

# Гибкие методологии разработки ПО

Часть 1

**Ксения Шипина**  
Системный аналитик



# Что мы узнаем?

- 1 Зарождение гибких методологий
- 2 Бережливое производство
- 3 Agile-манифест



# Зарождение гибких методологий

Модели и методологии  
Цикл Шухарта — Деминга

1



# Модели и методологии

**Модель разработки программного обеспечения** описывает, какие **стадии жизненного цикла** должна проходить разработка, какие **процессы** должны происходить на каждой стадии, и какие **артефакты** должны быть получены в результате их завершения.

**Методология** — это **набор практик** по управлению разработкой программного продукта: правила, техники и принципы, **позволяющие выстраивать эффективное взаимодействие** и **прозрачную коммуникацию между всеми участниками разработки**.



# Модели и методологии

**Модель разработки программного обеспечения** описывает, какие **стадии жизненного цикла** должна проходить разработка, какие **процессы** должны происходить на каждой стадии, и какие **артефакты** должны быть получены в результате их завершения.

**Методология** — это набор практик по управлению разработкой программного продукта: правила, техники и принципы, позволяющие выстраивать эффективное взаимодействие и прозрачную коммуникацию между всеми участниками разработки.



Модель можно сравнить с рецептом вашего любимого яблочного пирога. Вы знаете набор ингредиентов, примерную последовательность действий и представляете, какой результат вы получите в конце.



# Модели и методологии

Модель разработки программного обеспечения описывает, какие **стадии жизненного цикла** должна проходить разработка, какие **процессы** должны происходить на каждой стадии, и какие **артефакты** должны быть получены в результате их завершения.

**Методология** — это **набор практик** по управлению разработкой программного продукта: правила, техники и принципы, **позволяющие выстраивать эффективное взаимодействие** и **прозрачную коммуникацию между всеми участниками разработки**.



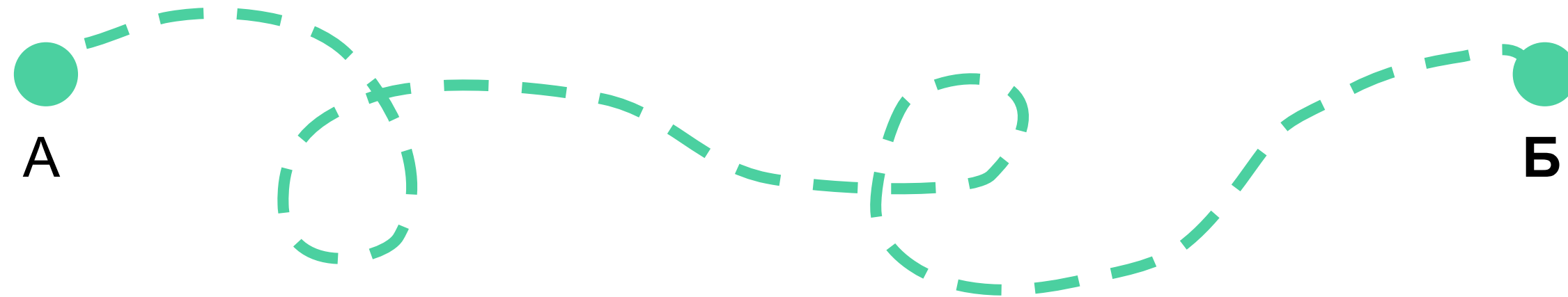
Методологию же можно представить как совокупность ваших навыков в кулинарии и тех инструментов, которые имеются у вас на кухне.

От того, насколько умело вы владеете ножом для нарезки яблок, или от того, насколько хорошо ваша духовка держит температуру, зависит скорость и качество итогового продукта.



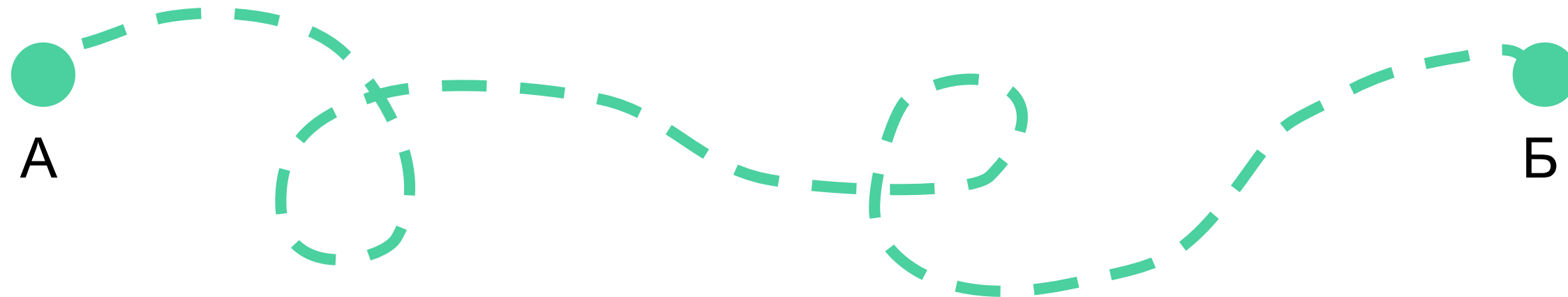
# Модели → методологии

Естественный путь развития

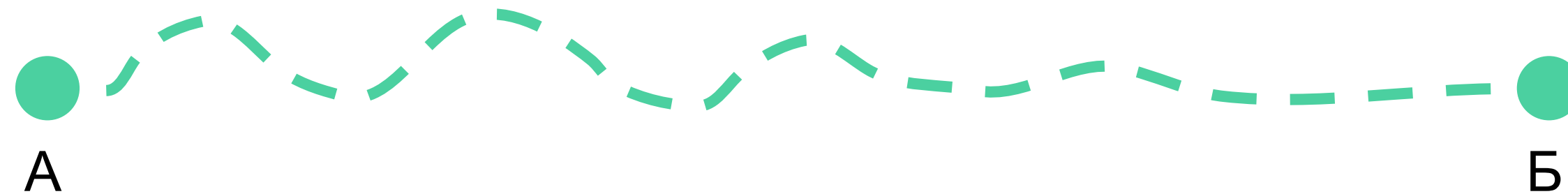


# Модели → методологии

Естественный путь развития



Заимствование практик





# Цикл Шухарта — Деминга



Уолтер Шухарт



Уильям Деминг



# Цикл Шухарта — Деминга

## Главная идея:

Чтобы что-то улучшить, нужно понять, что идёт не так, исправить это, а потом добиться стабилизации процесса — сохранить полученные изменения.

## Этапы цикла Шухарта — Деминга

PDCA

P — plan — планируй

D — do — делай

C — check — проверяй

A — act — действуй

PDSA

P — plan — планируй

D — do — делай

S — study — учись

A — act — действуй



# Бережливое производство

Философия Lean

Виды потерь при разработке ПО

2



# Toyota Production System — от убытков до лидерства на рынке автостроения



Тайити Оно

Выделил **7** видов потерь на производстве:

- потери из-за перепроизводства
- потери времени из-за ожидания
- потери при ненужной транспортировке
- потери из-за лишних этапов обработки
- потери из-за лишних запасов
- потери из-за ненужных перемещений
- потери из-за выпуска дефектной продукции



# Философия Lean

**Создание продукта — это **путь ценности** от поставщика до потребителя.**

Необходимо оптимизировать процесс производства, чтобы создавать ценности, уменьшая издержки.



# Философия Lean

Основные проблемы, с которыми борется философия бережливого производства:

- **муда** — потери
- **мура** — неравномерность выполнения операций
- **мури** — перегрузка рабочих, сотрудников, оборудования при работе с повышенной интенсивностью



# Концепция Lean в разработке ПО



Мэри и Том Поппендик

Адаптировали философию бережливого производства к процессам разработки ПО и выделили следующие принципы:

- Исключение потерь
- Акцент на обучении
- Предельно отсроченное принятие решений
- Предельно быстрая доставка заказчику
- Мотивация команды
- Интегрирование



# Виды потерь при разработке ПО

- Недоделанная работа
- Ненужная функциональность
- Повторное изучение
- Передача
- Переключение между задачами
- Ожидание
- Дефекты





# Agile-манифест

3



# История манифеста

**Agile Manifesto** — документ, созданный 17 независимыми практиками в феврале 2001 года, содержащий описание ценностей и принципов гибкой разработки программного обеспечения.



# Ценности

Люди и взаимодействие

Работающий продукт

Сотрудничество с клиентом

Готовность к изменениям

Важнее

важнее

процессов и инструментов

исчерпывающей документации

согласования условий контракта

следования первоначальному плану

Важно



# 12 принципов Agile

- 1 Наивысший приоритет — 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



# 12 принципов Agile

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



# 12 принципов Agile

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



# Для каких проектов подходит Agile?



# Преимущества Agile

- Гибкость процессов
- Возможность оценить потенциал продукта за счёт быстрого выпуска MVP
- Лёгкое управление приоритетами и низкий порог вхождения для заказчика за счёт применения пользовательских историй, а не спецификаций
- Дельта между ожиданием заказчика и реальностью сокращается с каждым релизом за счёт частых демонстраций, обратной связи и непрерывного улучшения
- Минимизация числа дефектов за счёт непрерывного тестирования





# Недостатки Agile

- Большое количество встреч
- Не все заказчики знают, когда стоит остановиться: постоянная обратная связь может сильно растянуть сроки работы над проектом
- Быстрое устаревание документации или её полное отсутствие
- Необходимость в высококвалифицированных и мотивированных на работу кадрах



# Итоги занятия

1

## **Модель разработки ПО**

описывает, какие стадии жизненного цикла должна проходить разработка, какие процессы должны происходить на каждой стадии, и какие артефакты должны быть получены в результате их завершения

2

## **Методология —**

это набор практик по управлению разработкой программного продукта: правила, техники и принципы, позволяющие выстраивать эффективное взаимодействие и прозрачную коммуникацию между всеми участниками разработки

3

## **Цикл улучшений Шухарта-Деминга:**

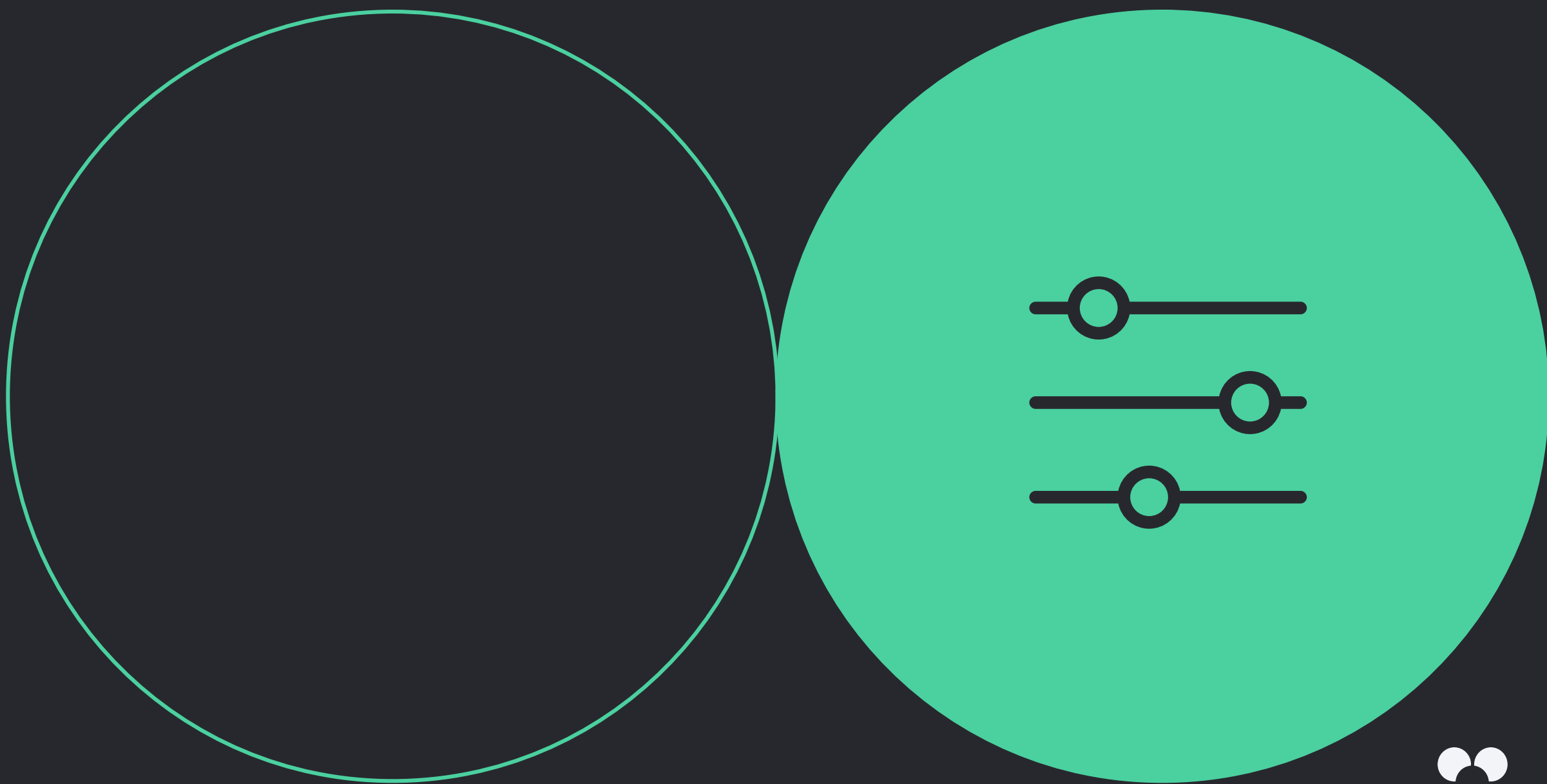
чтобы что-то улучшить, нужно понять, что идёт не так, исправить это, а потом добиться стабилизации процесса — сохранить полученные изменения

4

**Lean-подход и Agile-манифест:** принципы и ценности гибких методологий разработки ПО



# Дополнительные ресурсы



# Что почитать, посмотреть, вдохновиться

1. [История Agile-манифеста](#)
2. [Манифест agile все еще имеет вес?](#)
3. [Feature driven development](#)
4. [Cleanroom Software Engineering](#)



# Гибкие методологии разработки ПО

Часть 1

**Ксения Шипина**  
Системный аналитик

