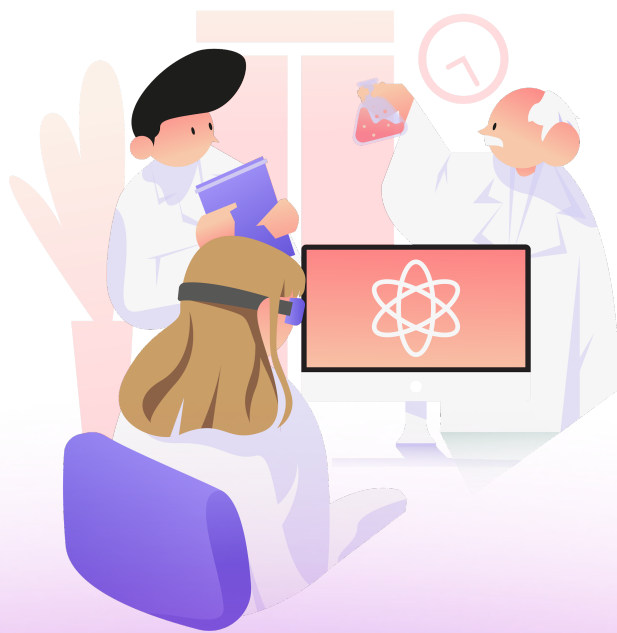


Командная разработка по Agile-концепции с использованием Scrum

Работа в команде



На этом уроке

1. Разберём, чем команда отличается от группы.
2. Рассмотрим, какие этапы развития проходит каждая команда.
3. Узнаем, какие пороки преследуют команды.
4. Рассмотрим, какие роли есть в Scrum-команде и каковы их обязанности.
5. Разберём основные события в Scrum и узнаем, как они помогают команде справиться с пороками.

Оглавление

[Команда: определение, признаки, принципы организации командной работы](#)

[Динамика развития команд по Такману](#)

[5 пороков команды по Ленсиони](#)

[Взаимное недоверие](#)

[Боязнь конфликта](#)

[Отсутствие приверженности \(безответственность\)](#)

[Избегание ответственности \(нетребовательность\)](#)

[Безразличие к результатам](#)

[Scrum-команда \(Scrum Team\)](#)

[Самоорганизация](#)

[Кросс-функциональность](#)

[Функциональные роли в Scrum](#)

[Владелец продукта \(Product Owner\)](#)

[Команда разработки \(Development Team\)](#)

[Scrum-мастер \(Scrum Master\)](#)

[События в Scrum](#)

[Спринт](#)

[Планирование спринта](#)

[Ежедневный Scrum](#)

[Обзор спринта](#)

[Ретроспектива спринта](#)

[Заключение](#)

[Дополнительные материалы](#)

[Используемые источники](#)

[Практическое задание](#)

На прошлом уроке мы подробно разобрали организацию рабочего процесса по фреймворку Scrum. Рассмотрели его элементы, их общее взаимодействие, основные принципы, ценности и отличия от другого популярного метода оптимизации проектных работ — Kanban.

На этом уроке мы подробнее обсудим, что такое команда, каковы её признаки и особенности. Разберём роли, входящие в Scrum-команду, и их зоны ответственности. Рассмотрим циклы развития команд, их пороки, а также события Scrum, которые помогают командам стать успешными.

Команда: определение, признаки, принципы организации командной работы

Идея командных методов работы была заимствована из мира спорта. Понятие команды применительно к организационному поведению появилось в США в 60-е годы прошлого века в рамках исследования [групповой динамики](#). Впоследствии это понятие получило серьёзное развитие, поскольку установлено, что команда в сравнении с обычной группой способна достичь значительно более высоких результатов.

Команда — это общность людей, связанных единой целью и принципами взаимодействия.

Обычно цели, стоящие перед командой, не могут быть достигнуты отдельными её членами в силу ограниченности времени и ресурсов, а также из-за невозможности одного члена команды овладеть всеми необходимыми знаниями и умениями. Это положение наглядно подтверждают примеры спортивных команд. Можно утверждать, что выполнять проекты, требующие глубоких знаний и компетентности, предпочтительнее командой: это позволяет распределить задачи между её членами и достичь цели совместными усилиями в установленные сроки.

На практике нередко происходит смешение понятий группы и команды, хотя различия между ними весьма существенны и команды обладают рядом особенных характеристик.

Признаки, присущие команде:

1. Команда состоит из небольшого количества человек: в среднем от 2 до 10.
2. Автономна, чаще всего не имеет прямой зависимости от других команд и подразделений, самостоятельна и имеет собственную организацию труда.
3. Объединена для достижения конкретной цели.
4. Члены команды равноправны и обладают взаимодополняющими навыками и качествами: технические и функциональные навыки, способность выявлять проблемы и принимать решения, навыки межличностного взаимодействия, способность к обучению и т. д.
5. Члены команды вырабатывают общую стратегию работы и командные правила — правила взаимодействия, которые разделяют все участники.

Считается, что именно наличие объединяющей всех участников команды цели — характерное и главное

её отличие от обычной группы людей, объединённых общими интересами, но не имеющих заинтересованности в общем результате. Ценности и цели в обычной группе людей могут быть совершенно разными.

Ключевые принципы организации командной работы:

1. Коллективное выполнение работ, распределение задач между участниками.
2. Коллегиальное принятие решений, поддержка и исполнение этих решений каждым участником команды.
3. Взаимозависимость — каждый член команды вносит свой индивидуальный вклад в общую работу, поэтому другие члены команды и результат зависят от его работы.
4. Коллективная ответственность — эффективность работы определяется прежде всего по коллективным показателям и общему результату.

Благодаря этим принципам командная работа становится значительно эффективнее не только групповой, но и индивидуальной работы. Каждый член команды стремится к достижению общей цели, генерирует новые идеи, активно взаимодействует с другими участниками для координации совместной деятельности. Это способствует возникновению эффекта синергии, суть которого состоит в том, что результат совместных усилий членов команды оказывается заметно больше суммы отдельных результатов её членов при условии индивидуального выполнения работы. Кроме того, в командах в значительно большей степени проявляются внутренняя сплочённость, преданность общему делу, сотрудничество и доверие.

Динамика развития команд по Такману

У всех команд помимо общих характерных признаков и принципов взаимодействия есть динамика развития, которая проходит несколько стадий за период существования команды. Только что сформированной команде требуется время на развитие своей эффективности через прохождение определённых стадий. Стадии развития команды по Такману представляют собой модель групповой динамики, которая подробно описывает этапы трансформации.

Впервые психолог Брюс Такман описал свою модель развития команд в статье «Последовательность развития в малых группах» (1965) и дал этим стадиям запоминающиеся названия на английском языке: forming, storming, norming, performing. На русском языке: формирующая стадия, стадия конфронтации (бурление), нормирующая стадия (упорядочивание), стадия функционирования (продуктивности). Именно такие стадии и в таком порядке проходит каждая команда в ходе своего эволюционного развития на пути к высокой производительности. Позже Такман добавил пятую стадию Adjourning («расставание/ропуск»).

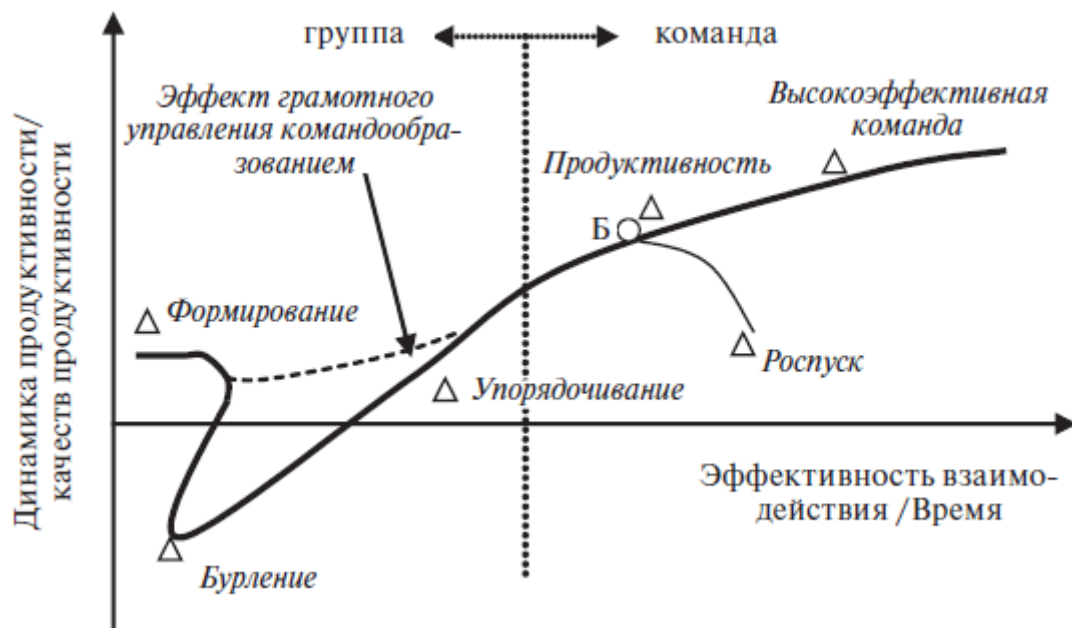


Рис. 1. Модель развития команды по Такману

Рассмотрим каждую стадию развития подробнее.

Формирующая стадия — это начальный этап развития команды, на котором происходит знакомство участников, изучение оснований для совместной работы, постановка цели, распределение ролей и задач. На этом этапе, несмотря на возможное волнение от некоторой неопределённости, большинство членов группы позитивны и доброжелательны, но осторожны в проявлении своего мнения. Этот этап может продлиться какое-то время, пока члены команды не приступят к непосредственному выполнению своих обязанностей и не начнут лучше узнавать друг друга.

Стадия конфронтации (бурление). После начала работ неминуемо появление конфликтов. Характеризуется обострением конкуренции, конфликтом интересов или расхождением во взглядах на организацию работ и решение задач. Конфликт может проявляться по-разному: носить открытую форму и выражаться вербально или быть пассивным и выражаться в нежелании сотрудничать и принимать активное участие в работе. Всё это может привести к сопротивлению при выполнении стоящих перед командой задач или общей усталости сотрудников от психологического напряжения и того, что рабочие процессы нестабильны, а поддержки от коллег и сплочённости они не ощущают. Чтобы перейти на следующую стадию, необходимо уйти от стратегии доказывания и выяснения к работе над непосредственным решением проблемы, то есть перейти к сотрудничеству.

Нормирующая стадия (упорядочивание). Если команда успешно прошла стадию конфронтации, то постепенно наступит нормирующий этап. Это этап разрешения противоречий, признания сильных и слабых сторон коллег, уважения чужого мнения и принятия различий в стиле работы. На этой стадии закладываются организационные принципы будущей командной работы и налаживается взаимодействие между участниками. Теперь, когда сотрудники знают друг друга лучше, они могут беспрепятственно коммуницировать и предоставлять конструктивную обратную связь. Команда начинает полностью фокусироваться на своей цели, а лидеры могут сосредоточиться на стратегических вопросах. Переход от конфликтной стадии к нормирующей — этап небыстрый, и при появлении новых

задач или нового члена команды может произойти возврат к стадии конфликта.

Стадия функционирования (продуктивность). Это стадия эффективного сотрудничества. Достигая этого уровня, команда начинает активно продвигаться к поставленной цели. Это этап коллективных решений и взаимопомощи, доверия и максимальной сплочённости. На этом этапе команда может выйти на максимально возможный уровень производительности за счёт слаженности действий и взаимопонимания.

Расставание (ропуск). Это неизбежная стадия большинства проектов. Проектные команды работают вместе в течение фиксированного периода, и после завершения проекта даже успешные команды могут быть расформированы. Для сотрудников, которые тяжело переносят подобные перемены из-за сформировавшейся привязанности, этап может показаться болезненным. Однако, если есть возможность сохранять состав команды от проекта к проекту, это будет способствовать дальнейшему переходу команды на новый высокоэффективный уровень, который может поддерживаться довольно долгий промежуток времени, до тех пор, пока у членов команды будет сохраняться мотивация.

Одни команды могут очень быстро и безболезненно преодолеть стадию конфронтации, другие могут из неё не выйти и прекратить свою деятельность, а третьи могут навсегда застрять на стадии нормирования, показывая довольно средний результат продуктивности.

Каким образом команда пройдёт этапы своего развития и насколько она будет успешной, с одной стороны, зависит от самой команды, с другой стороны — от таланта руководителя или лидера команды. Именно для помощи командам в их развитии многие компании используют разные методики командообразования (тимбилдинга), которые направлены не только на само формирование команд, но и на формирование и развитие навыков командной работы и укрепление командного духа.

Как показал опыт, мероприятия по тимбилдингу способны принести пользу и произвести необходимый эффект только в том случае, если они работают с так называемыми пятью дисфункциями (пороками) команды, которые на фундаментальном уровне мешают эффективному взаимодействию внутри команды, способствуют развитию неблагоприятной обстановки и конфликтной среды.

5 пороков команды по Ленсиони

Патрик Ленсиони — американский автор книг по управлению бизнесом и командами. Наиболее известен как автор популярной [бизнес-сказки «Пять дисфункций команды»](#), в которой исследуется динамика развития команды и предлагаются решения, помогающие командам работать лучше. Рассмотрим каждую дисфункцию, которую описывает Ленсиони, обозначим их взаимосвязь и пути устранения.



Рис. 2. Пять пороков команды по Ленсioni

Взаимное недоверие

Первая и базовая дисфункция, с которой начинается развитие всей цепочки.

Признаки взаимного недоверия:

- скрывают свои слабости и ошибки от других;
- не критикуют коллег;
- неохотно обращаются за помощью;
- быстро делают выводы о намерениях и убеждениях других, не пытаясь разобраться;
- не интересуются опытом и навыками коллег и не прислушиваются к их советам;
- много времени и сил тратят не на работу, а на то, чтобы произвести впечатление;
- не высказывают недовольство;
- боятся совещаний и ищут причины, чтобы не проводить время вместе с коллегами.

Недоверие в первую очередь основано на страхе показать коллегам свою слабость или некомпетентность. Конкурентная среда, которая нас окружает с самого рождения, не позволяет открыто признавать свои ошибки или просить помощи. Человеку, который всю жизнь заботился только о своих личных достижениях, сложно перестроиться на командные условия, где взаимное доверие должно быть превыше всего.

Доверие — это убежденность членов команды, что действия их коллег продиктованы благими намерениями. Члены команды не боятся демонстрировать друг другу своё несовершенство и уверены, что коллеги не воспользуются этим в корыстных целях. Под несовершенством понимается демонстрация слабостей, отсутствие каких-то навыков, проблемы в общении, ошибки, а также открытое высказывание просьб о помощи.

Боязнь конфликта

Недоверие в коллективе создаёт почву для развития второго порока — боязни конфликта, которая приводит к мнимой гармонии.

В командах, которые боятся конфликта:

- проводятся скучные совещания;
- процветают интриги, копятсь взаимные обиды;
- не решаются трудные вопросы, мнения по которым разделились;
- работники не прислушиваются друг к другу;
- время и силы расходуются впустую, потому что применяются методы управления, сводящие к минимуму риск межличностных конфликтов.

Люди, не доверяющие друг другу, неспособны к искренней, открытой дискуссии. Они боятся задеть кого-либо или оказаться задетыми. Вместо этого они ведут обмен нейтральными замечаниями, осторожными комментариями, порой переходя к колкостям. В результате они принимают единогласное решение, которое никого не устраивает и ни к чему не ведёт. Руководители — противники конфликтов обычно просят подчинённых высказывать своё мнение в частном порядке и не выносят проблему на общее обсуждение, в результате чего принятие важных решений откладывается, а проблема не решается.

В современном обществе конфликт считается абсолютно недопустимым во многих ситуациях, особенно на работе. Однако следует отличать продуктивный конфликт от деструктивного соперничества и интриг. В конструктивном конфликте обсуждаются только идеи, но не личные качества участников. Такому конфликту тоже могут быть свойственны накал страстей, эмоции, повышенный тон и взаимные упрёки. Сторонний наблюдатель вполне может принять конструктивный конфликт за банальную ссору. Однако в командах, способных на конструктивный конфликт, понимают, что его назначение — найти оптимальное решение проблемы за кратчайший период времени. Участники быстро и всесторонне обсуждают все вопросы, и после дискуссии у них не остаётся неприятного осадка или ощущения обиды.

Отсутствие приверженности (безответственность)

Мнимая гармония крайне опасна, потому что порождает третий порок команды — отсутствие приверженности к работе команды и общей цели (безответственность).

Команда, у которой отсутствует приверженность делу:

- не может чётко сформулировать свои цели и задачи;
- упускает возможности, тратя время на обдумывание и достижение ясности;
- боится поражения;
- снова и снова обсуждает свои планы и решения;
- заставляет членов команды принимать решения на свой страх и риск.

Так как в команде царит атмосфера недоверия, никто открыто не высказывает свою точку зрения, а

коллективное принятие решений носит формальный характер, члены команды начинают считать, что не несут никакой ответственности за всё то, что не затрагивает их лично.

Избегание ответственности (нетребовательность)

Это четвёртый порок команды, который объясняется стремлением членов команды избежать дискомфорта, возникающего при попытках указывать коллегам на их недочёты.

Нетребовательная команда:

- производит товары и услуги невысокого качества;
- страдает от взаимных обид;
- срывает сроки;
- взваливает всё бремя ответственности за дисциплину на лидера.

Принимая решения, к выполнению которых они равнодушны, члены команды не считают себя связанными какими-либо обязательствами. Даже понимая, что какие-то действия (или, наоборот, бездеятельность) других людей вредны для компании, сотрудники часто исходят из принципа «Это не моё дело». Как ни парадоксально, практика показывает, что самый надёжный метод поддержания высокой эффективности команды — взаимное давление членов команды друг на друга. Благодаря этому уменьшается необходимость добиваться повышения производительности и командного поведения административными методами. Именно боязнь подвести членов своей команды заставляет людей лучше работать.

Безразличие к результатам

Безответственность и нетребовательность создают почву для развития пятого порока. Безразличие к результатам возникает, когда члены команды ставят свои личные потребности (честолюбие, карьеру, признание) или потребности своих отделов выше общих целей команды или компании.

Команда, которая безразлична к результатам:

- не развивается или деградирует;
- редко побеждает конкурентов;
- теряет амбициозных работников;
- косвенно стимулирует работников к достижению исключительно личных целей;
- легко теряет направление и фокус.

Многие команды существуют и функционируют не для того, чтобы достичь значительных результатов, а просто для того, чтобы жить и работать. К сожалению, ни высокий уровень доверия в этих командах, ни конструктивные конфликты, ни высокая требовательность не могут компенсировать отсутствие желания победить. Чтобы отслеживать свою эффективность, команда должна знать свои цели, показатели деятельности и конечный результат. Широкое информирование о будущих результатах полезно для

стимулирования достижения выдающихся успехов в бизнесе. Команды, которые публично берут на себя обязательство достичь поставленных целей, будут трудиться усерднее и приложат все усилия, чтобы не опозориться в глазах коллег. Команды, которые не берут на себя определённые обязательства, а только обещают сделать всё от них зависящее, подсознательно обрекают себя на поражение.

Ленсиони говорит, что эти дисфункции в той или иной степени присутствуют во всех командах, однако существуют способы и конкретные рекомендации, каким образом с ними работать и предотвратить их развитие. Сам автор предлагает следующее решение — **«Модель пяти добродетелей»**:

- создание доверительных отношений;
- конструктивные конфликты — открытое обсуждение идей и проблем;
- ответственность — заинтересованность в принимаемых решениях и планах;
- требовательность к выполнению решений и планов;
- нацеленность на достижение общей цели — гордость за результат.

Эти постулаты, призванные помочь командам в борьбе с дисфункциями, тесно перекликаются с основными идеями и принципами Agile-концепции. С конкретными рекомендациями, которые предлагает автор к использованию на практике, можно ознакомиться в его книге «Пять пороков команды». В рамках курса нам гораздо интереснее рассмотреть инструменты Scrum, которые мы сможем использовать в практической части.

Scrum-команда (Scrum Team)

Из прошлого урока мы уже узнали, что Scrum-команды используют итеративно-инкрементальный подход к разработке, то есть работают короткими дистанциями по 1–4 недели, чтобы реализовать готовый к запуску функционал, который добавит продукту ценности. Это отличительная особенность фреймворка в части организации работ. Что же касается самой команды, её главные отличительные особенности в рамках Scrum: самоорганизация и кросс-функциональность. Рассмотрим подробнее эти два понятия.

Самоорганизация

Командная самоорганизация — это один из ключевых принципов Agile: «Самые лучшие требования, архитектурные и технические решения рождаются у самоорганизующихся команд».

Самоорганизация — процесс упорядочения элементов одного уровня в системе за счёт внутренних факторов, без внешнего специфического воздействия (изменение внешних условий может также быть стимулирующим либо подавляющим воздействием). Результат — появление единицы следующего качественного уровня.

В отношении команд это определение можно применить следующим образом:

Самоорганизующиеся команды — это команды, способные самостоятельно организовать свою работу для достижения поставленной цели без внешних указаний и контроля. Результат — появление синергии в виде высокой эффективности.

Примером и признаком самоорганизации будет не только самостоятельное распределение задач между участниками, но и выбор способа контроля качества разработки, наставничество и курирование менее опытных специалистов, взятие на себя ответственности за результат.

Кросс-функциональность

Кросс-функциональные команды должны обладать всеми необходимыми компетенциями для выполнения работы и не зависеть от людей, которые не входят в команду. Совокупных компетенций должно хватать для всего комплекса работ в сочетании с оптимальным способом его реализации. Это значит, что в команду должны входить не только технические специалисты (бэкенд-, фронтенд-разработчики и QA-специалисты), но и бизнес-аналитик, маркетолог, UX/UI-дизайнер и любые другие специалисты, участия которых требует проект. Всё это делается для достижения следующего немаловажного пункта из Scrum Guide: модель команды в Scrum предполагает минимизацию внешних зависимостей, располагая к гибкости, творчеству и продуктивности.

Второе, не столь очевидное, значение кросс-функциональности — это пересечение компетенций, то есть расширение знаний и навыков в смежных с основной экспертизой дисциплинах. Это не значит, что программист должен быть UX/UI-дизайнером, а маркетолог — освоить навыки альфа- и бета-тестирования. Но иметь базовые знания о смежных специализациях — это основа не только для личного и профессионального развития, но и для эффективного взаимодействия с коллегами. Например, бэкенд-разработчик должен понимать основные принципы работы с фронтендом и иметь базовые навыки создания интерфейсов с использованием UI-библиотек, а веб-дизайнер должен обязательно на базовом уровне разбираться в вёрстке, чтобы понимать, как тот или иной дизайн отразится на времени разработки. Специалистов, которые не только развивают свою основную экспертизу, но и осваивают дополнительные навыки из смежных областей, называют T-shaped People.

T-shaped-специалист противопоставляется I-специалисту (эксперту только в одной области) и дженералисту — человеку, который знает обо всём понемногу, но ни в одну область не углубляется.



Рис. 3. Характеристика нескольких типов личности по профессиональной специализации

Вертикальная линия Т показывает глубину знаний и экспертности в основной области. Горизонтальная показывает те области, где специалист тоже имеет знания и небольшой опыт.

Если команда состоит только из I-shaped-специалистов, то задачи приходится распределять по компетенциям и все участники команды будут находиться в зависимости друг от друга без возможности помочь в случае необходимости. Если же команда в основном состоит из T-shaped специалистов, задачи приоритизируются по важности и только после этого определяется тот, кто готов этой задачей заняться. Например, если в приоритете стоит тестирование и тестировщик не успевает с ним справиться, то ему на помощь может прийти разработчик, чья задача на данный момент менее приоритетна. То же самое произойдет, если кто-то из участников заболел: другой участник команды может временно его заменить.

Рассмотрим, какие роли представлены во фреймворке Scrum и каким образом они взаимодействуют между собой, чтобы как можно быстрее преодолеть зону конфронтации, успешно бороться с пороками и развивать T-Shaped-специалистов.

Функциональные роли в Scrum

Scrum-команда состоит из команды разработки, владельца продукта и Scrum-мастера.

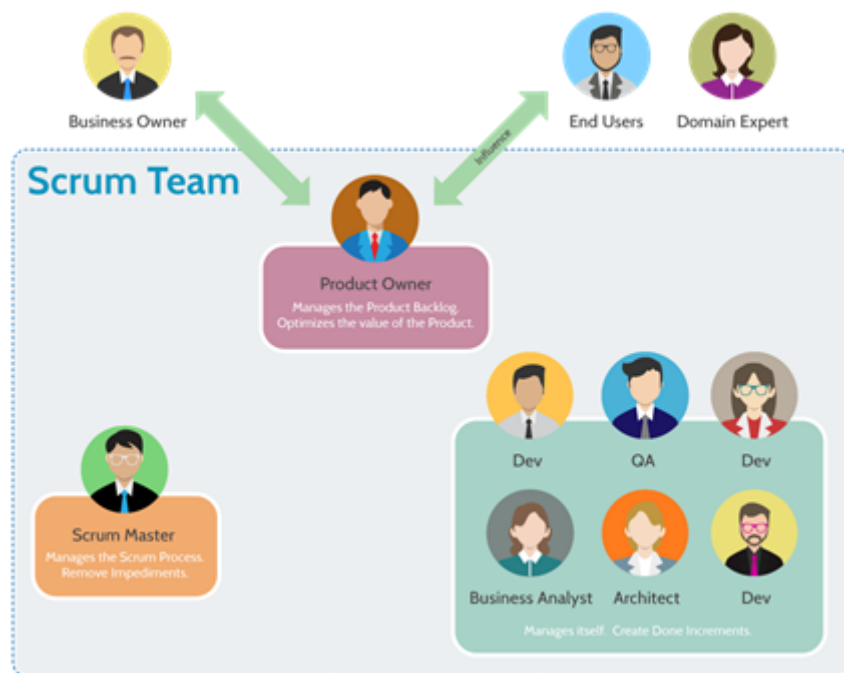


Рис. 4. Роли участников Scrum-команды

Владелец продукта (Product Owner)

Владелец продукта — это представитель заказчика, который отвечает за общую цель проекта, за состав и приоритизацию функциональных требований, за определение критериев приёмки результата. Он несёт ответственность за получение максимальной ценности продукта и его коммерческую успешность.

Это единственное лицо, ответственное за бэклог продукта, его полноту, ясность и приоритетность задач. Владелец продукта — полноценный участник Scrum-команды, но в силу разных обстоятельств, например отсутствия необходимой компетенции, он имеет право делегировать часть своих задач по управлению бэклогом продукта команде разработки, при этом ответственность перед бизнесом по-прежнему остаётся на нём.

Роль владельца продукта всегда выполняет один человек, даже несмотря на то, что со стороны заказчика может быть сформирована целая проектная команда. Одна из задач владельца продукта — сбор и аккумуляция мнений, решений и пожеланий со стороны команды клиента. Владелец продукта — это транслятор и главное ответственное лицо, которое демонстрирует продукт стейкхолдерам.

Стейкхóлдер (англ. stakeholder — дословно «акционер, заинтересованная сторона, причастная сторона») — физическое лицо, группа или организация, имеющая права, долю, требования или интересы в проекте. Иными словами, это лицо или группа лиц, которые иницируют проект, — бизнес-заказчики. Они не являются полноценными членами команды и вовлечены в Scrum только во время обзорного совещания по спринту при демонстрации продукта (Sprint Review). Именно с ними

регулярно взаимодействует владелец продукта, передавая им информацию о статусе проекта и получая от них обратную связь.

Команда разработки (Development Team)

Размер команды разработки должен быть оптимальным (3–9 человек), чтобы обеспечить достаточную гибкость и продуктивность работы в ходе спринта. Когда в команде два человека, производительность сильно снижается из-за нехватки необходимых навыков у её членов. Команды размером более 9 человек испытывают трудности с координацией работы.

Разработчик — единственная роль в команде разработки, независимо от типа выполняемых задач, то есть и аналитик, и дизайнер, и маркетолог, войдя в Scrum-команду, называются общим словом «разработчики».

Команда разработки состоит из профессионалов, работающих над созданием продукта, выпускаемого по частям (инкрементам) от одного спринта к другому, и несёт коллективную ответственность за результат — создание инкремента продукта наивысшего качества к концу спринта.

Команда разработки самоорганизующаяся и кросс-функциональная. Члены команды обладают всеми необходимыми навыками для командного создания инкремента продукта и координируют свою работу без внешних указаний. В результате возникает синергетический эффект — рост эффективности и производительности.

Разработчики принимают функциональные требования к разработке только от владельца продукта и отвечают за планирование спринта, постановку цели спринта, бэклог спринта, оценку задач, демонстрацию и запуск инкремента или релиза.

Scrum не признаёт подкоманд в команде разработки, независимо от областей, над которыми идёт работа (масштабирование Scrum — другой случай).

Scrum-мастер (Scrum Master)

Scrum-мастер несёт ответственность за должное функционирование фреймворка Scrum, а также следит за тем, чтобы все члены Scrum-команды правильно понимали ценности Scrum и практику его применения. Scrum-мастер — лидер, который помогает наладить взаимодействие в команде, помогает с планированием, обучает команду необходимым техникам, выступает в роли наставника, тренера и вдохновителя, поддерживая таким образом командный дух. Обеспечивает необходимые условия для эффективной работы команды разработки.

Функции Scrum-мастера по отношению к владельцу продукта:

- обеспечивает условия, при которых Scrum-команда как можно лучше понимает цели, объём работ и предметную область;
- помогает найти наиболее эффективные техники для управления бэклогом продукта;
- объясняет Scrum-команде необходимость кратких и понятных элементов бэклога продукта;
- объясняет особенности планирования продукта в эмпирической среде;
- помогает владельцу продукта упорядочить бэклог продукта.

Функции Scrum-мастера по отношению к команде разработки:

- устраняет препятствия, мешающие прогрессу команды разработки;
- налаживает сотрудничество между участниками команды;
- помогает команде разработки создавать продукты с высокой ценностью;
- развивает команду разработки, помогает быть самоорганизующейся и кросс-функциональной;
- содействует личностному росту и обучению членов команды;
- фасилитирует события Scrum и участвует в медиации конфликтов;
- развивает команду разработки в тех областях, в которых Scrum не полностью понят и принят.

Исходя из описания ролей и их зон ответственности, можно сделать вывод, что они сами по себе уже обеспечивают выполнение многих принципов Agile и одну из главных идей всей концепции: тесное взаимодействие с заказчиком. Пришла очередь рассмотреть основные события в Scrum, посредством которых осуществляется организация всех работ фреймворка.

События в Scrum

SCRUM



Рис. 5. Схема работы по Scrum

Предписанные в Scrum события — это основные встречи, которые проводятся регулярно в рамках каждого спринта и призваны обеспечить три основных принципа Scrum: прозрачность, инспекцию и адаптацию. Каждое событие обязательно для всей команды, имеет временное ограничение, свою цель и задачи.

Спринт

Sprint (спринт) — это итерация, временной отрезок длиной от 1 до 4 недель, в течение которого команда создаёт функционирующий и готовый к использованию и выпуску инкремент продукта. Продолжительность спринта не может быть изменена после его старта, и желательно поддерживать одинаковую длительность итераций на протяжении всего процесса разработки. Спринт — ядро Scrum, он включает в себя все остальные события: планирование спринта, ежедневный Scrum, обзор спринта и ретроспективу.

Во время спринта:

- изменения, которые могут поставить под угрозу цель спринта, не допускаются;
- качество продукта не должно снижаться;
- объём работ может быть уточнён и пересмотрен только совместно с владельцем продукта.

Каждый спринт можно считать отдельным проектом длительностью не более одного месяца, который состоит из чёткой цели, концепции реализации, адаптивного плана по её достижению, реализации и

демонстрации конечного продукта — инкремента.

Такой подход позволяет ограничить риски и увеличить прогнозируемость работ за счёт регулярной инспекции в течение спринта и адаптации в последующих итерациях.

Владелец продукта имеет право завершить спринт досрочно только в случае потери актуальности цели спринта. Это случается крайне редко, потому что приводит к полному запуску нового цикла от пересмотра бэклога продукта до запуска нового спринта.

Рассмотрим подробнее события, которые входят в рамки спринта.

Планирование спринта

Sprint planning meeting (планирование спринта) — составление перечня задач (бэклога спринта), над которыми будет трудиться команда разработки во время спринта. В рамках этого события команда определяет цель спринта, обсуждает объём работ и порядок действий. Встреча ограничена по времени: для спринта в 4 недели — 8 часов, соответственно, для спринта в 2 недели — 4 часа. По результатам планирования спринта команда должна получить ответы на два основных вопроса:

1. Что будет сделано?
2. Как мы это сделаем?

Ответ на первый вопрос зависит от потребностей бизнеса, поэтому владельцу продукта необходимо вынести на обсуждение бизнес-задачи, которые необходимо решить, и перечень функциональности, посредством которой эта цель может быть достигнута. На основании этих данных команда разработки формирует единое понимание работы и её объём. Результатом обсуждения становится цель спринта.

При ответе на второй вопрос команда детализирует объём работ, разделив элементы бэклога на конкретные задачи, и совершает их оценку для последующего планирования своих действий. По результатам этой работы команда разработки поясняет остальным участникам команды способы достижения цели спринта и берёт на себя обязательство по выполнению конкретного объёма работ. Результатом обсуждения становится бэклог спринта с предполагаемым объёмом работ, которые команда готова выполнить за спринт. Далее команда самоорганизуется при распределении задач и приступает к работе.

План работ во время спринта, как и объём работ, может измениться в процессе реализации и получения новых знаний. Возникшие и обоснованные изменения в этом случае не считаются препятствием и должны восприниматься лояльно владельцем продукта и командой, если это способствует достижению цели.

Ответственность за организацию и проведение встречи по планированию спринта берёт на себя Scrum-мастер. Ответственность за соответствие цели спринта интересам бизнеса лежит на владельце продукта. Ответственность за выполнение обязательств по объёму работ и достижению цели спринта берёт на себя команда разработки.

Ежедневный Scrum

Daily standup meeting (ежедневный Scrum) — ежедневная встреча команды разработки (не более 15 минут), в ходе которой команда планирует свою работу на ближайшие 24 часа. Ежедневный Scrum следует проводить в одно и то же время, в одном и том же месте.

Каждый участник команды разработки отвечает на следующие вопросы:

1. Что я сделал с момента прошлой встречи?
2. Что я собираюсь сделать сегодня?
3. Какие препятствия замедляют достижение цели спринта для меня и команды?

Ответы на эти вопросы позволяют определить прогресс работы и выявить препятствия, которые возникли на пути к достижению цели спринта. Если возникшие проблемы требуют более детальной проработки, то команда договаривается о назначении дополнительных встреч, чтобы найти решение и скорректировать планирование.

Ответственность за проведение ежедневных встреч и их результат несёт команда разработки, встречи созданы для неё с целью совместной кооперации и координации действий. Задача Scrum-мастера — обучить команду укладываться во временной регламент и убедиться в том, что все участники команды понимают предназначение встреч.

Необходимо понимать, что ежедневный Scrum помимо основных своих задач по инспекции и адаптации работ несёт в себе очень важную функцию — налаживание связи внутри команды и установление доверия между её участниками. Такие встречи способствуют быстрому принятию решений и повышают уровень осведомлённости и прозрачности процессов. Именно на этих встречах каждый участник имеет право высказать своё мнение, поделиться своими успехами и неудачами, обратиться за помощью или самостоятельно предложить выход из ситуации.

Обзор спринта

Sprint review (обзор спринта) проводится в конце спринта с целью демонстрации инкремента продукта стейкхолдерам, получения обратной связи и адаптации бэклога продукта. Это неформальная встреча, но самая важная с точки зрения получения обратной связи от заинтересованных лиц. Временные рамки встречи — четыре часа для спринтов длительностью один месяц. Короткие спринты предполагают сокращение временных рамок.

За организацию встречи и понимание всеми участниками её цели отвечает Scrum-мастер. Владелец продукта озвучивает цель спринта и рассказывает участникам, что было сделано. Команда разработки проводит демонстрацию и рассказывает, какие возникли проблемы во время реализации и как они решались, отвечает на вопросы. Далее обсуждаются прогресс на пути к достижению главной цели проекта, прогнозируемые сроки, план работ с учётом новых обстоятельств и знаний.

Результат обзора спринта — пересмотренный бэклог продукта, в котором может измениться приоритизация и состав работ.

Ретроспектива спринта

Sprint retrospective (ретроспектива спринта) проводится после обзора спринта перед планированием следующего спринта с целью внутреннего аудита и создания плана внедрения улучшений для будущего спринта.

Ретроспектива спринта — это возможность для команды разработки оценить свою эффективность и создать план улучшений своей работы для следующего спринта.

Максимальная продолжительность ретроспективы — три часа для спринта длительностью один месяц. Для более коротких спринтов, как правило, отводится меньше времени.

Scrum-мастер отвечает за организацию встречи и соблюдение временного регламента. На этот раз он принимает участие в процессе наравне со всеми участниками.

Цели ретроспективы спринта:

- инспекция степени успешности спринта;
- обнаружение и упорядочение того, что прошло хорошо, и того, что нуждается в улучшении;
- создание плана внедрения улучшений в процесс работы Scrum-команды.

По результатам встречи Scrum-мастер помогает команде разработки улучшить организацию процессов и оптимизировать используемые в рамках фреймворка практики. Основные цели этого содействия: повышение эффективности работы команды, улучшение качества продукта и повышение степени удовлетворённости всех участников процесса от работы и её результата.

Результатом ретроспективы должен быть согласованный всеми участниками команды список улучшений и план действий по их воплощению в следующем спринте.

Заключение

Исходя из рассмотренных нами событий, которые поддерживают организацию всей работы фреймворка, и ролей участников, можно сделать вывод, что все элементы системы отвечают философии Agile и его принципам: сфокусированность на продукте, тесное сотрудничество с заказчиком, ориентация на командное взаимодействие и максимальную адаптивность к среде.

Подведём итоги и обобщим преимущества, которые даёт командам использование фреймворка Scrum:

1. Самоорганизующиеся команды предполагают активное включение каждого участника в рабочий процесс, что способствует объединению команды вокруг общей цели и создаёт мощную групповую динамику в ходе её достижения.
2. Кросс-функциональность команды делает её самодостаточной, а самоорганизация способствует тому, что каждый чувствует свою нужность и важность своего вклада в общее дело. В результате возникает чувство гордости за свой труд.
3. Коллективная ответственность способствует тому, что в команде практически отсутствуют слабые звенья, развивается наставничество, взаимопомощь, самообучаемость, непрерывное

совершенствование и развитие командного духа.

4. Правильное использование инструментов Scrum помогает командам быстрее выйти на уровень максимальной эффективности, преодолев кризисные моменты, связанные со сложностями взаимодействия на старте проекта.
5. События Scrum способствуют устранению таких дисфункций команды, как недоверие, мнимая гармония, безответственность, нетребовательность и отсутствие интереса к результату.
6. Обратная связь от коллег, даже негативная, при таком подходе воспринимается позитивно. Все понимают, что цель критики — найти и исправить недоработки в продукте, а не наказать виновного.
7. Атмосфера открытости и быстрая поставка продукта после каждого спринта способствуют великолепным взаимоотношениям с заказчиками, потому что отсутствует один из самых серьёзных стрессов — состояние неопределённости.

Команды, которые понимают смысл именно такой организации фреймворка, с удовольствием соблюдают все его правила и участвуют во всех событиях.

На следующем уроке мы начнём готовиться к проекту: детально разберём, из чего состоит планирование спринта, научимся работать с бэклогом продукта и бэклогом спринта.

Глоссарий

Синергия (синергетический эффект) (греч. *συνεργία* «сотрудничество, содействие, помощь, соучастие, сообщничество» от др.-греч. *σύν* «вместе» + *ἔργον* «дело, труд, работа, (воз)действие») — усиливающий эффект взаимодействия двух или более факторов, характеризующийся тем, что совместное действие этих факторов существенно превосходит простую сумму действий каждого из указанных факторов.

Тимбилдинг (от английского *team building*, буквально — «строительство команды») — особый тип мероприятий, цель которых — объединение сотрудников в команду единомышленников.

Фасилитация команды (от англ. *facilitate* — облегчать) — профессиональная организация процесса групповой работы, направленная на прояснение и достижение группой поставленных целей. Приводит к повышению эффективности групповой работы, вовлечённости и заинтересованности участников, раскрытию их потенциала.

Дополнительные материалы

1. [Работа в команде — модели, теории, алгоритмы \(статья\)](#).
2. [Принципы и особенности командообразования \(статья\)](#).
3. [Кто такой Scrum-мастер? \(статья\)](#).
4. [Sprint Review vs Sprint Retrospective \(статья на англ. яз.\)](#).

Используемые источники

1. [Руководство по Scrum \(Кен Швабер, Джефф Сазерленд\)](#).
2. [Этапы развития команды по Такману \(статья\)](#).
3. [Пять пороков команды. Патрик Ленсиони](#).
4. [Why T-shaped people? \(статья\)](#).

Практическое задание

1. Пройти [тестирование к уроку](#).
2. Изучить методический материал.
3. Поделиться на команды.
4. Завести Telegram-чаты.
5. Изучить ТЗ.