

SCRUM

Без ТЗ - результат ХЗ

«УТВЕРЖДАЮ»
Генеральный директор
<Компания-заказчик>
_____ А. Иванов
« ____ » _____ 2007 г.

«УТВЕРЖДАЮ»
Генеральный директор
<Компания-исполнитель>
_____ Б. Петров
« ____ » _____ 2007 г.

ТЕХНИЧЕСКОЕ ЗАДАНИЕ

на разработку официального сайта
для компании _____

«СОГЛАСОВАНО»
<Представители Заказчика>
« ____ » _____ 2007 г.

«СОГЛАСОВАНО»
<Представители Исполнителя>
« ____ » _____ 2007 г.

Москва, 2007

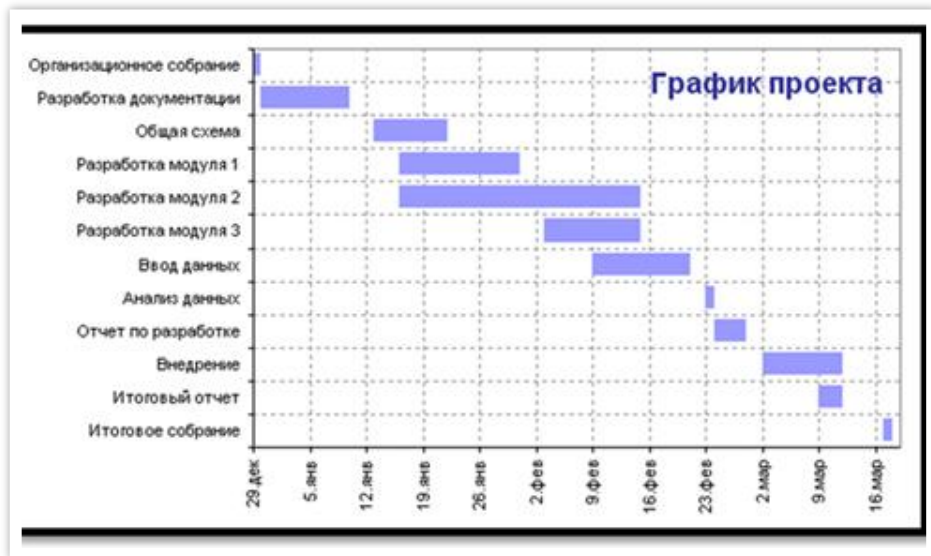
Содержание

1. Общие сведения.....	3
1.1. Полное наименование сайта и его условное обозначение	3
1.2. Наименование предприятий Разработчика и Заказчика	3
1.3. Основание для разработки.....	3
1.4. Плановые сроки начала и окончания работ.....	3
1.5. Сведения об источнике и порядке финансирования работ.....	3
1.6. Порядок оформления и предъявления Заказчику результатов работ.....	3
2. Назначение и цели создания сайта.....	4
2.1. Назначение сайта.....	4
2.2. Цели создания сайта.....	4
3. Сведения о компании.....	5
3.1. Общие сведения о компании	5
3.2. Сведения об условиях эксплуатации сайта	5
4. Требования к сайту.....	6
4.1. Требования к сайту в целом.....	6
4.2. Требования к функциям сайта.....	6
4.3. Требования к видам обеспечения	7
5. Состав и содержание работ по созданию сайта.....	8
6. Порядок контроля разработки сайта и его приемки.....	9
6.1. Общие требования к приемке работ.....	9
6.2. Требования к испытанию сайта	9
7. Требования к составу и содержанию работ по подготовке площадки Заказчика к вводу сайта в эксплуатацию.....	10
7.1. Технические мероприятия.....	10
7.2. Организационные мероприятия.....	10
7.3. Информационное обеспечение	10
8. Требования к документированию.....	11
8.1. Перечень предоставляемой документации	11
8.2. Требования к стандартам и программно-техническим средствам документирования	11
9. Источники разработки.....	12
Приложение 1. Тексты для веб-сайта.....	13
Приложение 2. Схема функциональной структуры сайта.....	14

Классический подход

Ожидаемый способ выполнения проекта –

соблюдение диаграммы Ганта



- Сроки не соблюдаются
- Всё делается в последний момент
- Заказчик меняет условия на противоположенные после каждой демонстрации
- В **итоге**: с перенапряжением делается какая-то часть ожидаемого результата с которой как-то мирится Заказчик

Есть другой подход? Их множество!

Incremental



Iterative



Философия, которая помогает предпринимателям делать работающий продукт

	2015	2016	2017	2018
Через %опу	35%	30%	18%	35%
Как получится	21%	18%	15%	23%
Scrum	14%	18%	21%	13%
Agile собственного приготовления	11%	18%	27%	16%
Водопад (Waterfall)	---	5%	8%	4%
RUP-based	5%	5%	5%	1%
XP	3%	1%	1%	0%
MSF	1%	1%	1%	0%
CMM/CMMI	2%	1%	---	---
Kanban	---	---	---	4%
Feature Driven Development	---	---	---	2%
Другое	8%	3%	4%	2%



Сотни Agile Фреймворков от "именитых" авторов

Программисты, военные, банки, промышленность,...

Все говорят про Agile!

Современный тренд –
"я внедрил Agile и стало только хуже, почему???"

Разберемся в понятиях

Agile это не SCRUM. SCRUM это не Agile

Философия

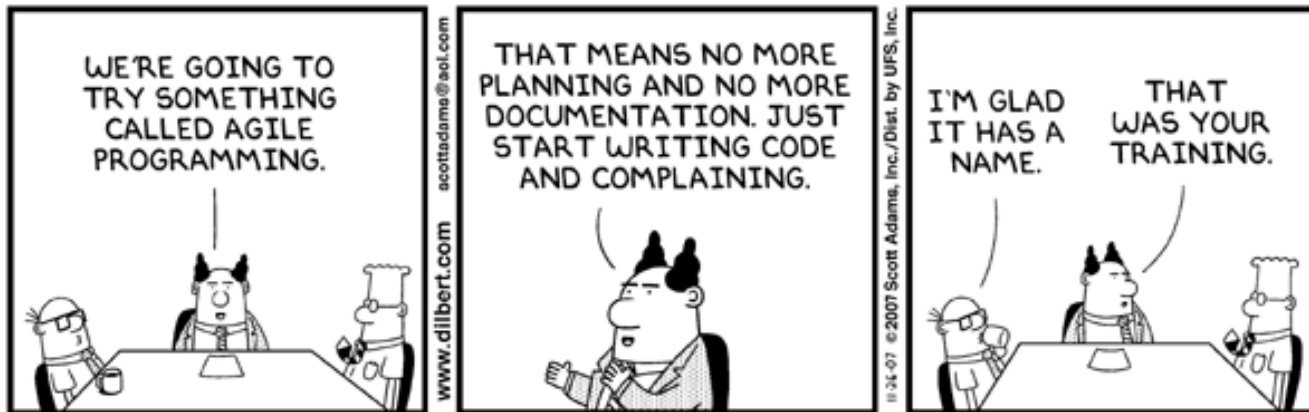
Не дает ответы на вопрос «как?»
Дает миллион ответов на вопрос «почему»

Дает возможность почувствовать,
что ты не один во Вселенной

Практика

Не дает ответы на вопросы «почему?»
Дает конкретные рецепты и строгие инструкции «как».

Заставляет чувствовать себя
«неудачником» который всё делает не так



© Scott Adams, Inc./Dist. by UFS, Inc.

А вы Agile?

(Гибкая методика разработки)

Люди и взаимодействия

Процессы и инструменты

Работающий продукт

Исчерпывающая документация

Сотрудничество с Заказчиком

Условия контракта

Готовность к изменениям

Первоначальный план

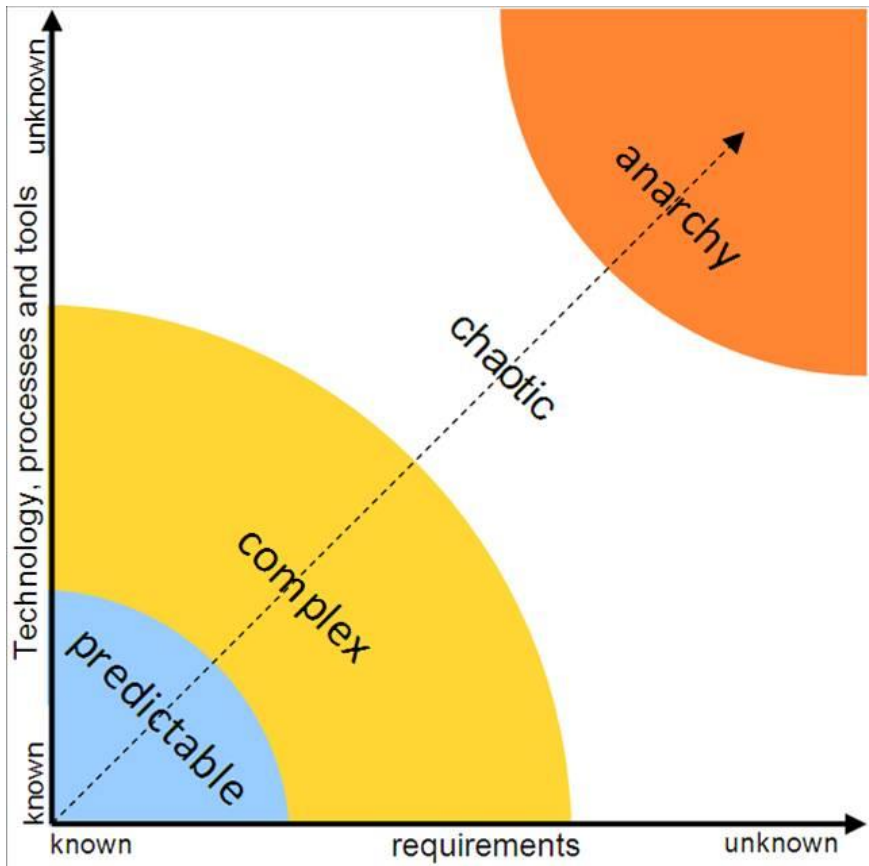
и ещё 12 принципов для разработчиков программного обеспечения

Когда нужен Agile

Всем хочется перейти от **хаоса** к **порядку**.

Но в реальной жизни **порядка нет**.

Методология должна выбираться в зависимости от уровня упорядоченности задач, а не потому, что так Греф сказал.



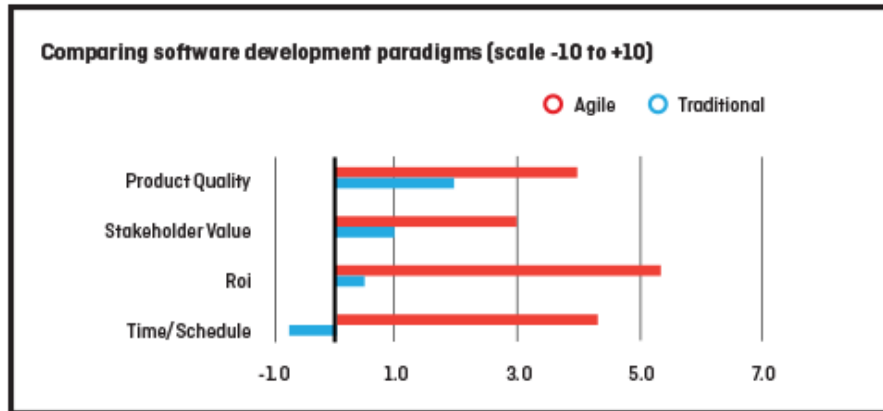
Agile подходит когда:

Нет «лучших практик» решения.
В задаче слишком много переменных чтобы предложить решение.

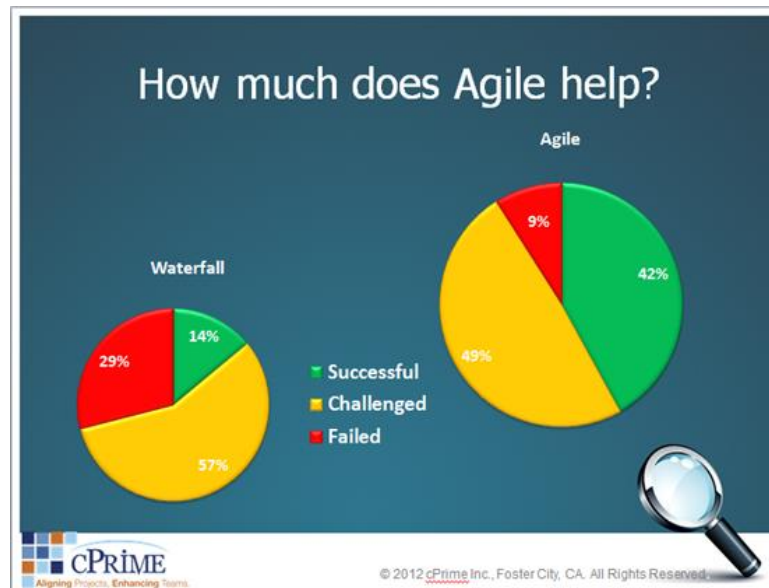
Есть возможность для итеративности действий.

Зачем это нужно на самом деле

В большинстве случаев Agile не нужен! Но считается, что ...



1. С ним легче соблюдать сроки и планировать.
2. Выше удовлетворенность заказчиков.
3. Меньше затрат ресурсов на лишнюю работу.



Но это не "серебряная пуля".

Работать всё равно придётся!

SCRUM

SCRUM – это про команды и спринты. В основе – самодостаточная команда которая сама решает как достичь результата. Это Agile фреймверк.



- Команда в рамках **фиксированного процесса** реализует «фичи» за **фиксированное время**.
- Есть **жесткие правила** для работы всего коллектива.
 - Мы добавляем **конкретную ценность** Потребителю за отведенное время.
 - Мы регулярно **пересматриваем** требования к проекту.
 - Есть **общее виденье** проекта.
 -
- В самом простом варианте: 3 митинга, 3 роли 3 артефакта **ОБЯЗАТЕЛЬНО**.

Роли

Спойлер: в SCRUM команде нет лида.

Product Owner (PO)

Финальный арбитр
Имеет виденье (vision) продукта.
Знает «что» но может не знать «как».
Отвечает за «результативность вложений» (ROI).

Scrum Development Team (SDT)

Кроссфункциональная команда выполняющая «скрам».
Нацеленная на создание продукта каждый «спринт».
Самоорганизовывается и взаимодействует.

ScrumMaster (SM)

Это не руководитель команды.
Это не менеджер проекта.
Это роль отвечающего за правильность «скрама» и только за это.
Фасилитатор!
Пользуется ScrumMaster Checklist'ом:

- Часто ли используются техники XP
- Поддерживается ли «доска»
- «*Do team members celebrate each other's success?*»
- ...

Митинги (активности)

SoS выделилось в отдельную масштабируемую методологию

Daily (Stand Up)

Ежедневное собрание
Проводит SM но **нужен для членов команды**

На митинге озвучивают

- Что я сделал
- Что я буду делать
- Какие были проблемы

~10-15 минут

Sprint

Цикл продуктовой разработки
Четкие задачи
Ограниченный срок
~ 1/2 недели

Retrospective

Проводится SM и РО
Выполняется в конце спринта

На митинге:

- **Проблемы**
- **Предложения** по улучшению
- На чем надо **сфокусироваться**

Sprint (Demo) Review

Демонстрируются **результаты**, а не то как они были достигнуты
Основной участник митинга Stakeholder – **пользователь** результатов
Корректируем Product Backlog
~ 1 час

Backlog Refinement (Sprint Planing)

Выбираем задачи на спринт
Делим и мерджим задачи на конечные части (Grooming)
Оцениваем сроки выполнения
Исполнители **сами выбирают** себе задачи
До 10% времени спринта

Scrum of Scrums

Часть собрания когда **взаимодействуют команда и ambassadors** (представитель) из других команд
Основная цель – понимание общих целей и задач на осознание

Артефакты

Product Backlog (PB)

Все требования к проекту которые нужно будет выполнить
Составляется и приоретизируется PO

Hobby Site Product Backlog Items chosen for this Sprint

Item ID	Bug ID	Summary	Rank	Category	Accomplished?
1		Search other hobby sites: bring back results in a list	1	Search	X
2		Search hobby content & bring back results in a list	3	Search	X
3		Show direct display of hobby content in results	14	Search	
4		Use variety of methods for relevancy	4	Search	X
5		Keep track of queries for buckets	37	Search	X
6		Include user-generated content (message boards) in index	19	Search	X
7		Include Ads on search	16	Search	X
8		Include sites to index	2	Search	X
9		Search infrastructure/ops/machines		Search	X
10		"Did you mean?" support for misspellings, etc.	18	Search	X
11		Media for Message boards	23	Boards	
12		View/post to boards	6	Boards	X
13		View list of forums		Boards	X
14		See a list of threads on this board	7	Boards	X
15		Show different views of threads	40	Boards	X
16		Show messages/all messages in thread		Boards	X
17		Sort list of threads	43	Boards	X
18		Sort postings by rating	42	Boards	X
19		Signed in users can rate another user's posted message	41	Boards	X
20		Signed in users can start a thread/message	8	Boards	X
21		Signed in users can preview messages before submitting	22	Boards	X
22		Find all posts by another user	45	Boards	
23		Show user info for each message	24	Boards	

Sprint Backlog (SB)

Задачи на неделю, состоят из отдельных задач
Существуют правила как правильно формулировать User Stories

User Story	Tasks	Day 1	Day 2	Day 3	Day 4	Day 5	...
As a member, I can read profiles of other members so that I can find someone to date.	Code the ...	8	4	8	0		
	Design the ...	16	12	10	4		
	Meet with Mary about ...	8	16	16	11		
	Design the UI	12	6	0	0		
	Automate tests ...	4	4	1	0		
	Code the other ...	8	8	8	8		
As a member, I can update my billing information.	Update security tests	6	6	4	0		
	Design a solution to ...	12	6	0	0		
	Write test plan	8	8	4	0		
	Automate tests ...	12	12	10	6		
	Code the ...	8	8	8	4		

Product Increment

Новая ценность в продукте
Состоит из PBI выполненных в рамках последнего спринта + ранее добавленные «increments»

Definition of Done (DoD)

Новая ценность в продукте
Состоит из PBI выполненных в рамках последнего спринта + ранее добавленные «increments»

Как это выглядит

The Agile: Scrum Framework at a glance

Inputs from Executives,
Team, Stakeholders,
Customers, Users



Product Owner



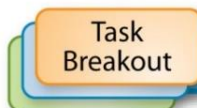
The Team



Product Backlog

Team selects starting at top as much as it can commit to deliver by end of Sprint

Sprint Planning Meeting



Sprint Backlog



Как это сейчас

MVP (минимальный жизнеспособный продукт) & MFF (минимальный ценный продукт)

Превращаем продукт в минимальный по размерам вариант (LEAN это не SCRUM!)

Forecast

Оценка времени только примерная и никогда не точная

NoEstimate

Точная оценка и не нужна, строим процесс постоянного выпуска продукции

Технический SCRUM

Строгое соблюдение правил не приводит к Agile, люди забывают зачем они это делали



Пример

Строим баню по SCRUM

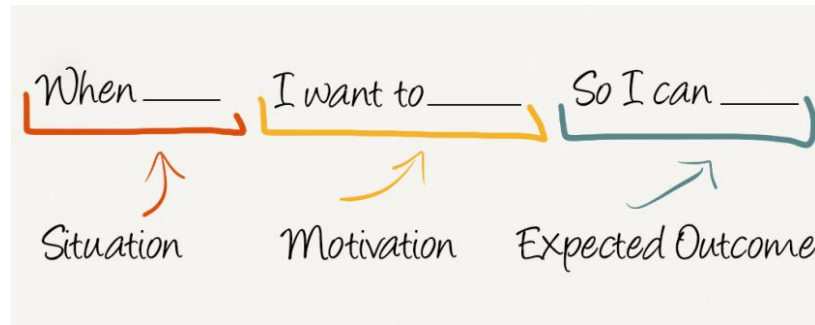
Алгоритм

- 1. Соберите все требования (user stories).
- 2. Проведите декомпозицию полученных задач, добавьте своих «технических» задач и оцените трудоемкость их выполнения.
- 3. Приоритезируйте полученный список по «нужности», а не по «сложности».
- 4. Оцените свои силы и выберите самые нужные задачи которые вы сможете реализовать в какой-то период времени.
- 5. Реализуйте за итерацию времени.
- 6. Продемонстрируйте результат и начните заново.

Полезный навык

Как формулировать User Stories?

Например, вот так:



- или так:



Для больших проектов на старте или в тупике стоит использовать USM (книга Паттона)

Итеративная работа



Как пользователь бани, я **хочу** треугольную крышу, **чтобы** дождь стекал на землю

Дом 8x8 м.

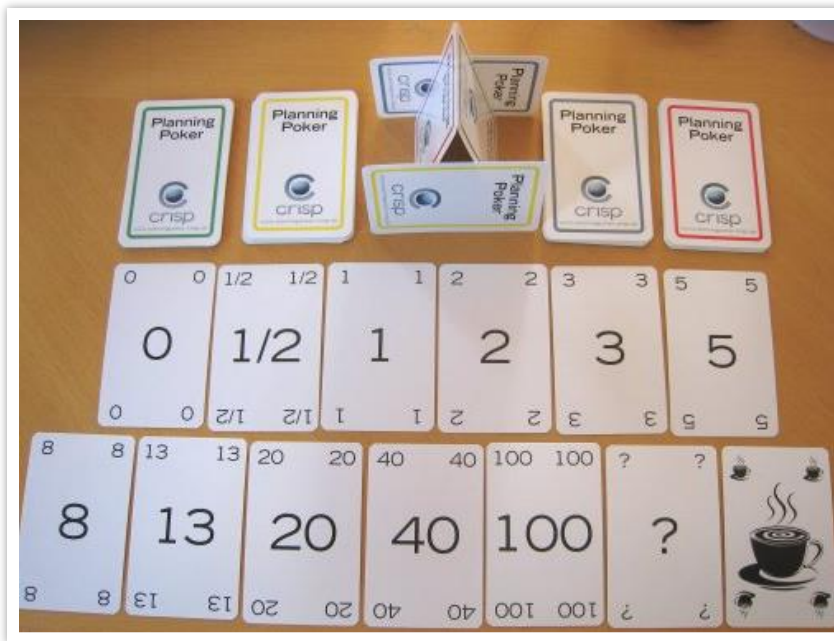
Высота 3 м.

Площадь крыши 80 кв.м.

Сдаётся мне джентльмены, это был покер.. planning poker

Цель планирования – выработка общего понимания как решать задачу,
а не составление плана

Но можно воспользоваться и для оценки времени выполнения проекта.

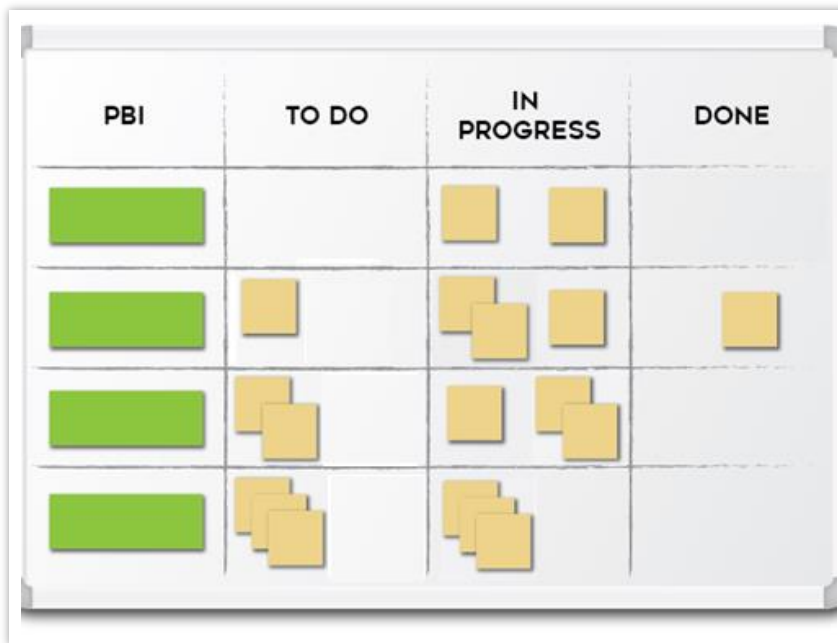


- В начале используется оценка в идеальных "человеко-днях".
- Когда команда стабилизируется в оценках в "стори-поинтах".
- "Agile estimating and planning"
M. Cohn

Прогресс задач

Ещё немного про управление задачами, но вообще-то это про Kanban.

Kanban не SCRUM, но SCRUM команды могут использовать Kanban и часть Kanban итераций может быть сделана по SCRUM



- Основная "фишка" не должно быть много выполняемых одновременно задач
- Задача зависла в выполнении – её надо разбить, переформулировать, навалится всей толпой, выкинуть из спринта и т.п.

Простые советы по формированию спринта

В результате спринта должен быть результат

Спринт самодостаточная итерация. Не стоит переносить задачи между спринтами.

Баги должны быть закрыты в спринте.

Декомпозируйте или берите меньше user stories.

Забронируйте "себе" время

10% уйдет на баги.

10-20% должно уйти на технический долг.

20% времени лучше иметь в запасе с учетом качества оценки.

Это ваше время, а не заказчика, но заказчик должен про него знать

Кстати, подготовка к демо тоже требует время.

На демо - демонстрируй

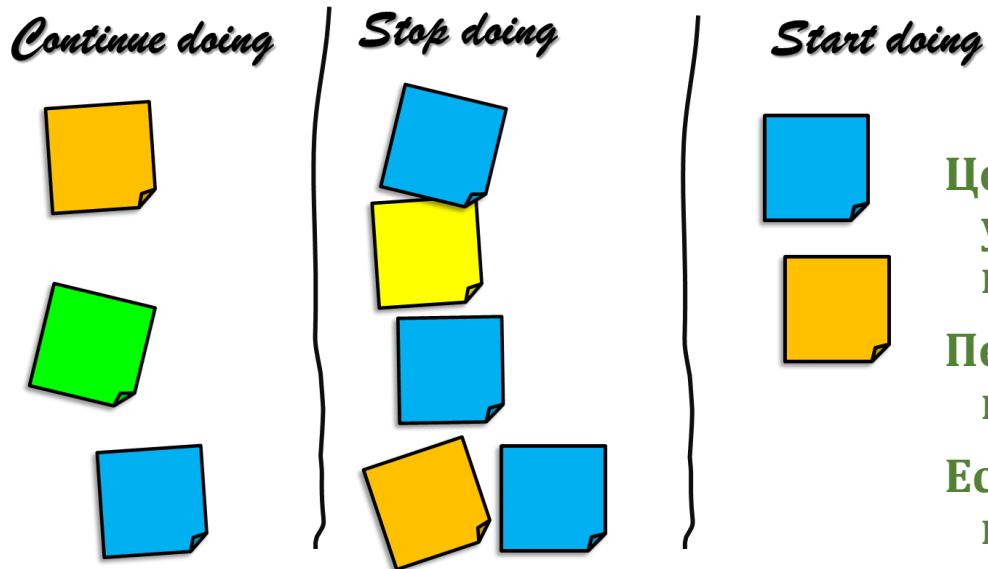
Цель демо не отчет, а уточнение виденья проекта у потребителя.

Спринт должен добавлять инкремент.

Релиз должен добавлять инкремент.

На демо должен быть виден инкремент.

Немного про практику ретроспектив



Цель ретроспективы –
улучшение процесса для
команды, а не психотерапия

Первые разы она идёт долго,
потом быстро

Есть множество практик
которые помогут даже если вы
не SCRUM и не Agile

WHAT MAKES JESUS...

