



ГИБКИЕ МЕТОДОЛОГИИ РАЗРАБОТКИ ПО

Лыжин Иван

22.04.2016

Agile Software Development

- Интерактивная разработка
- Динамическое формирование требований
- Постоянное взаимодействие между программистами и с заказчиком

Agile Manifesto

Разработан в феврале 2001 года на встрече 17 независимых практиков нескольких методик программирования в местечке Snowbird в штате Юта.

1. Kent Beck
2. Mike Beedle
3. Arie van Bennekum
4. Alistair Cockburn
5. Ward Cunningha
6. Martin Fowler
7. James Grenning
8. Jim Highsmith
9. Andrew Hunt
10. Ron Jeffries
11. Jon Kern
12. Brian Marick
13. Robert C. Martin
14. Steve Mellor
15. Ken Schwaber
16. Jeff Sutherland
17. Dave Thomas

Ценности Agile

- **Люди и их взаимодействие** важнее процессов и инструментов;
- **Готовый продукт** важнее документации по нему;
- **Сотрудничество с заказчиком** важнее жестких контрактных ограничений;
- **Реакция на изменения** важнее следования плану.

Сравнение с водопадной моделью



Принципы Agile

1. Наивысшим приоритетом является **удовлетворение потребностей заказчика.**
2. Изменение требований **приветствуется.**
3. Продукт следует выпускать **как можно чаще.**
4. Разработчики и представители бизнеса **должны ежедневно работать вместе.**

Принципы Agile

5. Над проектом должны работать **мотивированные профессионалы.**
6. **Непосредственное общение** является наиболее практичным и эффективным способом обмена информацией.
7. **Работающий продукт** — основной показатель прогресса.
8. **Поддерживать постоянный ритм** бесконечно.

Принципы Agile

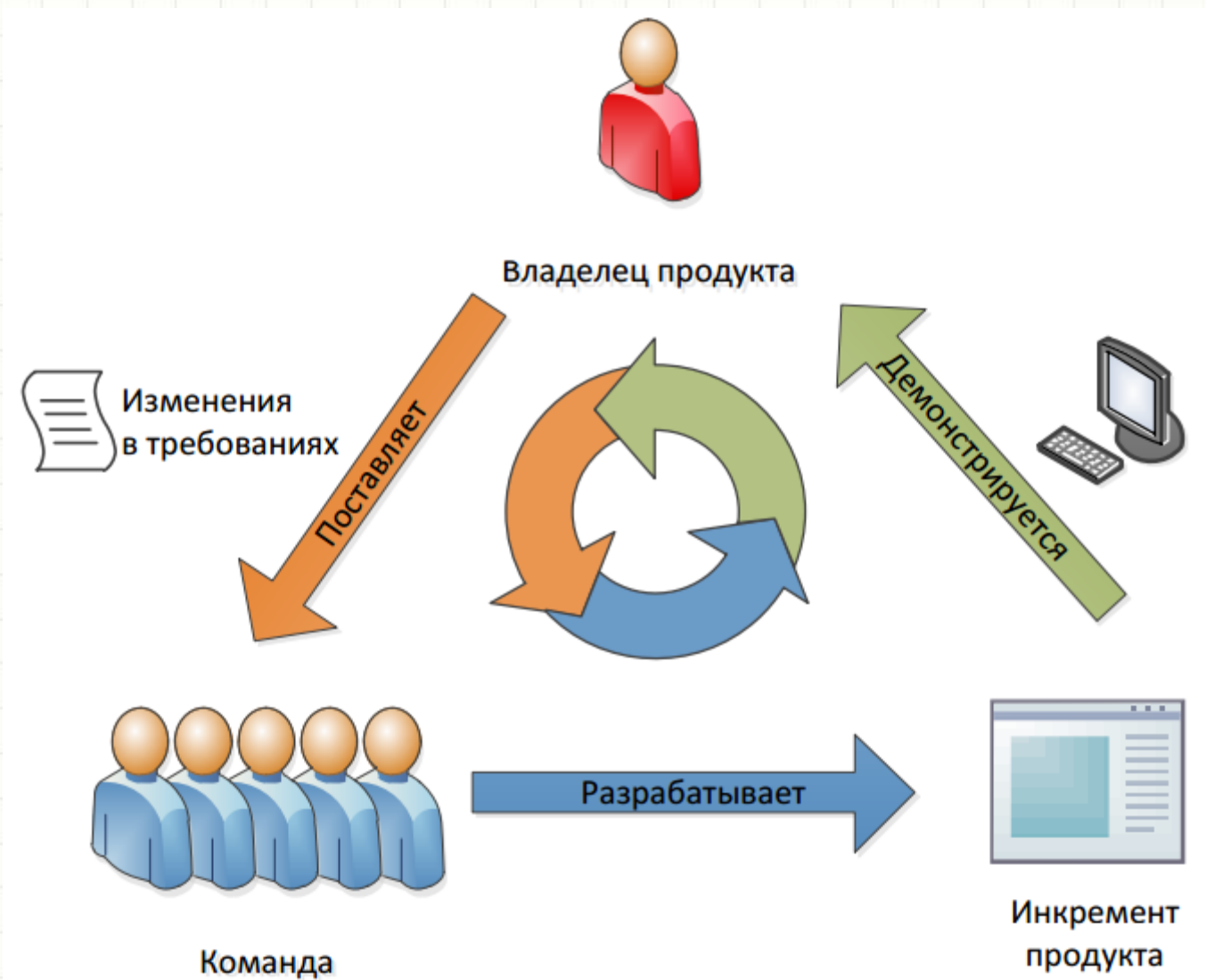
9. Постоянное внимание к **техническому совершенству**.


10. Простота — искусство минимизации лишней работы — крайне необходима.

11. Самоорганизующиеся команды.

12. Команда должна систематически анализировать возможные **способы улучшения эффективности**.

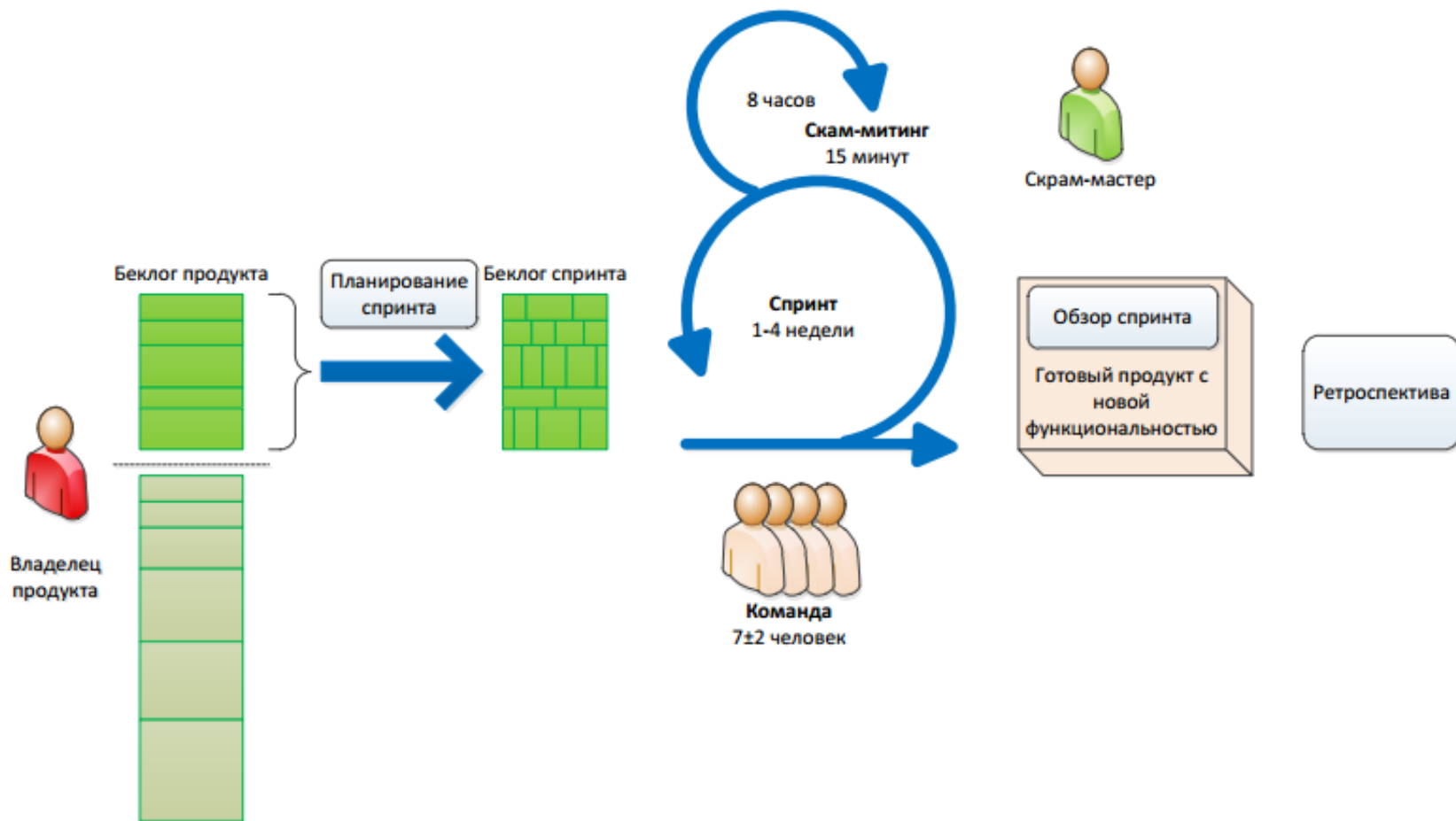
Общий цикл разработки



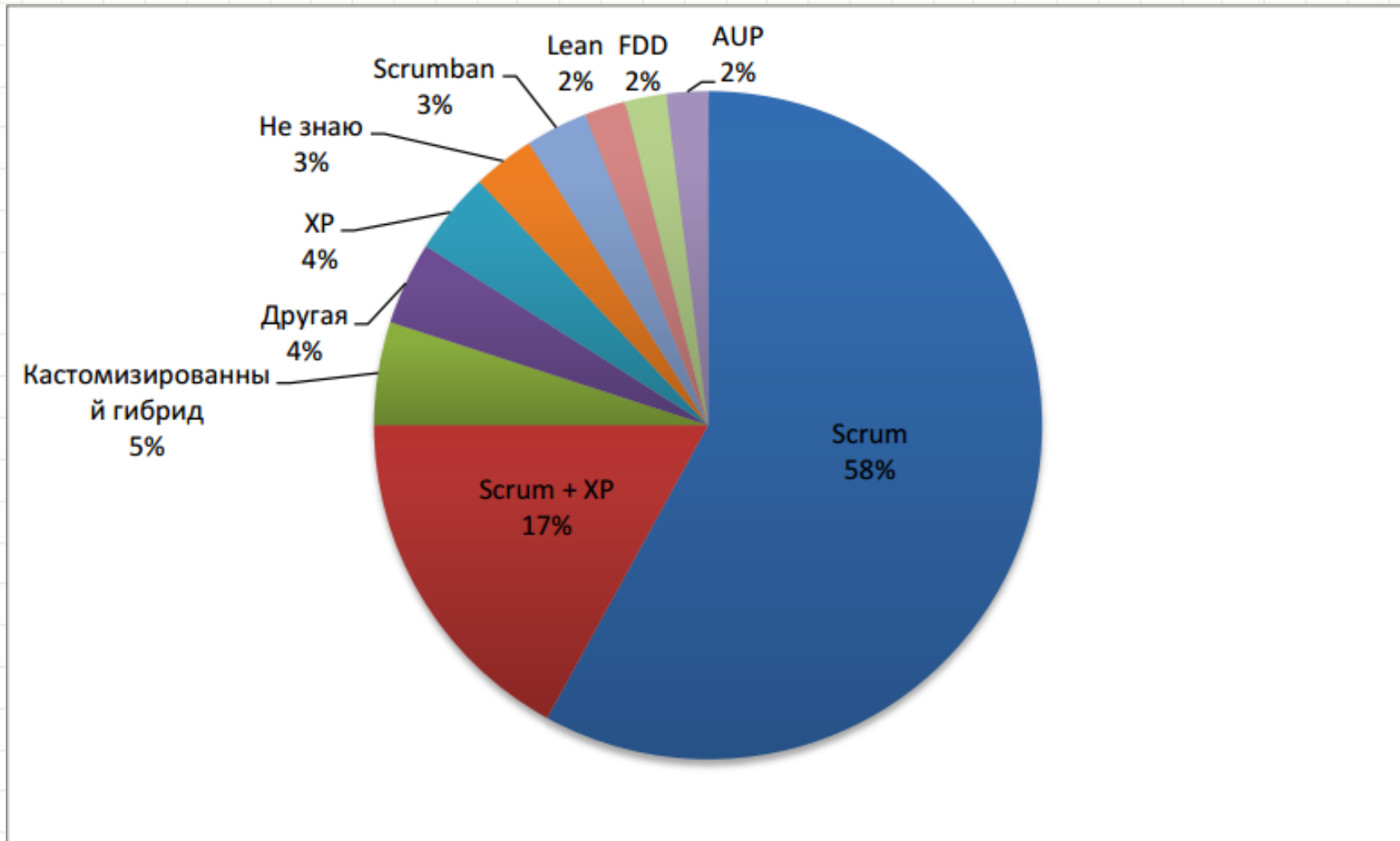
The background features a light gray grid pattern. On the right side, there are several overlapping, wavy blue lines that curve from the top right towards the bottom right, creating a sense of motion and depth. The lines vary in opacity and thickness, with some appearing as thin outlines and others as solid, vibrant blue bands.

Scrum

Схема Scrum



Статистика использования гибких методологий



Компоненты Scrum

Роли

- Владелец продукта
- Скрам-мастер
- Команда

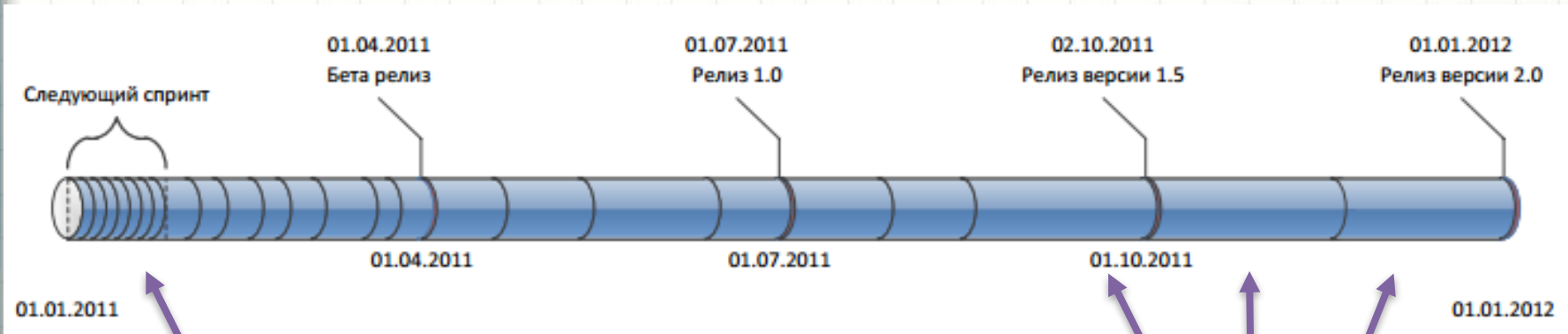
Артефакты

- Беклог продукта
- Беклог спринта
- Инкремент продукта

Процессы

- Планирование спринта
- Обзор спринта
- Ретроспектива
- Скрам-митинг
- Спринт

Беклог и метод набегающей волны



Беклог спринта

Эпики

User Story

Идентификатор

1453

*Неавторизованный
посетитель вводит логин и
пароль для того, чтобы
авторизоваться на сайте*

Название

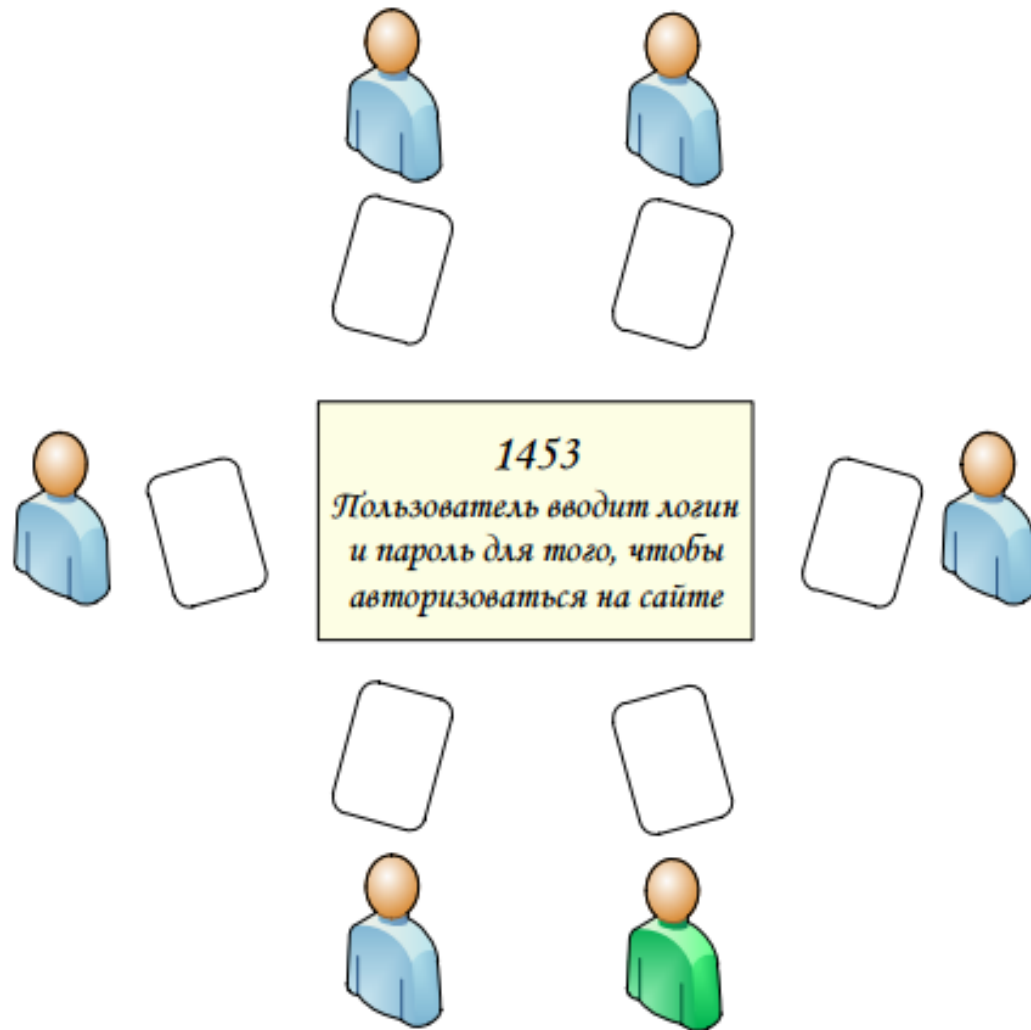
Важность

200

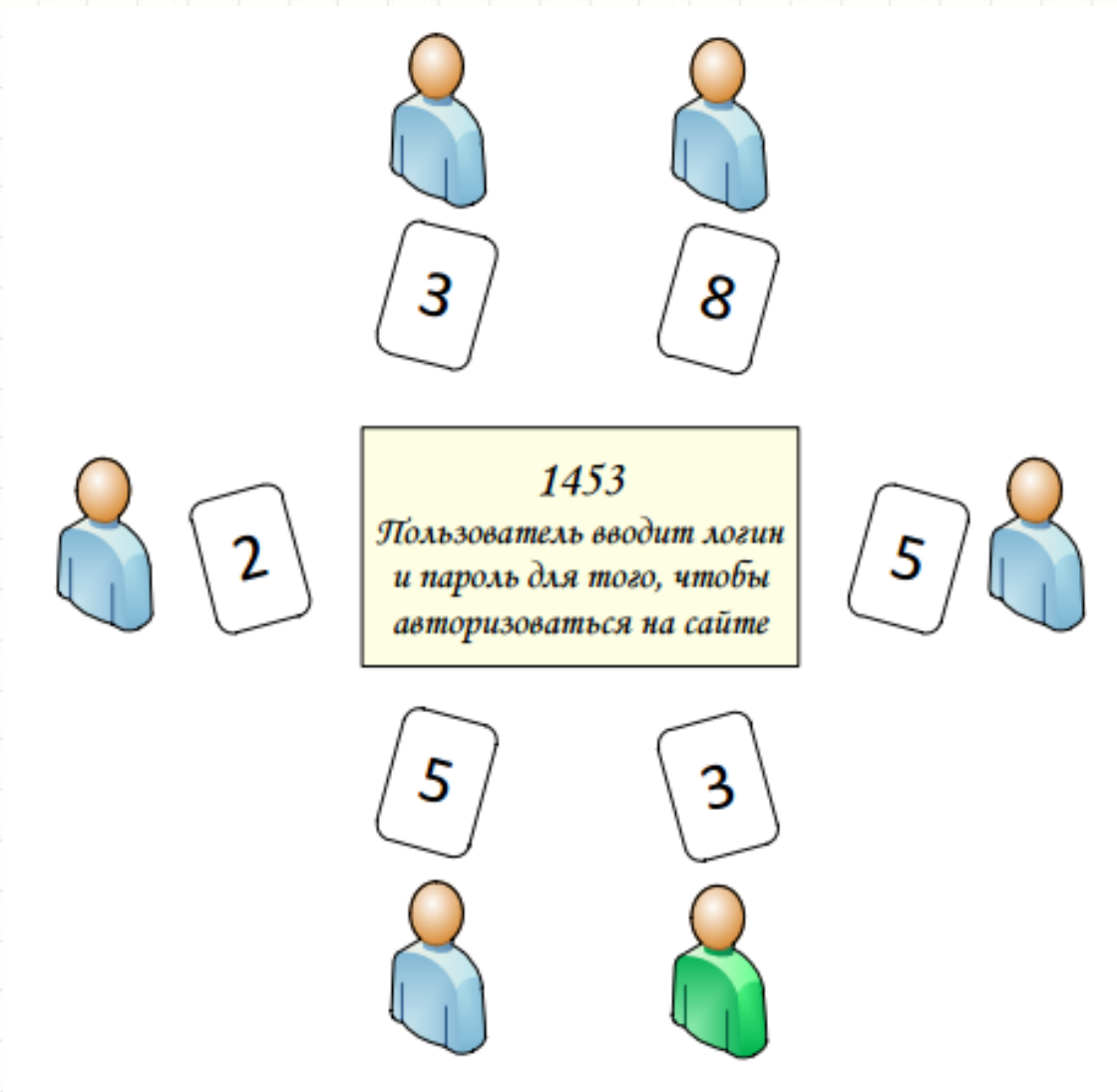
10

Оценка

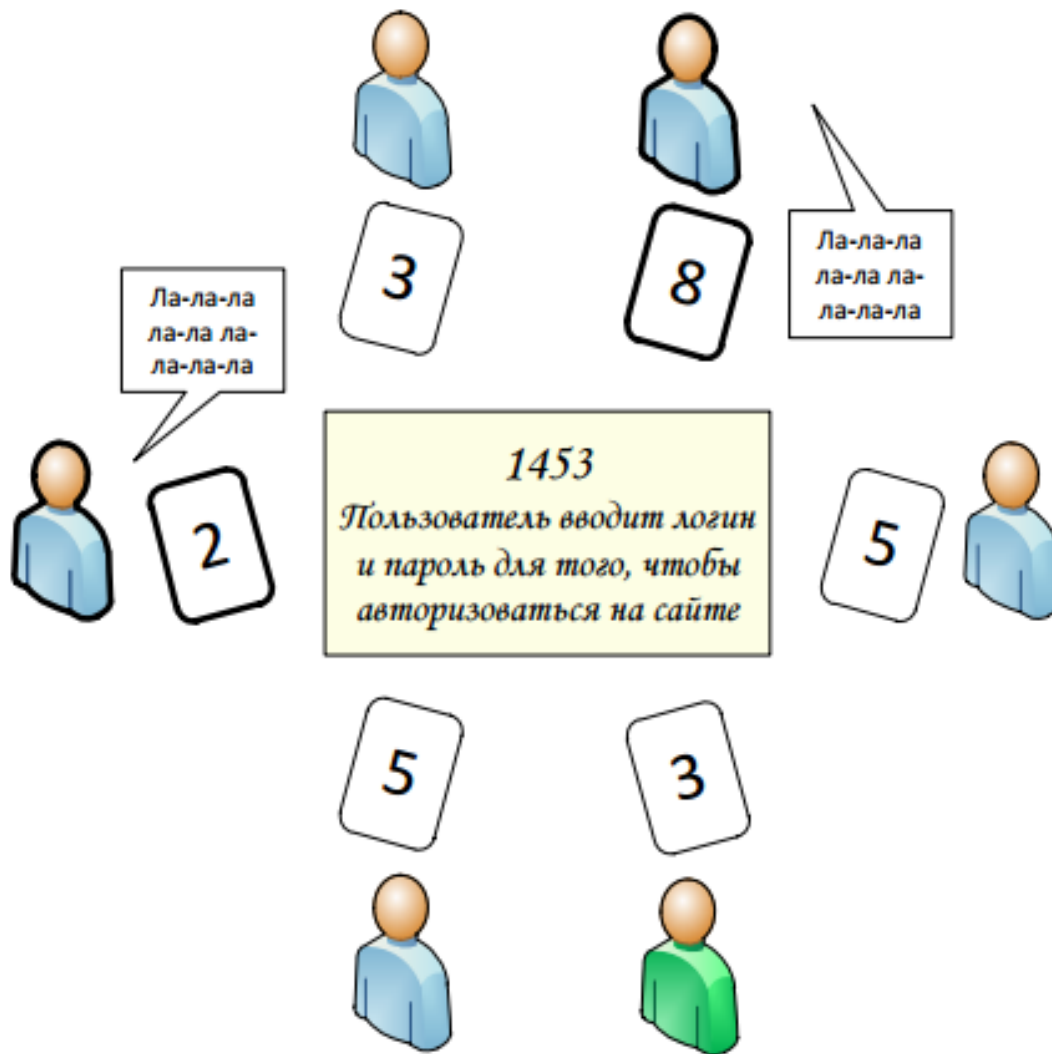
Покер-планирование



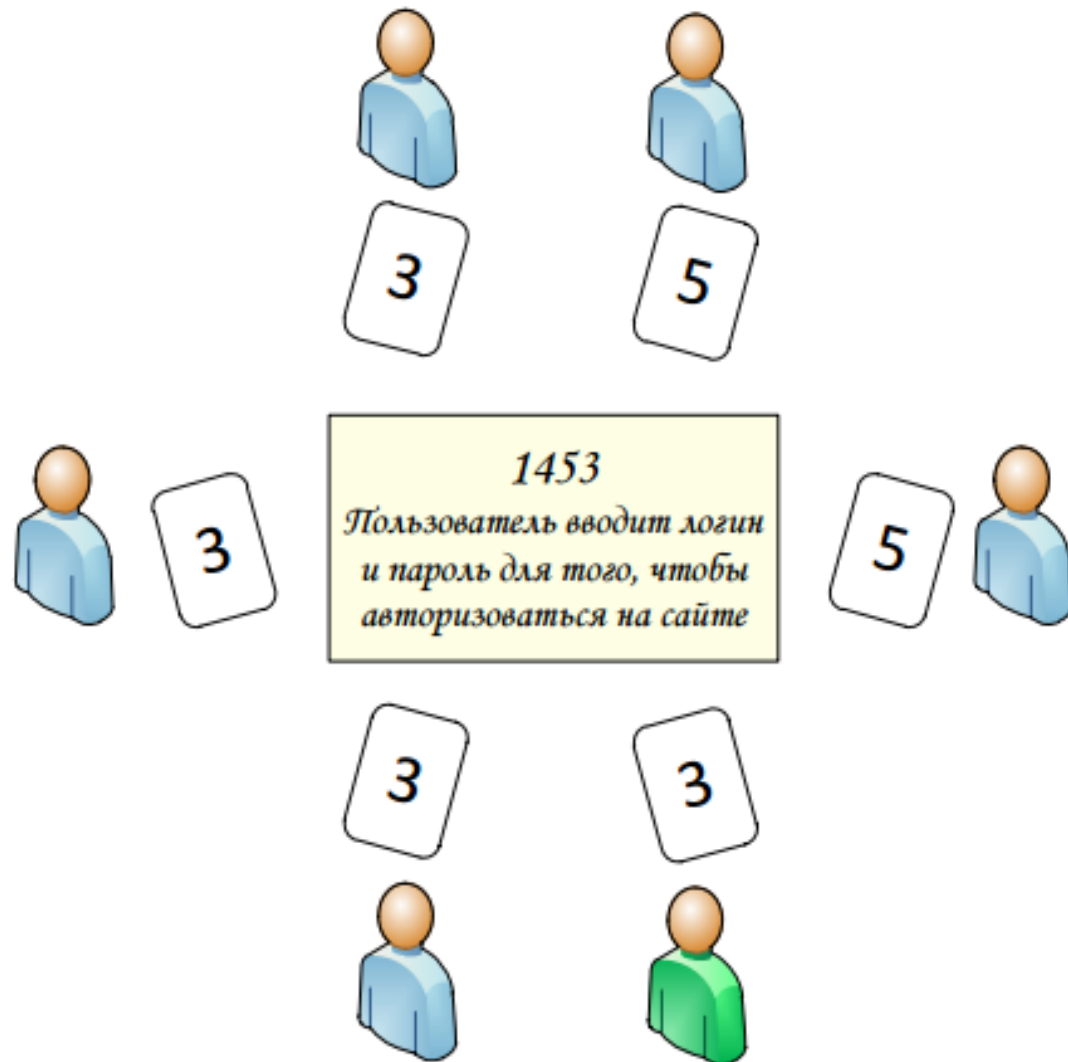
Покер-планирование (1 раунд)



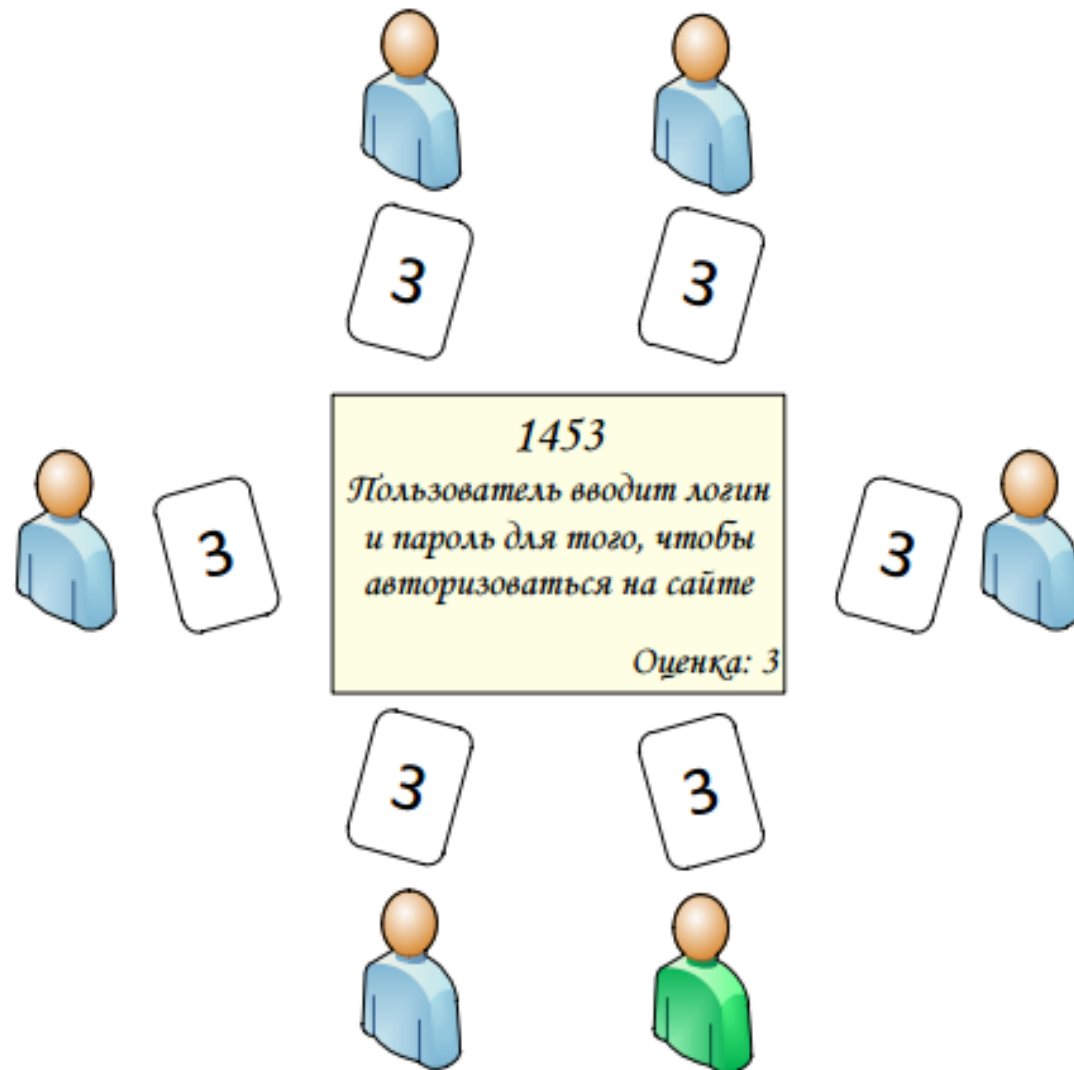
Покер-планирование (обсуждение)



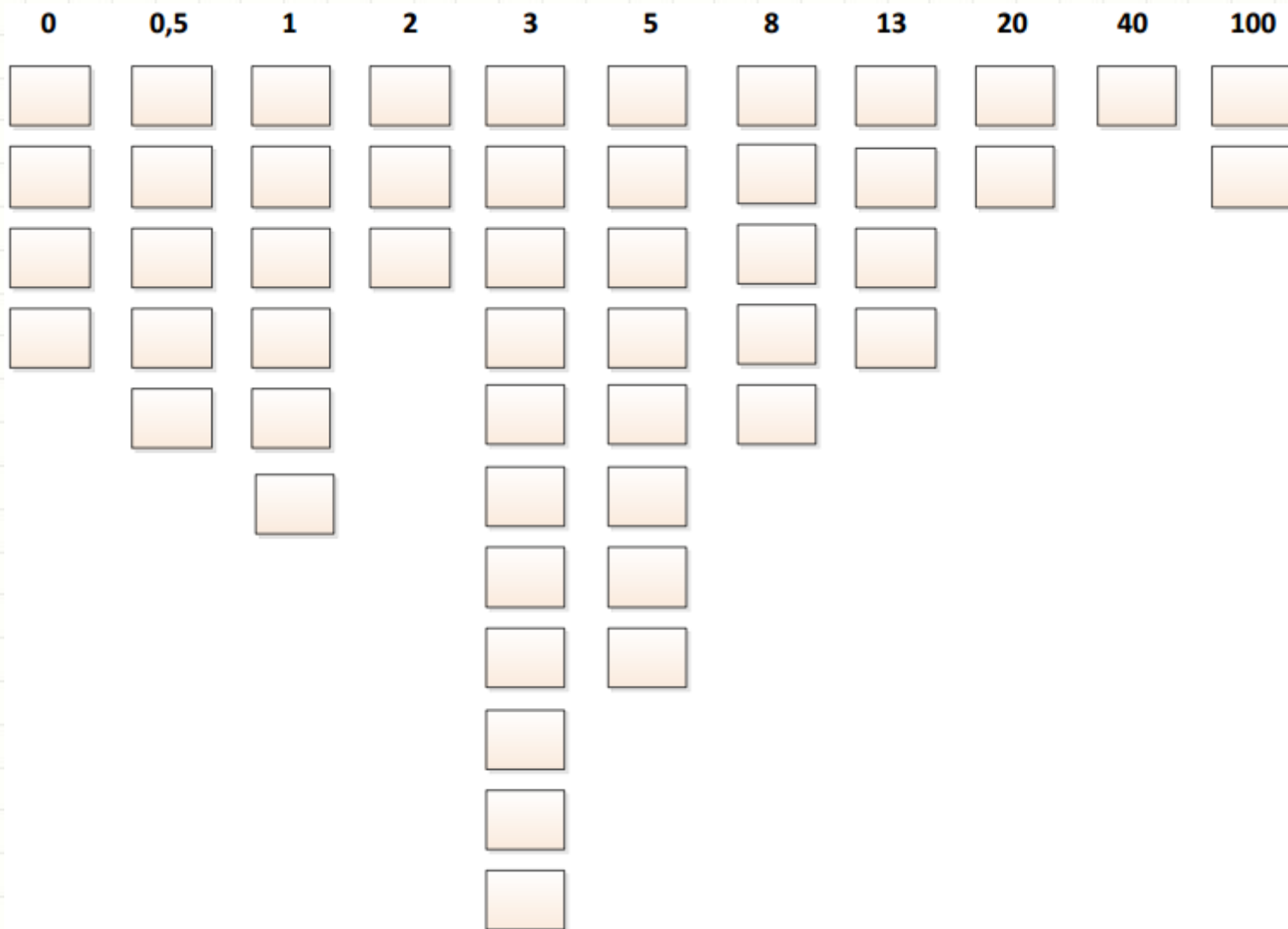
Покер-планирование (2 раунд)



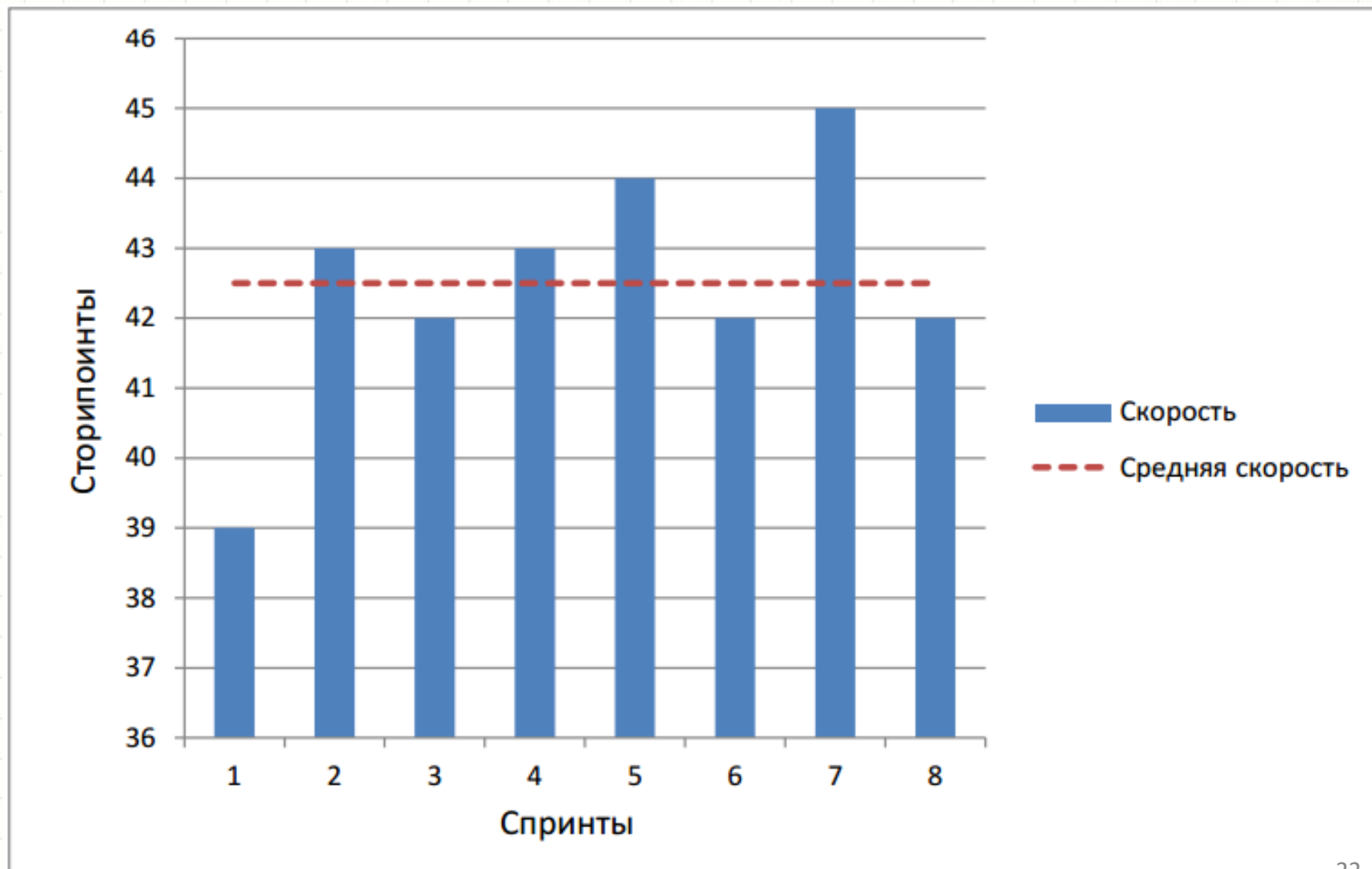
Покер-планирование (3 раунд)



Итоги оценки всех задач



Скорость работы команды



Отбор задач на спринт

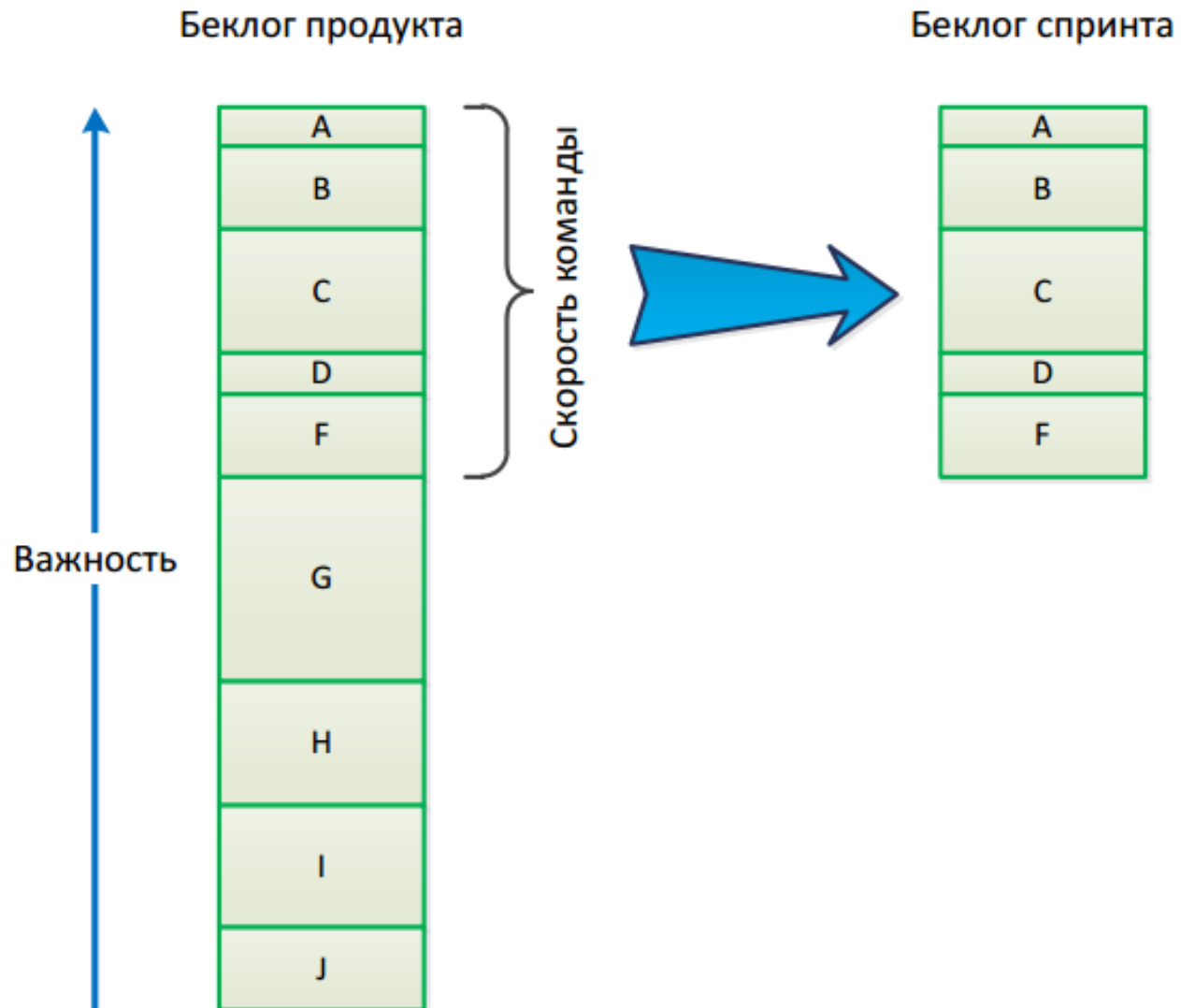
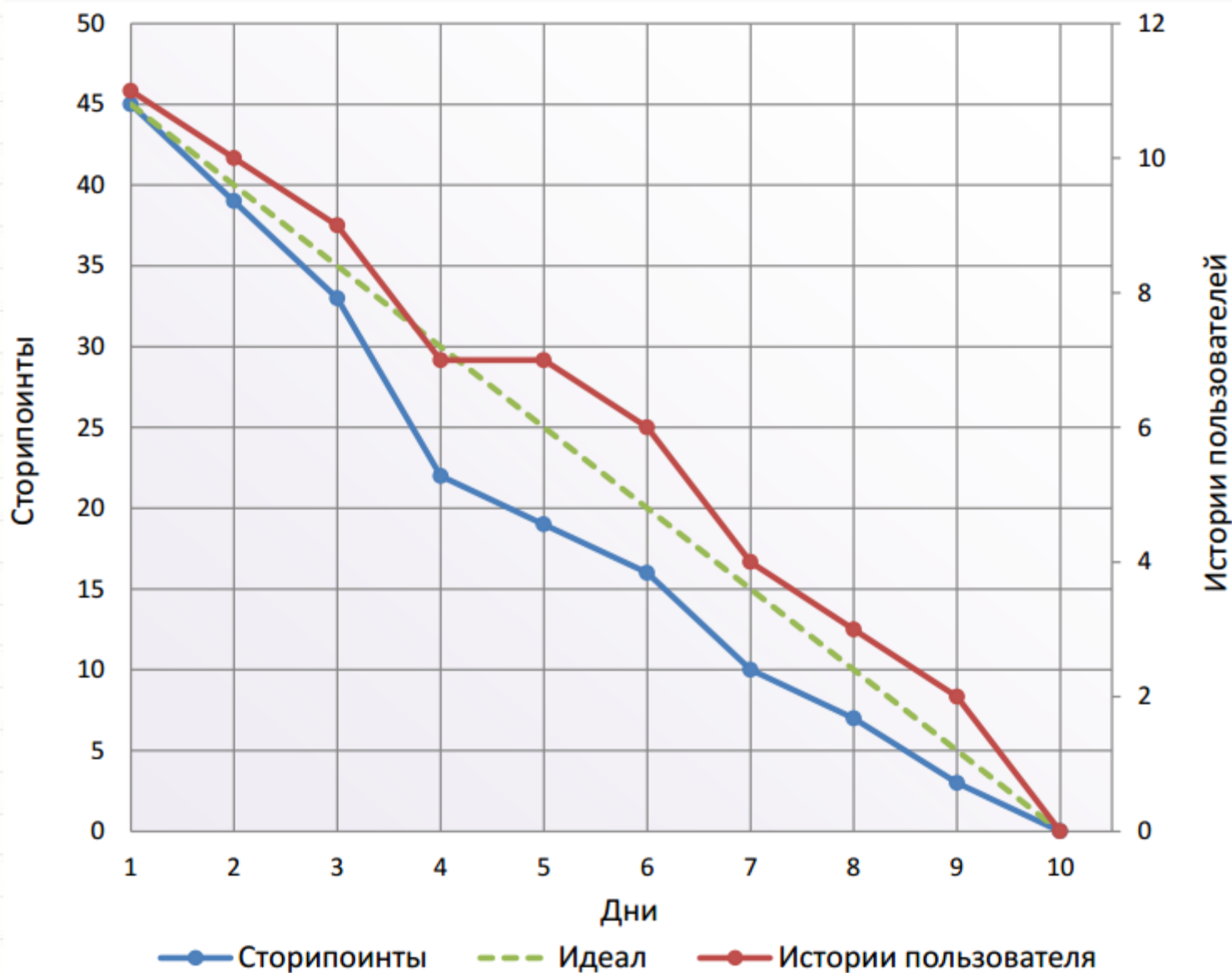



Диаграмма сгорания



The background features a light gray grid pattern. On the right side, there are several overlapping, wavy blue lines that curve from the top right towards the bottom right, creating a sense of motion and depth. The lines vary in opacity and thickness, with some appearing as thin outlines and others as solid, vibrant blue bands.

Kanban

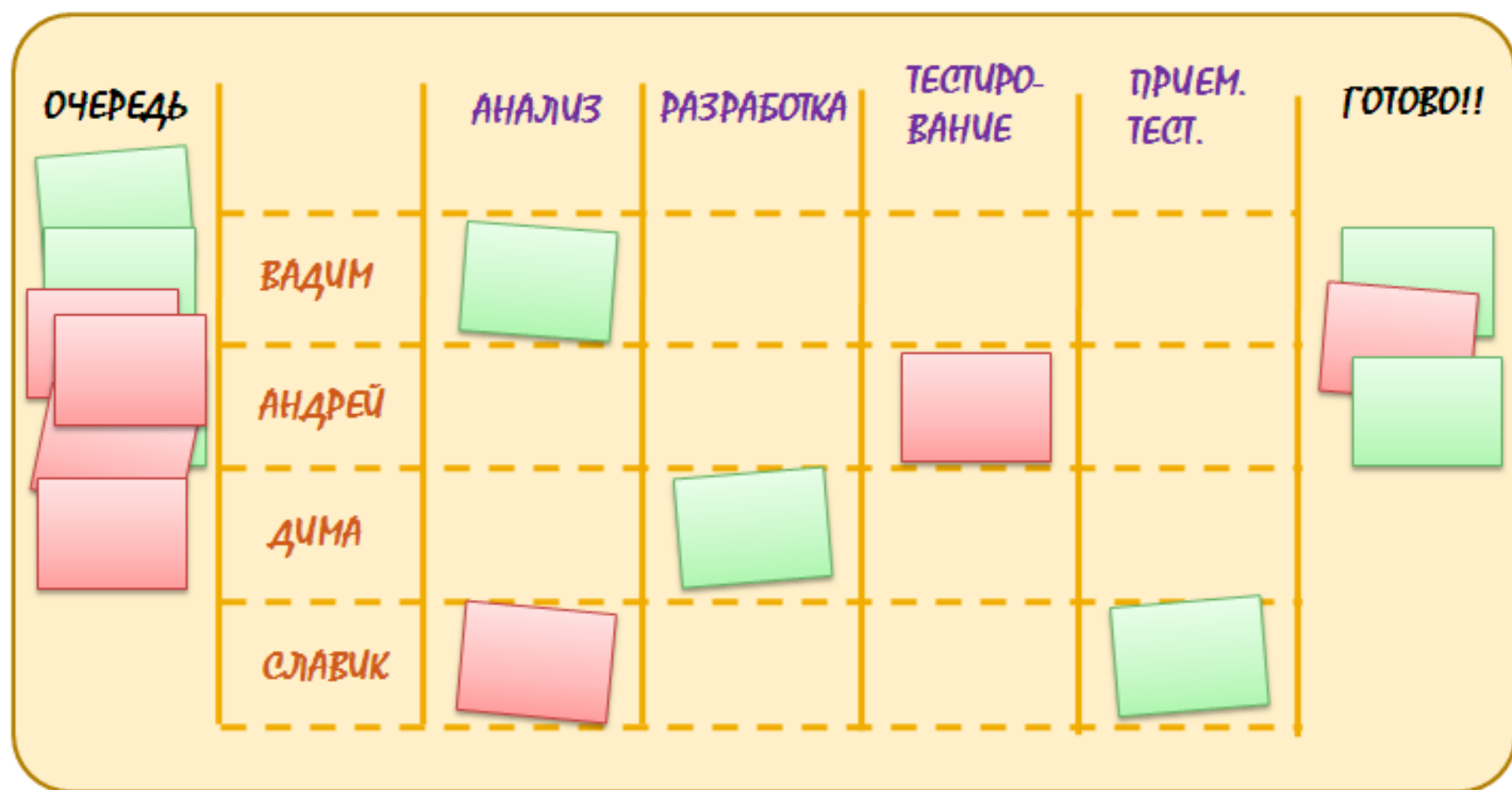
Происхождение

- «Кан» - видимый/визуальный
- «Бан» - карточка/доска
- Появился благодаря производственной системе Тойоты (Toyota Production System)

Принципы Kanban

- 1. Визуализируйте производство**
- 2. Ограничивайте WIP (work in progress)**
- 3. Измеряйте время цикла**

Доска задач в Канбан



The background features a light gray grid pattern. On the right side, there are several overlapping, wavy blue lines that curve from the top right towards the bottom right. The lines vary in opacity, with some being solid blue and others being semi-transparent, creating a sense of depth and movement.

XP

Авторы XP



Kent
Beck



Howard
Cunningham



Martin
Fowler

Практики XP.

1. Разработка через тестирование
2. Игра в планирование
3. Заказчик всегда рядом
4. Парное программирование
5. Непрерывная интеграция
6. Рефакторинг

Практики XP.

7. Частые небольшие релизы

8. Простота

9. Метафора системы

10. Коллективное владение кодом или
выбранными шаблонами проектирования

11. Стандарт кодирования

12. 40-часовая рабочая неделя

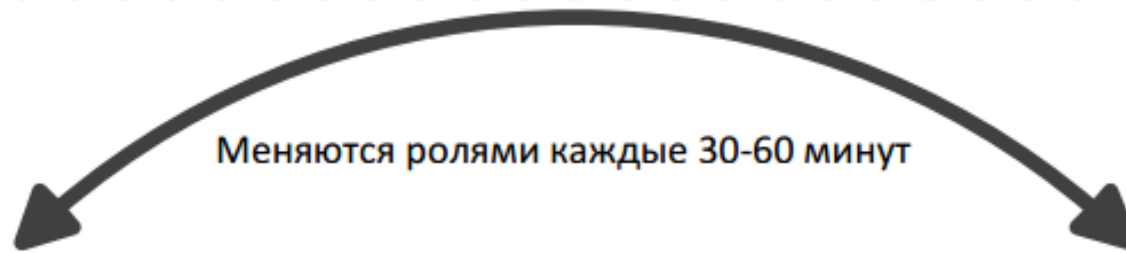
Разработка через тестирование



Рефакторинг



Парное программирование



Пилот
(тактика)

- Реализует методы
- Занимается отладкой
- Кодировать тесты



Штурман
(стратегия)

- Прорабатывает архитектуру на уровне классов
- Проводит инспекцию кода
- Проверяет код на ошибки
- Придумывает тесты



ВОПРОСЫ?