

Методологии управления проектами. Введение в SCRUM

SDM – методологии разработки программного обеспечения

Методологии управления проектами. Введение в SCRUM

Автор



Наталия Бурдина



После урока обязательно



Повторите этот урок в видео формате на [ITVDN.com](http://itvdn.com)



Проверьте как Вы усвоили данный материал на [TestProvider.com](http://testprovider.com)

Тема

SDM – методологии разработки
программного обеспечения

Содержание



SDM – это ...



Change-driven VS plan-driven SDM



Agile философия



Agile фреймверки

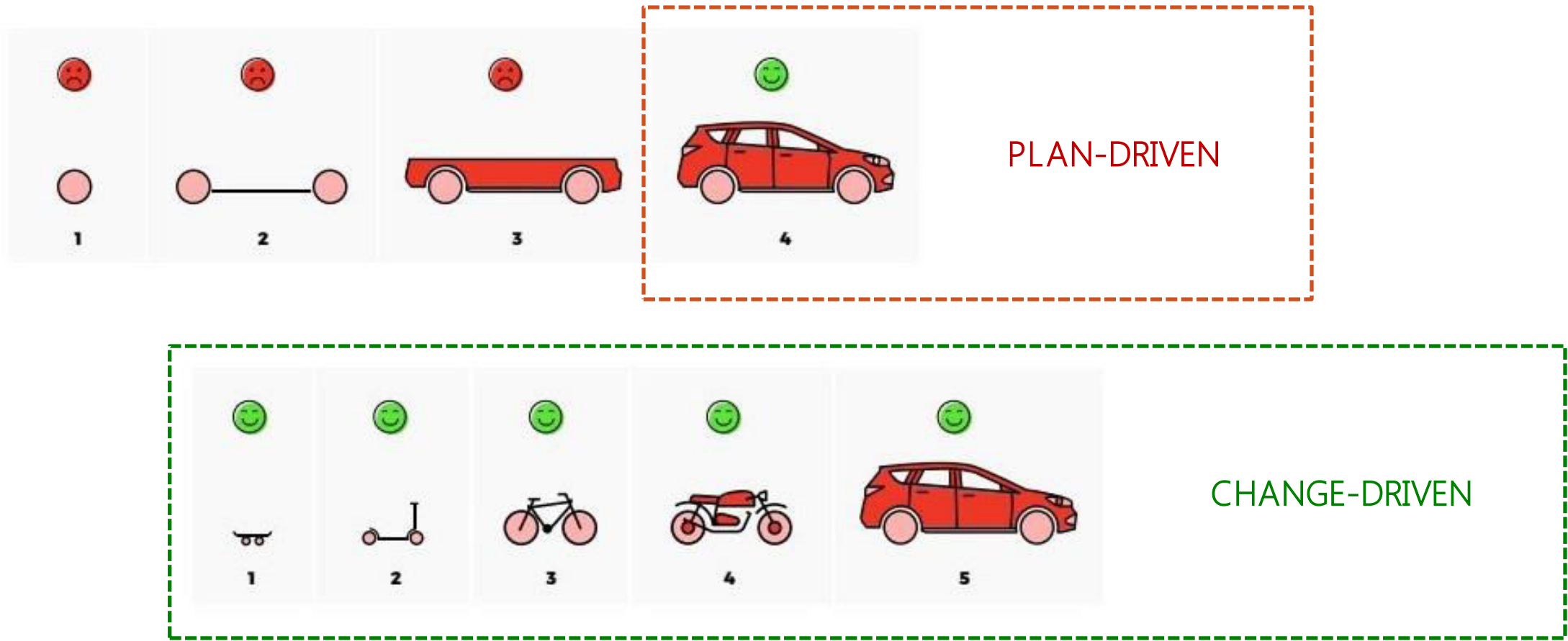
SDM - это



Software Development
Methodologies
Методологии Разработки
Программного Обеспечения

- это фреймверки или подходы, которые используются для структурирования, планирования и контроля процесса разработки ПО.

Plan-driven VS Change-driven SDM



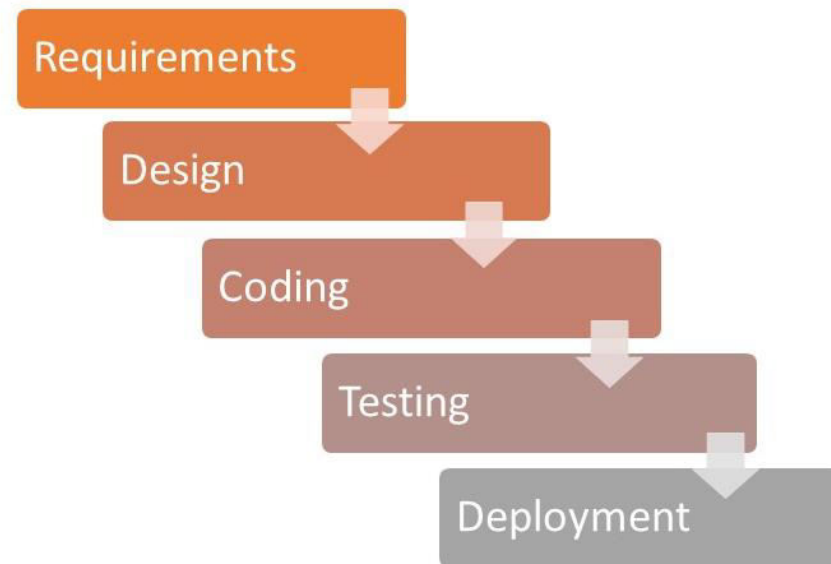
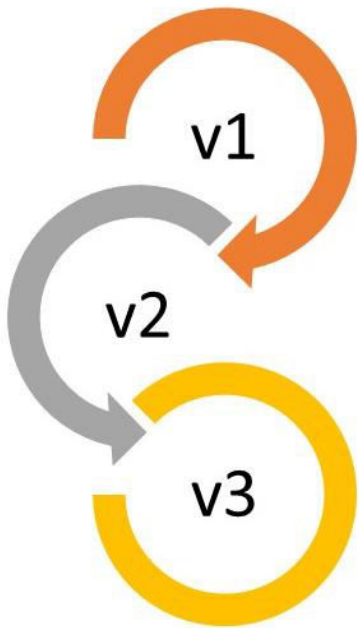
Plan-driven VS Change-driven SDM

	Plan-driven	Change-driven
Выявление и анализ требований	Цель состоит в том, чтобы полностью определить требования до начала разработки. Определение требований обычно завершается в качестве первой фазы проекта.	Требования изменчивы и нет необходимости исчерпывающе определять весь скоуп сразу. Разработка проекта начинается с требований высокого уровня, в процессе добавляются новые требования, которые также идут в разработку.
Последовательность задач	Последующая фаза разработки обычно начинается только после того, как завершена предыдущая.	Фазы проекта часто пересекаются - например, новые требования могут быть определены при разработке или тестировании функциональности приложения.
Сроки	Поставка готового решения (продукта) обычно занимает больше времени, чем при подходе Change-driven, т.к. зависит от сложности и комплексности разрабатываемого решения. Продукт выходит на рынок, только в полной готовности.	Поставка готового решения (продукта) обычно быстрая, так как разработка разделена на итерации, которые ограничены по времени (длительность итерации составляет 1-4 недели). Это возможно, потому что этапы разработки продукта выполняются параллельно.
Документация	Документация оформляется подробно, исчерпывающе и как правило утверждается до начала разработки (внесение изменений в утвержденный скоуп невозможно).	Документация полезна, но содержит более низкий уровень детализации и может быть предоставлена после реализации. Цель состоит в том, чтобы минимизировать излишнюю спецификацию, вовлекая заказчика/пользователя через постоянное общение и обратную связь.

Plan-driven VS Change-driven SDM

	Plan-driven	Change-driven
Приоритеты	Высокой важностью обладает часть планирования и оценки рисков, связанных с выявлением и согласованием требований к ожидаемому продукту.	Высокий приоритет отдается приоритезации требований, чтобы клиенты могли получать ценное программное обеспечение как можно быстрее.
Коммуникация с заказчиком и заинтересованными сторонами	Коммуникация обычно формальная и письменная.	Приоритет отдается неформальному общению (личной беседе). Подход подразумевает частое общение непосредственно с клиентом, заинтересованными лицами и пользователями с целью получения обратной связи и формирования скоупа новых требований, улучшений.
Обработка изменений	Изменения возможны только после реализации основного утвержденного скоупа продукта и обрабатываются как мини-проекты, которые должны быть одобрены заказчиком.	Изменения ожидаются и, как правило, осуществляются на основе приоритета от бизнеса.

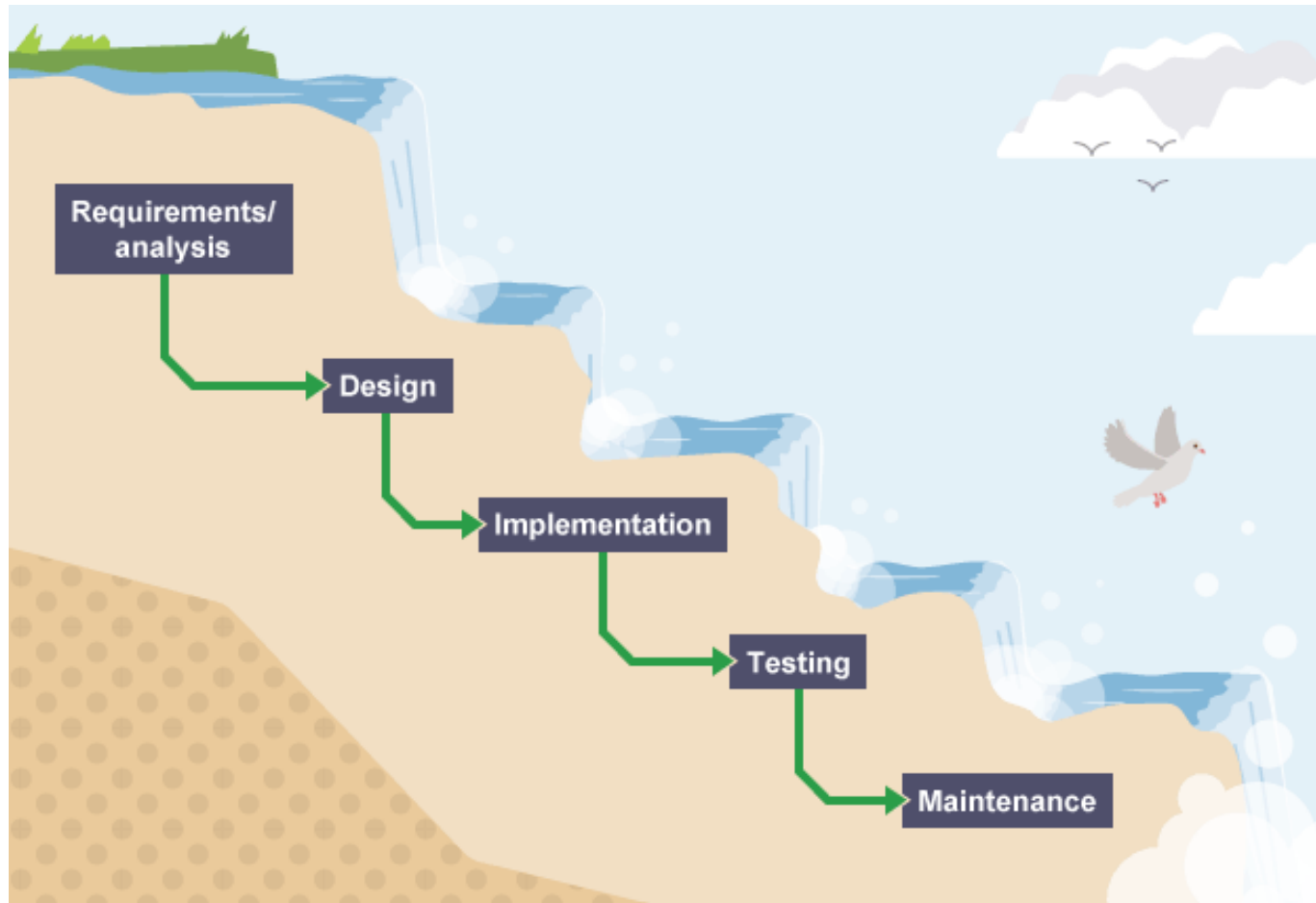
Plan-driven SDM



Plan-driven SDM:

- Waterfall;
- Итерационная модель;
- RUP

Plan-driven SDM: Waterfall



1

Заранее спланирован объем работ

2

Исчерпывающая документация

3

Понятная структура модели

4

Тщательная проверка результата на каждом этапе

5

Фокус на конечной цели

Plan-driven SDM: Waterfall



1

Сложность внести изменения

2

Исчерпывающий дизайн и документация

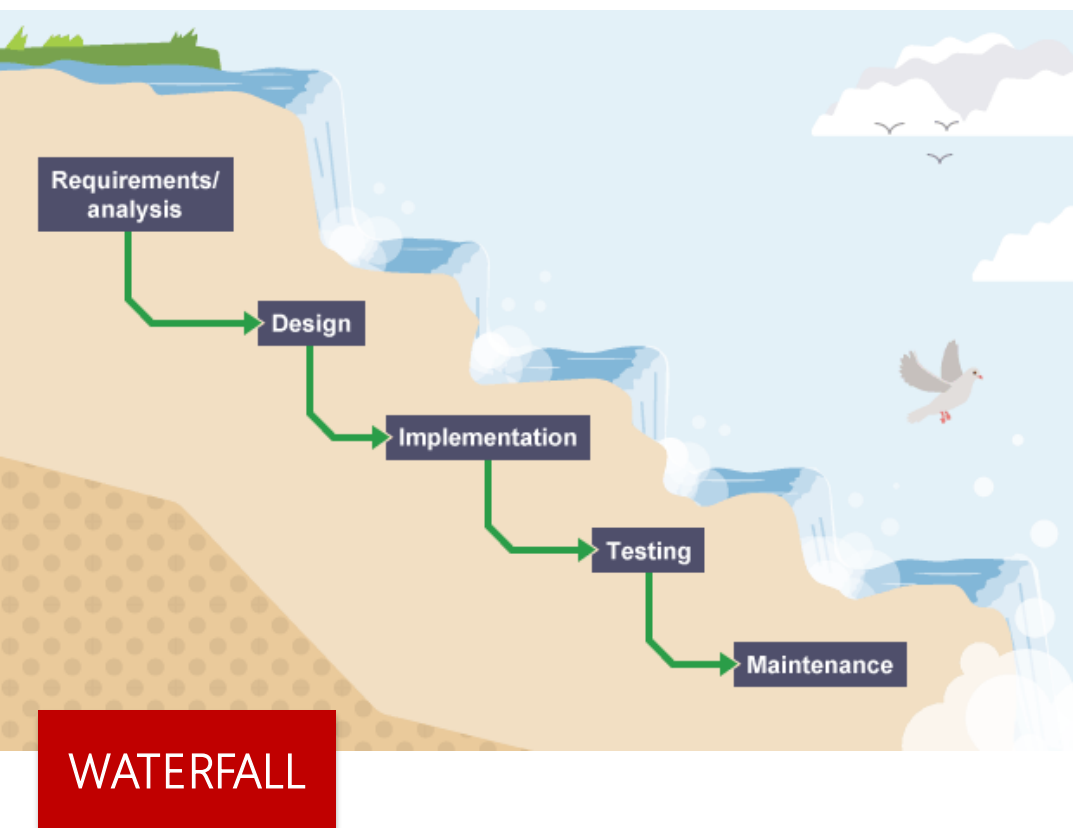
3

Удорожание продукта

4

Опасность выпустить продукт, который уже не будет актуальным для потребностей конечного пользователя

Plan-driven SDM: Итерационная модель



DISCIPLINES

BUSINESS MODELLING

REQUIREMENTS

ANALYSIS & DESIGN

IMPLEMENTATION

TEST

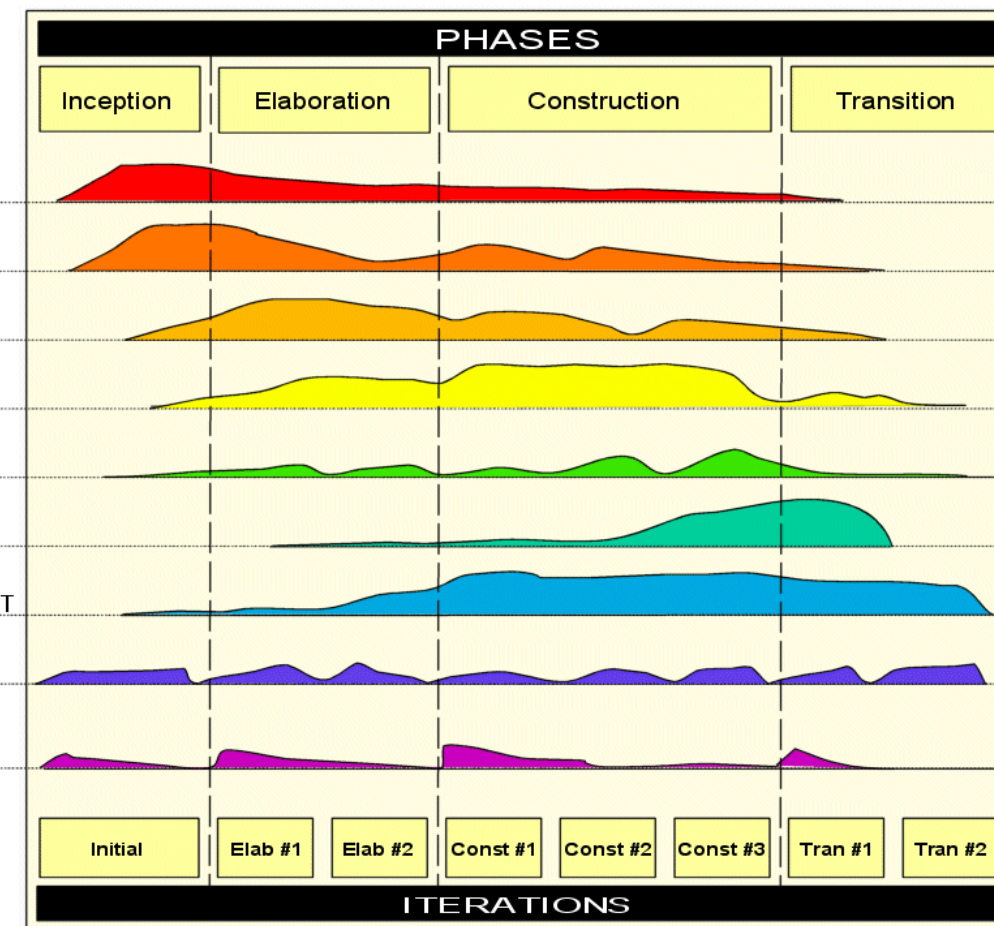
DEPLOYMENT

CONFIGURATION & CHANGE MANAGEMENT

PROJECT MANAGEMENT

ENVIRONMENT

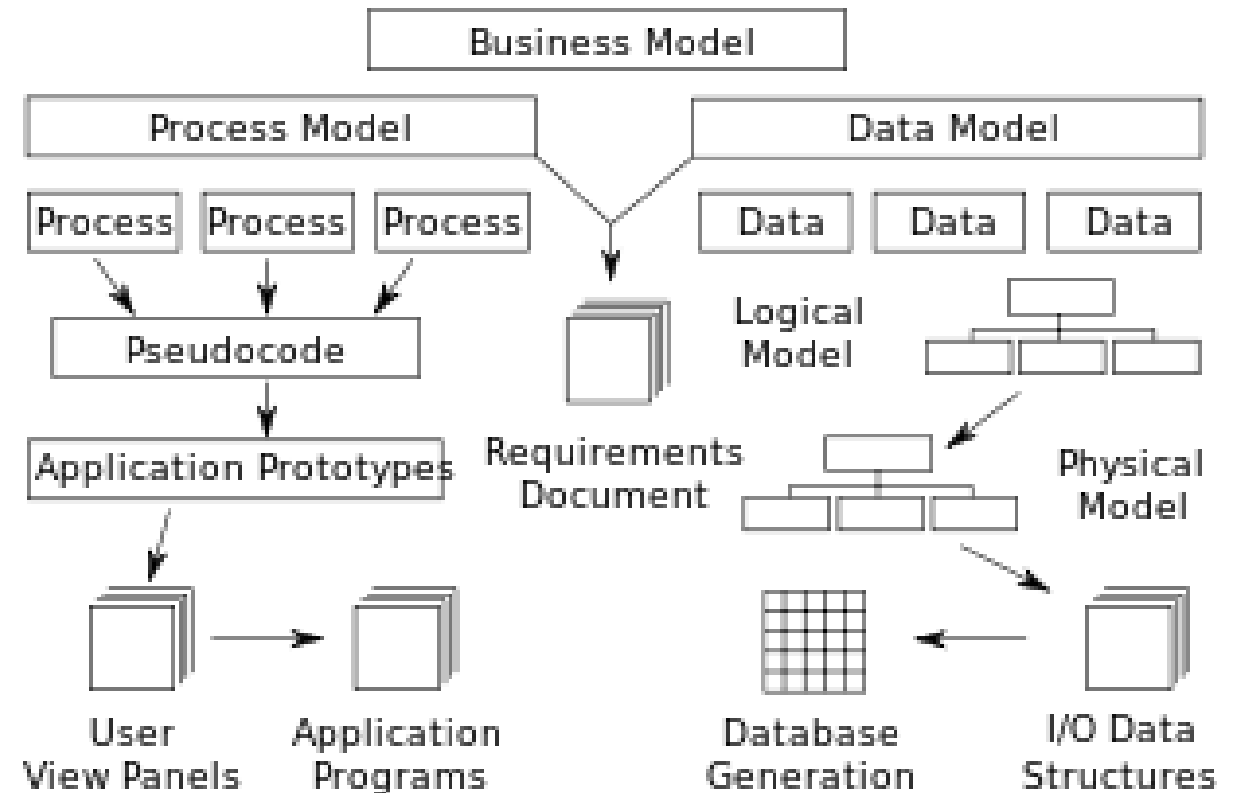
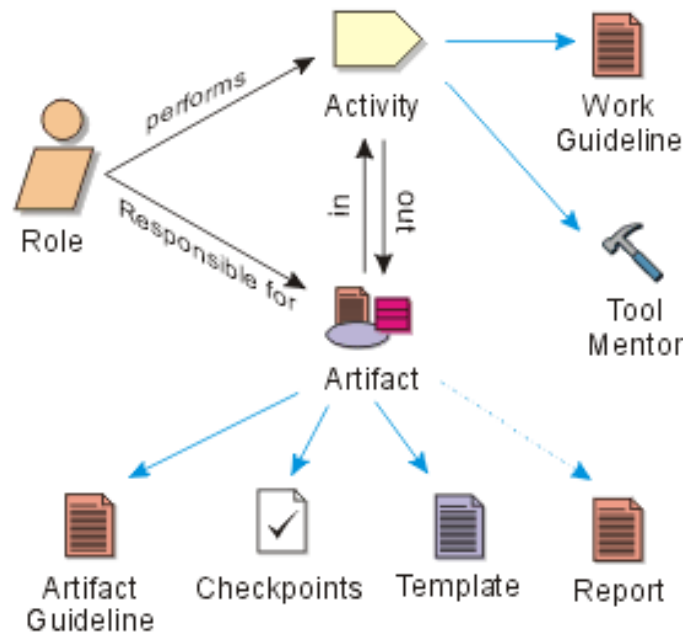
Итерационная
модель



Plan-driven SDM: RUP

1

Опирающийся на модели



Plan-driven SDM: RUP

2

Варианты использования: фокус на том, что делает система для каждого пользователя

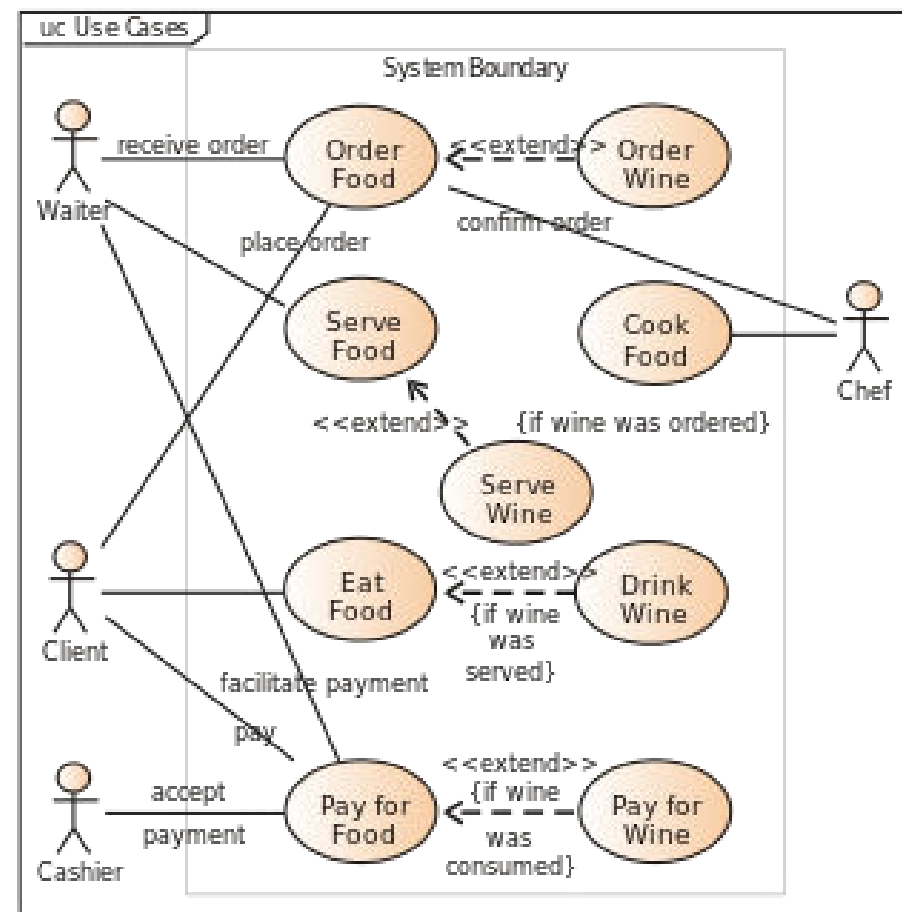
Использование Use Cases:

Описывают бизнес-процесс и область приложения

Позволяют обнаружить неточности и неясности требований на раннем этапе

Помогают приоритизировать требования

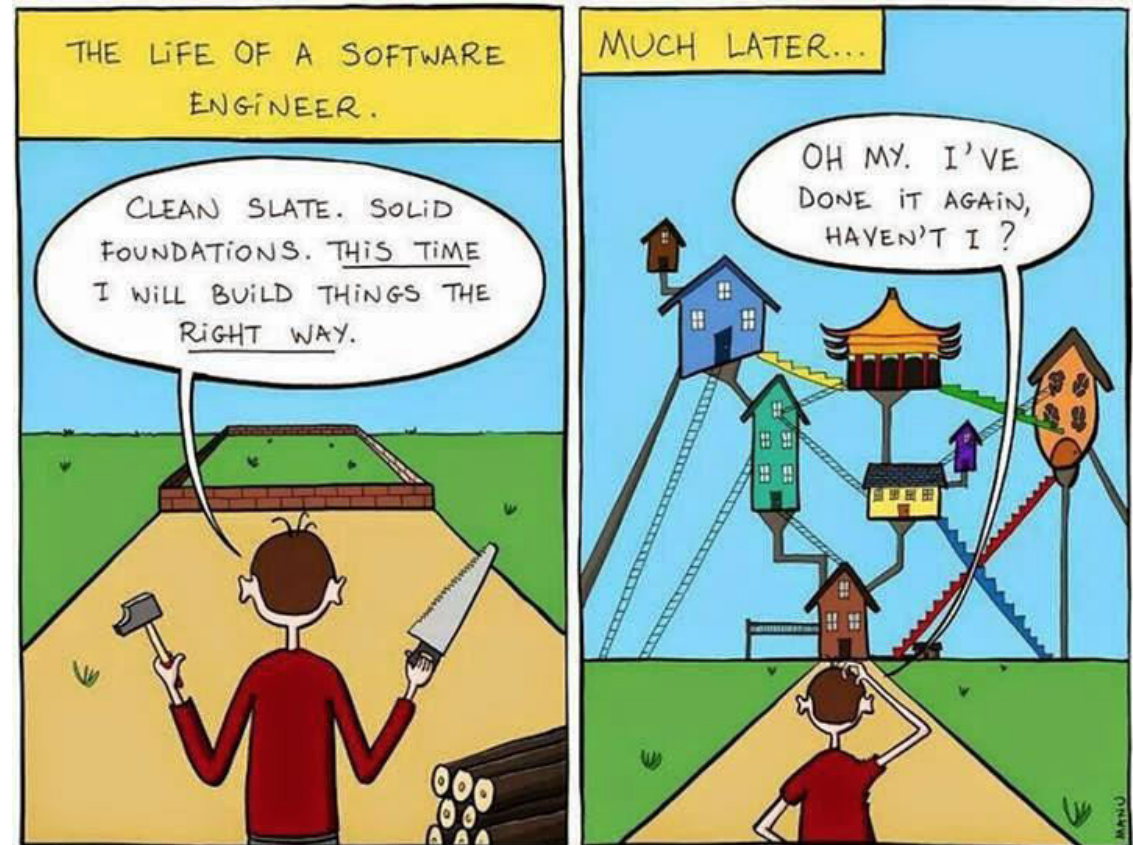
Хорошо трассируются (показывают пошаговое выполнение действий в программе)



Plan-driven SDM: RUP

3

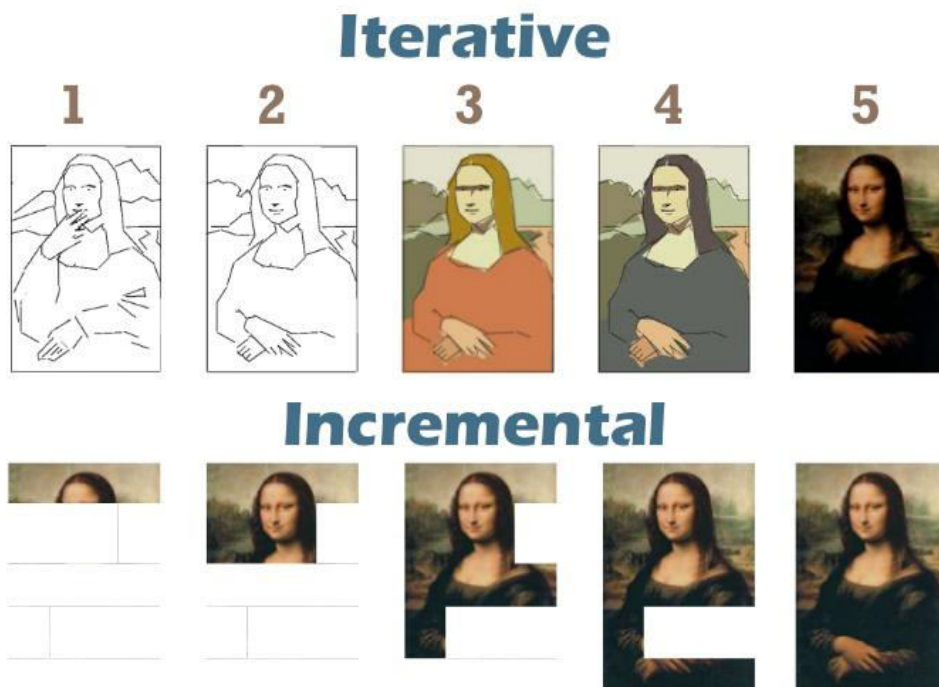
Архитектурно-ориентированный



Plan-driven SDM: RUP

4

Итеративный и инкрементный



Product Vision

«Что пытаемся сделать?»

Software Development Plan

«Как будем делать?»

Risk List

«Что может этому помешать?»

Iteration Plan

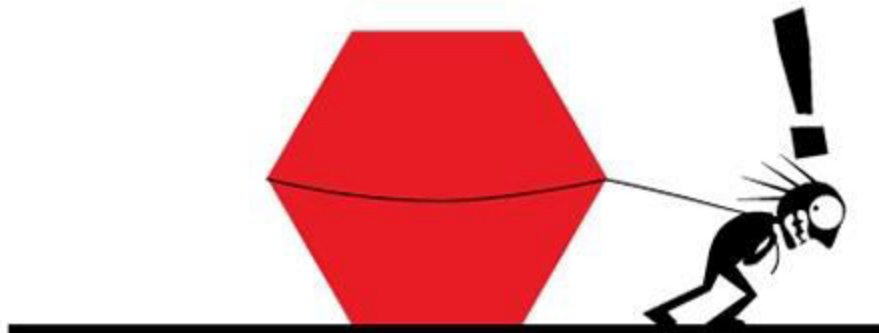
«Наши ближайшие задачи?»

Use-case model

«Как будут пользователи работать с продуктом?»

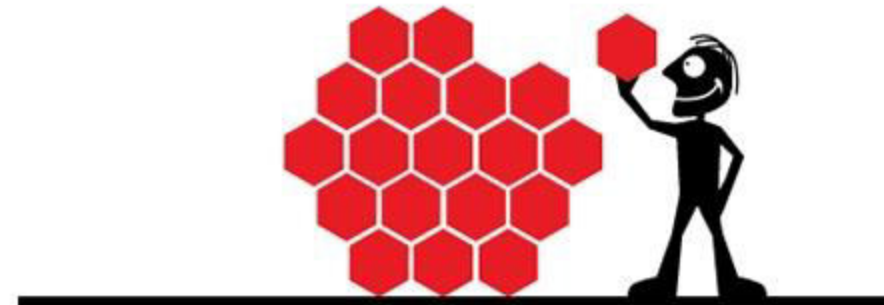
Plan-driven VS Change-driven SDM

THE WATERFALL PROCESS



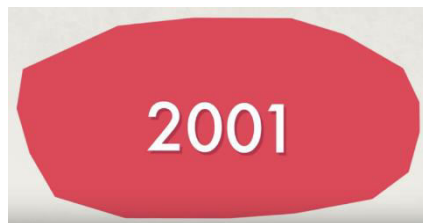
*'This project has got so big,
I'm not sure I'll be able to deliver it!'*

THE AGILE PROCESS



*'It's so much better delivering this
project in bite-sized sections'*

Change-driven SDM: Agile



Люди и взаимодействие важнее процессов и инструментов;
Работающий продукт важнее исчерпывающей документации;
Сотрудничество с заказчиком важнее согласования условий контракта;
Готовность к изменениям важнее следования первоначальному плану.

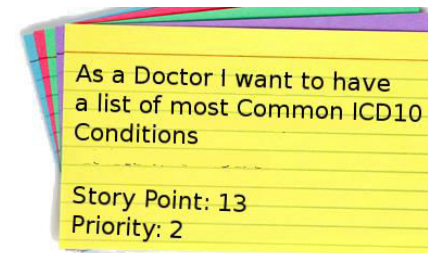
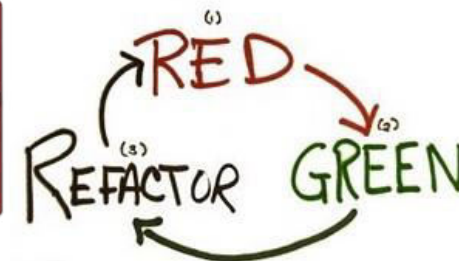
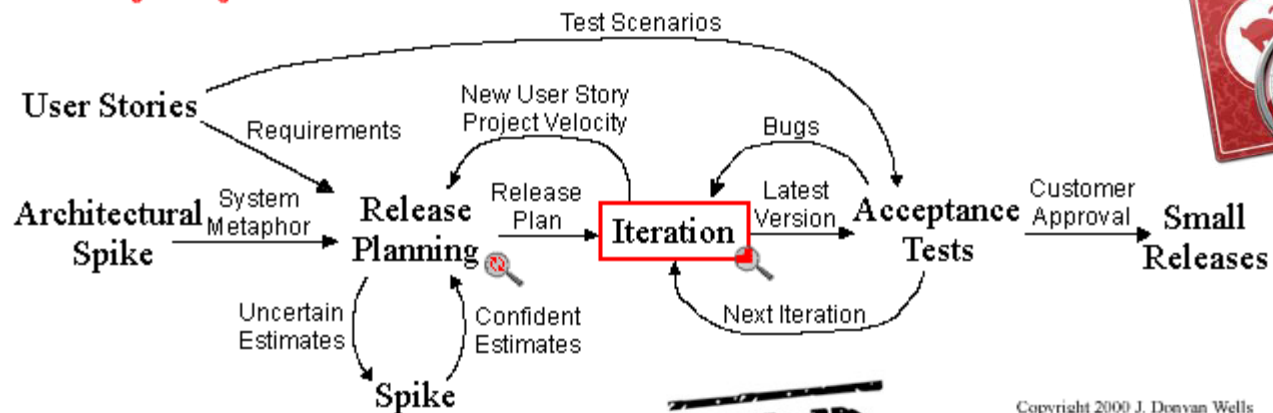
Agile Манифест

Методологии управления проектами. Введение в SCRUM

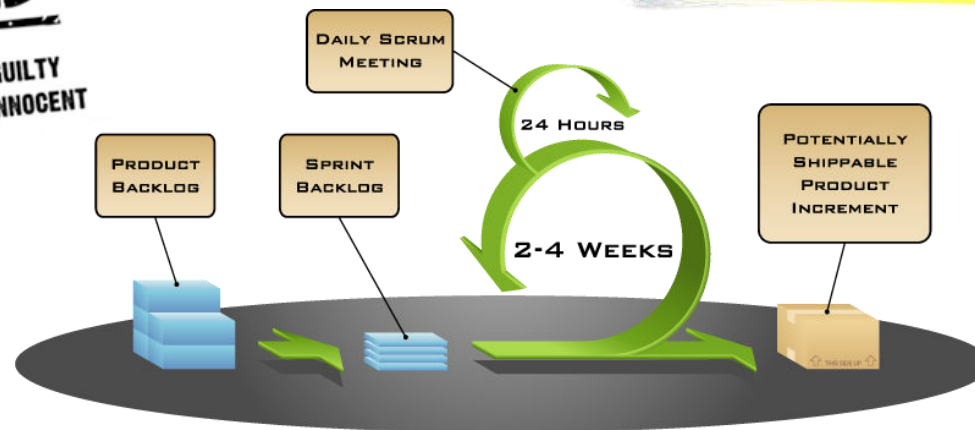
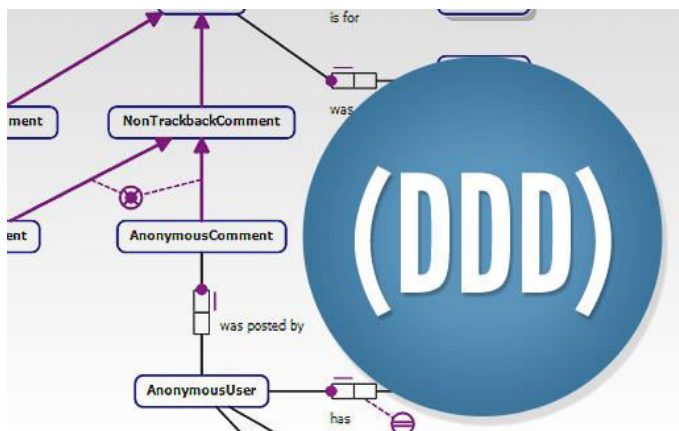
Change-driven SDM: Agile подходы (методологии)



Extreme Programming Project



Copyright 2000 J. Donovan Wells

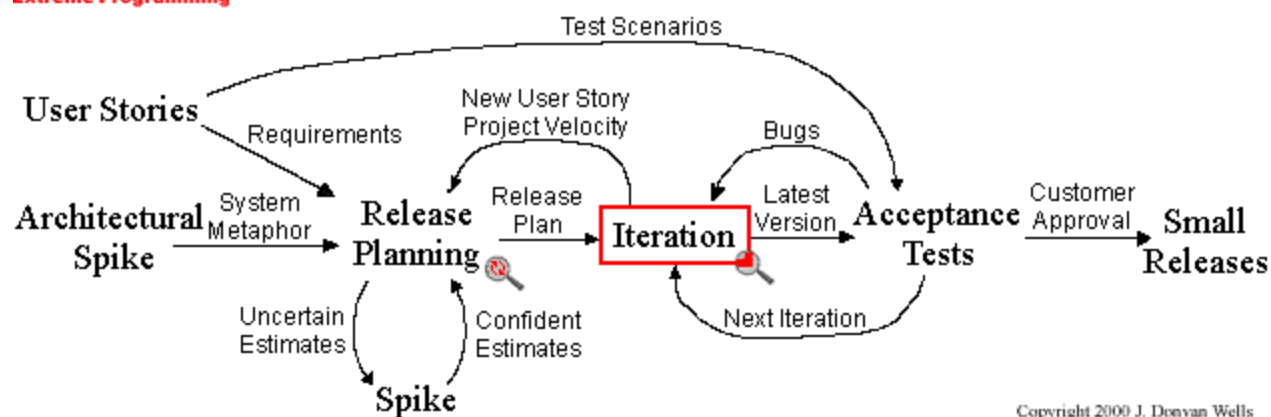


COPYRIGHT © 2005, MOUNTAIN GOAT SOFTWARE

Agile: Экстремальное программирование

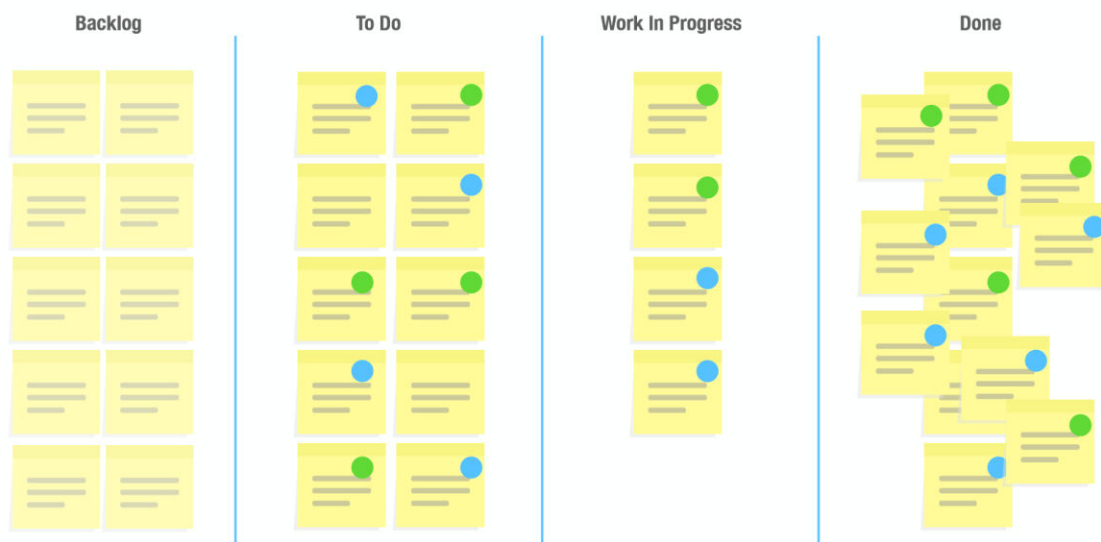


Extreme Programming Project



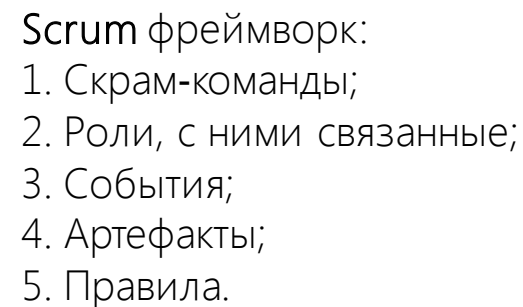
1. Экстремальное программирование позволяет разработчикам **реагировать на изменяющиеся** требования клиентов даже на поздних этапах жизненного цикла;
2. Менеджеры, клиенты и разработчики являются равноправными партнерами в совместной команде (**Короткий цикл обратной связи**);
3. Экстремальное программирование улучшает проект программного обеспечения пятью основными способами: **общение, простота, обратная связь, уважение и смелость**;
4. Парное программирование;
5. Непрерывная интеграция;
6. Рефакторинг и коллективное владение кодом.

Agile: Kanban



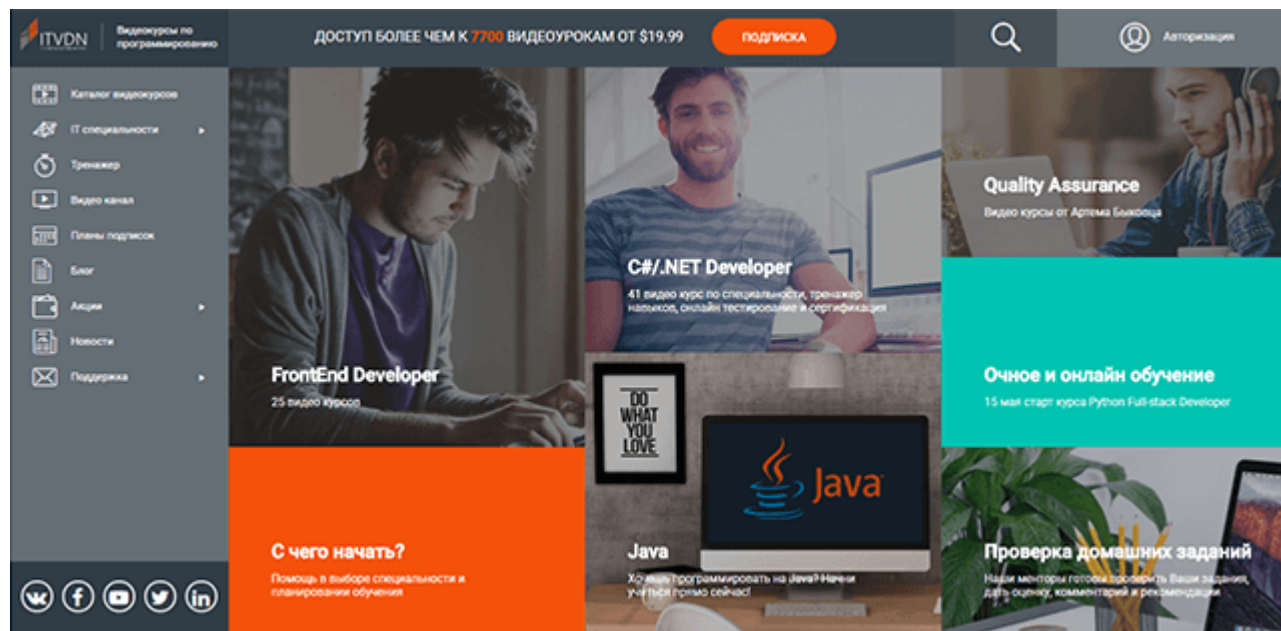
1. Ответственность разделяется на всю команду;
2. Использование средств визуализации: Канбан Доска и карты;
3. Нет заранее определенного графика выполнения задач, но работа разбита и отображается визуально;
4. Постоянный и непрерывный процесс разработки, который ограничен только рабочей нагрузкой членов команды;
5. Оценивание задач: «как можно скорее»;
6. Постоянное совершенствование;
7. Очень гибкая и благоприятная методология для работы с изменениями;
8. НО! Не для больших команд;
9. НО! Недостаток планирования времени на определенную задачу (работа над конкретной задачей продолжается до тех пор, пока она не будет завершена, поэтому сложно предсказать сроки и контрольные точки).

Agile: SCRUM



Смотрите наши уроки в видео формате

ITVDN.com



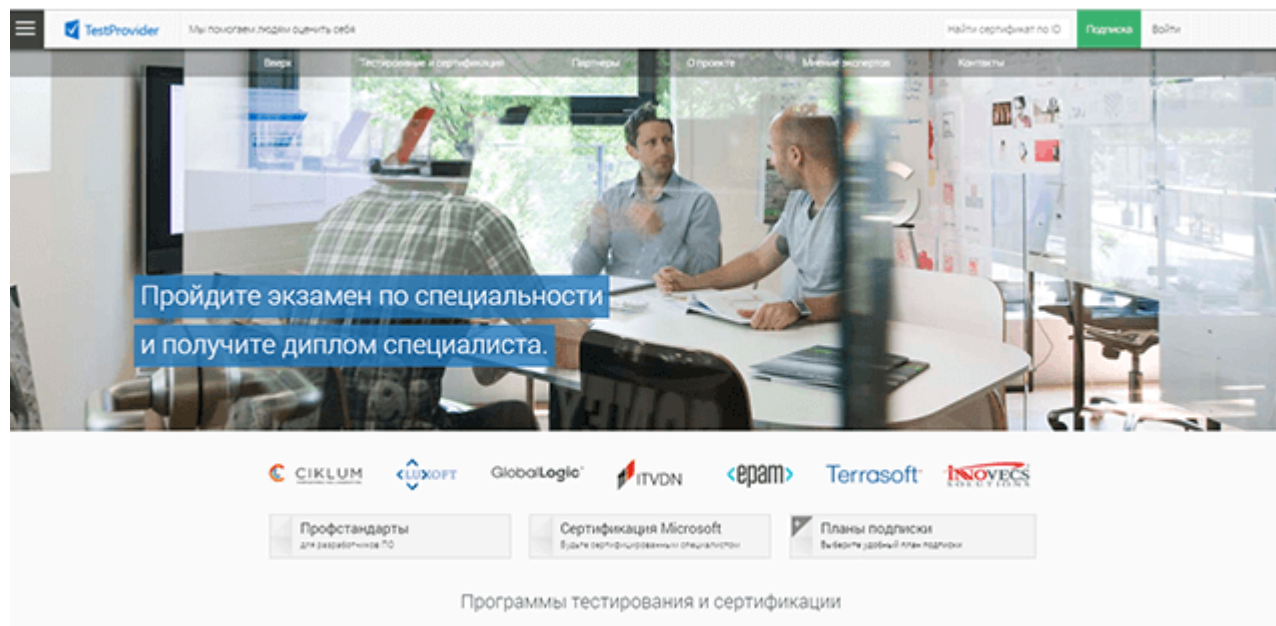
Посмотрите этот урок в видео формате на образовательном портале ITVDN.com для закрепления пройденного материала.

Курсы записаны сертифицированными тренерами, которые работают в учебном центре CyberBionic Systematics и другими высококвалифицированными разработчиками.



Проверка знаний

TestProvider.com



TestProvider – это online сервис проверки знаний по информационным технологиям. С его помощью Вы можете оценить Ваш уровень и выявить слабые места. Он будет полезен как в процессе изучения технологии, так и для общей оценки знаний IT специалиста.

После каждого урока проходите тестирование для проверки знаний на [TestProvider.com](https://testprovider.com)

Успешное прохождение финального тестирования позволит Вам получить соответствующий Сертификат.



Q&A

Информационный видеосервис для разработчиков программного обеспечения

