

# Программная инженерия, лекция 10

## Введение в Agile, бизнес-модель

Пожидаев М. С.

21 ноября 2022 г.

Agile

Agile software development — это набор *принципов* и правил, в рамках которого осуществляется разработка ПО.

**не единственный вариант**

Agile — это семейство процессов, а не один конкретный подход.

Agile не описывает конкретных практик, а фиксирует ценности и принципы, которыми руководствуются команды.

# Agile Manifesto

## *Ценности и принципы*

Суть Agile методологии сформулирована в документе «Agile Manifesto». разработан и принят 11–13 февраля 2001 г. на лыжном курорте The Lodge at Snowbird в шт. Юта. Подписали представители следующих методологий:

- ▶ Extreme programming;
- ▶ Scrum;
- ▶ DSDM;
- ▶ Adaptive Software Development;
- ▶ Crystal Clear;
- ▶ Feature-Driven Development;
- ▶ Pragmatic Programming.

# Короткие циклы

*Снова итерации, куда без них?*

Гибкие методологии сфокусированы на минимизацию рисков путём сведения разработки к серии коротких циклов, называемых итерациями, ориентированными обычно на 1–2 недели. Итерация воспринимается как минимальный проект и включает все задачи, направленные на получение инкремента: планирование, анализ требований, проектирование, реализацию и тестирование.

**Всегда готов!**

Хотя отдельная итерация, как правило, недостаточна для выпуска новой версии продукта, подразумевается, что гибкий программный проект готов к выпуску в конце каждой итерации.

После окончания итерации производится ревизия целей разработки.

# Ориентирование на небольшие команды

## *Взаимодействие*

Методы Agile делают упор на живое общение участников разработки. Большинство команд расположены в одном офисе, иногда называемом bullpen.

### Вместе с заказчиком!

Эту роль может выполнять менеджер проекта, бизнес-аналитик или клиент.

### В офисе также присутствуют:

тестеры, дизайнеры интерфейсов, технические писатели и менеджеры.

# Ценности

*Результат — это работающий программный продукт.*

Воспринимая работающий программный продукт как единственный показателя работы команды за период времени, сформулированы следующие ценности:

- ▶ личности и их взаимодействия важнее, чем процессы и инструменты;
- ▶ работающее программное обеспечение важнее, чем полная документация;
- ▶ сотрудничество с заказчиком важнее, чем контрактные обязательства;
- ▶ реакция на изменения важнее, чем следование плану.

# Принципы

## *Первая половина принципов Agile*

1. Удовлетворение клиента за счёт ранней и бесперебойной поставки ценного ПО.
2. Приветствие изменения требований, даже в конце разработки.
3. Частая поставка рабочего ПО (каждый месяц или неделю или ещё чаще).
4. Тесное, ежедневное общение заказчика с разработчиками.
5. Проектом занимаются мотивированные личности, которые обеспечены нужными условиями работы, поддержкой и доверием.
6. Рекомендуемый метод передачи информации — личный разговор.

# Принципы (продолжение)

## *Вторая половина принципов Agile*

1. Работающее ПО — лучший измеритель прогресса.
2. Спонсоры, разработчики и пользователи должны иметь возможность поддерживать постоянный темп на неопределенный срок.
3. Постоянное внимание на улучшение технического мастерства и удобный дизайн.
4. Простота — искусство НЕ делать лишней работы.
5. Лучшие архитектура, требования и дизайн получаются у самоорганизованной команды.
6. Постоянная адаптация к изменяющимся обстоятельствам.



# Scrum

## *Пример Agile с конкретизации практик*

Scrum («свалка» или «драка») делает акцент на качественном контроле процесса разработки. Это набор принципов, на которых строится процесс разработки, позволяющий в фиксированные промежутки времени (спринты, 2–4 недели) предоставлять конечному пользователю работающее ПО с инкрементом.

1. Функционал в очередном спринте определяется до его начала на этапе планирования и не может далее изменяться.
2. Строго фиксированная небольшая длительность спринта придаёт процессу разработки предсказуемость и гибкость.

# Бизнес-модель

*По аналогии с моделью вариантов использования*

Бизнес-модель описывает бизнес-процессы компании в терминах бизнес-вариантов использования и бизнес-актантов для бизнес-процессов и клиентов. Бизнес-модель вариантов использования представляет систему в разрезе ее использования и объясняет, как она обеспечивает получение результата ее пользователями (в данном случае клиентами и партнерами).

# Бизнес-понятия

Каждый бизнес-вариант использования реализуется сотрудниками, использующими бизнес-объекты и рабочие модули. Каждая реализация бизнес-варианта использования может быть описана диаграммами взаимодействия и диаграммами активности.

Бизнес-сущность представляет собой что-то вроде счета, к которому сотрудники имеют доступ и могут проверять, осуществлять операции, создавать или использовать в бизнес-варианте использования. Рабочий модуль — это набор таких сущностей, представляющий конечному пользователю единым целым.

# Разработка

## *Как получить бизнес-модель?*

Бизнес-модель разрабатывается в два приема:

1. Разработчики бизнес-модели должны создать бизнес-модель вариантов использования, идентифицирующую актантов, и бизнес-варианты использования, в которых участвуют эти актанты. Эта модель позволит разработчикам лучше понять, какой результат приносит бизнес его участникам.
2. Разработчики модели должны разработать объектную бизнес-модель, состоящую из сотрудников, бизнес-сущностей и рабочих модулей, которые совместно реализуют бизнес-варианты использования.

Спасибо за внимание!

Всё о курсе: <https://marigostra.ru/materials/engineering.html>

E-mail: [msp@luwrain.org](mailto:msp@luwrain.org)

Канал в Телеграм: <https://t.me/MarigostraRu>