

## Лекция 3: Scrum

Юрий Литвинов

y.litvinov@spbu.ru

13.03.2025

# Scrum

- ▶ Процессный фреймворк для разработки программных продуктов
- ▶ Не акроним (на самом деле, что-то про командную работу из рэгби)
- ▶ Ken Schwaber/Jeff Sutherland, 1995 (на самом деле, аж в 1986 году)
- ▶ Инкрементально-итеративный эмпирический подход
  - ▶ Источник знаний — опыт
- ▶ Прозрачность, инспекции, адаптация

# Роли в команде

- ▶ Основные

- ▶ Product owner
- ▶ Scrum master
- ▶ Команда разработки

- ▶ Вспомогательные

- ▶ Пользователи
- ▶ Заказчики
- ▶ Вспомогательный менеджмент
- ▶ Эксперты, консультанты



# Product owner

- ▶ Представляет интересы пользователей
  - ▶ Product manager, project manager, представитель заказчика, ...
  - ▶ Заботится о генерации ценности для пользователя
- ▶ Формирует видение проекта и доносит его команде
- ▶ Предоставляет требования
- ▶ Приоритезирует задачи
- ▶ Обеспечивает понятность и ясность задач
- ▶ Отвечает за приёмку кода в конце итерации
- ▶ Единая точка принятия решений о проекте
  - ▶ Отдельный человек
- ▶ Часть команды, работает у исполнителя
- ▶ НЕ начальник



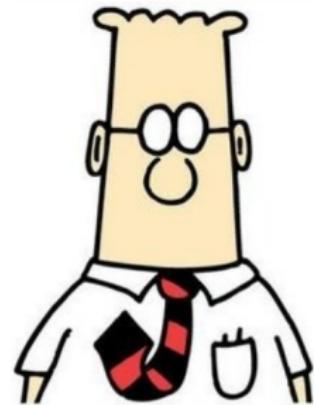
# Scrum master

- ▶ Ответственен за функционирование Scrum-процесса
  - ▶ Следит за соблюдением принципов
  - ▶ Организует митинги, организует процессы, считает метрики и т.п.
  - ▶ Разрешает противоречия
  - ▶ Защищает команду от внешних факторов
- ▶ Лидер-служитель, не имеет особой власти
  - ▶ Коуч, фасилитатор, наставник, тренер, вдохновитель, ...
- ▶ Помогает команде самоорганизовываться
  - ▶ Не управляет командой



# Команда разработки

- ▶ Те, кто на самом деле пишет код
- ▶  $7 \pm 2$  человек
  - ▶ Все в одном месте
  - ▶ Принцип двух пицц
- ▶ Самоорганизация
- ▶ Кроссфункциональность
- ▶ Коллективная ответственность
  - ▶ Полный контроль разработки
- ▶ Отсутствие подкоманд



# Product Backlog

- ▶ Упорядоченный источник всех задач
  - ▶ Фичи, баги, нефункциональные требования, мероприятия, ...
- ▶ Единственный источник требований
- ▶ Ведётся Product owner'ом
  - ▶ Задачи оцениваются командой
  - ▶ Совместное уточнение задач
- ▶ Постоянно пополняется
- ▶ Единый для нескольких команд

	A	B	C	D	E	Team Velocity	F
1	Product Backlog						25
2	Priority	Estimate	Sprint	User Type	Story		Story Type
3	1	1	1	Customer	I can see when the next show will begin for the show page I am on		Story
4	2	2	1	Editor	I can select what I want to display for each "section" within the editorial content section of the page. My options include last episode, next episode, selected forum posts, selected editorial articles (tvg generated), no selection and free form text		Story
5	3	2	1	Editor	I can select what picture (if any) I want to display for the corresponding content section		Story
6	4	5	1	Editor	I can select the default tab for the user to see upon visit to the page, for each show		Story
7	5	5	1	Customer	I can roll over the fields in the media player and see the various tabs change		Story
8	6	13	2	Editor	I can modify the existing headline for any show page		Story
9	7	1	2	Customer	I can select another show page in the drop down list next to the countdown clock		Story
10	8	1	2	Customer	I can click "remote record" and have the show for the show page I am on record on my two device		Story
11	9	1	2	Customer	I can click "join the discussion" button (or link) on the show page which takes me to the appropriate forum page for that show		Story
12	10	1	2	Customer	I can see how many recent posts have been posted in the forum for the show page I am on		Story
13	11	3	2	Customer	I can see how many recent replies have been posted in the forum for the show page I am on		Story
14	12	5	2	Customer	I can blog about the show for the show page I am on (I need to be signed in to see this)		Story
15	13	13	3	Customer	If I am not signed in, I can see a link to sign in		Story
16	14	13	3	Customer	If I am logged in, I can click "favorites" and have the show page added to my favorites menu on the site		Story
17	15	13	4	Customer	If I have not contributed to the poll, I can see the poll questions and submit to the poll		Epic

Практики

The screenshot shows the TrikStudio (Public) application interface. On the left, there's a sidebar with various icons. The main area has two boards:

- Current Iteration/Backlog**: A Kanban board with columns for '0 of 10 points' and '383 + 12 - 18 Jul'. It contains several stories, some with 'Start' and 'Finish' buttons. Stories include:
  - EV3: Глажение листа при закрытии среды при безуспешных попытках подключения [20]
  - EV3: Интерпретация по lab как-то работает [20]
  - В Данных возможность послать сообщение на робот [20]
  - надеждка в онлайн [20]
  - Замедлить поддержку котиков [20]
  - Хотите в папире [20]
  - перевод
  - Сделать более user-friendly документацию по системе проверки заданий  
алгоритма или проверка заданий
  - Implementardt installer
- Icebox**: A Kanban board with columns for '0 Add Story' and 'Start' buttons. It contains stories like:
  - Автономный режим Leo на Linux
  - Погребовать сборку на QBS
  - Погребовать собирать Рэйк
  - Избавится от запуска под рутом на линуксе
  - Добавить подгружаемую пишеру и переседу
  - Добавить группу ev3-motors, с дополнительными опциями типа повернуть на угол и прочее (есть арты)
  - Добавить файл скрип, для выдачи, чтобы выполнено в стиле 10-ки
  - Android легкое приложение с выдачей блоками
  - Обновление всех третьясторонних компонент
  - Сборка инсталляторов с помощью Cl
  - Масштабирование в 2d с сохранением пропорций 2d-элементов
  - Разгон/погремушки
  - Проецировать на
  - Хотела комментарии и тд текстовый режим
  - Подсветка js 2d проплат?
  - Плагин для ЦНИИ РТК, базовые блоки цикла пре этап
  - Базовая генерация в C++

# Спринты

- ▶ Итерация в 1-4 недели
- ▶ Результат — готовая версия (инкремент)
  - ▶ Или хотя бы что-то осязаемое
- ▶ Фиксируется объём работ, практически не меняется во время спринта
- ▶ Жизненный цикл спринта
  - ▶ Планирование
  - ▶ Разработка
  - ▶ Демо
  - ▶ Ретроспектива

# Планирование спрингта

- ▶ 3-8 часов
- ▶ Определение целей спрингта и набора задач
  - ▶ Из бизнес-задач формируется sprint backlog
  - ▶ Обычно просто как верхушка Product backlog, которая лезет по объёму в спрингт
- ▶ Оценка и упорядочивание задач
  - ▶ Декомпозиция, добавление технических задач
  - ▶ Planning poker
- ▶ Коллективная оценка — команда не знает, кто какую задачу будет делать
- ▶ Оценка скорости команды (team velocity)

# Planning poker

- ▶ Условные единицы, не привязанные напрямую к линейному времени
  - ▶ team velocity позволяет весьма точно пересчитать в линейный срок
- ▶ Фибоначчиева шкала (1 — сейчас сяду и сделаю, держите меня семеро; 8 — без идей как делать)
- ▶ Каждый член команды выбирает свою оценку самостоятельно
- ▶ Потом оценки оглашаются и берётся среднее (почему и покер)
- ▶ Если есть сильные расхождения, задачу обсуждают, может, декомпозириуют, и повторяют голосование
- ▶ И так по каждой задаче

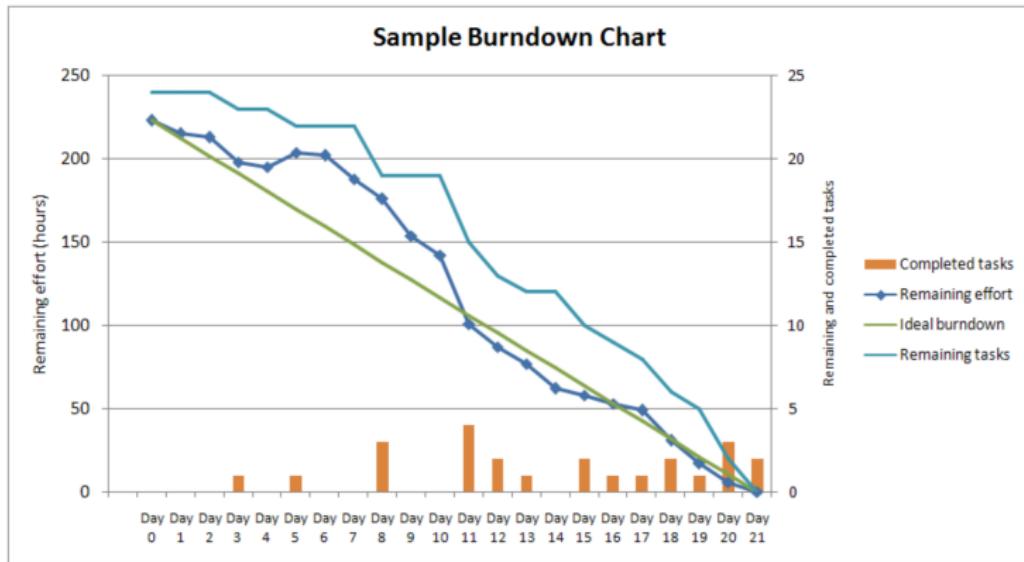


# GitHub Projects

The screenshot shows the GitHub Projects interface for the repository `yuriii-litvinov / REAL.NET`. The interface is divided into three main columns: `TODO`, `In progress`, and `Done`.

- TODO:** Contains three items:
  - Сделать уже нормальное название для проекта :)
  - Зафиксировать ширину панелек справа, иначе если там что-то широкое, то панелька растягивается на полэкрана
  - Drag-and-drop-рисование линков. Так привычнее, чем как сейчас.
- In progress:** Contains one item:
  - Редактирование свойств. В качестве первого варианта все свойства можно считать строками и давать их редактировать просто как строки. Более продвинутый редактор свойств должен уметь содавать разные редакторы для разных типов.
- Done:** Contains zero items.

At the bottom right, there is a button labeled `+ Add column` to add more columns to the board.



# Ежедневные встречи

- ▶ Ежедневно, не более 15 минут
- ▶ Всегда в одно и то же время в одном и том же месте
- ▶ Участвуют только члены команды разработки
- ▶ Три вопроса каждому
  - ▶ Что сделал?
  - ▶ Что буду делать?
  - ▶ Какие есть сложности?
- ▶ Только делимся информацией, не решаем проблемы
- ▶ Налаживание связей и распространение информации в команде
  - ▶ Ключевое мероприятие для инспекции и адаптации
- ▶ Scrum of Scrums

# Ревью спринта/демо

- ▶ Не более 4 часов, в конце спринта
  - ▶ Плюс подготовка, не более двух часов
- ▶ Неформальное мероприятие
- ▶ Демонстрация реализованного инкремента
- ▶ Обсуждение
  - ▶ Что делали, что сделали
  - ▶ Что не сделали и почему
  - ▶ Как улучшить процесс далее
- ▶ Приглашаются все участники проекта
  - ▶ Обратная связь
- ▶ Результат — обновлённый product backlog

# Ретроспектива спрингта

- ▶ 1-3 часа, в конце спрингта
  - ▶ После демо и до планирования следующего спрингта
- ▶ Команда разработки и Scrum master
- ▶ Вопросы:
  - ▶ Что было сделано хорошо?
  - ▶ что надо улучшить в следующем спрингте?

## The Agile: Scrum Framework at a glance

Inputs from Executives,  
Team, Stakeholders,  
Customers, Users



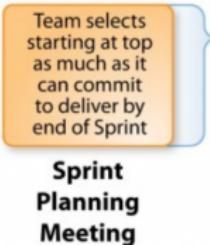
Product Owner



The Team



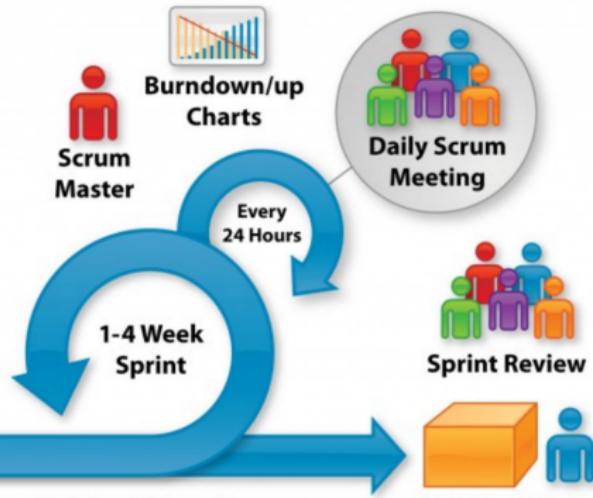
Product Backlog



Sprint Planning Meeting



Sprint Backlog



# ScrumAnd (Scrum++)

- ▶ Нулевой спринт
- ▶ Практики XP
- ▶ Ревью кода
- ▶ Высокий процент test/code coverage
- ▶ All the team testing
- ▶ Кроссфункциональные работники
- ▶ Недельные спринты
- ▶ Широкая аудитория на демо
- ▶ Автоматизация

## ScrumBut (Scrum- -)

- ▶ Персонализация задач/спринтов/багов
- ▶ “водопад” внутри спрингта
- ▶ Ориентация на тулы вместо прямой коммуникации
- ▶ 6-12-недельные спрингты, перерывы между спрингтами
- ▶ Big Design Up-Front (BDUF)
- ▶ Отсутствие Scrum master'a
- ▶ Демо по принципу “Да мы ничего особого не сделали в этом спрингте”
- ▶ Наличие тимлида

ScrumBut	Причина	Решение/Оправдание
Мы используем Scrum, <u>НО</u>	Делать Daily Scrum для нас слишком накладно	Daily Scrum будет только один раз в неделю
Мы используем Scrum, <u>НО</u>	Ретроспективы для нас — пустая тратя времени	Мы решили их не проводить
Мы используем Scrum, <u>НО</u>	Мы не можем доставить инкремент продукта за месяц	Устанавливаем спрингт длиной шесть недель
Мы используем Scrum, <u>НО</u>	Иногда наши менеджеры дают специальные таски	Готовые таски не всегда соответствуют Definition Of Done

# Когда Scrum может работать плохо

- ▶ Fixed-cost/fixed-time проекты
- ▶ Безответственные, низко мотивированные работники
- ▶ Слишком узкоспециализированные работники
- ▶ Неполноставочники
- ▶ Большое количество внешних зависимостей
- ▶ Legacy или системы повышенной надёжности
- ▶ Распределённые команды
  - ▶ Все более-менее научились работать удалённо



Мотофотопрограммист  
@iv4ngrigoryev



– Команда, поздравляю! Мы с вами только что закрыли очередной спринт!

– Но ты же просто взял и перенес все незавершенные задачи в новый спринт в джире...

– Ты че пес, я сертифицед скрам-мастер!

# Альтернативные подходы к разработке

- ▶ Deadline driven development
- ▶ Job security driven development
- ▶ Hate driven development
- ▶ Hope driven development
- ▶ Conference driven development
- ▶ GitHub Stars driven development
- ▶ Résumé driven development