



> Конспект > 3 урок > **РАЗВИТИЕ ПРОДУКТА**

> **Оглавление урока по Развитию продукта**

1. Что такое управление проектами и зачем для этого методологии
2. Ключевые фазы проекта
3. Методологии управления проектами
4. Waterfall (каскадная модель, «водопад»)
5. Философия Agile
6. 12 принципов гибкой методологии Agile
7. Scrum
8. Kanban

Презентация

> **Что такое управление проектами и зачем для этого методологии**

Ключевые характеристики проекта: ограниченность во времени и уникальный готовый продукт на выходе.

Ограничения проекта – сроки, стоимость и содержание.

Методология = стандартизация проведения проектов.

Выбор методологии определяет этапы и процессы взаимодействия проектных команд.

> Ключевые фазы проекта

1. Инициация — выдвижение идеи и подготовка проектных документов.
2. Планирование — определение сроков и этапов реализации проекта, назначение исполнителей и ответственных лиц.
3. Исполнение — реализация всех намеченных действий.
4. Контроль – слежка за ходом реализации проекта.
5. Завершение — анализ полученных данных и контроль на предмет соответствия факта плану.

> Методологии управления проектами

Две самые популярные методологии в IT – Waterfall («водопад» / «каскадная») и Agile (в частности её ветвь - Scrum).

Также существуют такие методологии как:

- APF (Adaptive Project Framework).
- BF (Benefit Realization).
- CPM (Critical Path Method) — метод критического пути.
- ECM (Event Chain Methodology) — моделирование событий.
- XP (Extreme Programming) — экстремальное программирование.
- Lean. Бережливое производство.
- Six Sigma (6 сигм).
- PRINCE2 (Projects in Controlled Environments).
- Проект в контролируемой среде.

- PRISM — Устойчивые методы.
- PBPM — Процессно-ориентированный метод.

> **Waterfall (каскадная модель, «водопад»)**

Методология **Waterfall** – самая «старая» из всех. Изложена Уинстоном Уокером Ройсом в 1970 году в ответ на потребность управления все более усложняющимся процессом разработки программного обеспечения.

Ключевые особенности: Последовательность и ориентация на требования. **Основные этапы:** Сбор и анализ требований, разработка и внедрения решения и исправление проблем. **Каждый этап этого процесса автономен:** Чтобы перейти к следующему, необходимо завершить предыдущий этап.

Пример:

Проект: разработать сайт. Этапы:

1. Написать техническое задание
2. Нарисовать дизайн
3. Сверстать дизайн
4. Закодировать
5. Протестировать
6. Запустить проект

Чтобы двигаться по водопаду, нужно иметь четкое техническое задание и понимание шагов, следующих друг за другом.

Преимущества и недостатки

Преимущества:

- Простота использования.
- Структура.
- Документация.

Недостатки:

- Повышенный риск.
- Расходы выше запланированного.
- Длинные сроки.
- Устаревшие решения.

Для каких проектов лучше всего подойдет Waterfall

- Коротких несложных проектов.
- Проектов с чётко установленными требованиями без противоречий.
- Проектов, в которых меняются ресурсы, зависимые от подробной документации.

> Философия Agile

Agile (agile software development, от agile – проворный) — гибкая методология разработки.

Возник в феврале 2001 года, когда 17 разработчиков опубликовали Манифест гибкой разработки программного обеспечения, с рассказом о «лучших способах разработки программного обеспечения».

Характеризуется небольшими циклическими изменениями, которые внедряют в ответ на изменение требований. Возник в IT-среде, но затем распространился и в другие сферы – от промышленной инженерии до искусственного интеллекта.

Ключевые особенности: поощряет постоянную обратную связь от конечных пользователей вместо тщательного планирования. **Процесс:** кросс-функциональные команды работают над итерациями продукта в течение определенного периода времени. **Цель:** цель каждой итерации — создать рабочий продукт.

Ценности

1. Люди и взаимодействие важнее процессов и инструментов.
2. Работающий продукт важнее исчерпывающей документации.
3. Сотрудничество с клиентом важнее согласования условий контракта.

4. Готовность к изменениям важнее следования первоначальному плану.

Преимущества и недостатки

Преимущества:

- Гибкость и свобода.
- Пониженный риск.

Недостатки:

- Отсутствие четкого плана.
- Сложность взаимодействия.

Для каких проектов лучше всего подойдет

- Когда нет уверенности в том, каким должен быть конечный результат, но есть общее представление о продукте.
- Когда проект нужно быстро подстраивать под изменения.
- Если взаимодействие и коммуникация — сильные стороны команды, а планирование — нет.

Какому продукту точно нужен agile?

- Большой и технологически сложный. Когда дешевле делать все постепенно и постоянно тестировать, чем переделывать уже готовый продукт.
- Длительный по времени. Чем дольше проект будет функционировать, тем тяжелее представить его развитие — например, интернет-магазин.
- С высокой неопределённостью. Когда проект инновационный, невозможно заранее продумать все функции, проще делать его маленькими рывками и тестировать.
- Когда идей много и непонятно, какие из них окажутся удачными. Внедрять все сразу — рискованно и экономически неоправданно.
- С идеальным заказчиком. Когда клиент настолько заинтересован в продукте, что хочет сам во всём участвовать.

> 12 принципов гибкой методологии Agile

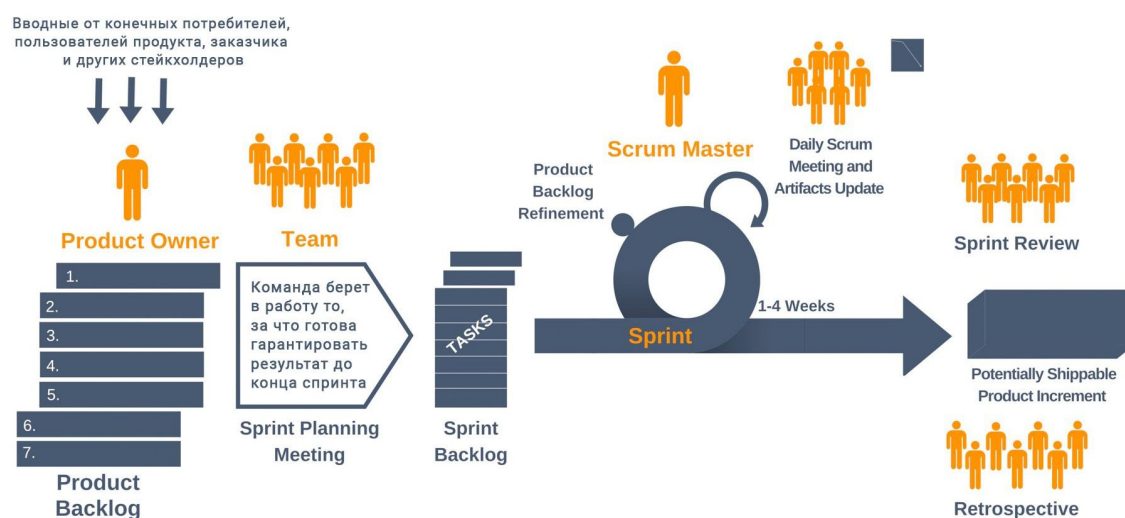
1. Наивысшим приоритетом является удовлетворение потребностей клиента, благодаря регулярной и ранней поставке ценного программного обеспечения.
2. Изменение требований приветствуется, даже на поздних стадиях разработки.
3. Работающий продукт следует выпускать как можно чаще, с периодичностью от пары недель до пары месяцев.
4. На протяжении всего проекта разработчики и представители бизнеса должны ежедневно работать вместе.
5. Над проектом должны работать мотивированные профессионалы. Чтобы работа была сделана, создайте условия, обеспечьте поддержку и полностью доверьтесь им.
6. Непосредственное общение является наиболее практичным и эффективным способом обмена информацией как с самой командой, так и внутри команды.
7. Работающий продукт — основной показатель прогресса.
8. Инвесторы, разработчики и пользователи должны иметь возможность поддерживать постоянный ритм бесконечно.
9. Постоянное внимание к техническому совершенству и качеству проектирования повышает гибкость проекта.
10. Простота — искусство, минимизации лишней работы — крайне необходима.
11. Самые лучшие требования, архитектурные и технические решения рождаются у самоорганизующихся команд.
12. Команда должна систематически анализировать возможные способы улучшения эффективности и соответственно корректировать стиль своей работы.

> Scrum

Scrum – это «подход структуры». Над каждым проектом работает универсальная команда специалистов (+ владелец продукта и scrum-мастер)

- Product owner соединяет команду с заказчиком и следит за развитием проекта (= куратор).
- Scrum-мастер помогает организовать бизнес-процесс и следит за соблюдением scrum-подхода.

Scrum-подход делит рабочий процесс на равные спринты.



> Kanban

Kanban – это «подход баланса». Основная задача: сбалансировать разных специалистов внутри команды и избежать ситуаций неоптимального расходования времени. Вся команда едина. Бизнес-процесс делится не на универсальные спринты, а на стадии выполнения конкретных задач: «Планируется», «Разрабатывается», «Тестируется», «Завершено» и другие. Главный показатель эффективности – среднее время прохождения задачи по доске.

