначальному плану	5.2.1 Ретроспективы 5.2.3 Уточнение бэклога 5.2.5 Демонстрации/обзоры

Таблица A2-2. Картирование «*Agile: практическое руководство*» по принципам, вытекающим из Agile-манифеста

Принцип	Обзор «Agile: практическое руководство»
Наивысшим приоритетом является удовлетворение потребностей заказчика за счет ранней и непрерывной поставки ценного продукта.	3.1 Характеристики жизненных циклов проектов 5.2.7 Практики исполнения, помогающие командам поставлять ценность
Изменение требований приветствуется, даже на поздних стадиях разработки. Процессы agile используют изменения в целях создания конкурентных преимуществ для заказчика.	5.2.3 Уточнение бэклога
Работающий программный продукт следует выпускать как можно чаще, с периодичностью от пары недель до пары месяцев, отдавая предпочтение более коротким срокам.	5.2 Общепринятые практики agile
На протяжении всего проекта разработчики и представители бизнеса должны ежедневно работать вместе.	4.2 Обслуживающее лидерство усиливает команду 5.2.2 Подготовка бэклога 5.2.3 Уточнение бэклога
Над проектом должны работать мотивированные люди. Чтобы работа была сделана, создайте условия, обеспечьте поддержку и полностью доверьтесь им.	4.3 Состав команды 5.1 Устав проекта и устав команды 5.2.1 Ретроспективы
Личное общение является наиболее эффективным и результативным способом обмена информацией как с самой командой, так и внутри команды.	4.3.4 Структуры команд 5.2.4 Ежедневные летучки
Работающий продукт — основной показатель хода исполнения.	5.2.7 Практики исполнения, помогающие командам поставлять ценность 5.2.8 Как итерации и инкременты помогают в поставке рабочего продукта
Процессы agile способствуют устойчивой разработке. Спонсоры, разработчики и пользователи должны иметь возможность поддерживать постоянный ритм неопределенно долго.	5.1 Устав проекта и устав команды
Постоянное внимание к техническому совершенству и качеству проектирования повышает гибкость	5.2 Общепринятые практики agile
Простота — искусство максимизации объема работы, которую не нужно выполнять — крайне необходима.	5.2.2 Подготовка бэклога 5.2.3 Уточнение бэклога
Самые лучшие архитектуры, требования и проектные решения рождаются у самоорганизующихся команд.	4.3 Состав команды
Команда должна систематически анализировать возможные способы улучшения результативности и соответственно корректировать стиль своей работы.	5.2.1 Ретроспективы

Приложение А3. Обзор фреймворка agile и бережливого фреймворка

В настоящем приложении дано описание некоторых часто используемых подходов agile. Эти подходы можно использовать в оригинальном виде или в определенных сочетаниях с целью их адаптации так, чтобы обеспечить их максимальное соответствие определенной среде или ситуации. Если необходимость использовать какие-либо из них отсутствует, подход agile можно разработать с самого начала при условии, что он будет соответствовать образу мышления, ценностям и принципам Agile-манифеста. В случаях, когда принципы agile соблюдаются для поставки ценности устойчивыми темпами, а разработанный подход способствует сотрудничеству с заказчиком, применять особый подход не требуется. Ссылку

на источники с дополнительной информацией о каждом подходе можно найти в разделе «Список использованной литературы» настоящего Руководства.

А3.1 КРИТЕРИИ ВЫБОРА ДЛЯ РАССМОТРЕНИЯ В «AGILE: ПРАКТИЧЕСКОЕ РУКОВОДСТВО»

Различных подходов и методов agile слишком много, чтобы их все можно было подробно описать в настоящем Практическом руководстве. На рис. А3-1 представлены примеры подходов agile на основе глубины указаний и ширины жизненных циклов этих подходов. Конкретные выбранные для обсуждения подходы являются широко распространенными примерами, которые:

♦ **Предназначены для комплексного использования.** Некоторые подходы agile направлены на какую-то одну деятельность по проекту, например, оценку или осмысление. Приведенные примеры включают только наиболее комплексные фреймворки agile. Некоторые из них включают больше свойств, чем другие, но все выбранные для рассмотрения подходы предназначены для организации широкого набора деятельности по проектам.

♦ **Формализованы для общего пользования.** Некоторые фреймворки agile по своей природе являются частными и предназначены для использования в особых случаях в одной единственной организации или в рамках одного единственного контекста. Фреймворки, описанные в разделах с А3.2 по А3.14, предназначены для обычного использования в разнообразных контекстах.

♦ **Широко используются в настоящее время.** Некоторые фреймворки agile имеют комплексный характер и хорошо формализованы, но просто не распространены в большинстве проектов и организаций. Описанные в этом приложении фреймворки agile, как показывают результаты ряда недавних отраслевых исследований, были приняты для применения в значительном числе отраслей.

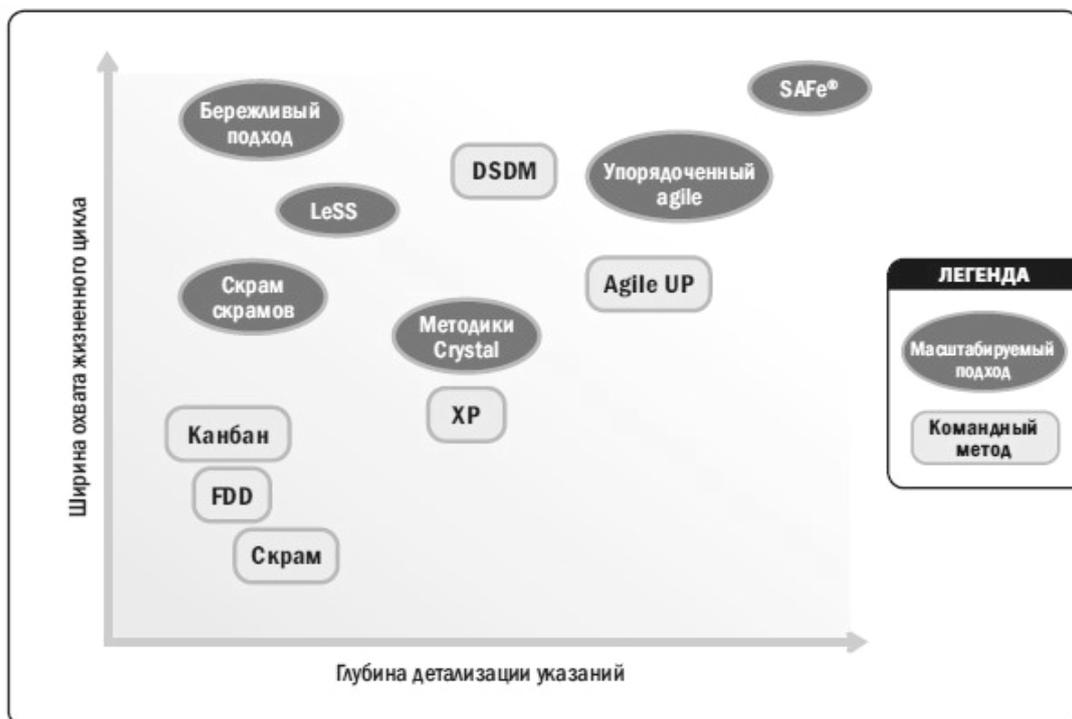


Рис. А3-1. Подходы agile, представленные по ширине охвата и детализации

А3.2 СКРАМ

Скрам – это фреймворк процесса с участием одной команды, используемый для управления разработкой продукта. Этот фреймворк состоит из ролей, событий, артефактов и правил скрам и используется как итеративный подход для поставки работающего продукта. Скрам осуществляется в пределах временных рамок с фиксированной длительностью до 1 месяца, называемых «спринтами», в течение которых производятся инкременты продукта с потенциальной возможностью релиза. В таблице А3-1 перечислены события и артефакты скрам, используемые для исполнения проекта.

Скрам-команда состоит из владельца продукта, команды разработчиков и скрам-мастера.

- ◆ Владелец продукта отвечает за максимизацию ценности продукта.
- ◆ Команда разработчиков является кроссфункциональной самоорганизующейся командой, состоящей из членов, которые имеют внутри своей команды все необходимое для поставки работающего продукта и не зависят при этом ни от кого вне команды.
- ◆ За то, чтобы процесс скрам был правильно организован, а также за соблюдение скрам-командой практик и правил, отвечает скрам-мастер, который, среди прочего, обучает команду приемам преодоления препятствий.

Таблица А3-1. События и артефакты скрам

События	Артефакты
Спринт	Бэклог продукта
Планирование спринта	Бэклог спринта
Ежедневный скрам	Инкременты
Анализ спринта	
Ретроспектива спринта	

А3.3 ЭКСТРЕМАЛЬНОЕ ПРОГРАММИРОВАНИЕ

Экстремальное программирование (eXtreme Programming, XP) – это методология разработки программного обеспечения, основанная на частом повторении циклов разработки. Название вытекает из концепции выделения имеющейся передовой практики в ее наиболее чистой, простой форме и ее постоянного применения на всем протяжении проекта.

Метод XP получил известность, главным образом, благодаря популяризации ряда комплексных практик, предназначенных для улучшения результатов проектов по разработке программного обеспечения. Эта методология была впервые формализована в виде двенадцати основных практик, но в процессе последующего поступательного развития в нее были включены несколько других связанных с ними практик. Они приводятся в таблице А3-2.

Таблица А3-2. Практики экстремального программирования

Область практики XP	Основная	Вторичная
Организационная	<ul style="list-style-type: none"> • совместное размещение • команда как одно целое • информативное рабочее пространство 	<ul style="list-style-type: none"> • привлечение реального заказчика • преемственность в команде • устойчивый темп
Техническая	<ul style="list-style-type: none"> • парное программирование • программирование «сначала тесты» • инкрементное проектирование 	<ul style="list-style-type: none"> • общий код / коллективное владение • документация по коду и тестам • рефакторинг
Планирования	<ul style="list-style-type: none"> • пользовательские истории • недельный цикл • квартальный цикл • slack (временной резерв) 	<ul style="list-style-type: none"> • анализ первопричины • сжимающиеся команды • плата по мере использования • договор с обсуждаемым содержанием • ежедневные летучки
Интеграции	<ul style="list-style-type: none"> • 10-минутная сборка • непрерывная интеграция • «сначала тесты» 	<ul style="list-style-type: none"> • единая база кода • инкрементное развертывание • ежедневное развертывание

Это поступательное развитие стало результатом разработки и адаптации методов с использованием фильтра основных ценностей (коммуникация, простота, обратная связь, смелость и уважение) и было основано на ключевых принципах (человечность, экономика, взаимная выгода, сходство, совершенствование, разнообразие, осмысление, поток, благоприятные возможности, избыточность, неудачи, качество, мелкие шаги, принятая ответственность).

А3.4 МЕТОД «КАНБАН»

Метод «канбан» в бережливом производстве – это система для контроля расписания и пополнения ресурсов. Этот процесс пополнения ресурсов «точно в срок» первоначально возник в гастрономах, где пополнение продуктов на полках производилось по мере появления на полках свободных мест, а не в зависимости от запасов у поставщика. На примере этих систем управления запасами по принципу «точно в срок» Тайити Оно (Taiichi Ohno) разработал метод «канбан», который был применен на главном производственном объекте компании Toyota в 1953 году.

Слово канбан буквально означает в переводе «рекламный щит» или «карточка». Реальные доски «канбан» с карточками создают необходимые условия и способствуют визуализации и потоку работы через систему так, чтобы он был представлен наглядно для всех. Информационная доска (большой дисплей) состоит из колонок, представляющих состояния, через которые поток работы должен пройти, чтобы работа была выполнена. Самые простые доски могут иметь три колонки (а именно: «нужно сделать», «делается» и «сделано»), но их можно адаптировать с указанием состояний, которые, по мнению использующих их команд, необходимы.

Метод «канбан» применяется и предназначен для использования во многих средах. Он позволяет осуществить непрерывный поток работы и ценности для заказчика. Метод «канбан» является менее директивным в сравнении с некоторыми подходами agile и, соответственно, менее деструктивным на начальном этапе его применения, поскольку является оригинальным принципом «начинай прямо там, где находишься». Организации могут сравнительно просто начать применять метод «канбан» с последующим переходом к его полному внедрению, если сочтут это необходимым или целесообразным.

В отличие от большинства подходов agile, метод «канбан» не предусматривает ограниченных временными рамками итераций. Итерации могут использоваться в рамках метода «канбан», однако принцип непрерывного проведения отдельных элементов через процесс и ограничения объемов незавершенных работ для оптимизации потока должен неукоснительно соблюдаться. Метод «канбан» лучше всего можно использовать в случаях, когда команде или организации необходимы следующие условия:

♦ **Гибкость.** Команды, как правило, не связаны временными рамками и занимаются имеющими наивысший приоритет работами, включенными в бэклог.

♦ **Фокус на непрерывной поставке.** Команды считают главной задачей организацию потока работы через систему до ее завершения и не начинают новую работу до завершения текущих работ.

♦ **Повышенные производительность и качество.** Производительность растет и качество улучшается благодаря ограничению объемов текущих работ.

♦ **Увеличенная эффективность.** Проверка каждого задания на предмет наличия добавляющих ценность операций и устранение операций, которые ценность не добавляют.

♦ **Сфокусированность членов команды.** Ограничение объема текущей работы позволяет команде сосредоточить усилия на непосредственно выполняемой работе.

♦ **Вариативность рабочей нагрузки.** Когда существует непредсказуемость в порядке поступления работы и у команд больше нет возможности принимать прогнозируемые обязательства – даже на короткие периоды времени.

♦ **Сокращение потерь.** Прозрачность делает потери наглядными, так что их можно устранить.

Метод «канбан» вытекает из принципов бережливого мышления. Определяющие принципы и основные свойства метода «канбан» перечислены в таблице А3-3.

Метод «канбан» – это комплексный фреймворк, предназначенный для инкрементного, последовательно развивающегося процесса и системных изменений для организаций. В данном методе для продвижения работы в рамках процесса используется принцип «вытягивающая система» (pull system). По завершении командой отдельного элемента она может вытянуть другой элемент в данный шаг.

Таблица А3-3. Определение принципов и свойств метода «канбан»

Определяющие принципы	Основные свойства
Начинайте с текущего состояния	Визуализируйте поток работ
Соглашайтесь на инкрементные, эволюционные изменения	Установите ограничение объема текущей работы
Учитывайте текущий процесс, роли, ответственность и должности	Управляйте потоком
Поощряйте проявление лидерства на всех уровнях	Сделайте политики процесса явно выраженными
	Установите петли обратной связи
	Совершенствуйтесь коллективно

Доски «канбан», подобные представленной на рис. А3-2, являются неавтоматизированной технологией низкого уровня, которая на первый взгляд может создать впечатление чересчур простой, однако те, кто их практически использует, очень скоро понимают их большие возможности. Доски «канбан» при использовании правил входа в колонки и выхода из них, а также ограничений, подобных ограничениям объемов текущих

работ, обеспечивают ясное наглядное представление о потоке работ, узких местах, блокерах и общем статусе. Кроме этого, доска служит в качестве информационной доски для всех желающих, предлагая актуальную информацию о состоянии хода работ в команде.

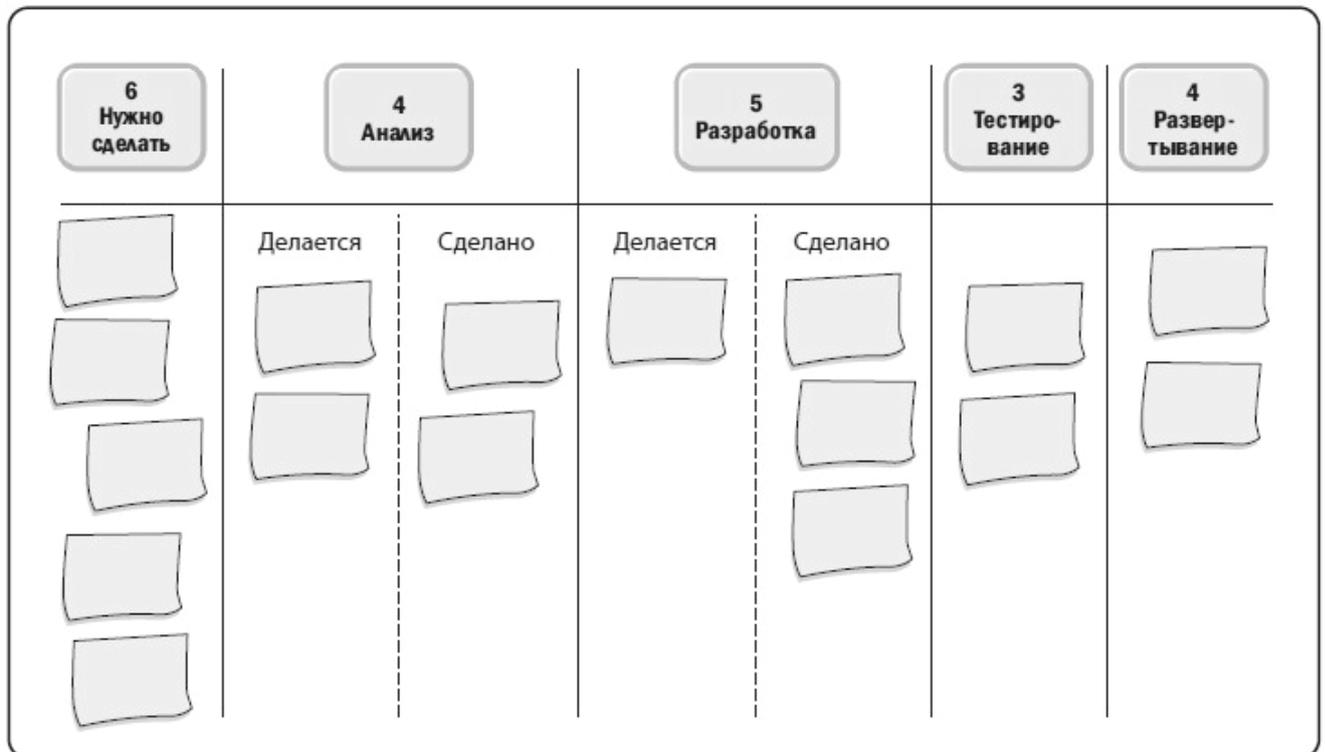


Рис. А3-2. Доска «канбан», демонстрирующая ограничения объемов текущих работ и вытягивающую систему для оптимизации потока работ

В методе «канбан» самое важное – завершить работу, а не начать новую. Незавершенная работа не производит никакой ценности, поэтому команда работает коллективно, чтобы применять и поддерживать лимиты объемов текущей работы (work in progress, WIP) и провести все части работы через систему до состояния «сделано».

А3.5 МЕТОДИКИ CRYSTAL

Методики Crystal – это семейство методологий. Методологии Crystal предназначены для масштабирования и предусматривают выбор средств обеспечения строгости методологии в зависимости от размера (количество участвующих в проекте людей) и критичности проекта.

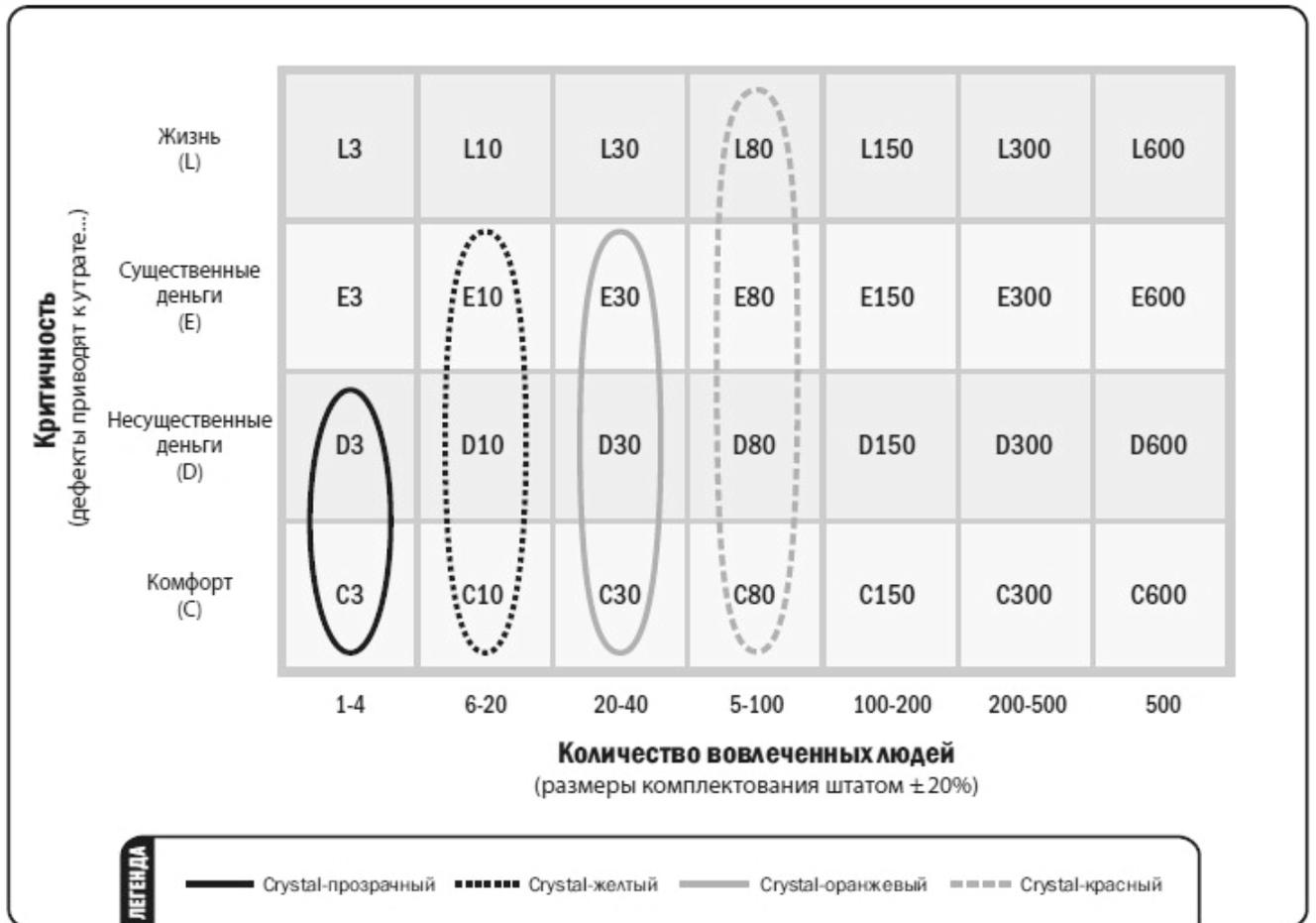


Рис. А3-3. Семейство методологий Crystal

Методология Crystal предусматривает, что каждый проект может потребовать применения ряда незначительно адаптированных политик, практик и процессов, чтобы обеспечить соответствие уникальным характеристикам проекта. В этом семействе методологий для определения методологии, которую следует использовать, применяются различные цвета в зависимости от «веса». Использование слова crystal (кристалл) связано с драгоценным камнем, где разные «грани» представляют главные основополагающие принципы и ценности. Эти грани служат представлением методов, инструментов, стандартов и ролей, перечисленных в таблице А3-4.

Таблица А3-4. Главные ценности и наиболее распространенные свойства методологии Crystal

Основные ценности	Общие свойства ^А
Люди Взаимодействие Сообщество Навыки Таланты Коммуникации	Частые поставки Совершенствование через рефлексию Тесная или осмотическая коммуникация Личная безопасность Фокусировка Легкий доступ к экспертам Техническая среда с автоматизированным тестированием, управлением конфигурацией и частой интеграцией

^А Чем больше таких свойств в проекте, тем выше вероятность его успешного завершения.

А3.6 СКРАМБАН

Скрамбан – это подход agile, изначально предназначенный для перехода от метода «скрам» к методу «канбан». По мере формирования дополнительных фреймворков и методологий agile скрамбан стал самостоятельным поступательно развивающимся гибридным фреймворком, когда команды используют метод «скрам» в качестве фреймворка, а метод «канбан» в целях совершенствования процесса.

При подходе скрамбан работа организована в небольших спринтах, а доски «канбан» применяются для наглядного представления и мониторинга работы. Истории помещаются на доску «канбан», и команда организует свою работу с учетом ограничений на объемы текущей работы. Для поддержания сотрудничества между членами команды и устранения препятствий проводятся ежедневные совещания. Определяется триггер планирования, чтобы команда знала, когда в следующий раз заниматься планированием. Обычно это происходит тогда, когда уровень текущей работы опускается ниже установленного предварительно ограничения. В методе скрамбан отсутствуют предварительно определенные роли – команда сохраняет свои текущие роли.

А3.7 РАЗРАБОТКА НА ОСНОВЕ ФУНКЦИОНАЛЬНОСТИ

Разработка на основе функциональности (Feature-Driven Development, FDD) была создана для удовлетворения особых потребностей крупных проектов по разработке программных продуктов. Функциональные свойства соотносятся с возможностью создания небольшой бизнес-ценности.

В проекте по разработке на основе функциональности имеется шесть основных ролей, когда человек может играть одну или несколько из них:

- ◆ руководитель проекта,
- ◆ главный архитектор,
- ◆ руководитель разработки,
- ◆ главный программист,
- ◆ владелец класса, и (или)
- ◆ эксперт в прикладной области.

Организация проекта по разработке на основе функциональности осуществляется на основе пяти процессов или действий, которые осуществляются итеративно:

- ◆ разработка общей модели,
- ◆ создание списка свойств,
- ◆ планирование по свойствам,
- ◆ проектирование по свойствам,
- ◆ создание по свойствам.

Поток жизненного цикла и взаимодействия этих пяти процессов показаны на рис. А3-4. Обеспечением операций разработки на основе функциональности служит основной набор передовых практик разработки программного обеспечения, а именно:

- ◆ объектное моделирование предметной области,
- ◆ разработка по свойствам,
- ◆ владение индивидуальными классами,
- ◆ закрепленные за свойствами команды,
- ◆ инспекции,
- ◆ управление конфигурацией,
- ◆ регулярные сборки,
- ◆ наглядность прогресса и результатов.

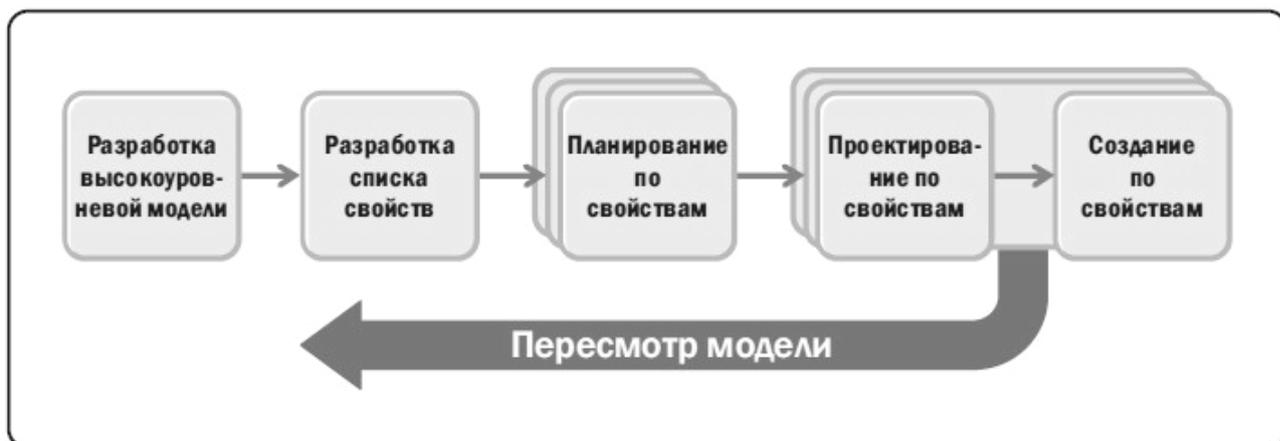


Рис. А3-4. Жизненный цикл проекта по разработке на основе функциональности

А3.8 ДИНАМИЧЕСКИЙ МЕТОД РАЗРАБОТКИ СИСТЕМ

Динамический метод разработки систем (Dynamic Systems Development Method, DSDM) – это фреймворк поставки проекта agile, изначально предназначенный для придания большей строгости существующим итеративным методам, получившим широкое распространение в 1990-е годы. Он был разработан как форма некоммерческого сотрудничества среди отраслевых лидеров.

DSDM известен, главным образом, своим упором на поставку на основе ограничений. Этот фреймворк устанавливает стоимость, качество и время с самого начала, и затем использует формальную приоритизацию содержания, чтобы обеспечить соблюдение этих ограничений, как показано на рис. А3-5.

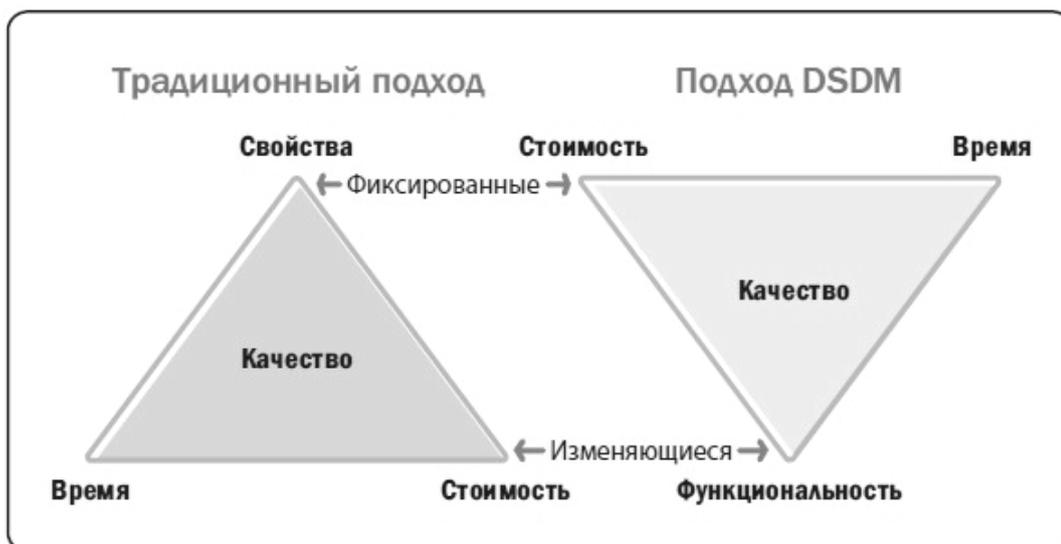


Рис. А3-5. Подход DSDM к гибкости на основе ограничений

Порядок использования фреймворка DSDM определяют восемь принципов:

- ◆ фокус на потребности бизнеса,
- ◆ поставка в срок,
- ◆ сотрудничество,
- ◆ недопустимость снижения уровня качества,
- ◆ инкрементное создание продукта на твердой основе,
- ◆ итеративная разработка,
- ◆ непрерывные и ясные коммуникации,
- ◆ демонстрация контроля (использование соответствующих методов).

А3.9 УНИФИЦИРОВАННЫЙ AGILE-ПРОЦЕСС

Унифицированный agile-процесс (Agile Unified Process, AgileUP) является ответвлением унифицированного процесса (Unified Process, UP) для проектов по разработке программных продуктов. В сравнении с предшествовавшим ему унифицированным процессом он отличается ускоренными циклами и менее тяжеловесными процессами. Цель состоит в совершении большего числа итеративных циклов во всех семи ключевых дисциплинах, а также инкорпорировании полученной обратной связи до формальной поставки. Эти дисциплины, наряду с руководящими принципами, перечислены в таблице А3-5.

Таблица А3-5. Ключевые элементы унифицированного agile-процесса

Дисциплины в рамках релиза	Принципы, определяющие дисциплины
Модель	Команда знает, что делает
Реализация	Простота
Тестирование	Гибкость
Развертывание	Фокус на действиях с высокой ценностью
Управления конфигурацией	Независимость инструментов
Управление проектом	Адаптация для использования
Среда	Ситуационная специфика

A3.10 МАСШТАБИРУЕМЫЕ ФРЕЙМВОРКИ

A3.10.1 СКРАМ СКРАМОВ

Скрам скрамов (Scrum of Scrums, SoS), известный также под названием «мета-скрам» (meta Scrum) – это метод, используемый в случаях, когда у двух и более скрам-команд в составе от трех до девяти членов в каждой возникает необходимость в координации своей работы, чтобы не создавать одну большую скрам-команду. Представитель от каждой команды участвует в совещаниях с представителями других команд, потенциально каждый день, но на практике два-три раза в неделю. Такие ежедневные совещания проводятся аналогично ежедневным летучкам в скраме, когда представитель докладывает о выполненных работах, предстоящей на следующем этапе части работы, а также потенциальных предстоящих препятствиях, которые могут помешать работе других команд. Цель состоит в обеспечении того, чтобы команды координировали работу и устраняли препятствия для оптимизации эффективности всех команд.

Результатом крупных проектов с участием нескольких команд может стать проведение «скрам-скрама-скрамов» (Scrum of Scrum of Scrums), который имеет такую же схему, как и «скрам скрамов» (SoS), когда представитель каждого «скрама скрамов» докладывает на совещании большей группы представителей, как показано на рис. A3-б.

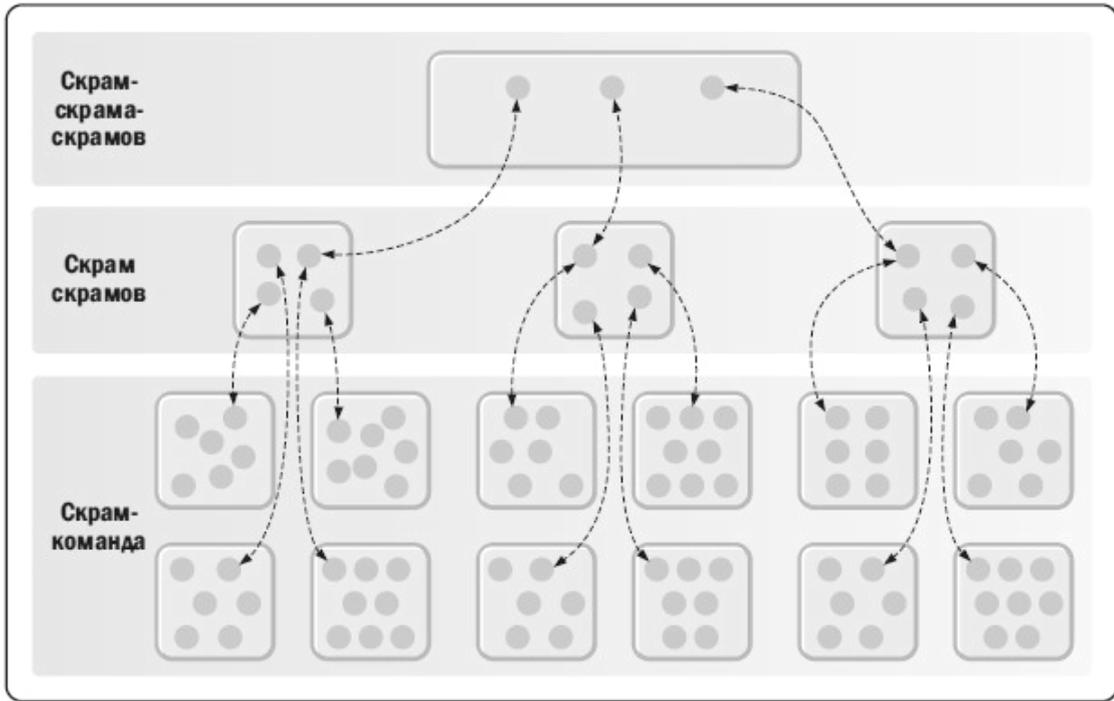


Рис. А3-6. Представители скрам-команд, участвующие в командах скрама скрамов

А3.11 МАСШТАБИРОВАННЫЙ AGILE-ФРЕЙМВОРК

Основная задача масштабированного agile-фреймворка (Scaled Agile Framework, SAFe®) состоит в формировании базы знаний о схемах масштабирования работы по разработке на всех уровнях предприятия.

В основе SAFe® лежат следующие принципы:

- ◆ учет экономических факторов;
- ◆ применение системного мышления;
- ◆ принятие изменчивости, сохранение вариантов;
- ◆ создание инкрементов в рамках быстрых, интегрированных циклов обучения;
- ◆ базовые контрольные события по объективной оценке работающих систем;
- ◆ наглядное представление и ограничение объема текущих работ, уменьшение размера порций работ и управление длинами очередей;
- ◆ применение каденции, синхронизация с межобластным планированием;
- ◆ высвобождение внутренней мотивации работников интеллектуального труда;
- ◆ децентрализация процесса принятия решений.

В фокусе SAFe® находятся практики детализации, роли и действия на уровнях портфеля, программы и команды с упором на организацию предприятия вокруг потоков ценностей, главная задача которых состоит в поставке непрерывной ценности заказчику.

А3.12 КРУПНОМАСШТАБНЫЙ СКРАМ (LESS)

Крупномасштабный скрам (Large Scale Scrum, LeSS) – это фреймворк, предназначенный для организации работы нескольких команд разработки для достижения общей цели, расширяющий метод скрам, представленный на рис. А3-6. Главный организационный принцип состоит в максимальном сохранении элементов традиционной модели фреймворка

скрам для одной команды. Это помогает минимизировать расширения этой модели, которые могут стать причиной возникновения ненужной путаницы и сложности. В таблице А3-6 дано сравнение LeSS со скрамом.

Таблица А3-6. Сравнение LeSS со скрамом

Сходства LeSS и скрам	Методы LeSS добавленные к скраму
<p>Один бэклог продукта</p> <p>Один набор критериев выполнения для всех команд</p> <p>Один потенциально поставляемый инкремент продукта по окончании каждого спринта</p> <p>Один владелец продукта</p> <p>Полные кроссфункциональные команды</p> <p>Один спринт</p>	<p>Планирование спринта более формально делится на две части: «что» и «как»</p> <p>Органическая координация между командами</p> <p>Общее кросс-командное совершенствование</p> <p>Общая ретроспектива, направленная на улучшение взаимодействия между командами</p>

Для расширения скрама без утраты его сущности LeSS поощряет применение определенных принципов здравого смысла, например, системного мышления, фокуса на продукте в целом, прозрачности и других.

А3.13 СКРАМ ПРЕДПРИЯТИЯ

Скрам предприятия – это фреймворк, предназначенный для применения метода скрам на более глобальном организационном уровне, а не на уровне единичной разработки продукта. В частности, этот фреймворк рекомендует руководителям организаций:

- ◆ распространить использование метода скрам на все аспекты деятельности организации;
- ◆ обобщить методы скрам, чтобы их можно было без труда применять к этим разнообразным аспектам;
- ◆ масштабировать метод скрам с использованием, по мере необходимости, дополнительных методов.

Цель состоит в использовании подходов agile за пределами исполнения проекта путем создания условий для применения прорывных инноваций.

А3.14 УПОРЯДОЧЕННЫЙ AGILE

Упорядоченный agile (Disciplined Agile, DA) – это фреймворк процесса принятия решений, где несколько передовых практик подхода agile интегрированы в одной комплексной модели. Фреймворк DA был разработан с целью предложить баланс между теми популярными методами, которые считались либо имеющими слишком узкий фокус (например, скрам), либо слишком директивными по детализации (например, AgileUP). Чтобы обеспечить этот баланс, в данном фреймворке объединяются воедино различные методы agile на основе следующих принципов:

- ◆ **Сначала люди.** Определение ролей и элементов организации на различных уровнях.
- ◆ **Ориентация на обучение.** Поощрение совместного совершенствования.

♦ **Жизненный цикл полной поставки.** Использование нескольких подходящих по назначению жизненных циклов.

♦ **Ориентация на цель.** Адаптация процессов для достижения конкретных результатов.

♦ **Осознание интересов организации в целом.** Предложение рекомендаций по управлению организацией на основе взаимодействия ее подразделений.

♦ **Возможность масштабирования.** Охват множества измерений сложностей программы.

Приложение X1. Авторы и рецензенты

X1.1 ОСНОВНОЙ КОМИТЕТ «*AGILE: ПРАКТИЧЕСКОЕ РУКОВОДСТВО*»

В состав членов Основного комитета, ответственного за подготовку черновика данного руководства, включая обзор и одобрение предложений рецензентов, входили следующие специалисты:

X1.1.1 ПРЕДСТАВИТЕЛИ ИНСТИТУТА УПРАВЛЕНИЯ ПРОЕКТАМИ (PROJECT MANAGEMENT INSTITUTE, PMI):

Mike Griffiths, PMP, PMI-ACP, (председатель Комитета)

Jesse Fewell, CST, PMI-ACP

Horia Slușanschi, PhD, CSM

Stephen Matola, BA, PMP

X1.1.2 ПРЕДСТАВИТЕЛИ AGILE ALLIANCE:

Johanna Rothman, MS (заместитель председателя Комитета)

Becky Hartman, PMI-ACP, CSP

Betsy Kaufman, ICP-ACC, PMI-ACP

X1.2 РЕЦЕНЗЕНТЫ – ЭКСПЕРТЫ ПО ПРЕДМЕТНЫМ ОБЛАСТЯМ «*AGILE: ПРАКТИЧЕСКОЕ РУКОВОДСТВО*»

Следующие специалисты работали в качестве приглашенных экспертов по предметным областям, которые отвечали за рецензирование черновика и давали рекомендации в процессе SME-рецензирования.

Joe Astolf, PMP, PSM

Maria Cristina Barbero, PMI-ACP, PMP

Michel Biedermann, PhD, PMI-ACP

Zach Bonaker

Robert Bulger, PfMP, CSM

Sue Burk

Shika Carter, PMP, PMI-ACP

Lauren Clark, PMP, CSM

Linda M Cook, CSM, CSPO

Pamela Corbin-Jones, PMI-ACP, CSM

Jef Covert

Alberto Dominguez, MSc, PMP

Scott P. Duncan, CSM, ICP-ACC

Sally Elatta, PMI-ACP, EBAC

Frank R. Hendriks, PMP, PMI-ACP

Derek Huether

Ron Jefries

Fred Koos

Philippe B. Kruchten, PhD, PEng

Steve Mayner, SPCT4, PMP

Michael S. McCalla, PMI-ACP, CSP

Don B. McClure, PMP, PMI-ACP

Anthony C. Mersino, PMI-ACP, CSP
 Kenneth E. Nidifer, PhD, PMP
 Michael C. Nollet, PMP, PMI-ACP
 Laura Paton, MBA, PMP
 Yvan Petit, PhD, PMP
 Dwayne Phillips, PhD, PMP
 Piyush Prakash, PMP, Prince2
 Dave Prior, PMP, CST
 Daniel Rawsthorne, PhD, PMP
 Annette D. Reilly, PMP, PhD
 Stephan Reindl, PMI-ACP, PMP
 Reed D. Shell, PMP, CSP
 Cindy Shelton, PMP, PMI-ACP
 Teresa Short
 Lisa K. Sieverts, PMP, PMI-ACP
 Christopher M. Simonek, PMP, CSM
 Robert "Sellers" Smith, PMP, PMI-ACP
 Ram Srinivasan, PMP, CST
 Chris Stevens, PhD
 Karen Strichartz, PMP, PMI-ACP
 Rahul Sudame, PMI-ACP
 Joanna L. Vahlsing, PMP
 Erik L. van Daalen
 Annette Vendelbo, PMP, PMI-ACP
 Dave Violette, PMP, PMP
 Anton Vishnyak, PMI-ACP, CSM
 Chuck Walrad, MA, MS

X1.3 ФОКУС-ГРУППА ПО ФОРМАТИРОВАНИЮ

Следующие специалисты участвовали в разработке нового стиля представления содержания и элементов форматирования для *«Agile: практическое руководство»*.

Goran Banjanin, PgMP, PMP
 Andrew Craig
 Cătălin-Teodor Dogaru, PhD, PMP
 Jorge Espinoza, PMP
 Jennifer M. Forrest, CSM, PMP
 Helen Fotos, PMP, PMI-ACP
 Dave Hatter, PMP, PMI-ACP
 Christopher Healy, PMP
 Mike Hofmann, MBA, PMP
 Chadi Kahwaji, PMP
 Rajaraman Kannan, PMP, MACS CP
 Amit Khanna PMP, PMI – ACP
 Ariel Kirshbom, PMI-ACP, CSP
 Bernardo Marques, PMP
 Noura Saad, PMI-ACP, CSPO
 Kurt Schuler, PMP
 Demetrius L. Williams, MBA, PMP
 Liza Wood
 Melody Yale, CSP, SPC4

X1.4 ГРУППА ЧЛЕНОВ-КОНСУЛЬТАНТОВ (MEMBER ADVISORY GROUP, MAG) ПО СТАНДАРТАМ PMI

Следующие специалисты работали в составе Группы членов-консультантов по стандартам PMI, которые давали рекомендации по стандартам PMI и окончательное одобрение от имени PMI в ходе подготовки «*Agile: практическое руководство*».

Maria Cristina Barbero, PMI-ACP, PMP

Brian Grafsgaard, PMP, PgMP

Hagit Landman, PMP, PMI-SP

Yvan Petit PhD, PMP

Chris Stevens, PhD

Dave Violette, MPM, PMP

John Zlockie, MBA, PMP, руководитель по стандартам PMI

X1.5 СОВЕТ ДИРЕКТОРОВ AGILE ALLIANCE®

Следующие специалисты являются членами Совета директоров Agile Alliance, которые давали рекомендации и окончательное одобрение от имени Agile Alliance в ходе подготовки «*Agile: практическое руководство*».

Juan Banda

Phil Brock (исполнительный директор)

Linda Cook

Stephanie Davis

Ellen Grove

Paul Hammond (председатель)

Victor Hugo Germano

Rebecca Parsons (секретарь)

Craig Smith

Declan Whelan

X1.6 ТЕХНИЧЕСКИЙ ПЕРСОНАЛ PMI И НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЕ ОБЕСПЕЧЕНИЕ

Следующие специалисты обеспечивали поддержку Основного комитета в ходе разработки и одобрения проекта Руководства, при оказании поддержки группы, ответственной за форматирование, и работы PMI по маркетингу.

Melissa M. Abel, специалист по маркетинговым коммуникациям

Karl F. Best, PMP, CStd, специалист по стандартам

Alicia C. Burke, MBA, CSM, руководитель продукта, подтверждение квалификации

Edivandro C. Conforto, PhD, консультант PMI по исследованиям в области Agile

Dave Garrett, CSPO, вице-президент по трансформации

Erica Grenfell, административный помощник вице-президента, организационные отношения

M. Elaine Lazar, MA, MA, AStd, специалист по проектам

Andrew Levin, PMP, руководитель проекта

Tim E. Oglie, дизайнер пользовательского взаимодействия

Stephen A. Townsend, директор сетевых программ

Michael Zarro, PhD, исследователь пользовательского взаимодействия

X1.7 ПРОИЗВОДСТВЕННЫЙ ПЕРСОНАЛ PMI

Donn Greenberg, менеджер, Отдел публикаций

Kim Shinnars, сотрудник отдела подготовки публикаций

Roberta Storer, редактор конечного продукта

Barbara Walsh, ответственный издатель

X1.8 ЧЛЕНЫ КОМИТЕТА ПО ПРОВЕРКЕ ПЕРЕВОДА НА РУССКИЙ ЯЗЫК

Valerii Funtov, DrSc, PMP

Mikhail Lazarev, PMP

Kirill Melnikov, PhD, PMP

Konstantin Trunin, PMP

X1.9 КОМИТЕТ ПО ПРОВЕРКЕ ПРАВИЛЬНОСТИ ПЕРЕВОДА

Barbara Walsh, ответственный издатель

Margaret Lyons, разработчик экзаменационных материалов

Stephen Townsend, директор, Network Programs

Vivian Isaak, президент бюро переводов Magnum Group, Inc.

Brian Middleton, менеджер по стратегическим решениям бюро переводов Magnum Group, Inc.

Приложение X2. Свойства, влияющие на адаптацию

X2.1 ВВЕДЕНИЕ

Настоящее приложение содержит высокоуровневые указания о том, когда и как требуется осуществлять адаптацию подходов agile. Его можно использовать для определения обстоятельств, которые могут послужить основанием для внесения изменений или применения новых методов. Кроме того, здесь содержатся некоторые рекомендации для принятия во внимание.

Модель приобретения навыков «сюхари» (Shu-Ha-Ri) описывает порядок последовательного перехода от соблюдения правил («Сю» 守 означает повиновение и защиту) через осознанное отступление от правил («Ха» 破 означает изменение или отступление) и, наконец, определение в процессе последовательной практики и совершенствования собственного пути («Ри» 離 означает отделение или отход). Сначала нужно приступить к деятельности и пройти практику на уровне «Сю», чтобы подготовиться к переходу на уровень «Ха» для адаптации процесса или на уровень «Ри» для создания нового кастомизированного процесса.

X2.2 СНАЧАЛА НЕСКОЛЬКО ПРЕДУПРЕЖДЕНИЙ

Адаптация – это продвинутая тема обсуждения для опытных специалистов-практиков, которые, прежде чем рассматривать возможность адаптации подходов agile, ранее успешно использовали такие подходы в соответствии с оригинальным описанием их применения в разнообразных средах. Другими словами, сначала нужно приобрести опыт и добиться успеха в применении того или иного подхода, прежде чем пытаться его адаптировать.

Обычно предмет дискуссии о применении практики agile состоит в поиске ответа на вопрос, стоит это делать или нет. Заявление типа «Ретроспективы были непопулярны, поэтому мы решили отказаться от них» иллюстрирует эту проблему и указывает на наличии более серьезной проблемы в команде, которую вряд ли можно будет решить путем адаптации метода. Ситуация станет еще хуже в результате пренебрежения к ретроспективной деятельности, которая направлена на совершенствование процесса.

Наконец, адаптацией следует заниматься совместно с коллегами по команде или лицами, которых изменение, вероятно, коснется. Людям нужно участвовать в процессе обдумывания и принятия решений о процессах внесения изменений, чтобы они приняли изменения и поддерживали их для успешного перехода. Отстранение людей от адаптации процесса, скорей всего, приведет к сопротивлению изменениям и недовольству ими, даже если с технической точки зрения они являются целесообразными. Часто задачу вовлечения людей можно результативно решить с помощью опытных коучей или руководителей.

X2.3 КАК ИСПОЛЬЗОВАТЬ ЭТО ПРИЛОЖЕНИЕ

Чтобы можно было извлечь выгоду от содержащихся в настоящем приложении указаний, рекомендуем сначала добиться успешного использования подходов agile в исходном виде.

Затем следует изучить указания по адаптации, приведенные в таблице X2-1, которые применимы к данной ситуации, и ознакомиться с соответствующими рекомендациями. Далее обсудите изменения и согласуйте порядок действий с людьми, на которых они окажут воздействие.

Как сказано в разделе 5, хорошим способом оценки предложенного изменения для начала станет его практическая проверка в течение одной или двух итераций до его окончательного принятия. Как вариант можно рассмотреть потоковый подход, попытавшись осуществить поставку нескольких свойств. Затем подумать в ретроспективе и выполнить оценку снова.

Когда люди знают, что у них есть возможность экспериментировать и давать обратную связь по результатам эксперимента, вероятность того, что они будут пробовать новшества, выше. Проведя испытания в ограниченный временными рамками период, команда должна рассмотреть вопрос его результативности в ретроспективе, чтобы определить, можно ли продолжать использовать новшество «как есть», нужно ли внести в него изменения для усовершенствования или отказаться от него.

Наконец, успешно принятые адаптированные подходы можно принять официально, введя их в число стандартных процессов, используемых в проектах, которые имеют общие характеристики. Рекомендуется также соблюдать приведенные в разделе 5 указания, которые описывают порядок принятия (или адаптации) новых подходов.

X2.4 РЕКОМЕНДАЦИИ ПО АДАПТАЦИИ

Ниже перечислены некоторые хорошие практики, которые следует рассмотреть, прежде чем приступить к адаптации подхода.

X2.4.1 НУЖНА ОСТОРОЖНОСТЬ ПРИ ОТКАЗЕ ОТ ЧЕГО-ЛИБО

Многие практики подхода agile действуют как взаимозависимые пары. Например, совместное расположение и регулярные деловые обсуждения позволяют принять облегченные требования, поскольку пробелы во взаимопонимании можно быстро ликвидировать. Схожим образом строгое тестирование в рамках XP позволяет смело осуществлять рефакторинг, поскольку одна практика поддерживает другую. Отказ от чего-либо без понимания или принятия мер в отношении уравнивающей практики, скорей всего, создаст больше проблем, чем позволит решить.

X2.4.2 ИСПОЛЬЗОВАНИЕ ТАБЛИЦЫ С УКАЗАНИЯМИ ПО АДАПТАЦИИ

С помощью таблицы X2-1 найдите обстоятельства, которые соответствуют определенной ситуации и рассмотрите рекомендации по адаптации. Обсудите любые изменения с теми, на кого они окажут влияние, и для начала запланируйте краткосрочное испытание наряду с последующим добросовестным анализом по результатам испытаний, прежде чем окончательно принять предлагаемые изменения.

Таблица X2-1. Указания по адаптации

Ситуация	Рекомендации по адаптации
Очень большие команды проекта	<p>Измените структуру проекта так, чтобы он состоял из нескольких более мелких проектов. Сначала осуществите испытательный проект для проверки технологии, а затем проект внедрения.</p> <p>Рассмотрите более частые релизы с меньшим числом свойств в каждом. Это позволит использовать команды проектов меньшего размера.</p> <p>Попробуйте сократить численность команды до минимального числа ключевых членов. Во многих случаях слишком большое количество людей не способствует, а мешает осуществлению процесса. Сокращение численного состава команды позволяет уменьшить «текучку», а также затраты.</p> <p>Разбейте большие команды на несколько команд меньшего размера и используйте управление программой для синхронизации и координации их работы.</p> <p>Используйте agile и бережливое управление программой при организации более крупных работ.</p> <p>Попробуйте масштабированный agile-фреймворк или бережливый фреймворк, например DA, SAFe® или LeSS. Каждый из них предлагает некоторые полезные идеи и сопряжен с рисками реализации и определенным соотношением весомости и стоимости процесса.</p>

Таблица X2-1. Указания по адаптации (прод.)

Ситуация	Рекомендации по адаптации
<p>Рассредоточенные команды</p>	<p>Многие проекты имеют (несколько) рассредоточенных членов команды. Такие инструменты, как мгновенные сообщения, видео-конференции и электронные доски команды, помогают устранить многие коммуникационные проблемы.</p> <p>Когда есть вероятность сохранения стабильного состава команды, как можно скорее проведите совещания с личным присутствием ее членов, чтобы сделать удаленное общение между ними в будущем более результативным. Люди, которые встречались лично, благодаря наличию большего доверия между ними скорее вступят в открытую дискуссию.</p> <p>При проведении совещаний с удаленными участниками, когда нельзя наблюдать за мимикой и языком тела, подумайте о проведении опроса участников «по кругу», чтобы обеспечить их участие и проверку консенсуса при принятии решений.</p> <p>Подумайте также об использовании подходов итерационного agile. Когда члены команды находятся на расстоянии нескольких часовых поясов друг от друга, стоит подумать о менее частых взаимодействиях в составе всей команды, поощряя в то же время более частые контакты в более частном порядке (с участием двух-трех человек).</p>
<p>Некоторые имеющие критическое значение для безопасности продукты могут требовать подготовки дополнительной документации и проверки соответствия сверх того, что по умолчанию предлагают процессы agile.</p>	<p>В таких средах, тем не менее, подходы agile можно использовать, но они должны предусматривать соответствующие дополнительные уровни проверки соответствия, документацию и сертификацию, которые требует данная область применения. В этом случае документация может быть частью того, что команда поставляет вместе с завершенными свойствами. Свойства не могут быть выполнены, пока не закончена подготовка документации.</p> <p>Попробуйте гибридный подход (несколько подходов agile), чтобы получить выгоды от совершенствования сотрудничества и коммуникаций, которые дает agile, при усилении строгости, которая требуется в связи со средой применения продукта. Разработчики систем управления полетом воздушных судов, а также фармацевтические компании используют подходы agile в сочетании со своими собственными дополнительными процессами для использования всех выгод при сохранении соответствующих средств контроля.</p>
<p>Стабильность требований и процесса исполнения</p>	<p>Действительно ли нужен agile? При низком уровне неопределенности в требованиях, низких темпах изменений или минимальном риске исполнения необходимость в применении подходов agile в полном объеме может отсутствовать. Несмотря на то, что любому проекту пойдет на пользу усиление сотрудничества и прозрачности, некоторые итеративные циклы разработки и обзора могут быть излишними.</p> <p>Если циклы разработки/обратной связи не обеспечивают в обычном порядке обнаружение и уточнение требований, попробуйте увеличить продолжительность циклов для минимизации воздействия на стоимость времени, затраченного на обзор.</p> <p>Если проект характеризуется высокими темпами изменений на стадии проектирования и разработки, а его поэтапная поставка заказчиком является устоявшимся и повторяемым процессом, более целесообразным может быть применение гибридных подходов, в которых используется модель соответствующего жизненного цикла для каждой фазы проекта.</p>
<p>Команды находятся в функциональной изоляции внутри функциональных организаций</p>	<p>В основе agile лежит идея кроссфункциональных команд. Попробуйте предложить людям создать кроссфункциональные команды самостоятельно, без участия руководства и посмотрите, что произойдет.</p> <p>Если система вознаграждения основана на признании и вознаграждении по функциональным областям, попробуйте сначала изменить ее. Действия людей могут не учитывать интересы продукта или команды до тех пор, пока они так или иначе не оказывают влияние на вознаграждение за их труд.</p>

Таблица X2-1. Указания по адаптации (прод.)

Ситуация	Рекомендации по адаптации
Прозрачность может вызывать страх	<p>Agile создает культуру прозрачности: люди демонстрируют результаты своей работы и делятся ими на всем протяжении разработки. Прозрачность — это обмен промежуточными поставляемыми результатами, а также открытость и достоверность информации об успехах, неудачах и текущем состоянии дел. Прозрачность требует мужества.</p> <p>Подавайте пример и демонстрируйте прозрачность в процессах принятия решений, используя информационную доску или белую доску.</p>
Многие члены команды имеют ограниченные знания в технической области	<p>Подходы agile поощряют создание и работу самоорганизующихся команд для принятия локальных решений по рабочим вопросам, например о последовательности исполнения заданий, а также о том, какой подход надо использовать при решении проблемы. Когда большинство членов команды не обладает нужным опытом, применение подходов, требующих единогласия, повлечет за собой проблемы и доработки. Поэтому в таких командах может быть необходима дополнительная помощь по «назначению» и «управлению», пока команда не приобретет необходимые навыки. Другими словами, нельзя взять и объявить о введении agile и поручить неопытным членам команды решать самим все вопросы просто потому, что им предоставлены необходимые полномочия и право на самоорганизацию. Подумайте о создании центров компетенций, чтобы помочь в руководстве и приобретении знаний в предметной области проекта.</p>
Отсутствие поддержки высшего руководства	<p>При отсутствии поддержки высшего руководства в командах возникает противоречие между образом мышления и подходами agile и более предиктивными образом мышления и подходами.</p> <p>Найдите общее основание, области совершенствования, исходя из потребностей организации, и затем используйте эксперименты и ретроспективы для продвижения вперед.</p> <p>Подумайте о проведении обучения/подготовки высших руководителей. Подумайте о разъяснении agile в терминах бережливого мышления: короткие циклы, небольшие размеры порций работ, частые обзоры и ретроспективы с небольшими улучшениями.</p>
Термины и язык agile не подходят к культуре организации	<p>Измените терминологию так, чтобы люди понимали и согласовывали саму деятельность, пусть и не посредством языка agile. Давайте конкретные определения каждому термину.</p> <p>Например, если организация считает, что слово «игра» звучит непрофессионально, не используйте такие понятия, как «игра в планирование». Вместо него можно попробовать использовать понятие «семинар по планированию».</p>

Приложение X3. Инструменты фильтров применимости agile

X3.1 ВВЕДЕНИЕ

В литературе по agile предлагается много инструментов фильтров, используемых для оценки того, при каких обстоятельствах может быть целесообразно применять подход agile. В 1994 г. в рамках динамического метода разработки систем (DSDM) были созданы «Анкета оценки применимости agile к проекту» (Agile Project Suitability Questionnaire) и «Анкета оценки применимости к организации» (Organizational Suitability Questionnaire), предназначенные для измерения вероятности соответствия требованиям, а также для определения потенциальных проблемных областей.

В семействе подходов Crystal также применяются критерии оценки применимости: проекты располагаются в порядке их размера и важности разрабатываемого продукта или услуги. Согласно Crystal, рекомендуется, чтобы проекты меньшего размера и не имеющие решающего значения осуществлялись с облегченным управлением и применением более простых подходов. При осуществлении крупных, имеющих критическое значение для миссии или жизни проектов было рекомендовано применять больше строгости и валидации.

За период после разработки этих подходов появилось еще много моделей, предназначенных для определения условий и времени применения подхода agile. Бем (Boehm) и Тернер (Turner) взяли на вооружение некоторые элементы из DSDM и Crystal для разработки популярной модели оценки, помогающей определить при осуществлении проектов целесообразность использования agile или более традиционных подходов.

Основываясь на указанных предшествующих моделях и расширяя рассмотрение до компромисса из гибридных подходов, предлагается следующая модель. Она представляет

собой синтез нескольких свойств фильтров применимости, призванных помочь организациям оценить и обсудить, насколько целесообразно использовать предиктивные, гибридные подходы или подходы agile.

Х3.2 КРАТКИЙ ОБЗОР МОДЕЛИ

Оценка свойств организации и проекта производится по трем основным категориям:

♦ **Культура.** Имеется ли благоприятная среда с поддержкой данного подхода и доверием внутри команды?

♦ **Команда.** Имеет ли команда подходящий размер для успеха в использовании agile, обладают ли ее члены необходимым опытом и доступом к представителям бизнеса для достижения успеха?

♦ **Проект.** Имеют ли место высокие темпы изменений? Возможна ли инкрементная поставка? Насколько критическое значение имеет проект?

На вопросы в каждой из указанных категорий даются ответы, а результаты изображаются на лепестковой диаграмме. Группы значений вокруг центра диаграммы показывают высокую степень приемлемости использования подходов agile. Результаты вдоль внешнего края показывают, что более целесообразным может быть предиктивный подход. Значения в средней части (в промежуточной области между подходом agile и предиктивным подходом) указывают, что хороший результат мог бы дать гибридный подход. Пример диаграммы показан на рис. Х3-1.

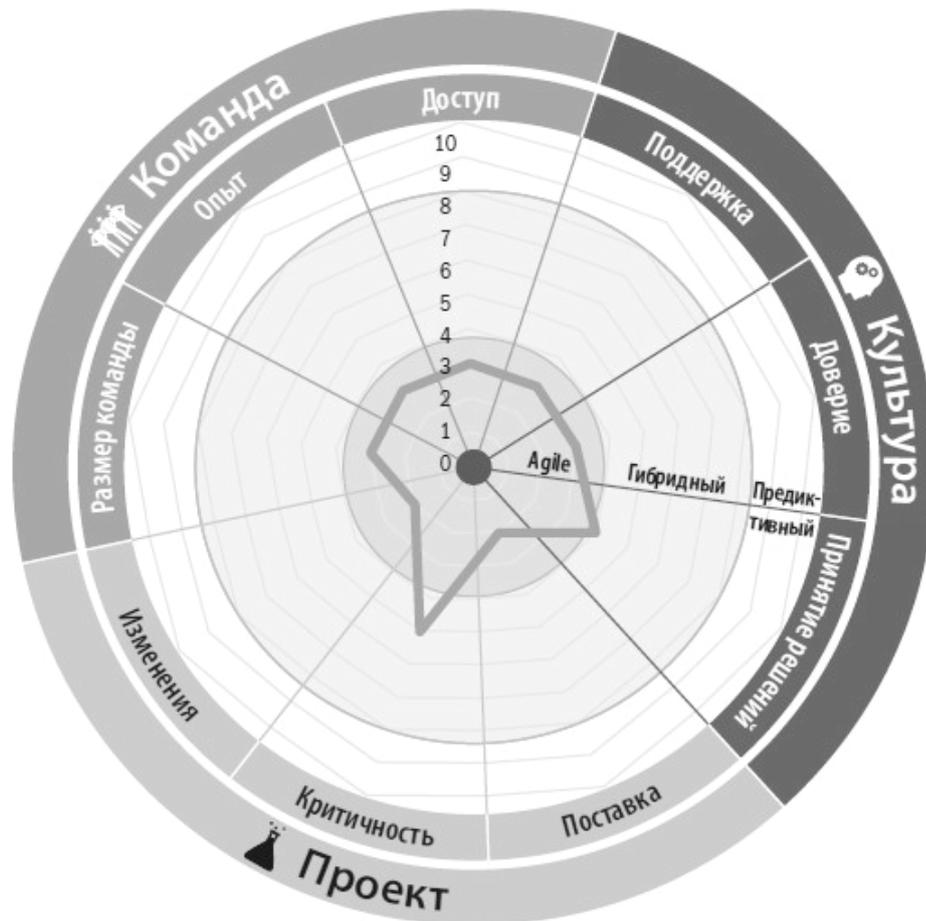


Рис. Х3-1. Модель применимости подхода agile

Х3.3 УКАЗАНИЯ ПО ПРИМЕНЕНИЮ

Х3.3.1 ЗАПОЛНИТЕ АНКЕТУ ВСЕЙ ГРУППОЙ

В небольших проектах группа может состоять лишь из спонсора, технического лидера и заказчика. Если проект большой, в группу могут входить представители спонсорских групп, команды исполнения проекта, испытывающих воздействие бизнес-групп, групп руководства проектом и сообщества заказчика. Идея состоит в том, что, точно так же, как ни одна заинтересованная сторона в одиночку не должна заниматься оценкой или планированием проекта, так как она представляет только одну точку зрения и неизбежно страдает субъективностью, так и ни одно лицо в одиночку не должно оценивать применимость того или иного подхода, поскольку поле зрения одного лица точно также будет неизбежно ограниченным, а оценка субъективной.

Вместо этого ценность данного инструмента состоит в поощряемом обмене мнениями между всеми заинтересованными сторонами проекта. Даже если результаты указывают на целесообразность использования гибридного подхода, но заинтересованные стороны выступают за использование главным образом agile или предиктивного подхода, следует действовать в соответствии с достигнутым между заинтересованными сторонами соглашением. Данный инструмент служит лишь для диагностики высокого уровня, а окончательное решение должно опираться на мнение вовлеченных людей и поддерживаться ими.

Х3.3.2 ДАЙТЕ ОЦЕНКУ ПО ВОПРОСАМ В БАЛЛАХ ОТ 1 ДО 10

В составе группы обсудите и согласуйте (или найдите компромиссное решение) балл, который наиболее точно отражает субъективную оценку для данного вопроса. Хотя конкретные варианты ответа предлагаются только для начальной, средней и конечной позиций, соответствующих баллам 1, 5 и 10 полного диапазона ответов на вопрос, вполне допустимо (и даже желательно) использовать такие баллы, как 2, в случае ответа «почти, но не совсем 1», или 7 при ответе «где-то между 5 и 10». Стоит повторить, что оценка – это инструмент дискуссии: мнения будут иметь субъективный характер, и следует ожидать наличия в них неопределенности.

В случае, если группа не в состоянии договориться о балле, следует открыто и откровенно обсудить вопрос. Прежде чем предлагать компромиссные решения (например, использование среднеарифметических значений для вычисления балла или обозначение баллов ОУП синим знаком «Х» в отличие от оценки команды разработчиков, которая обозначается зеленым знаком «О»), стоит подумать о том, какова вероятность успешного завершения проекта, если участники не в состоянии прийти к общему мнению при выполнении простой процедуры оценки. Если при обсуждении этого вопроса расхождения во мнениях можно точно определить, то все отлично – метод работает, остается только достичь согласия. Таким же образом, если оценка указывает на предиктивный подход, но все хотят попробовать применить подход agile (или наоборот), то и в этом случае все прекрасно – надо лишь разобраться с имеющимися проблемами и обсудить, как будут потом решаться вопросы влияния подхода.

Х3.3.3 ИНТЕРПРЕТИРУЙТЕ РЕЗУЛЬТАТЫ

Пометьте ответы на вопросы на бланке диаграммы оценки применимости и соедините точки. Группирование результатов вокруг центра в зоне agile показывает приемлемость использования подхода agile в чистом виде.

Группирование большинства результатов в зоне гибридных подходов говорит о том, что лучше всего будет работать с определенным сочетанием подхода agile и предиктивного подхода. Однако возможно также, что будет достаточно использовать подход agile в сочетании с дополнительными мерами по снижению уровня риска, например, расширенной подготовкой и обучением сотрудников или ужесточением процесса подтверждения и строгости документов в случае проектов с высокой степенью критичности. Как вариант, предиктивный подход с некоторой работой по проверке концепции или дополнительными процессами может оказаться работающим.

Результаты, сосредоточенные в основном в предиктивной зоне, указывают на хорошую приемлемость предиктивного подхода. Как ранее говорилось в разделе X3.3.2 (шаг «Дайте оценку по вопросам»), этот диагностический инструмент призван положить начало содержательным дискуссиям с участвующими сторонами по вопросам выбора наиболее подходящего подхода для применения. Если подход, предложенный в результате использования этого инструмента, оказывается неприемлемым, можно использовать другой подход. Используйте результаты в качестве входов в процесс управления рисками, поскольку этот инструмент показывает несоответствия, которые потребуется учитывать.

X3.4 ВОПРОСЫ ПО ФИЛЬТРУ ПРИМЕНИМОСТИ

X3.4.1 КАТЕГОРИЯ: КУЛЬТУРА

X3.4.1.1 ПОДДЕРЖКА ПОДХОДА

Понимает ли главный спонсор суть использования подхода agile для данного проекта и согласен ли он поддержать данный проект? См. рис. X3-2.

Да	Частично	Нет
1	5	10
Оценка = _____		

Рис. X3-2. Оценка поддержки подхода

X3.4.1.2 ДОВЕРИЕ В КОМАНДЕ

Стоит изучить мнение спонсоров и представителей бизнеса, которые будут работать с командой. Имеют ли эти заинтересованные стороны уверенность, что команда в состоянии претворить их видение и потребности в успешный продукт или услугу при постоянной двусторонней поддержке и обратной связи между сторонами? См. рис. X3-3.

Да	Вероятно	Маловероятно
1	5	10
Оценка = _____		

Рис. ХЗ-3. Оценка доверия в команде

ХЗ.4.1.3 ПОЛНОМОЧИЯ КОМАНДЫ НА ПРИНЯТИЕ РЕШЕНИЙ

Будет ли команда иметь самостоятельность в принятии своих собственных решений по вопросам выполнения работы? См. рис. ХЗ-4.

Да	Вероятно	Маловероятно
1	5	10
Оценка = _____		

Рис. ХЗ-4. Оценка полномочий команды на принятие решений

ХЗ.4.2 КАТЕГОРИЯ: КОМАНДА

ХЗ.4.2.1 РАЗМЕР КОМАНДЫ

Какой размер имеет основная команда? Используйте следующую шкалу: 1–9 = 1, 10–20 = 2, 21–30 = 3, 31–45 = 4, 46–60 = 5, 61–80 = 6, 81–110 = 7, 111–150 = 8, 151–200 = 9, 201+ = 10. См. рис. ХЗ-5.

1	5	10
Оценка = _____		

Рис. ХЗ-5. Оценка размера команды

ХЗ.4.2.2 УРОВНИ ОПЫТА

Рассмотрение уровней опыта и навыков по ролям основной команды. Несмотря на то, что наличие опытных и неопытных специалистов в определенном соотношении по ролям команды является нормальным, для того, чтобы в работе по проектам agile не возникало проблем, лучше, когда каждую из ролей представляет хотя бы один опытный член команды. См. рис. ХЗ-6.

Да	Частично	Нет
1	5	10
<input type="radio"/> ————— <input type="radio"/>		
Оценка = _____		

Рис. ХЗ-6. Оценка уровня опыта

ХЗ.4.2.3 ДОСТУП К ЗАКАЗЧИКУ/БИЗНЕСУ

Будет ли у команды ежедневный доступ, по крайней мере, к одному представителю заказчика/бизнеса для уточнения возникающих вопросов и обратной связи? См. рис. ХЗ-7.

Да	Частично	Нет
1	5	10
<input type="radio"/> ————— <input type="radio"/>		
Оценка = _____		

Рис. ХЗ-7. Оценка доступа к заказчику/бизнесу

ХЗ.4.3 КАТЕГОРИЯ: ПРОЕКТ

ХЗ.4.3.1 ВЕРОЯТНОСТЬ ИЗМЕНЕНИЙ

Каков процент требований, которые могут с определенной степенью вероятности измениться или проявиться на протяжении месяца? См. рис. ХЗ-8.

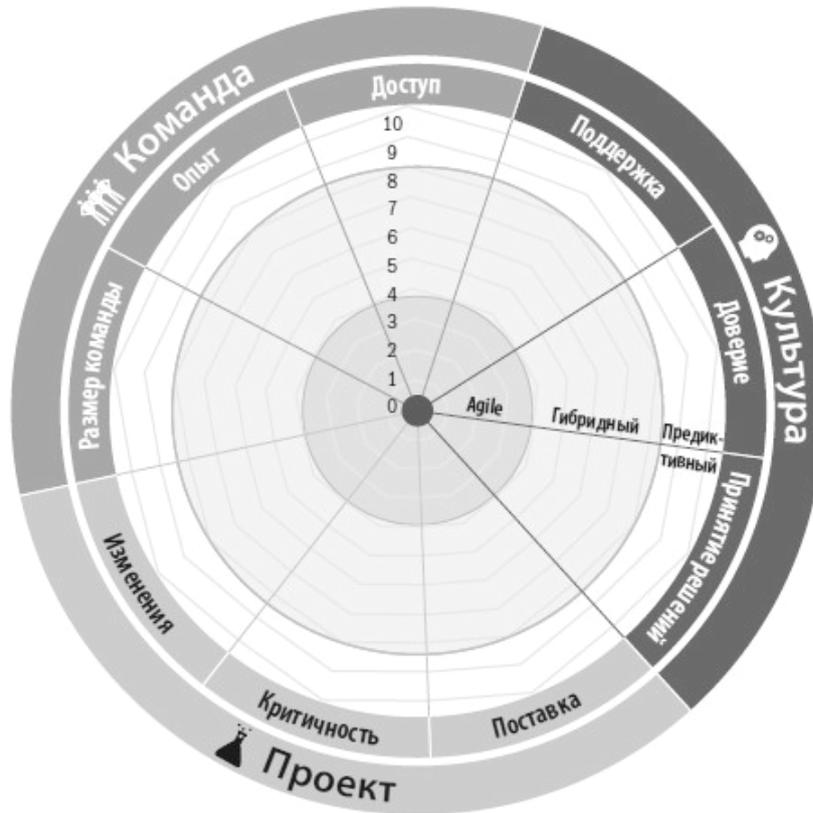


Рис. X3-11. Лепестковая диаграмма оценки применимости

X3.5.1 ПРАКТИЧЕСКИЕ ПРИМЕРЫ

Для иллюстрации того, как работает лепестковая диаграмма, рассмотрим два примера использования данной модели для оценки в баллах проектов различного типа. Первый является примером проекта интернет-аптеки (см. рис. X3-12), а второй (рис. X3-13) – военной системы обмена сообщениями. Эти два практических примера иллюстрируют некоторые различия, которые можно наблюдать при сравнении разных проектов. Концентрация значений в центре означает хорошую приемлемость использования подходов agile, разброс баллов по периферии говорит о том, что предиктивные подходы могут оказаться более целесообразными. Баллы некоторых проектов концентрируются в средней части, но имеют пики по одной или двум осям. Для таких проектов лучше всего подходит гибридный подход.

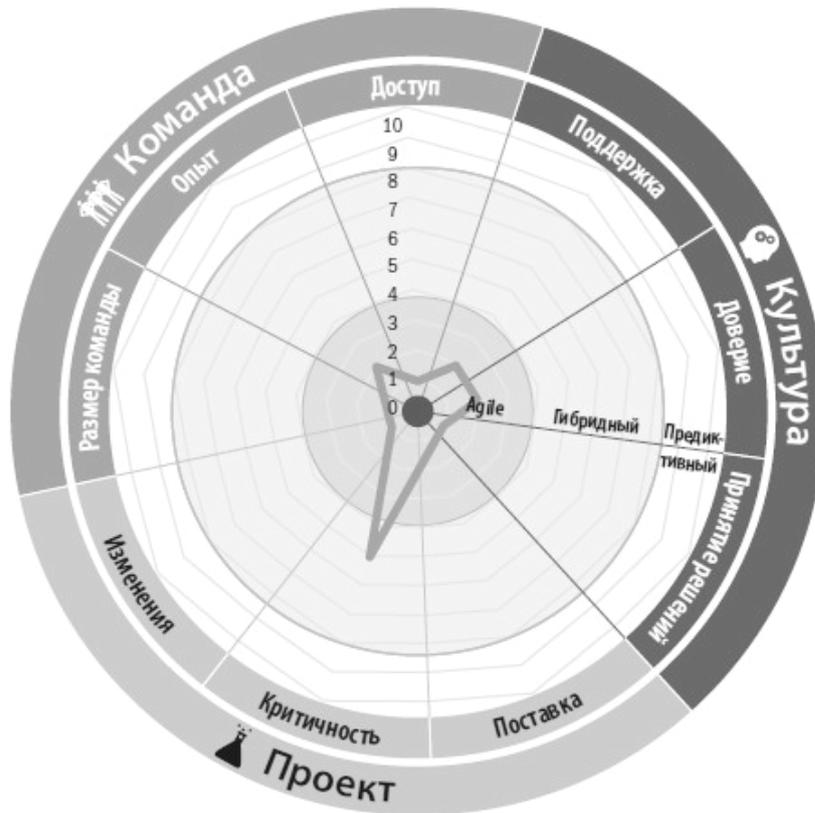


Рис. X3-12. Проект аптечного интернет-магазина

Х3.5.1.1 ПРИМЕР АПТЕЧНОГО ИНТЕРНЕТ-МАГАЗИНА

Проект был предпринят с целью создать интернет-аптеку для продажи более дешевых канадских рецептурных лекарств покупателям (преимущественно) в США. Продажа этих лекарств является предметом споров и в Канаде, и в США, и в результате для этой отрасли характерны быстрые изменения в государственно-правовом регулировании и острая конкуренция. Проект осуществлялся в условиях постоянно меняющихся требований, когда серьезные изменения происходили каждую неделю. Использовались очень короткие (двухдневные) итерации и еженедельные релизы, чтобы решить проблему высоких темпов изменений.

Как показано на рис. X3-12, высокие уровни поддержки и доверия очевидны для тех, кто работал, имея необходимые полномочия. Наглядный характер веб-сайта упростил задачу демонстрации новых инкрементов функциональности, однако критичность системы была довольно высокой с учетом существенных финансовых средств для фармацевтики, которые зависели от этого проекта. Как было сказано выше, имели место очень высокие темпы изменений, но небольшая, обладающая опытом команда хорошо справлялась с ними и имела удобный доступ к обладавшему необходимыми знаниями представителю бизнеса. Подход был очень успешным и в высшей степени agile.

Х3.5.1.2 ПРИМЕР ВОЕННОЙ СИСТЕМЫ ОБМЕНА СООБЩЕНИЯМИ

Сравните первый пример с крупным проектом по разработке военной системы обмена сообщениями, которая на момент проведения оценки, работала уже в течение 5 лет. См. рис. X3-13.

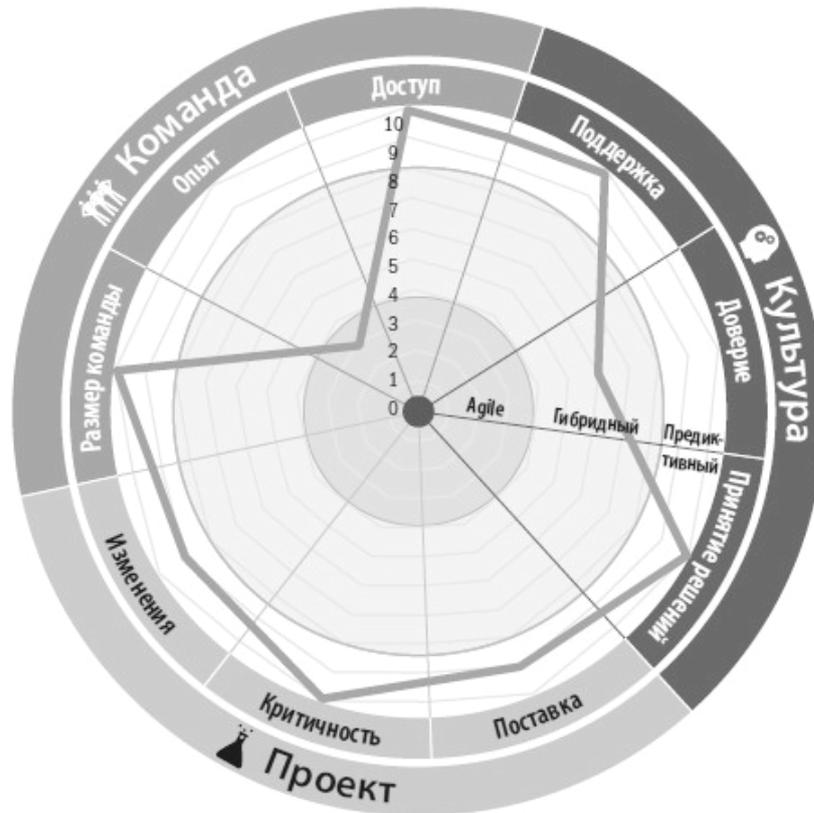


Рис. X3-13. Пример военной системы обмена сообщениями

Поддержка подхода agile отсутствовала, поскольку подход agile не рассматривался. Доверие поставщикам было смешанным, но в целом присутствовало. Решения принимались не на местном уровне, а комитетами по архитектуре и требованиям. Хотя имелась возможность тестировать элементы проекта инкрементно (по частям) в лабораторных условиях, их невозможно было собрать вместе для полномасштабной демонстрации функциональности. Потенциально существовал риск для жизни многих людей, поэтому уровень критичности был очень высоким. Требования были фиксированы, поскольку их изменения влияли на работу очень большого числа организаций-субподрядчиков.

Проект был крупным, с участием более 300 человек только от одного поставщика, но в исполнении каждой роли участвовало много опытных специалистов-практиков. И наконец, доступ к бизнесу/заказчику был невозможен, но можно было связаться со специалистами по договорам, которым можно было задать вопросы по спецификации, и они обычно давали ответы или задавали уточняющие вопросы в течение 10 дней. Часть проекта можно было выделить и осуществлять как проекты agile, но в центре работы был один единственный крупный проект.

X3.6 ВЫВОДЫ

Фильтры применимости подхода agile являются полезным инструментом для определения потенциальных соответствий и несоответствий для использования подхода agile. Их следует использовать не в качестве конкретных триггеров включения или исключения, а как темы для объективного обсуждения со всеми заинтересованными сторонами.

ССЫЛКИ

[1] *Agile-манифест разработки программного обеспечения*. (2001). Взято с веб-сайта <http://agilemanifesto.org/>.

[2] Project Management Institute. 2013. *Управление изменениями в организациях: практическое руководство*. Newtown Square, PA: Автор.

[3] Project Management Institute. 2017. *Руководство к своду знаний по управлению проектом (Руководство PMBOK®) – Шестое издание*. Newtown Square, PA: Автор.

[4] Project Management Institute. 2013. *Дополнение к Руководству PMBOK® по разработке программного обеспечения – пятое издание*. Newtown Square, PA: Автор.

Список использованной литературы

Ниже приводится список рекомендованной дополнительной литературы, разделенный на подразделы и (или) темы:

РАЗДЕЛ 2 – ВВЕДЕНИЕ В AGILE

Briggs, Sara. «Agile Based Learning: What Is It and How Can It Change Education?» *Opencolleges.edu.au* 22 февраля 2014 г., по данным веб-сайта <http://www.opencolleges.edu.au/informed/features/agile-based-learning-what-is-it-and-how-can-it-change-education/>.

Manifesto for Agile Software Development, 2001, <http://agilemanifesto.org/>.

Peha, Steve. «Agile Schools: How Technology Saves Education (Just Not the Way We Thought it Would).» *InfoQ*. 28 июня 2011 г., по данным веб-сайта <https://www.infoq.com/articles/agile-schools-education>.

Principles behind the Agile Manifesto, 2001, <http://agilemanifesto.org/principles.html>.

Rothman, Johanna. 2007. *Manage It! Your Guide to Modern, Pragmatic Project Management*. Raleigh: Pragmatic Bookshelf.

Sidky, Ahmed (Keynote). 2015. <https://www.slideshare.net/AgileNZ/ahmed-sidky-keynote-agilenz>.

Stacey Complexity Model. 2016. <http://www.scrum-tips.com/2016/02/17/stacey-complexity-model/>.

РАЗДЕЛ 3 – ВЫБОР ЖИЗНЕННОГО ЦИКЛА

«Agile Modeling (AM) Home Page: Effective Practices for Modeling and Documentation,» Agile Modeling, (без даты), <http://www.agilemodeling.com/>.

Anderson, David, and Andy Carmichael. 2016. *Essential Kanban Condensed*. Seattle: Blue Hole Press.

Anderson, David. 2010. *Kanban: Successful Evolutionary Change for Your Technology Business*. Seattle: Blue Hole Press.

Benson, Jim, and Tonianne DeMaria Barry. 2011. *Personal Kanban: Mapping Work / Navigating Life*. Seattle: Modus Cooperandi Press.

Burrows, Mike. 2014. *Kanban from the Inside: Understand the Kanban Method, connect it to what you already know, introduce it with impact*. Seattle: Blue Hole Press.

Domain Driven Design Community. 2016. <http://dddcommunity.org/>.

Gothelf, Jeff, and Josh Seiden. 2016. *Lean UX: Designing Great Products with Agile Teams*. Sebastopol: O'Reilly Media.

Hammarberg, Marcus, and Joakim Sunden. 2014. *Kanban in Action*. Shelter Island: Manning Publications.

«Kanban,» *Wikipedia*, последние изменения внесены 4 мая 2017 г., по данным 22 ноября 2016 г. с веб-сайта <https://en.wikipedia.org/wiki/Kanban>.

«Kanban (development),» *Wikipedia*, последние изменения внесены 4 мая 2017 г., по данным 29 ноября 2016 г. с веб-сайта [https://en.wikipedia.org/wiki/Kanban_\(development\)](https://en.wikipedia.org/wiki/Kanban_(development)).

Larsen, Diana, and Ainsley Nies. 2016. *Liftoff: Start and Sustain Successful Agile Teams*. Raleigh: Pragmatic Bookshelf.

«Learning Kanban,» *Leankit*, (без даты), <https://leankit.com/learn/learning-kanban/>.

Leopold, Klaus, and Siegrfried Kaltenecker. 2015. *Kanban Change Leadership: Creating a Culture of Continuous Improvement*. Hoboken: Wiley.

«Make a big impact with software products and projects!» *Impact Mapping*, (без даты), <https://www.impactmapping.org/>.

Patton, Jeff, and Peter Economy. 2014. *User Story Mapping: Discover the Whole Story, Build the Right Product*. Sebastopol: O'Reilly Media.

Reinertsen, Donald. 2009. *The Principles of Product Development Flow: Second Generation Lean Product Development*. Redondo Beach: Celeritas Publishing.

Rothman, Johanna. «Dispersed vs. Distributed Teams,» *Rothman Consulting Group, Inc.*, 25 октября 2010 г., <http://www.jrothman.com/mpd/2010/10/dispersed-vs-distributed-teams/>.

Schwaber, Ken, and Jeff Sutherland. «The Scrum Guide™,» *Scrum.org*, июль 2016 г. <http://www.scrumguides.org/scrum-guide.html> and <http://www.scrumguides.org/docs/scrumguide/v2016/2016-Scrum-Guide-US.pdf#zoom=100>.

Skarin, Mattias. 2015. *Real-World Kanban: Do Less, Accomplish More with Lean Thinking*. Raleigh: Pragmatic Bookshelf.

«The High Cost of Multitasking: 40 % of Productivity Lost by Task Switching,» *Wrike.com*, 24 сентября 2015 г., <https://www.wrike.com/blog/high-cost-of-multitasking-for-productivity/>.

Wells, Don. «Extreme Programming: A Gentle Introduction,» *Extreme Programming*, 8 октября 2013 г., <http://www.extremeprogramming.org/>.

РАЗДЕЛ 4 – РЕАЛИЗАЦИЯ AGILE:

Amabile, Teresa, and Steven Kramer. 2011. *The Progress Principle: Using Small Wins to Ignite Joy, Engagement, and Creativity at Work*. Boston: Harvard Business Review Press.

«Early Warning Signs of Project Trouble – Cheat Sheet, 2017, <https://agilevideos.com/wp-content/uploads/2017/02/WarningSignsOfProjectTrouble-CheatSheet.pdf>.

Dweck, Carol. 2006. *Mindset: The New Psychology of Success*. New York: Penguin Random House.

Kaner, Sam. *Facilitator's Guide to Participatory Decision-Making*. 3rd ed. 2014. San Francisco: Jossey-Bass.

Keith, Kent. *The Case for Servant Leadership*. 2008. Westfield: Greenleaf Center for Servant Leadership.

Rothman, Johanna. 2016. *Agile and Lean Program Management: Scaling Collaboration Across the Organization*. Victoria, British Columbia: Practical Ink.

Rothman, Johanna. „Dispersed vs. Distributed Teams,“ *Rothman Consulting Group, Inc.*, 25 октября 2010 г., <http://www.jrothman.com/mpd/2010/10/dispersed-vs-distributed-teams/>.

Rothman, Johanna. 2007. *Manage It! Your Guide to Modern, Pragmatic Project Management*. Raleigh: Pragmatic Bookshelf.

Rothman, Johanna. 2016. *Manage Your Project Portfolio: Increase Your Capacity and Finish More Projects*. Raleigh: Pragmatic Bookshelf.

Schwaber, Ken, and Jeff Sutherland. „The Scrum Guide™,“ *Scrum.org*, июль 2016 г., <http://www.scrumguides.org/scrum-guide.html> и <http://www.scrumguides.org/docs/scrumguide/v2016/2016-Scrum-Guide-US.pdf#zoom=100>.

Sinek, Simon. 2011. *Start with Why: How Great Leaders Inspire Everyone to Take Action*. New York: Portfolio, Penguin Random House.

„The High Cost of Multitasking: 40 % of Productivity Lost by Task Switching,“ *Wrike.com*, 24 сентября 2015 г., <https://www.wrike.com/blog/high-cost-of-multitasking-for-productivity/>.

EXPERIENCE REPORTS:

„Experience Reports,“ *Agile Alliance*, (без даты),
<https://www.agilealliance.org/resources/experience-reports/>.

БЛАГОПОЛУЧИЕ ПРОЕКТА И КОМАНДЫ:

„Early Warning Signs of Project Trouble – Cheat Sheet.“ 2017.
<https://agilevideos.com/wp-content/uploads/2017/02/WarningSignsOfProjectTrouble-CheatSheet.pdf>.

„TeamHealth Radar – Summary View,“ *Agilehealth*. 2014.
<http://agilityhealthradar.com/wp-content/uploads/2014/11/bigradar.gif>.

ЭФФЕКТИВНОСТЬ РАСХОДОВАНИЯ РЕСУРСОВ:

Modig, Niklas, and Pär Åhlström. 2015. *This is Lean: Resolving the Efficiency Paradox*. London: Rheologica Publishing.

Rothman, Johanna. „Resource Efficiency vs. Flow Efficiency, Part 5: How Flow Changes Everything,“ *Rothman Consulting Group, Inc.*, 20 сентября 2015 г.,
<http://www.jrothman.com/mpd/agile/2015/09/resource-efciency-vs-fow-efciency-part-5-how-fow-changes-everything/>.

SCALING:

Disciplined Agile 2.X – A Process Decision Framework. 2016.
<http://www.disciplinedagiledelivery.com/>.

Kniberg, Henrik. „Scaling Agile @ Spotify with Tribes, Squads, Chapters & Guilds,“ *Crisp*, 14 ноября 2012 г., <http://blog.crisp.se/2012/11/14/henrikkniberg/scaling-agile-at-spotify>.

„Overview – Large Scale Scrum,“ LeSS. 2016. <http://less.works/>.

„SAFe® for Lean Software and System Engineering,“ *SAFe®*. 2016.
<http://www.scaledagileframework.com/>.

НАВЫКИ:

Beck, Kent. *Paint Drip People*, 4 августа 2016 г.,
<https://www.facebook.com/notes/kent-beck/paint-drip-people/122670000696195/>.

„Generalizing Specialists: Improving Your IT Career Skills,“ *Agile Modeling*, (без даты),
<http://www.agilemodeling.com/essays/generalizingSpecialists.htm>.

Hunter, Brittany. „Of Software Designers & Broken Combs,“ *Atomic Object*, 27 июня 2013 г.,
<https://spin.atomicobject.com/2013/06/27/broken-comb-people/>.

РАЗДЕЛ 5 – РЕАЛИЗАЦИЯ AGILE: ПОСТАВКА В СРЕДЕ AGILE

Larsen, Diana, and Ainsley Nies. 2016. *liftoff: Start and Sustain Successful Agile Teams*. Raleigh: Pragmatic Bookshelf.

РЕТРОСПЕКТИВА:

Derby, Esther, and Diana Larsen. 2006. *Agile Retrospectives: Making Good Teams Great*. Raleigh: Pragmatic Bookshelf.

Gonçalves, Luis, and Ben Linders. 2015. *Getting Value out of Agile Retrospectives: A Toolbox of Retrospective Exercises*. Victoria, British Columbia: Leanpub.

БЭКЛОГ:

Adzic, Gojko, Marjory Bissett, and Tom Poppendieck. 2012. *Impact Mapping: Making a Big Impact with Software Products and Projects*. Woking, Surrey: Provoking Thoughts.

Patton, Jeff, and Peter Economy. 2014. *User Story Mapping: Discover the Whole Story, Build the Right Product*. Sebastopol: O'Reilly Media.

Rothman, Johanna. „We Need Planning; Do We Need Estimation?“ *Rothman Consulting Group, Inc.*, 21 января 2015 г.,

<http://www.jrothman.com/mpd/project-management/2015/01/we-need-planning-do-we-need-estimation/>.

ЛЕТУЧКИ:

Brodzinski, Pawel. „Effective Standups around Kanban Board,“ *Brodzinski.com*, 30 декабря 2011 г., <http://brodzinski.com/2011/12/effective-standups.html>.

Fowler, Martin. „It’s Not Just Standing Up: Patterns for Daily Standup Meetings,“ *Martinfowler.com*, 21 февраля 2016 г., <http://martinfowler.com/articles/itsNotJustStandingUp.html>.

Hefley, Chris. „How to Run Effective Standups and Retrospectives,“ *Leankit*, 15 сентября 2014 г., <https://leankit.com/blog/2014/09/run-effective-standups-retrospectives/>.

ОСВОЕННЫЙ ОБЪЕМ:

Griffiths, Mike. „A Better S Curve and Simplified EVM,“ *Leading Answers*, 6 июня 2008 г., http://leadinganswers.typepad.com/leading_answers/2008/06/a-better-s-curve-and-simplified-evm.html.

РАЗДЕЛ 6 – ОРГАНИЗАЦИОННЫЕ СООБРАЖЕНИЯ ДЛЯ ПРОЕКТОВ AGILE

Bankston, Arlen, and Sanjiv Augustine. *Agile Team Performance Management: Realizing the Human Potential of Teams*, 14 июня 2010 г., www.lithespeed.com/transfer/Agile-Performance-Management.pptx.

Browder, Justin, and Brian Schoeff. *Perfect Strangers: How Project Managers and Developers Relate and Succeed*. CreateSpace Independent Publishing Platform, 2016, <https://www.createspace.com/>.

Griffiths, Mike. „Agile Talent Management,“ *Leading Answers*, 14 октября 2015 г., http://leadinganswers.typepad.com/leading_answers/2015/10/agile-talent-management.html.

Kohn, Alfie. 1999. *Punished by Rewards: The Trouble with Gold Stars, Incentive Plans, A’s, Praise, and Other Bribes*. New York: Mariner Books.

Mar, Kane. „How to do Agile Performance Reviews,“ *Scrumology*, (без даты), <https://scrumology.com/how-to-do-agile-performance-reviews/>.

McChrystal, Stanley, Tatum Collins, David Silverman, and Chris Fussell. 2015. *Team of Teams: New Rules of Engagement for a Complex World*. New York: Portfolio, Penguin Random House.

Pink, Daniel. 2011. *Drive: The Surprising Truth About What Motivates Us*. New York: Riverhead Books.

РАЗДЕЛ 7 – ПРИЗЫВ К ДЕЙСТВИЮ (ИНСПЕКЦИЯ БЕЗ АДАПТАЦИИ БЕСПОЛЕЗНА)

Dennis, Pascal. 2006. *Getting the Right Things Done: A Leader’s Guide to Planning and Execution*. Cambridge: Lean Enterprise Institute.

Griffiths, Mike. „Introducing Agile Methods: Mistakes to Avoid – Part 3,“ *Leading Answers*, 15 марта 2007 г., http://leadinganswers.typepad.com/leading_answers/2007/03/introducing_agi_2.html.

Little, Jason. *Lean Change Management: Innovative Practices for Managing Organizational Change*. Happy Melly Express, 2014, <http://www.happymelly.com/category/hm-express/>.

Rising, Linda, and Mary Lynne Manns. 2004. *Fearless Change: Patterns for Introducing New Ideas*. Upper Saddle River: Addison-Wesley Professional.

„The IDEAL Model,“ *Software Engineering Institute, Carnegie Mellon*, 2006, <http://www.sei.cmu.edu/library/assets/idealmodel.pdf>.

ПРИЛОЖЕНИЕ А1 – КАРТИРОВАНИЕ РУКОВОДСТВА РМВОК®

Larsen, Diana and Ainsley Nies. 2016. *liftoff: Start and Sustain Successful Agile Teams*. Raleigh: Pragmatic Bookshelf.

ПРИЛОЖЕНИЕ А2 – КАРТИРОВАНИЕ AGILE-МАНИФЕСТА

Manifesto for Agile Software Development, 2001, <http://agilemanifesto.org/>.

Principles behind the Agile Manifesto, 2001, <http://agilemanifesto.org/principles.html>.

ПРИЛОЖЕНИЕ А3 – ОБЗОР ФРЕЙМВОРКА AGILE И БЕРЕЖЛИВОГО ФРЕЙМВОРКА

Agile Business Consortium, 2014, <https://www.agilebusiness.org/what-is-dsdm>.

Ambler, Scott. «The Agile Unified Process,» *Ambyssoft*, 13 мая 2006 г., <http://www.ambyssoft.com/unifiedprocess/agileUP.html>.

Anderson, David. 2010. *Kanban: Successful Evolutionary Change for Your Technology Business*. Seattle: Blue Hole Press.

Beedle, Mike. *Enterprise Scrum: Executive Summary: Business Agility for the 21st Century*, 7 января 2017 г., <http://www.enterprisescrum.com/enterprise-scrum/>.

Cockburn, Alistair. 2004. *Crystal Clear: A Human-Powered Methodology for Small Teams*. Upper Saddle River: Pearson Education.

Cockburn, Alistair. «Crystal Methodologies,» *alistair.cockburn.us*, 28 марта 2014 г., <http://alistair.cockburn.us/Crystal+methodologies>.

Disciplined Agile 2.X – A Process Decision Framework, 2016, <http://www.disciplinedagiledelivery.com/>.

Joint MIT-PMI-INCOSE Community of Practice on Lean in Program Management. 2012. *The Guide to Lean Enablers for Managing Engineering Programs*. Newtown Square, PA: Автор.

«Kanban,» *Wikipedia*, последние изменения внесены 4 мая 2017 г., по данным 22 ноября 2016 г. с веб-сайта <https://en.wikipedia.org/wiki/Kanban>.

«Kanban (development),» *Wikipedia*, последние изменения внесены 4 мая 2017 г., по данным 29 ноября 2016 г. с вебсайта [https://en.wikipedia.org/wiki/Kanban_\(development\)](https://en.wikipedia.org/wiki/Kanban_(development)).

Reddy, Ajay, and Jack Speranza. 2015. *The Scrumban [R]Evolution: Getting the Most Out of Agile, Scrum, and Lean Kanban*. Boston: Addison-Wesley Professional.

«Overview – Large Scale Scrum,» *LeSS*, 2016, <http://less.works/>.

«SAFe® for Lean Software and System Engineering,» *SAFe®*, 2016, <http://www.scaledagileframework.com/>.

Schwaber, Ken, and Jeff Sutherland. «The Scrum Guide™,» *Scrum.org*, июль 2016 г., <http://www.scrumguides.org/scrum-guide.html> и

<http://www.scrumguides.org/docs/scrumguide/v2016/2016-Scrum-Guide-US.pdf#zoom=100>.

«Scrum of Scrums,» *Agile Alliance*, (без даты), <https://www.agilealliance.org/glossary/scrum-of-scrums/>.

«Scrumban,» *Wikipedia*, 2 марта 2017 г., <https://en.wikipedia.org/wiki/Scrumban>.

«State of Agile Report: Agile Trends,» *Version One*, 2017, <http://stateofagile.versionone.com/>.

Sutherland Jeff. «Agile Can Scale: Inventing and Reinventing SCRUM in Five Companies.» *Cutter IT Journal* 14, no. 12 (2001): 5-11 <http://www.controlchaos.com/storage/scrum-articles/Sutherland200111proof.pdf>.

«The 2015 State of Agile Development,» *Scrum Alliance®*, 2015, <https://www.forrester.com/report/The+2015+State+Of+Agile+Development/-/E-RES122910>.

Wells, Don. «Extreme Programming: A Gentle Introduction,» *Extreme Programming*, 8 октября 2013 г., <http://www.extremeprogramming.org/>.

Why Scrum? State of Scrum Report, 2016, <https://www.scrumalliance.org/why-scrum/state-of-scrum-report/2016-state-of-scrum>.

ПРИЛОЖЕНИЕ Х2 – СВОЙСТВА, ВЛИЯЮЩИЕ НА АДАПТАЦИЮ

Griffiths, Mike. «Agile Suitability Filters,» *Leading Answers*, 2007, http://leadinganswers.typepad.com/leading_answers/files/agile_suitability_filters.pdf.

Jeffries, Ron. «We Tried Baseball and It Didn't Work,» *ronjeffries.com*, 2 мая 2006 г., <http://ronjeffries.com/xprog/articles/jatbaseball/>.

Rothman, Johanna. «One Experimental Possibility: Self-Organization from Component Teams to Feature Teams,» *Rothman Consulting Group, Inc.*, 23 сентября 2014 г., <http://www.jrothman.com/mpd/agile/2014/09/one-experimental-possibility-self-organization-from-component-teams-to-feature-teams/>.

Глоссарий

1. СОКРАЩЕНИЯ

ATDD – разработка на основе приемочных тестов

BDD – разработка на основе поведения

BRD – документы с бизнес-требованиями

DA – упорядоченный agile

DoD – критерии выполнения

DoR – критерии готовности

DSDM – динамический метод разработки систем

Evo – эволюционная поставка ценности

LeSS – крупномасштабный скрам

LSD – бережливая разработка программного обеспечения

PDCA – планирование-действие-проверка-корректировка

ОУП – офис управления проектами

ROI – окупаемость инвестиций

RUP – рациональный унифицированный процесс

SAFe® – Scaled Agile Framework® (масштабированный agile-фреймворк)

SBE – спецификация на примерах

XP – экстремальное программирование

2. ОПРЕДЕЛЕНИЯ

A3 / A3. Способ мышления и процесс систематического решения проблем, который предусматривает изложение соответствующей информации на одном листе бумаги формата A3.

Agile (гибкая разработка) / Agile. Термин, используемый для описания образа мышления, основанного на ценностях и принципах, изложенных в Agile-манифесте (манифесте гибкой разработки).

Agile-коуч / Agile Coach. Человек, имеющий знания и опыт в agile, который может обучать, выполнять функции наставника и руководителя в организациях и командах в процессе их трансформации.

Agile-манифест / Agile Manifesto. Исходный официальный документ, содержащий описание ценностей и принципов agile.

Agile-практик / Agile Practitioner. Человек, разделяющий философию agile, который сотрудничает с имеющими такой же образ мышления коллегами в кроссфункциональных командах. Именуется также эджайлистом (agilist).

DevOps (интеграция разработки и эксплуатации) / DevOps. Набор практик для обеспечения бесперебойной поставки результатов путем сотрудничества между разработчиками и эксплуатационным персоналом.

IDEAL / IDEAL. Модель организационного совершенствования, получившая свое название по пяти первым буквам слов, обозначающих ее пять составных фаз: инициирование (initiating), диагностика (diagnosing), становление (establishing), действие (acting) и обучение (learning).

I-образный / I-shaped. Описание человека с глубокой специализацией в одной сфере и отсутствием интересов или навыков в других областях, необходимых команде. См. также *T-образный* и *Сломанный гребешок*.

UX-дизайн / UX Design. Процесс улучшения удобства пользователя за счет уделения особого внимания улучшению удобства использования и доступности, которые необходимы для взаимодействия пользователя с продуктом.

Автоматизированный анализ качества кода / Automated Code Quality Analysis. Тестирование базы кода по сценарию с целью выявления ошибок и уязвимостей.

Антишаблон / Anti-Pattern. Известный неэффективный порядок работы, применять который не рекомендуется.

Бережливая разработка программного обеспечения (LSD) / Lean Software Development (LSD). Бережливая разработка программного обеспечения является адаптацией принципов и методик бережливого производства к сфере разработки программного обеспечения, которая базируется на наборе принципов и методик для обеспечения качества, оперативности и максимального соответствия требованиям заказчика.

Блокер / Blocker. См. *Препятствие*.

Бэклог продукта / Product Backlog. Упорядоченный список ориентированных на пользователя требований, который ведет команда в отношении продукта.

Бэклог спринта / Sprint Backlog. Перечень задач, выбранных скрам-командой для выполнения в скрам-спринте.

Бэклог / Backlog. См. *Бэклог продукта*.

Владелец продукта / Product Owner. Лицо, которое отвечает за обеспечение максимальной ценности продукта, а также несет конечную ответственность и отчитывается за находящийся в разработке конечный продукт. См. также *Руководитель запросов на обслуживание*.

Временные рамки / Timebox. Фиксированный период времени, к примеру, 1 неделя, 1 двухнедельный период, 3 недели или 1 месяц. См. также *Итерация*.

Гибридный подход / Hybrid Approach. Сочетание двух и более agile и не-agile элементов, имеющее конечный результат не agile.

Двойной цикл обучения / Double-Loop Learning. Процесс критического рассмотрения базовых ценностей и предположений, чтобы добиться лучшей оценки основных причин и определить лучшие контрмеры, вместо того чтобы сосредотачиваться только на симптомах.

Диаграмма выгорания / Burnup Chart. Графическое представление завершенной работы относительно выпуска продукта.

Диаграмма сгорания / Burndown Chart. Графическое представление объема оставшейся работы в сопоставлении с ограничениями времени на исполнение.

Динамический метод разработки систем (DSDM) / Dynamics Systems Development Model (DSDM). Фреймворк поставки результатов agile-проектов.

Документы с бизнес-требованиями (BRD) / Business Requirement Documents (BRD). Перечень всех требований для конкретного проекта.

Доска «канбан» / Kanban Board. Средство визуализации, которое позволяет совершенствовать рабочий процесс путем наглядного представления узких мест и объемов работы.

Доска «скрам» / Scrum Board. Информационная доска, которая используется для управления бэклогами и спринтами продукта и служит для отображения рабочего процесса и его узких мест.

«Дымовое» тестирование / Smoke Testing. Практика использования упрощенного набора тестов с целью убедиться, что наиболее важные функции разрабатываемой системы работают надлежащим образом.

Ежедневный скрам / Daily Scrum. Краткое ежедневное собрание с целью развития сотрудничества, в ходе которого команда анализирует результаты работы за прошлый день, определяет планы на текущий день и выявляет возникшие или предполагаемые в будущем препятствия. Также известен как ежедневная летучка.

Жизненный цикл agile / Agile Life Cycle. Подход, который является итеративным и, в то же время, инкрементным и предназначен для доработки перечня задач с целью частой поставки.

Жизненный цикл / Life Cycle. Процесс, в ходе которого создается замысел продукта, осуществляется его создание и ввод в эксплуатацию.

Изолированная организация / Siloed Organization. Организация, имеющая структуру, которая позволяет ей выполнять лишь некоторую часть работ определенной категории, необходимых для создания ценности для заказчиков. Для сравнения см. *Поток создания ценности*.

Инкремент / Increment. Функциональный, протестированный и принятый поставляемый результат, который является промежуточной частью конечного результата проекта.

Инкрементный жизненный цикл / Incremental Life Cycle. Подход, дающий конечные поставляемые результаты, которые клиент может использовать сразу же.

Информационная доска / Information Radiator. Наглядный физический дисплей с информацией для остальной части организации, обеспечивающий возможность обмена самыми последними знаниями без необходимости отвлекать команду.

Итеративный жизненный цикл / Iterative Life Cycle. Подход, позволяющий использовать обратную связь с целью доработки и уточнения незавершенной работы.

Итерация / Iteration. Ограниченный временными рамками цикл разработки продукта или поставляемого результата, в ходе которого выполняется вся работа, необходимая для создания ценности.

Каденция / Cadence. Цикличность выполнения. См. также *Временные рамки*.

Картирование воздействия / Impact Mapping. Метод стратегического планирования, служащий организации дорожной картой в процессе разработки новых продуктов.

Картирование пользовательских историй / User Story Mapping. Метод визуализации, служащий для организации работы в полезную модель с целью помочь определить обладающие высокой ценностью характеристики, которые должны быть созданы в последующем, выявить упущения в бэклоге и результативно спланировать выпуски продукта для поставки ценностей пользователям.

Картирование потока ценности / Value Stream Mapping. Метод бережливого производства, используемый для документирования, анализа и совершенствования потока информации или материалов, требуемых для производства продукта или услуги для заказчика.

Критерии выполнения (DoD) / Definition of Done (DoD). Контрольный список всех критериев, которые команде необходимо выполнить, чтобы поставляемый результат можно было считать готовым для использования заказчиком.

Критерии готовности (DoR) / Definition of Ready (DoR). Контрольный список команды по ориентированному на пользователя требованию, который содержит всю информацию, необходимую команде для начала работы по исполнению этого требования.

Кроссфункциональная команда / Cross-Functional Team. Команда, состоящая из практиков, обладающих всеми необходимыми навыками для поставки имеющих ценность инкрементных частей продукта.

Крупномасштабный скрам (LeSS) / Large-Scale Scrum (LeSS). Крупномасштабный скрам является фреймворком разработки продукта, который расширяет применение метода скрам за счет масштабируемых руководств, при этом сохраняя исходные цели метода скрам.

Мастер процесса / Flowmaster. Коуч для команды и руководитель запросов на обслуживание, работающий в условиях непрерывного потока или системы «канбан». Является эквивалентом *Скрам-мастера*.

Масштабированный agile-фреймворк (SAFe®) / Scaled Agile Framework (SAFe®). База знаний, состоящая из интегрированных моделей, используемых для бережливой agile-разработки в масштабе предприятия.

Мероприятия «кайдзен» / Kaizen Events. Мероприятия, направленные на совершенствование системы.

Метод «канбан» / Kanban Method. Agile-метод, разработанный на основе оригинальной системы контроля инвентарных запасов канбан, и используемый, как правило, в сфере умственного труда.

Моббинг / Mobbing. Метод, в рамках которого несколько членов команды одновременно фокусируются и ведут совместную работу над определенной рабочей задачей.

Непрерывная интеграция / Continuous Integration. Практика, в рамках которой производится периодическая интеграция и валидация всех рабочих продуктов команды друг с другом.

Непрерывная поставка / Continuous Delivery. Практика поставки имеющих определенные свойства частей продукта непосредственно заказчику, часто с использованием небольших пакетов работы и средств автоматизации.

Образ мышления agile / Agile Mindset. Образ мышления и поведения, в основе которого лежат четыре ценности и двенадцать принципов Agile-манифеста.

Обслуживающее лидерство / Servant Leadership. Практика лидерства, состоящая в служении команде путем фокусирования на понимании ее нужд и поиске средств для их удовлетворения, а также развития команды для достижения максимальной эффективности и результативности ее работы.

Одинарный цикл обучения / Single-Loop Learning. Практика, когда решение проблем осуществляется с использованием конкретных, предварительно определенных методов без их критического анализа с учетом опыта.

Организационное искажение / Organizational Bias. Представленные в виде шкалы предпочтения организации, служащие для сопоставления следующих ключевых ценностей: исследования против исполнения, скорость против стабильности, количество против качества, гибкость против предсказуемости.

Относительная единица / Story Point. Безразмерная оценка, используемая в методике сравнительной оценки пользовательской истории.

Офис управления проектами (ОУП) / Project Management Office (PMO). Управленческая структурная единица, стандартизирующая процессы руководства проектами и способствующая обмену ресурсами, методологиями, инструментами и методами.

Парное программирование / Pair Programming. Работа в парах, основной задачей которой является программирование.

Планирование методом набегающей волны / Rolling Wave Planning. Метод итеративного планирования, в соответствии с которым подробно планируется работа, которую надо будет выполнить в ближайшей перспективе, а работа будущих периодов планируется с меньшей степенью детализации.

Планирование спринта / Sprint Planning. Совместное скрам-мероприятие, в рамках которого скрам-команда планирует работу в текущем спринте.

Планирование-действие-проверка-корректировка (PDCA) / Plan – Do – Check – Act (PDCA). Итеративный управленческий метод, используемый в организациях для улучшения контроля и непрерывного совершенствования процессов и продуктов.

Поворотная точка / Pivot. Плановая поправка курса, предназначенная для проверки новой гипотезы о продукте или стратегии.

Подтек краски / Paint Drip. См. *Сломанный гребешок*.

Подход на основе плана / Plan-Driven Approach. См. *Предиктивный подход*.

Подходящий для использования / Fit for Use. Характеристика продукта, говорящая о том, что его можно использовать в существующей форме для получения результата по предусмотренному назначению.

Подходящий по назначению / Fit for Purpose. Характеристика продукта, говорящая о его соответствии предусмотренному назначению.

Пользовательская история / User Story. Краткое описание поставляемой ценности для конкретного пользователя. Служит предпосылкой для обсуждения с целью уточнения деталей.

Последовательное уточнение / Progressive Elaboration. Итеративный процесс повышения уровня детализации плана управления проектом по мере получения большего объема информации и более точных оценок.

Поток ценности / Value Stream. Организационный структурный элемент, направленный на процесс создания ценности для заказчика при поставке конкретных продуктов или услуг.

Предиктивный подход / Predictive Approach. Подход к управлению работами, в соответствии с которым на протяжении жизненного цикла проекта применяется рабочий план и осуществляется управление им.

Препятствие / Impediment. Препятствие, мешающее команде достичь поставленных целей. Также называется «блокером».

Принципы agile / Agile Principles. Предусмотренные в Agile-манифесте двенадцать принципов поставки agile-проектов.

Работа в парах / Pair Work. Метод объединения двух членов команды в паре для одновременной работы над одним и тем же рабочим заданием.

Разработка на основе поведения (BDD) / Behavior-Driven Development (BDD). Практика проектирования и валидации системы, в которой применяются принцип «сначала тесты» и скрипты на обычном языке общения.

Разработка на основе приемочных тестов (ATDD) / Acceptance Test-Driven Development (ATDD). Метод совместной разработки критериев приемочных тестов, используемых для создания приемочных тестов до начала поставки.

Разработка на основе тестов / Test-Driven Development. Метод, когда тесты определяются до начала работы, с тем чтобы проверка текущей работы осуществлялась непрерывно, что позволяет добиться производства работ в соответствии с философией нулевого количества дефектов.

Разработка на основе функциональности / Feature-Driven Development. Упрощенный agile-метод разработки программного обеспечения на основе набора функциональных характеристик, важных для клиента.

Ретроспектива / Retrospective. Регулярно проводимый семинар, в ходе которого участники анализируют свою работу и полученные результаты с целью улучшения как процесса, так и самого продукта.

Рефакторинг / Refactoring. Метод обеспечения качества продукта, в соответствии с которым совершенствование дизайна продукта осуществляется за счет улучшения его эксплуатационных и других желаемых свойств без изменения его ожидаемого поведения.

Роение / Swarming. Метод, когда несколько членов команды концентрируют общие усилия на устранении конкретного препятствия.

Руководитель запросов на обслуживание / Service Request Manager. Лицо, ответственное за упорядочивание запросов на обслуживание с целью максимизации ценности в условиях непрерывного потока или системы «канбан». Аналог владельца продукта.

Самоорганизующаяся команда / Self-Organizing Team. Кроссфункциональная команда, в которой участники принимают на себя лидерские позиции, по мере необходимости для достижения целей команды.

Семейство методологий Crystal / Crystal Family of Methods. Набор упрощенных agile-методов разработки программного обеспечения, направленных на адаптацию к конкретному обстоятельству.

Скрам (Scrum) / Scrum. Agile-фреймворк для разработки и поддержки сложных продуктов со специфическими ролями, событиями и артефактами.

Скрам скрамов / Scrum of Scrums. Применения метода скрам в масштабе нескольких команд, которые работают над одним продуктом, координируют обсуждение хода работы на

предмет взаимозависимостей и фокусируются на том, как интегрировать поставку программного обеспечения, особенно на пересекающихся участках.

Скрамбан (Scrumban) / Scrumban. Управленческий фреймворк, который возникает, когда команды применяют скрам в качестве выбранного подхода к работе и используют метод «канбан» как призму для анализа, осмысления и непрерывного совершенствования своей работы.

Скрам-команда / Scrum Team. Термин описывает используемое в фреймворке «скрам» сочетание команды разработки, скрам-мастера и владельца продукта.

Скрам-мастер / Scrum Master. Коуч команды разработки и владельца процесса в скрам-фреймворке. Устраняет помехи, улучшает продуктивность мероприятий и предотвращает сбои в работе команды. См. также *Мастер процесса*.

«Сломанный гребешок» / Broken Comb. Так называют человека, обладающего различными уровнями глубины специализации в нескольких навыках, необходимых команде. Называется также «подтек краски». См. также *T-образный и I-образный*.

Смешанный agile / Blended Agile. Два или более agile-фреймворка, метода, элемента или практики, используемых совместно. Например, скрам в сочетании с XP и методом «канбан».

Совершенствование бэклога / Backlog Refinement. Последовательное уточнение требований к проекту и/или последовательное уточнение текущей деятельности, в ходе которой команда совместными усилиями проверяет, обновляет и документально оформляет требования с целью удовлетворения потребности определенной запросом заказчика.

Совместное владение кодом / Collective Code Ownership. Метод ускорения исполнения и взаимодействия участников проекта, в соответствии с которым любой член команды имеет право вносить изменения в любой рабочий продукт или поставляемый результат, что подчеркивает коллективное участие и ответственность всех членов команды.

Составление пар / Pairing. См. *Работа в парах*.

Спайк / Spike. Короткий интервал в проекте, имеющий, как правило, фиксированную продолжительность, в течение которого команда проводит исследования или моделирует аспект решения для проверки его эффективности.

Спецификация на примерах (SBE) / Specification by Example (SBE). Подход совместного определения требований и проведения ориентированных на бизнес функциональных тестов продуктов программного обеспечения, основанный на выявлении и наглядном представлении требований с использованием реальных примеров, а не абстрактных описаний.

Спринт / Sprint. Термин используется для описания ограниченной временными рамками итерации в скрам.

Технический долг / Technical Debt. Отложенная стоимость работ, которые не были выполнены на более ранней точке жизненного цикла продукта.

Типовой пользователь / Personas. Типичный пользователь, представляющий определенную группу схожих конечных пользователей, которые характеризуются общими целями, мотивацией и типичными личными качествами.

T-образный / T-shaped. Термин означает человека с глубокой специализацией в одной области и широким спектром других навыков, необходимых команде. См. также *I-образный и Сломанный гребешок*.

Унифицированный agile-процесс / Agile Unifed Process. Упрощенный и доступный для понимания подход к разработке программных продуктов для бизнеса с использованием методов и концепций agile. Это упрощенная версия Рационального унифицированного процесса (RUP).

Упорядоченный agile / Disciplined Agile (DA). Фреймворк процесса принятия решений, который позволяет упрощенное принятие решений по инкрементным и итеративным поставкам продукта.

Управление организационными изменениями / Organizational Change Management. Комплексный, циклический, структурированный подход к переводу отдельных лиц, групп и

организаций из текущего состояния в будущее состояние с целью достижения предполагаемых выгод для бизнеса.

Фреймворк / Framework. Основная система или структура идей или фактов, которая лежит в основе того или иного подхода.

Функциональная спецификация / Functional Specification. Специфическая функция, которую система или приложение должны выполнять. Как правило представлена в документе, содержащем функциональные спецификации.

Функциональное требование / Functional Requirement. Специфическое поведение, которое должен иметь продукт или услуга.

Хосин канри (Hoshin Kanri) / Hoshin Kanri. Метод развертывания стратегии или политики.

Эволюционная поставка ценности (Evo) / Evolutionary Value Delivery (EVO). Открыто признается в качестве первого agile-метода, содержащего специфическую составляющую, которая отсутствует в других методах: фокус на поставке заинтересованным сторонам набора требований измеримых по их ценности.

Эджайлист (agilist) / Agilist. См. *Agile-практик*.

Экстремальное программирование / eXtreme Programming. Agile-метод разработки программного обеспечения, который обеспечивает более высокое качество, более высокую степень реагирования на изменения требований заказчика и более частые выпуски продукта с более короткими циклами.

1 Цифры в скобках относятся к списку литературы в конце настоящего Руководства.

2 Процесс разработки в порядке «следуя за солнцем» – это процесс, когда работа в конце каждого рабочего дня передается из одного рабочего места в другое, находящееся от него в нескольких часовых поясах, с целью сократить время разработки продукта.