

ЛЕКЦИЯ 11

«Жизненный цикл проекта»



Содержани

е:

- Модели ЖЦ проекта
- Гибкие методологии
- Скрам-процесс

Модели жизненного цикла проекта

- Моделирование жизненного цикла проекта по **принципу «водопада» (каскадная)**
- Моделирование жизненного цикла проекта по **итеративной модели (инкрементная)**

Водопадная модель

При моделировании по принципу «водопада» процесс разработки делится на четко определенные фазы, выполняемые строго последовательно:

- анализ требований (исследование среды);
- проектирование;
- разработка и реализация подпроектов;
- проверка подпроектов и проверка проекта в целом;
- внедрение.

Водопадная модель



Водопадная модель

Преимущества:

- Полная и согласованная документация на каждом этапе.
- Легко определить сроки и затраты на проект.

Недостатки:

- Процесс плохо работает в проектах с нечеткими требованиями.
- Сложно управлять рисками некоторых типов (например, риски, связанные с использованием новых технологий).

Итеративная модель

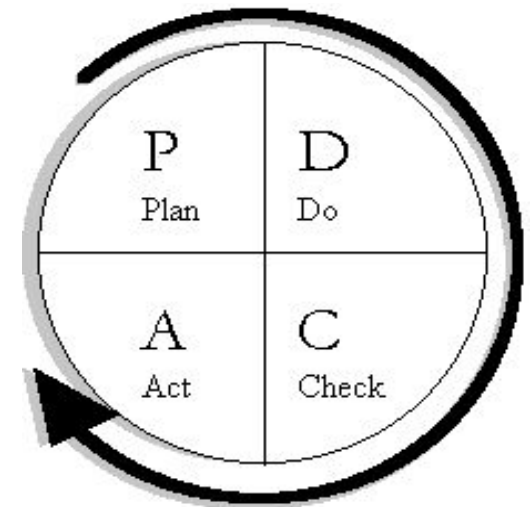
Проект при этом подходе в каждой фазе развития проходит повторяющийся цикл **PDCA**:

Планирование —

Реализация — Проверка —

Оценка

(англ. plan-do-check-act cycle).



Итеративная модель



Итеративная

Преимущества итеративного подхода:

- снижение воздействия серьёзных рисков на ранних стадиях проекта;
- организация эффективной обратной связи;
- акцент усилий на наиболее важные и критичные направления проекта;
- непрерывное итеративное тестирование;
- раннее обнаружение конфликтов между требованиями, моделями и реализацией проекта;
- более равномерная загрузка участников проекта;
- реальная оценка текущего состояния проекта и, как следствие, большая уверенность заказчиков и непосредственных участников в его успешном завершении.

Итеративная модель

Недостатки:

- целостное понимание возможностей и ограничений проекта долгое время отсутствует;
- при итерациях приходится отбрасывать часть сделанной ранее работы;
- добросовестность специалистов при выполнении работ всё же снижается («всё равно всё можно будет переделать и улучшить позже»).

Гибкие методологии

Гибкая методология разработки (Agile software development) – серия подходов к разработке ПО, ориентированных на использование итеративной разработки, динамическое формирование требований и обеспечение их реализации в результате постоянного взаимодействия внутри самоорганизующихся рабочих групп, состоящих из специалистов различного профиля.

*«Манифест гибкой методологии разработки программного обеспечения»,
Февраль 2001, Юта, США*

Гибкие МЕТОДОЛОГИИ

Основные идеи:

- **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.

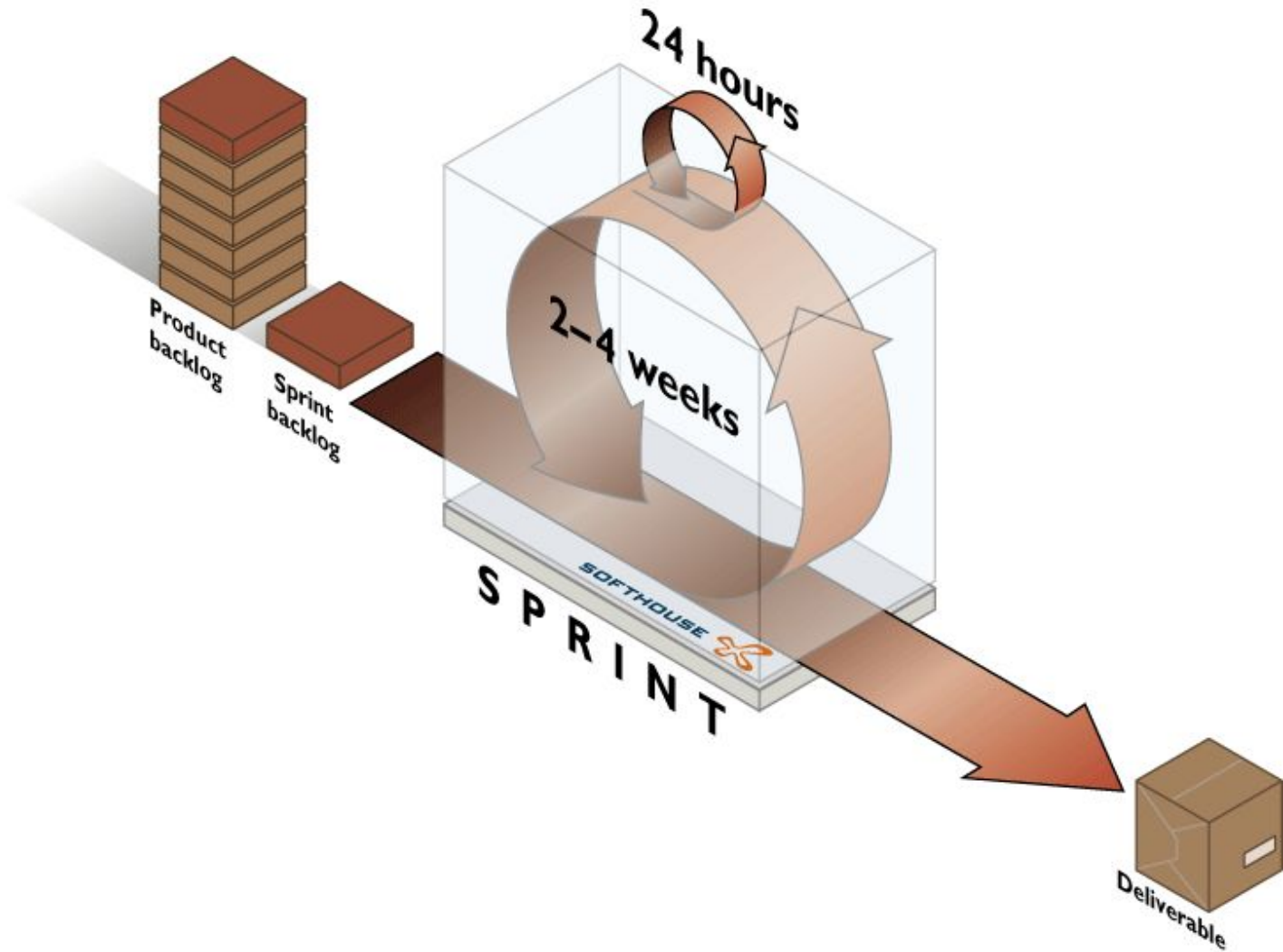
Основная метрика agile-методов – рабочий продукт.

Гибкие МЕТОДОЛОГИИ

Scrum (от англ. scrum «толкучка») — методология управления проектами, активно применяющаяся при разработке информационных систем для гибкой разработки программного обеспечения.

Это набор принципов, на которых строится процесс разработки, позволяющий в жёстко фиксированные и небольшие по времени итерации, называемые спринтами (sprints), предоставлять конечному пользователю работающее ПО с новыми возможностями, для которых определён наибольший приоритет.

Scrum



Scrum

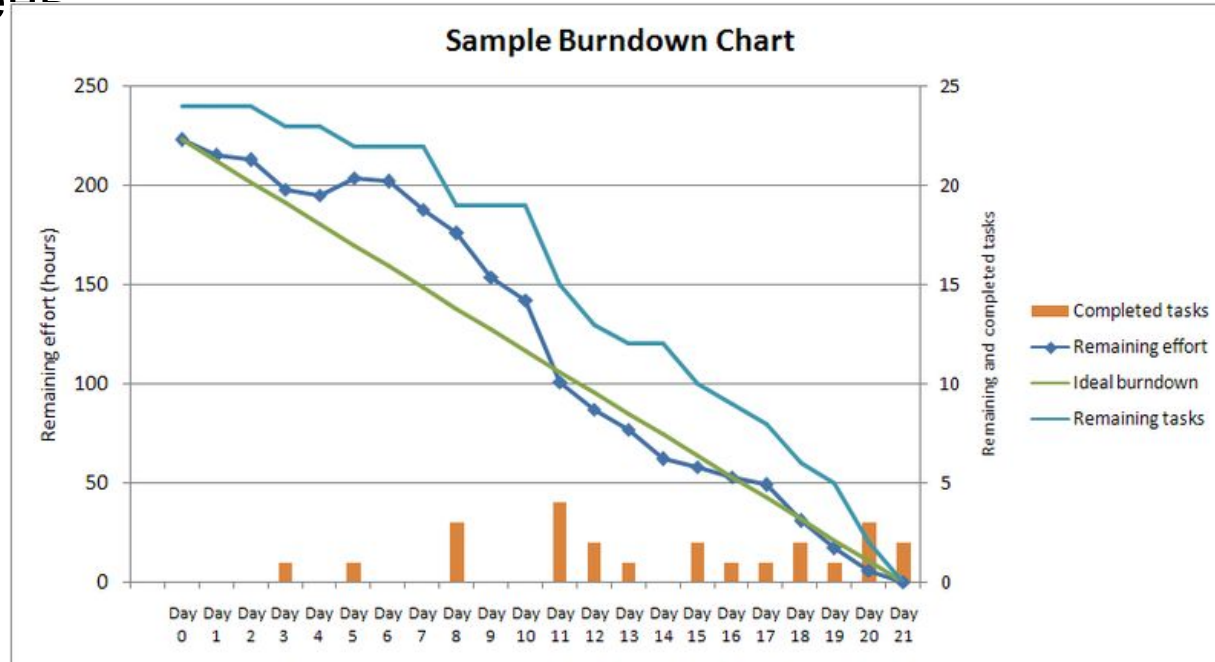
Спринт (Sprint) — итерация в скраме, в ходе которой создаётся функциональный рост программного обеспечения. Жёстко фиксирован по времени. Длительность одного спринта от 2 до 4 недель.

Бэклог проекта (Project backlog) — это список требований к функциональности, упорядоченный по их степени важности, подлежащих реализации.

Бэклог спринта (Sprint backlog) — содержит функциональность, выбранную владельцем проекта из Бэклога проекта. Все функции разбиты по задачам, каждая из которых оценивается скрам-командой.

Scrum

Диаграмма сгорания задач (Burndown chart) – отображает завершенный спринт: оставшиеся нерешенные задачи и трудозатраты, необходимые для их завершения в расчете на 21 рабочий день.



Роли в Scrum



Роли в

Scrum

Основные роли – «Свиньи»:

«Свиньи» полностью включены в проект и в скрам-процесс.

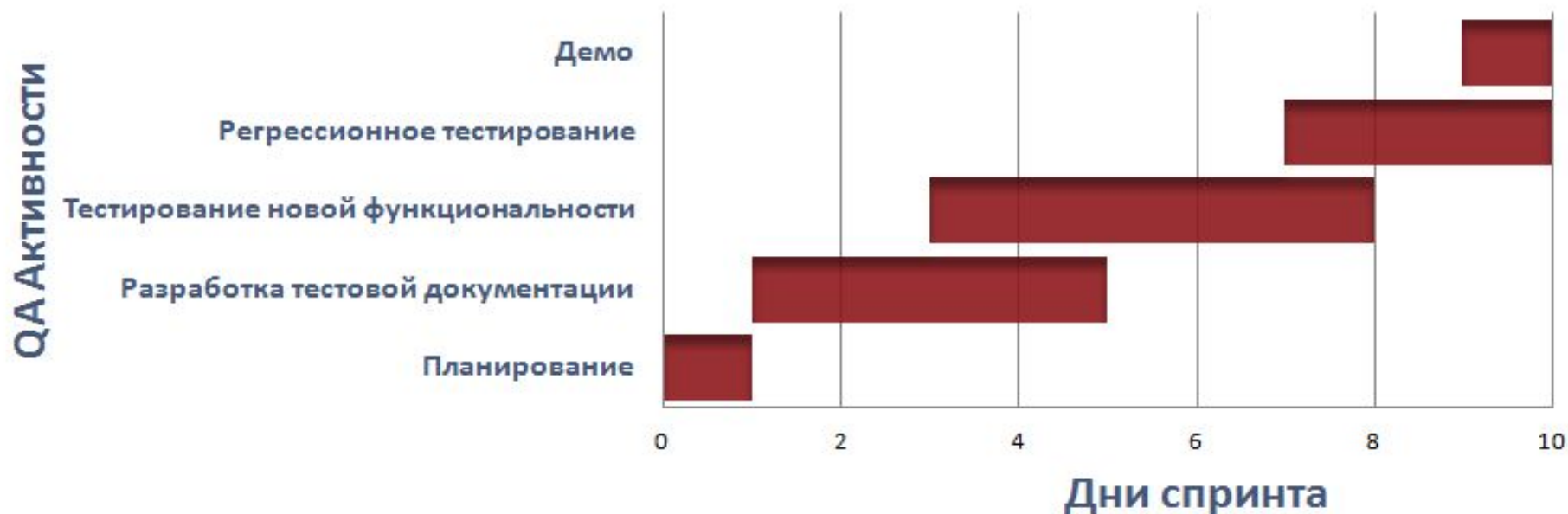
- Скрам-мастер (Scrum Master)
- Владелец продукта (Product Owner)
- Скрам-команда (Scrum Team)

Дополнительные роли – «Куры»:

- Пользователи (Users)
- Клиенты, Продавцы (Stakeholders)
- Управляющие (Managers)
- Эксперты-консультанты (Consulting Experts)

Scrum

QA активности в рамках Sprint



Scrum

Прочие QA активности:

- Планирование спринта (Sprint Planning meeting).
- Ежедневное совещание (Daily Scrum meeting).
- Ретроспективное совещание (Retrospective meeting).

Спасибо за
внимание! Жду
Ваших вопросов

