

## Знакомство со Scrum

JSP & Servlets  
15 уровень, 3 лекция

ОТКРЫТА

### История появления Scrum

С момента публикации в 1970 году отчета Уинстона Ройса “Управление разработкой крупных программных систем” многие пытались найти методику, которая могла бы устранить минусы модели разработки Waterfall. Альтернативой “водопаду” стал метод Scrum, о котором сейчас пойдет речь.

Scrum получил свое название в 1986 году благодаря работе Такеучи и Нонаки “Новые правила разработки новых продуктов”. В этом документе утверждается, что наиболее эффективный способ достигнуть поставленную цель — это дать разработчикам точный план действий.

В 1995 году появилось еще одно руководство “Разработка программного обеспечения по Scrum” авторства Сазерленда и Швейбера. Впоследствии эта публикация многократно обновлялась. Сейчас она считается основным пособием для разработки по данному методу. В настоящее время актуальная версия Scrum Guide содержит сведения, обновленные в 2020 году.

Основные положения руководства Scrum Guide предполагают, что шаблон управления проектами должен базироваться на том, что разработчики поставляют готовый продукт в оговоренные сроки — спринты. Для успешного применения Scrum рекомендуется использовать структуру, состоящую из нескольких элементов: ролей, событий, правил и артефактов.

### Роли в Scrum

В Scrum есть три роли, все они формируют скрам-команду:

Заказчик программного продукта — наиболее важный в проекте человек, поскольку только он полностью осознает его ценность для бизнеса. Заказчик объясняет потребности пользователей будущего продукта до разработчиков, однако он не отвечает за техническую часть процесса разработки. Заказчик также определяет приоритет при создании тех или иных элементов или функций в продукте.

На разработчиков возложено выполнение технических задач, кроссфункциональность которых зависит от сферы применения. Разработчики заняты созданием бэклога спринта, написанием кода, адаптацией проекта относительно цели спринта и другими задачами.

Scrum-мастер является координатором работы scrum-команды. Он оказывает помощь заказчику и разработчикам. Проще говоря, скрам-мастер занят коммуникацией между теми, кто не занят в проекте, и людьми, пишущими код. Иногда различные команды кодеров в одной крупной компании общаются и координируют действия на общих собраниях скрам-мастеров этих команд.

### События в Scrum

Есть 5 разновидностей scrum-событий:

Sprint — наиболее значимая часть Scrum. В нее входит планирование спринта, ежедневные стендапы (daily scrum), обзор и ретроспектива спринта.

Sprint Planning. В составлении плана будущего спринта принимают участие все участники скрам-команды. Именно здесь презентуют идею продукта и каждый участник команды может высказать свое мнение, что он думает по этому поводу. Затем на встрече определяются приоритеты и оглашаются дедлайны.

Daily Scrum — ежедневное короткое скрам-событие, длительностью не более 15 минут. Обычно оно проводится для планирования работы кодеров на сегодняшний или завтрашний день. На Daily Scrum можно обсудить текущие проблемы.

Участвовать в таком рабочем совещании обязаны все разработчики, занятые в проекте. Присутствие скрам-мастера допускается, но не обязательно.

Sprint Review (Demo) — показ результатов, созданных во время спринта. Обычно это событие проходит на завершающем этапе. В нем принимают участие все заинтересованные лица.

Sprint Retrospective — обсуждение результатов спринта. Участники команды делятся мнением относительно того, как они справилась с поставленными перед ними задачами и как улучшить результаты работы в дальнейшем.

Помимо этого иногда проводится уточнение бэклога — Backlog Refinement. На нем обсуждаются элементы бэклога, подготовка к будущему спринту и приоритеты текущих задач.

## Артефакты

Артефакты Scrum — это работа, выполнение которой происходит при завершении проекта или спринта. Есть три артефакта — бэклог продукта, бэклог спринта и инкремент. Каждый из них нужен для своевременной поставки ПО пользователям. Есть также вспомогательные артефакты (берн-даун чарты и другое).

### Компоненты, входящие в артефакты спринта:

Бэклог продукта — функции интерфейса и бэкенда.

Бэклог спринта — список заданий, которые требуется сделать во время итерации. Их согласовывают перед началом спринта.

Инкремент — общее количество элементов бэклога ПО, созданных во время спринта, и ценность инкрементов, которые были сделаны до него. Готовый новый инкремент нужно показать перед завершением спринта. Это означает наличие рабочей версии, соответствующей требованиям scrum-команды.

Элемент бэклога продукта — его необходимо выполнить за время итерации спринта. Как правило, элемент разбивают на несколько мелких заданий.

Цель спринта — задачи, которые необходимо выполнить (создание элемента бэклога или другое задание).

Берн-даун чарт спринта — работа, оставшаяся до окончания спринта. Берн-даун чарт бывает восходящим или нисходящим. Все зависит от трудностей, с которыми члены команды сталкиваются во время работы. Он не показатель прогресса, а лишь способ решения проблем и стимул.

Релиз продукта/берн-даун чарт продукта — график, который перед завершением очередного спринта рисует скрам-мастер. Горизонтальная ось — это спринты, вертикальная — количество оставшейся работы.

## Правила скрам-фреймворка

Роли, события и артефакты — основа Scrum, но кроме этого есть и иные правила. Все они усиливают эффективность процесса работы. Вот перечень этих правил:

- В scrum-команду входят заказчик ПО, scrum-мастер и разработчики.
- Все спринты должны быть одинаковыми по длительности.
- После завершения одного спринта сразу начинается работа над новым.
- Спринт всегда начинается с согласования плана.
- Участники команды в начале рабочего дня проводят утренний скрам.
- Во время каждого спринта выполняется его обзор. Это улучшает взаимодействие между командой и заинтересованными лицами.
- Не рекомендуется изменять бэклог спринта во время спринта.

## Ограничения в Scrum

Наряду с очевидными плюсами, у Scrum есть и недостатки:

- Scrum часто приводит к уменьшению объема выполненной работы по причине отсутствия общего дедлайна.
- При низкой вовлеченности или неготовности к сотрудничеству среди участников проекта есть немалые шансы провалить результат.

- Структуру Scrum сложно использовать в крупных командах, но все же это реально. Для этого есть фреймворки масштабирования: LeSS, SAgile, Nexus и другие.
- Уход одного или нескольких участников из команды в середине проекта не очень хорошо влияет на проект.

[← Предыдущая лекция](#)

[Следующая лекция →](#)

−

+13

+

Комментарии

популярные

новые

старые

JavaCoder

Введите текст комментария



У ЭТОЙ СТРАНИЦЫ ЕЩЕ НЕТ НИ ОДНОГО КОММЕНТАРИЯ

ОБУЧЕНИЕ

[Курсы программирования](#)

[Курс Java](#)

[Помощь по задачам](#)

[Подписки](#)

[Задачи-игры](#)

СООБЩЕСТВО

[Пользователи](#)

[Статьи](#)

[Форум](#)

[Чат](#)

[Истории успеха](#)

[Активности](#)

КОМПАНИЯ

[О нас](#)

[Контакты](#)

[Отзывы](#)

[FAQ](#)

[Поддержка](#)

JavaRush — это интерактивный онлайн-курс по изучению Java-программирования с нуля. Он содержит 1200 практических задач с проверкой решения в один клик, необходимый минимум теории по основам Java и мотивирующие фишки, которые помогут пройти курс до конца: игры, опросы, интересные проекты и статьи об эффективном обучении и карьере Java-девелопера.

ПОДПИСЫВАЙТЕСЬ

ЯЗЫК ИНТЕРФЕЙСА

Русский 

СКАЧИВАЙТЕ НАШИ ПРИЛОЖЕНИЯ

