

# Бизнес-анализ

Лекция 4

# Программа на сегодня

- Команда
- Методологии разработки
- Водопадная модель
- Инкрементная модель
- Agile
- Факторы успеха и неудачи проектов

Команда разработки

# Команда по Бруксу (1975)

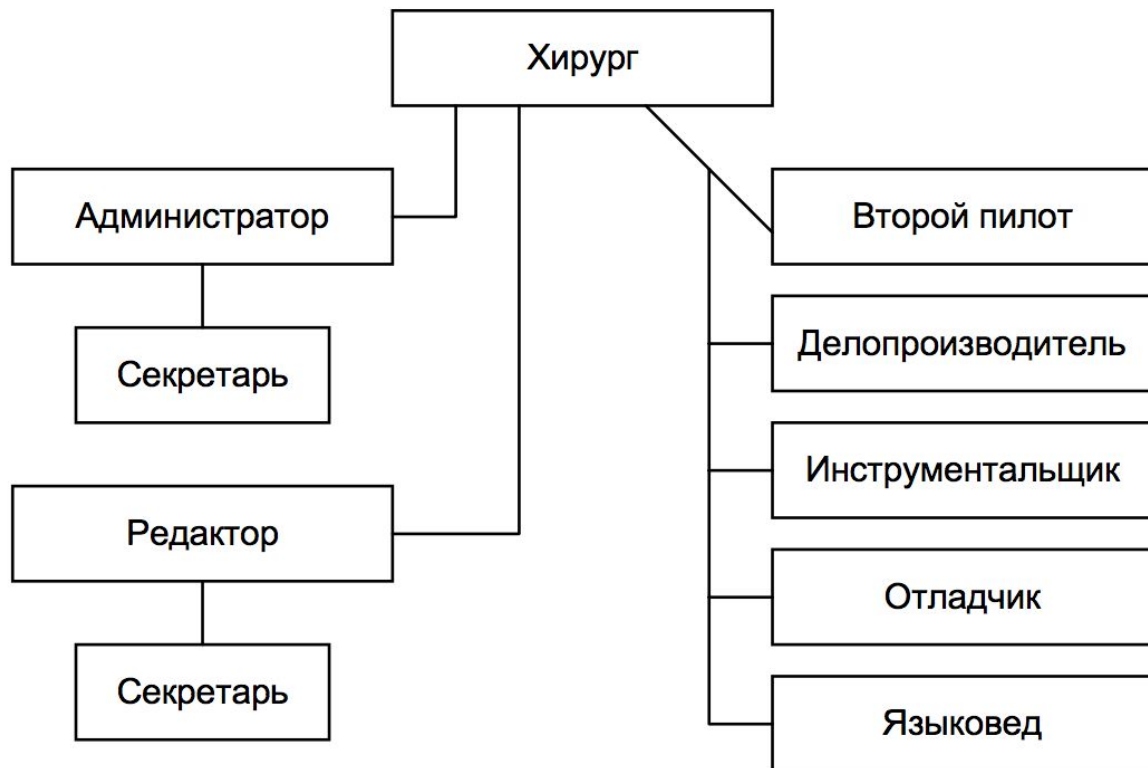
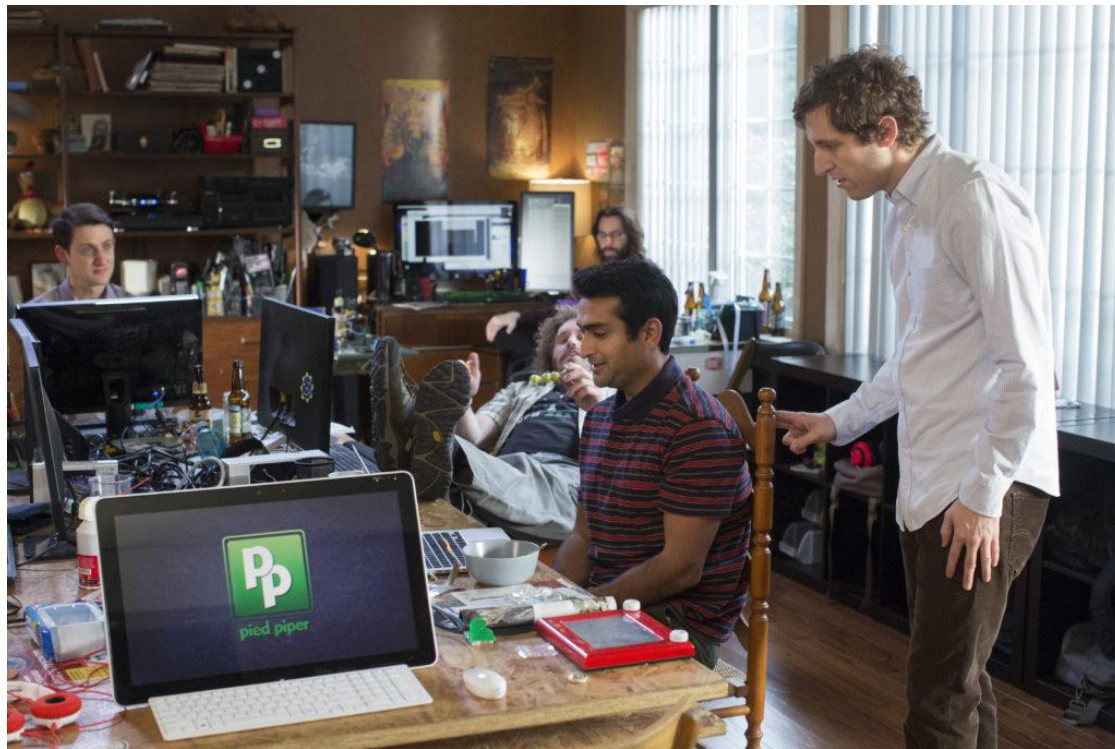


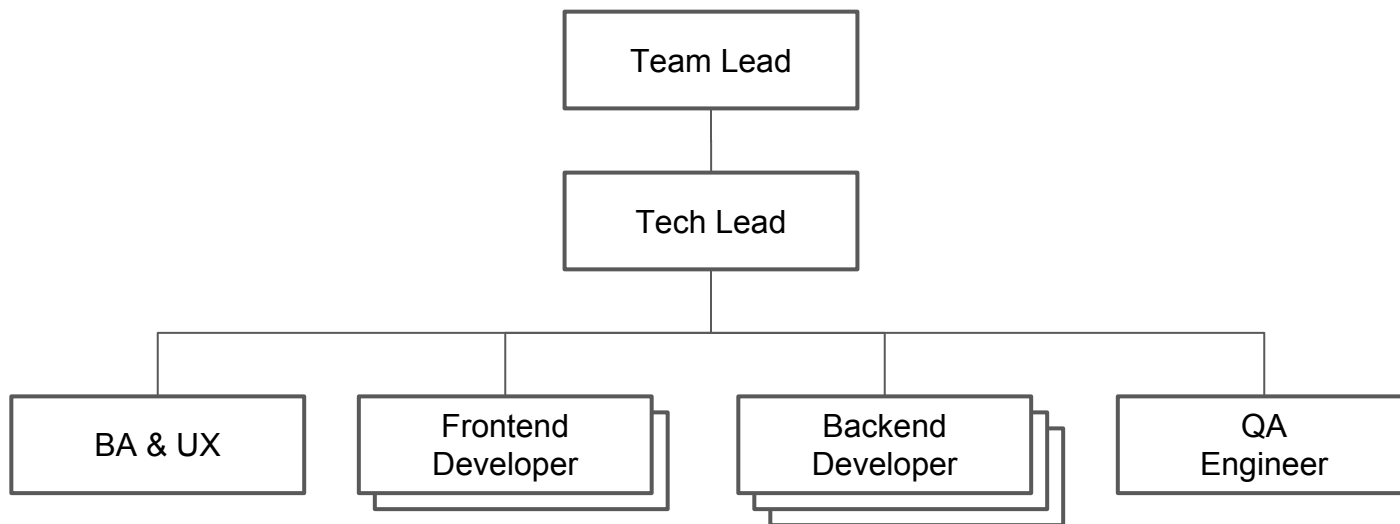
Схема контактов между сотрудниками в бригаде из 10 человек

# Эволюция команды: стартап

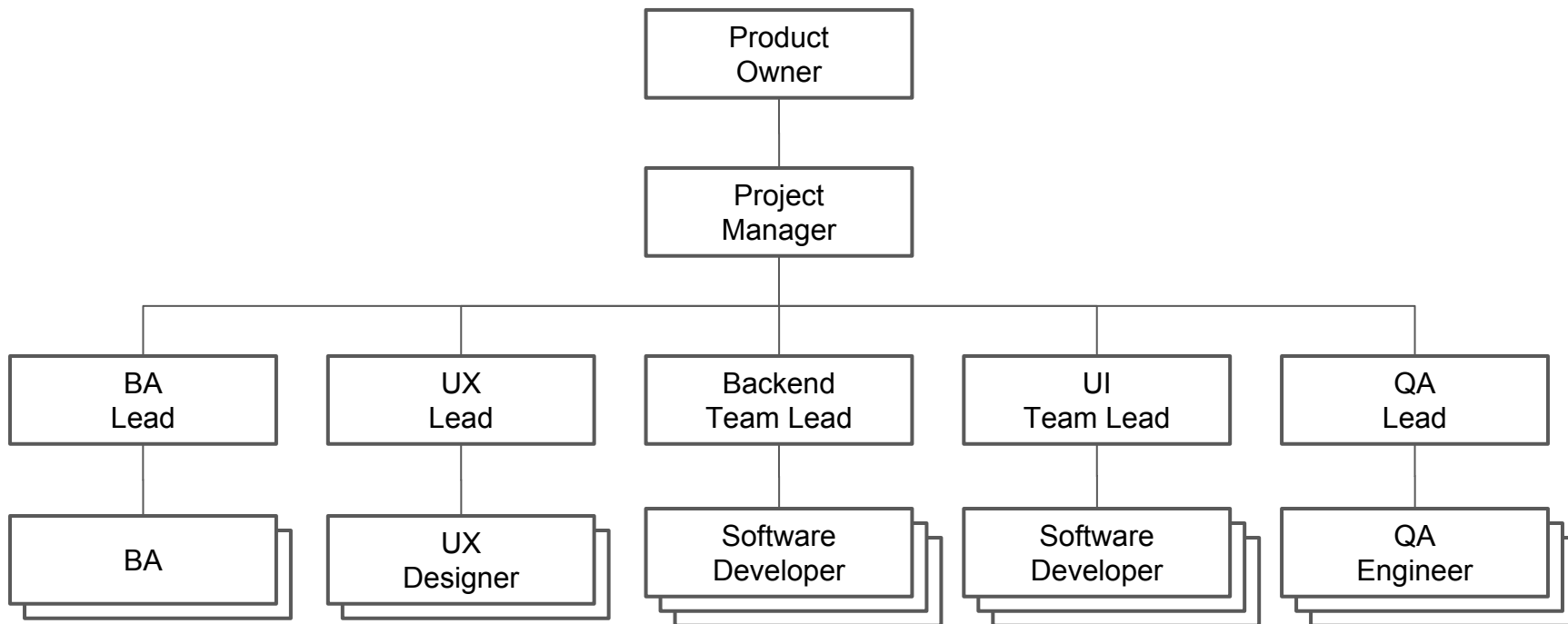


В сериале "Кремниевая долина (Silicon Valley)" стартап "Pied Piper" -- это: трое программистов (включая основателя), финансовый директор (он же биздев) и инвестор (он же хозяин дома)

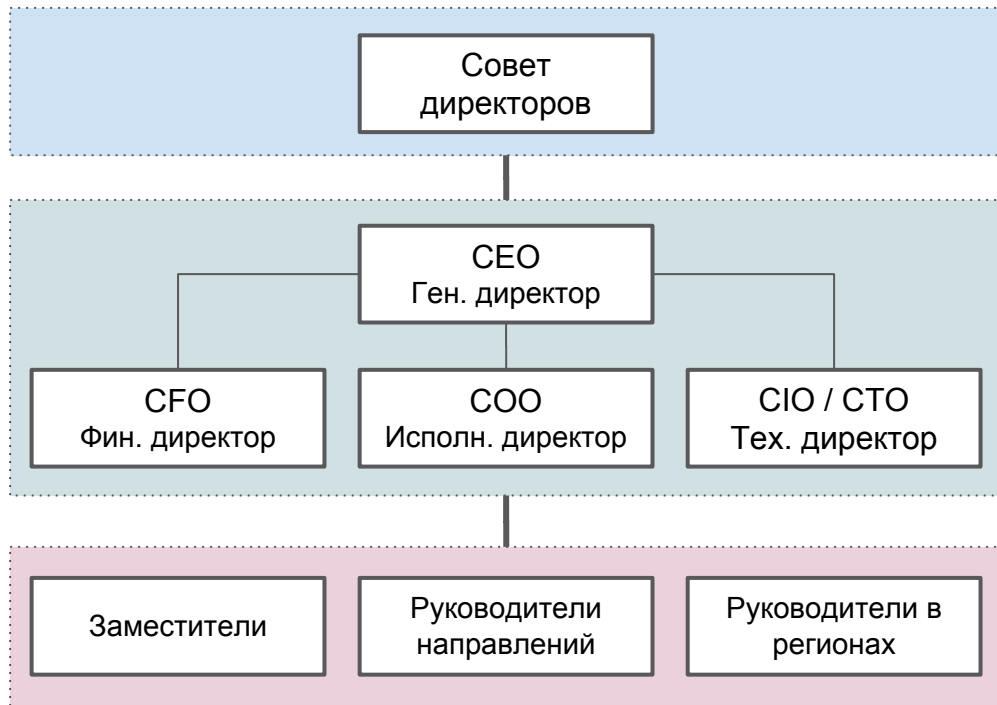
# Эволюция команды: небольшой проект



# Эволюция команды: один из проектов компании



# Эволюция команды: структура верхнего уровня



[Amazon](#)

[Apple](#)

[Facebook](#)

[Mail.ru](#)

[Microsoft](#)

[Яндекс](#)

[Yahoo](#)





# Роли, необходимые для создания продукта

## Планирование и контроль

Product Owner	Бизнес-цели, стратегия
Product Manager	Руководство продуктом, ведение roadmap
Project Manager	Планирование релизов, контроль исполнения
Tech Lead	Оценка осуществимости, применяемые технологии

## Непосредственная работа

Business Analyst	Выявление, анализ и описание требований, мокапы
UX Designer	Пользовательское взаимодействие, прототипы
Architect	Архитектура
Backend Developer	Разработка серверной части
Frontend Developer	Разработка фронтэнда
QA	Тестирование
IT Engineer	Техническая поддержка

# Сферы деятельности в ИТ

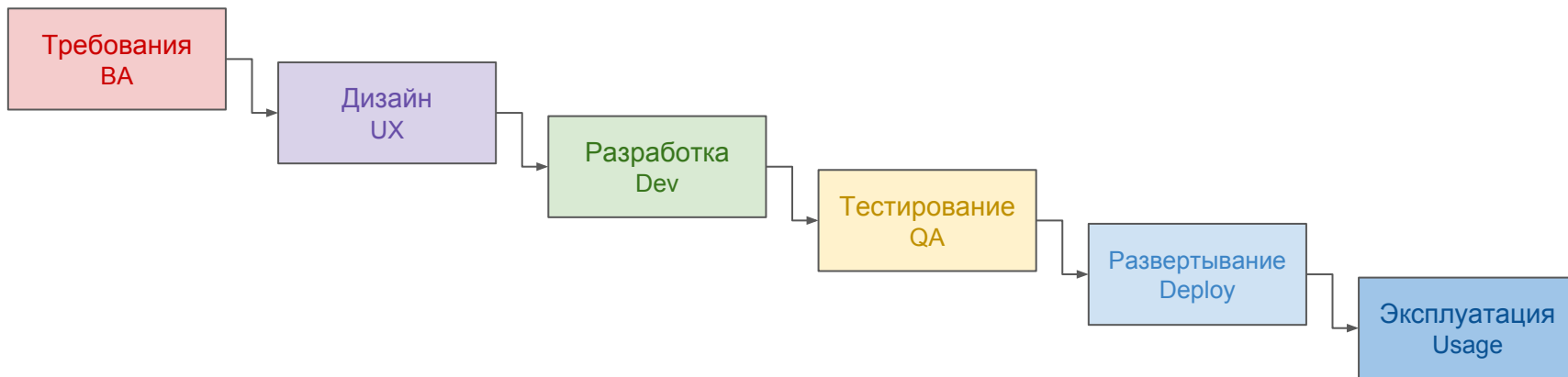
- Безопасность компьютеров и сетей
- Разработка и администрирование баз данных
- Обслуживание и поддержка
- Внедрение корпоративного ПО и консультирование
- Управление проектами
- Администрирование серверов и сетей
- Архитектура ПО и сетей
- Разработка ПО
- Системный анализ
- Опыт взаимодействия и информационная архитектура
- Дизайн интерфейсов и опыта взаимодействия

# Методологии разработки

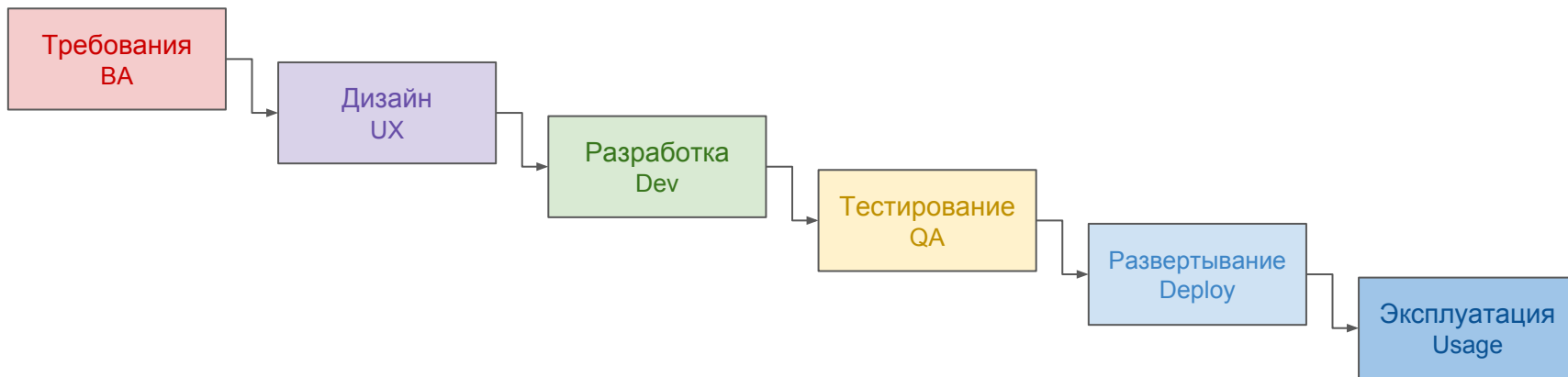
# Некоторые методологии разработки продуктов

<b>Водопадная (каскадная) модель</b>	Переход от одной фазы к другой происходит только после полного и успешного завершения предыдущей
<b>Инкрементная разработка</b>	Выпуск базовой версии продукта и его доработка в последующих версиях
<b>Итеративная модель</b>	Разновидность инкремента, создание рабочего прототипа будущего продукта, его итеративное улучшение
<b>Спиральная модель</b>	Разновидность инкремента с акцентом на анализ рисков на каждом этапе
<b>Быстрая разработка (RAD)</b>	Разновидность инкремента, на каждой итерации работа ведется параллельно с последующей интеграцией в единый прототип
<b>Agile</b>	Итеративная разработка короткими итерациями

# Водопад, каскад (Waterfall)



# Водопад, каскад (Waterfall)



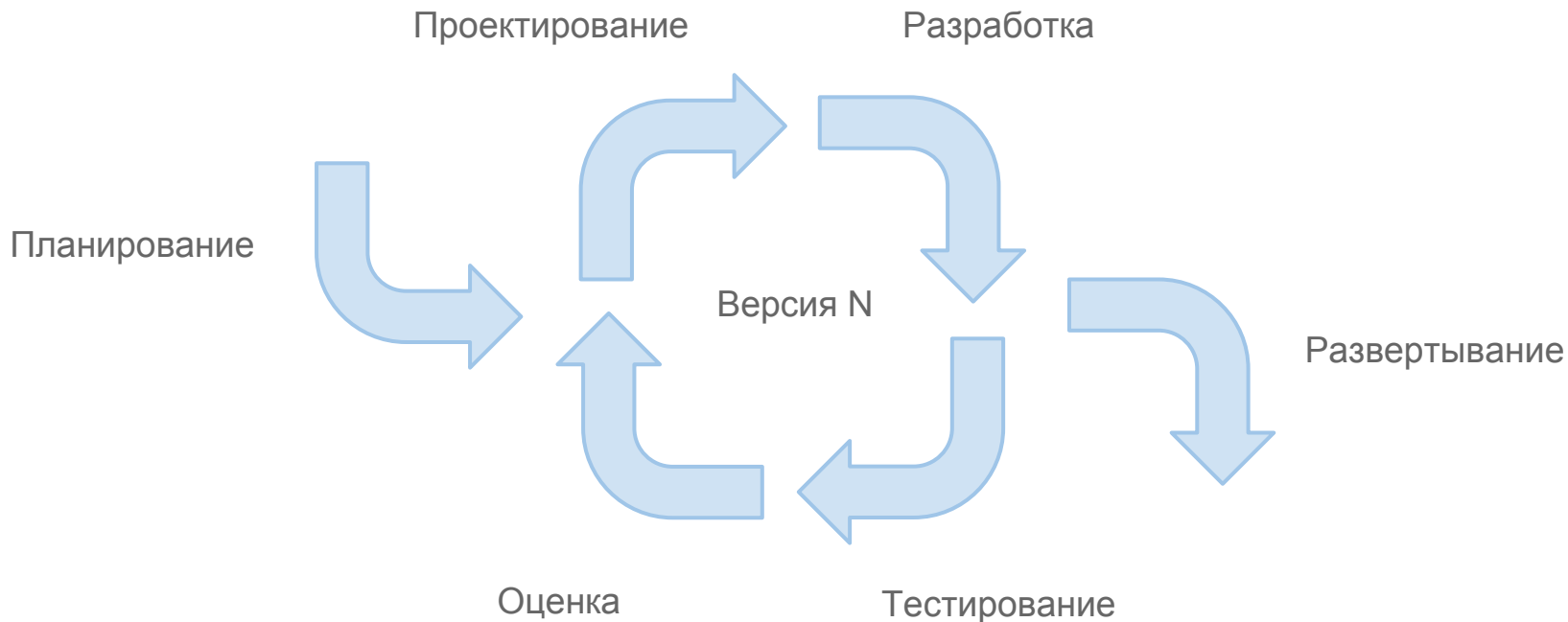
## Преимущества

- Фиксированная стоимость разработки
- Простое управление проектом

## Недостатки

- Нужна полная и четкая спецификация
- Нет возможности что-либо изменить в уже начатом проекте
- Стоимость внесения изменений высока

# Инкрементная VS итеративная модель





# Инкрементная VS итеративная модель

Промежуточные продукты  
могут не иметь ценности

Инкрементная разработка

Конечный продукт может  
быть не оптимален



Готовый продукт после  
каждой итерации

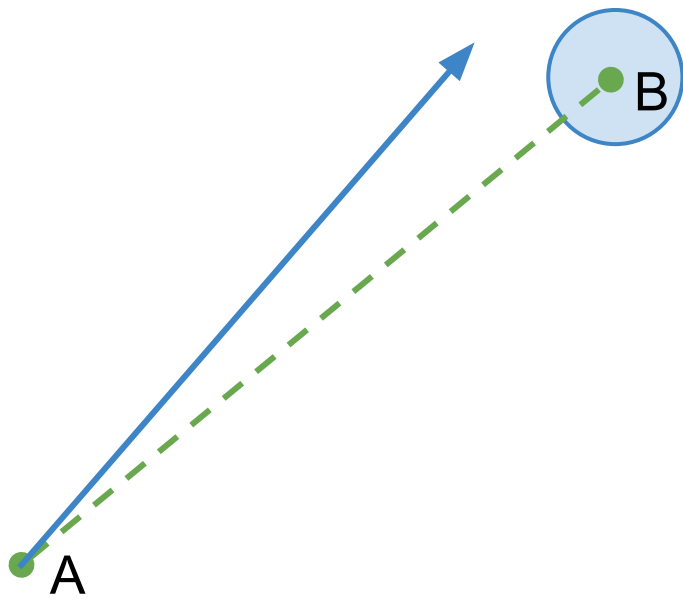


Итеративная разработка

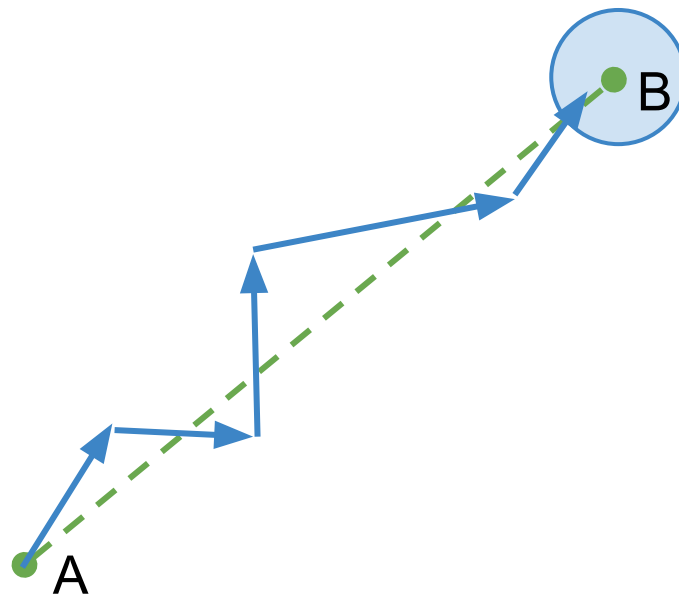


Продукт дорабатывается  
под нужды заказчика

# Водопад VS итеративный подход

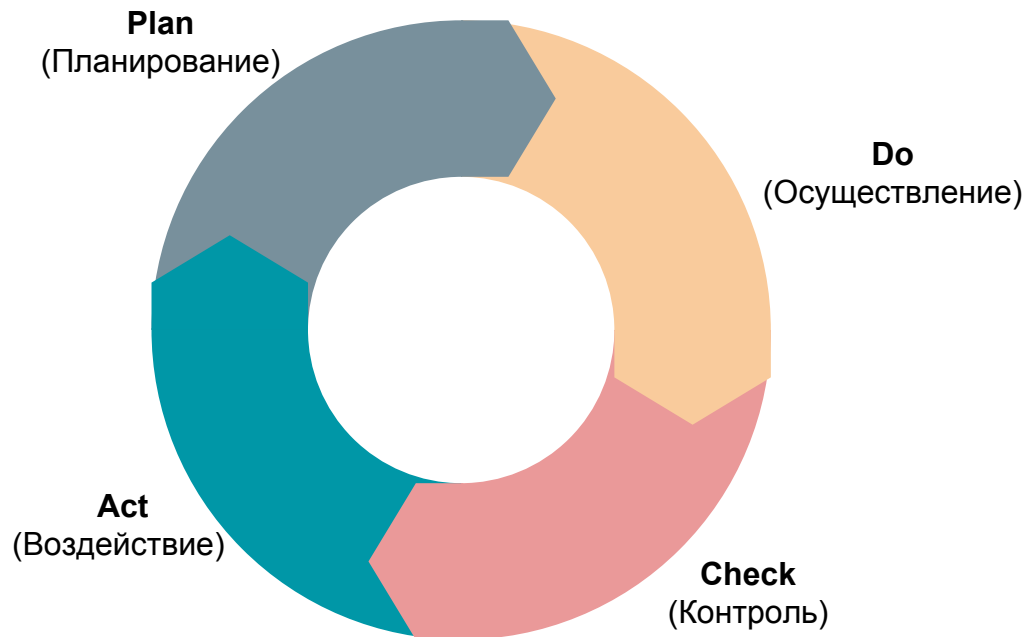


Водопад



Итеративный подход

# Цикл PDCA (Деминг, Шухарт)



# Принципы Agile

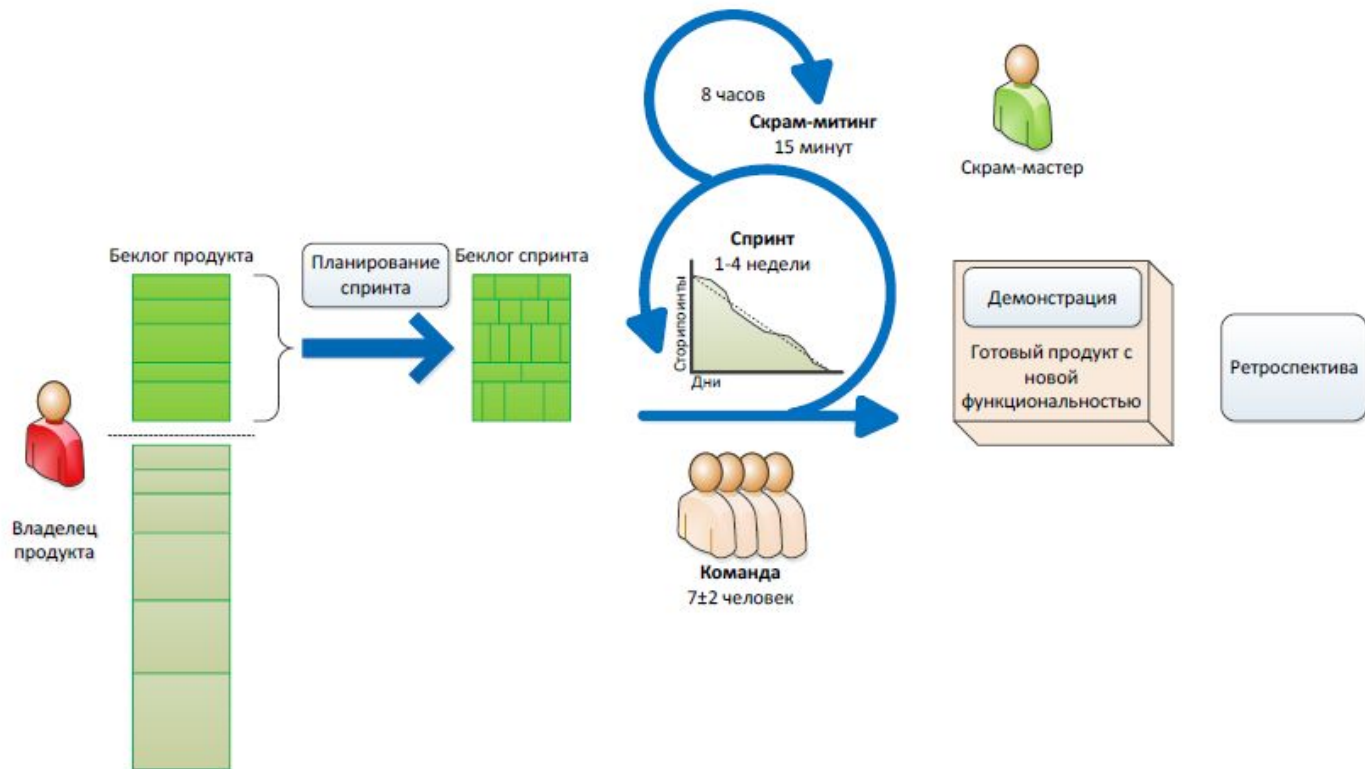
## Ценности

- Люди и взаимодействие между ними
- Работающий продукт важнее документации
- Сотрудничество и выстраивание партнерских отношений с заказчиком важнее контракта
- Готовность к изменениям

## Принципы

- Основная задача -- удовлетворить заказчика
- Требования могут быть изменены
- Постоянная обратная связь
- Лучший способ коммуникации -- лицом к лицу
- Постоянно работающий продукт
- Частые итерации
- PDCA-цикл
- KISS-подход
- Качественная команда = качественный код = качественный продукт
- Самоорганизация, а не управление

# Скрам



# Водопад VS итеративный подход VS agile

	<b>Водопад</b>	<b>Итерации</b>	<b>Agile</b>
<b>Цели</b>	Определяются для каждой фазы (входы/выходы)	Завершенные требования передаются в разработку и тестирование	В рамках спринта создается работающая фича
<b>Проектирование</b>	Объем работ проекта определяется РО, все фичи должны быть проработаны до их программирования	Объем каждой итерации определяется РМ и согласуется с РО, в итерацию попадают только проработанные фичи	Команда определяет объем исходя из приоритетов и ресурсов, в спринт попадает то, что не будет изменено и может быть сделано за время спринта
<b>Оценка затрат</b>	РМ рассчитывает оценку для всего проекта и согласует ее с РО	РМ дает оценку для каждой итерации	Оценка дается командой на митинге, проводимом скрам-мастером, оценка затем может быть пересмотрена
<b>График работ</b>	По фазам	По итерациям	По спринтам

# Водопад VS итеративный подход VS agile

	<b>Водопад</b>	<b>Итерации</b>	<b>Agile</b>
<b>Качество</b>	Фокус меняется от фазы к фазе, тестирование в конце проекта	Переход фокуса от проектирования к разработке/тестированию в рамках каждой итерации	Фокус на всем ЖЦ, тестирование внутри спринта
<b>Учет рисков</b>	Отсутствует, “тушение пожаров” на фазе тестирования	Присутствует, риски идентифицируются и учитываются в следующих итерациях	Присутствует, риски идентифицируются и учитываются в следующих спринтах
<b>Обратная связь</b>	По завершении проекта	После каждого релиза	После каждого спринта

# Водопад VS итеративный подход VS agile

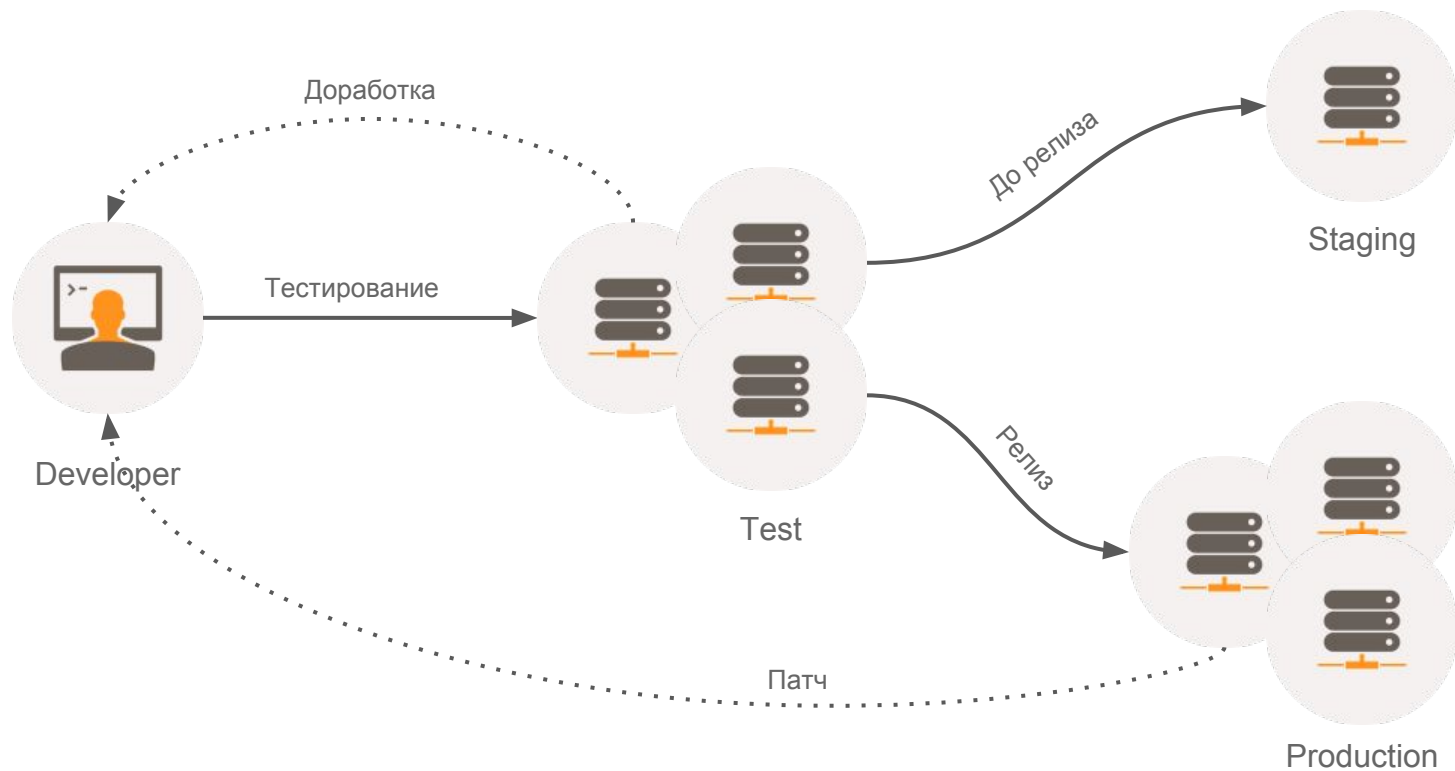
	<b>Водопад</b>	<b>Итерации</b>	<b>Agile</b>
<b>Процесс</b>	Последовательное выполнение фаз	Проектирование, разработка и тестирование ведутся одновременно с перекрытием	Над фичами работают миникоманды, которые делают весь ЖЦ
<b>Распределение ролей</b>	Четкое разделение ролей	Четкое разделение ролей, но подмена возможна	Каждый может работать над любой фазой задачи (проектирование, разработка, тестирование)
<b>Назначение задач и ответственность</b>	Задачи назначает РМ, он же ответственен за реализацию проекта	Назначение задач согласуется с РМ, он ответственен за выполнение итерации	За спринт ответственна вся команда, каждый может взять себе задачу по вкусу
<b>Отчетность</b>	Процент выполненного	Процент выполненного	График сгорания задач
<b>Совещания</b>	Еженедельные (ежедневные) отчеты о состоянии дел	Еженедельные (ежедневные) отчеты о состоянии дел	Ежедневные стендапы: что сделали, что будем делать, проблемы



# График сгорания задач (Burndown Chart)



# Схема развертывания



# Примерный график работ: водопад



# Примерный график работ: итерации

QA	QA	QA	D														
Dev	Dev	Dev	QA	QA	QA	D											
UX	UX	Dev	Dev	Dev	Dev	QA	QA	QA	D								
BA	BA	BA	UX	UX	Dev	Dev	Dev	Dev	QA	QA	QA	D					
			BA	BA	BA	UX	UX	Dev	Dev	Dev	Dev	QA	QA	QA	D		
						BA	BA	BA	UX	UX	Dev	Dev	Dev	Dev	QA	QA	QA

# Успешность проектов

Успешные проекты	Проекты с проблемами	Неудачные проекты
<b>29%</b>	<b>52%</b>	<b>19%</b>

Усредненные данные за 2011-2015 годы  
2015 CHAOS Report, Standish Group

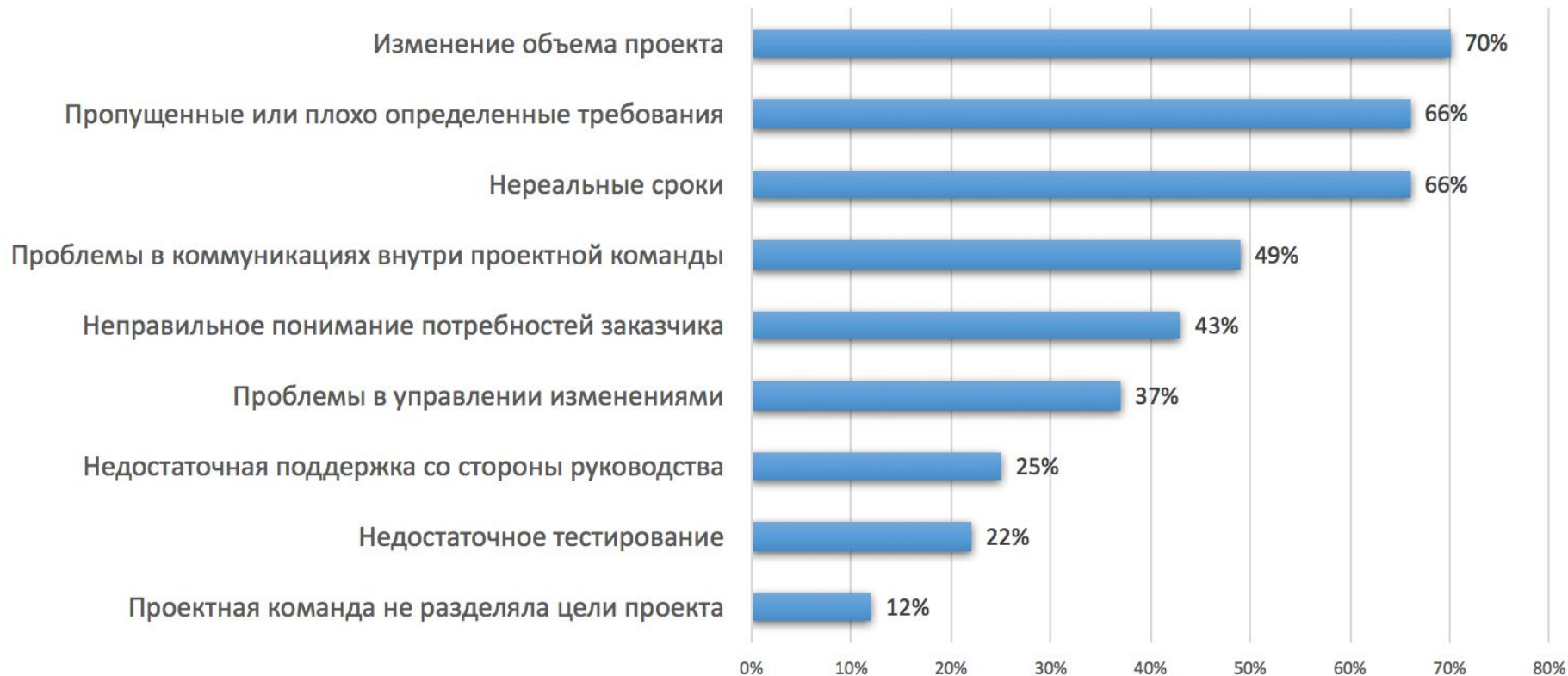
# Успешность проектов (в зависимости от его размера)

	Успешные проекты	Проекты с проблемами	Неудачные проекты
Огромные	<b>2%</b>	<b>7%</b>	<b>17%</b>
Большие	<b>6%</b>	<b>17%</b>	<b>24%</b>
Средние	<b>9%</b>	<b>26%</b>	<b>31%</b>
Небольшие	<b>21%</b>	<b>32%</b>	<b>17%</b>
Мелкие	<b>62%</b>	<b>16%</b>	<b>11%</b>

# Успешность проектов (Agile vs Waterfall)

	Agile vs Waterfall					
	Успешные проекты		Проекты с проблемами		Неудачные проекты	
Все	<b>39%</b>	<b>11%</b>	<b>52%</b>	<b>60%</b>	<b>9%</b>	<b>29%</b>
Большие	<b>18%</b>	<b>3%</b>	<b>59%</b>	<b>55%</b>	<b>23%</b>	<b>42%</b>
Средние	<b>27%</b>	<b>7%</b>	<b>62%</b>	<b>68%</b>	<b>11%</b>	<b>25%</b>
Мелкие	<b>58%</b>	<b>44%</b>	<b>38%</b>	<b>45%</b>	<b>4%</b>	<b>11%</b>

# Причины неудач проектов





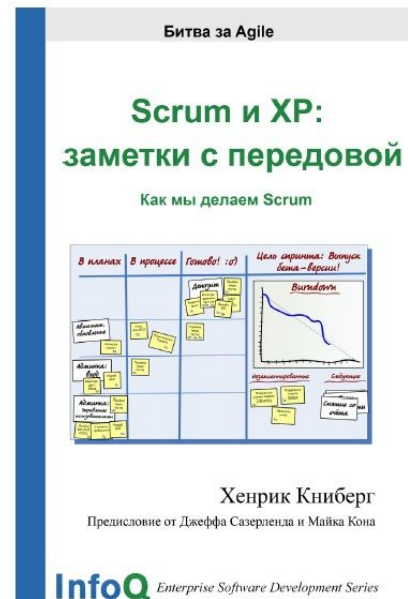
# Факторы успеха проекта

Фактор	Вес
Поддержка руководства	20
Вовлеченность пользователей	15
Бизнес-процессы	15
Компетентность команды	13
Управление проектом	12
Навыки Agile	10
Ясные цели	6
Зрелость команды	5
Инструменты и инфраструктура	1

Дж. Сазерленд  
Scrum. Революционный метод  
управления проектами



Х. Книберг  
Scrum и XP: заметки с передовой



# Интересно про ИТ

- [Мастер-класс Бориса Вольфсона. Основы Agile](#)
- [Ещё раз про семь основных методологий разработки](#)
- [Они пишут правильную вещь](#)
- [Agile vs Iterative vs Waterfall models](#)
- [Скрам Гайд \(PDF\)](#)