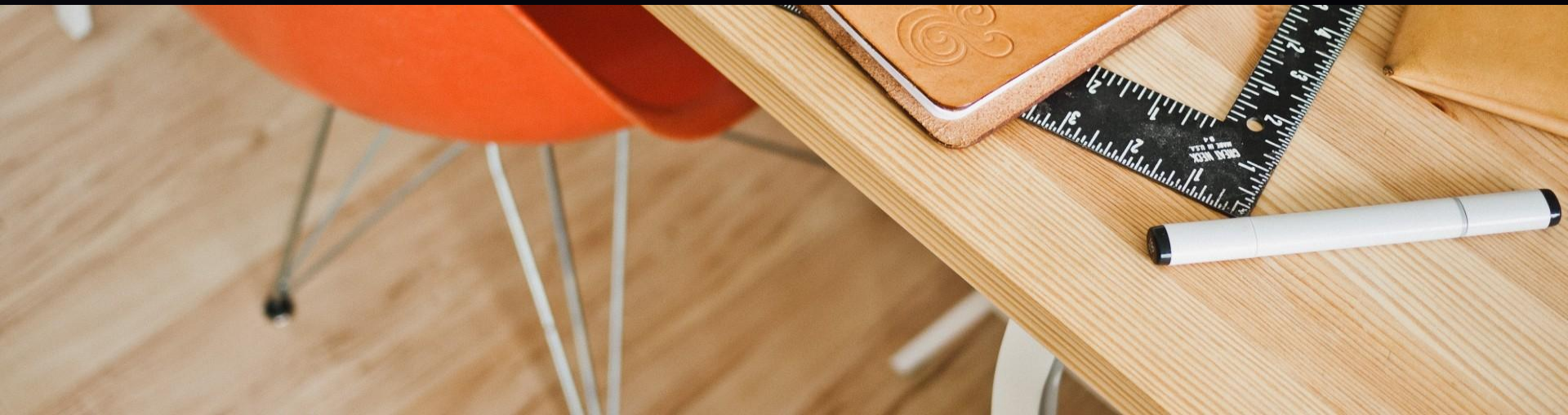


# От Waterfall к Agile





1

Урок 1

**Проектный менеджмент**

2

Урок 2

**От Waterfall к Agile**

3

Урок 3

**Свод знаний  
по управлению проектами**

4

Урок 4

**Требования к компетенциям  
менеджера**

5

Урок 5

**Гибкие методологии**

6

Урок 6

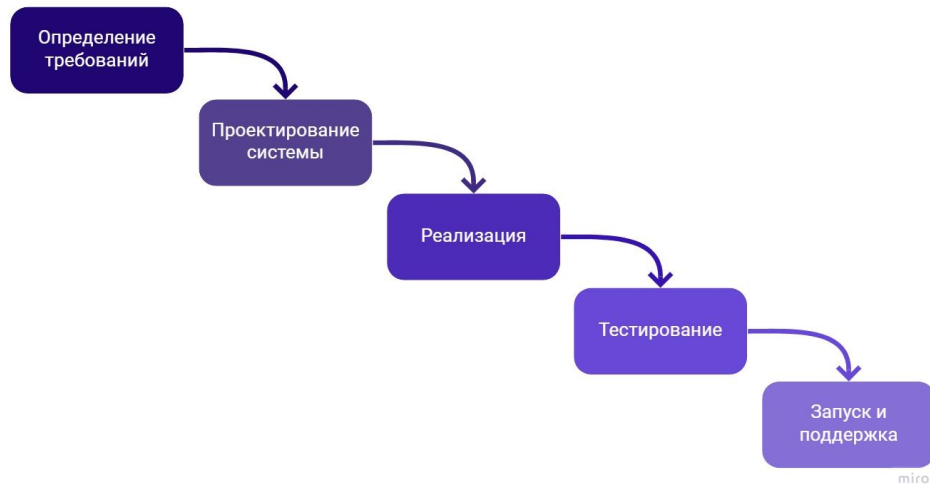
**Сертификации по управлению  
проектами**

# Предиктивный жизненный цикл

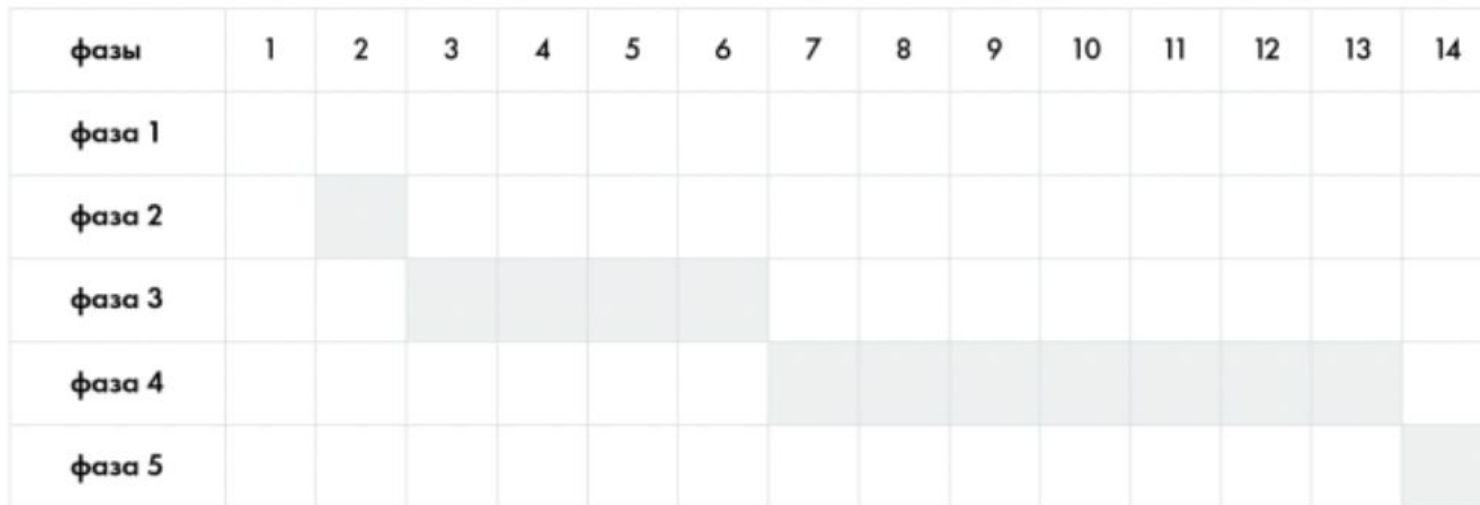
от англ. “predict” - “предсказывать”

Основная особенность предиктивного жизненного цикла состоит в том, что содержание, сроки и стоимость проекта рассчитываются и фиксируются **на самой первой стадии** проекта и далее не изменяются, а на результат проекта можно посмотреть только **в самом конце** во время финальной поставки.

# Каскадная модель



1. Проект проходит фазы строго последовательно, нельзя пропустить фазу
2. Проект переходит в следующую фазу только после **полного** и **успешного** завершения предыдущей фазы
3. Проект не может вернуться в уже завершённую фазу, движение по цепочке идет только в одну сторону





# Плюсы и минусы

## Плюсы:

- **Наличие инструкций и правил по всему процессу.** Работа начинается с подробного анализа требований заказчика и того, как будет реализован проект. Планы, этапы и процессы утверждаются заранее, фиксируются в документах и вопросов не вызывают. Исполнителю нужно просто им следовать.
- **Определенность в сроках и бюджете.** Стоимость продукта и сроки сдачи проекта рассчитаны и утверждены в самом начале и не меняются в процессе.
- **Отсутствие дополнительных трат на коммуникацию в команде.** Даже если придет новый разработчик или тестировщик, понять задачу и приступить к работе получится быстро: для всех процессов есть описанные правила.

## Минусы:

- **Отсутствие гибкости.** Невозможно предусмотреть все проблемы в проекте заранее. Из-за жесткой последовательности этапов недочеты станут известны только в конце проекта, придется делать дополнительные итерации и начинать работу заново, а это новые затраты и лишние рабочие часы.
- **Заказчик не допускается до разработки и тестирования.** Он не может комментировать макеты или прототипы и видит результат только в конце проекта. Если изменились требования или условия, заранее учесть это невозможно.
- **Проблемы всплывают только при тестировании.** Сделать часть работы и сразу протестировать или совместить разработку и тестирование, чтобы найти уязвимости, нельзя. Тестирование начинается после окончания разработки, поэтому часто недостатки обнаруживаются слишком поздно.



## Уинстон Ройс

директор центра информационных технологий авиастроительной компании Lockheed

Доктор Ройс в 1970 году выпустил статью, в которой критиковал каскадную методологию.

Ирония судьбы - эта статья считается моментом появления каскадной методологии.

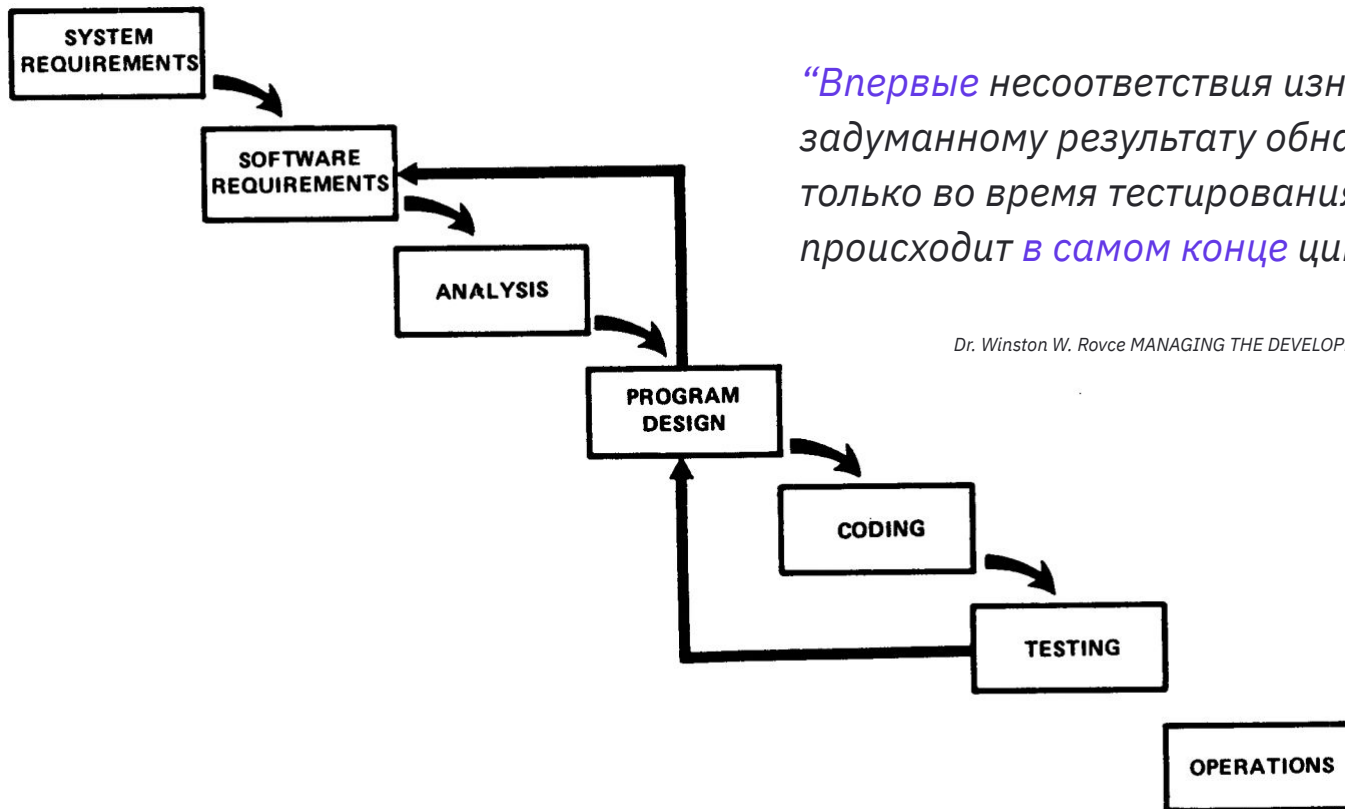
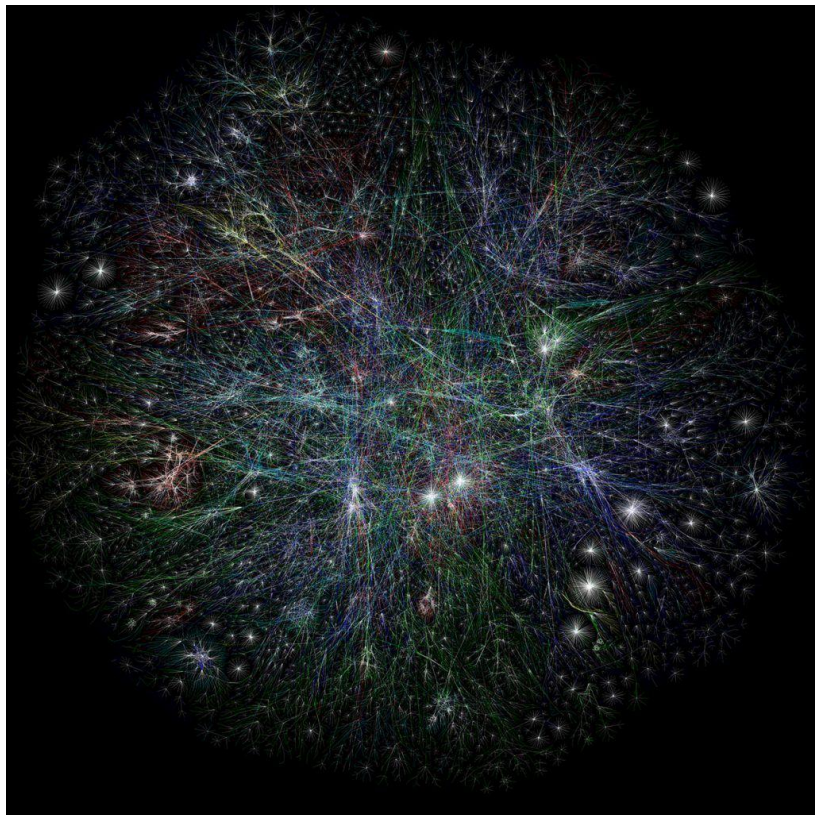


Figure 4. Unfortunately, for the process illustrated, the design iterations are never confined to the successive steps.





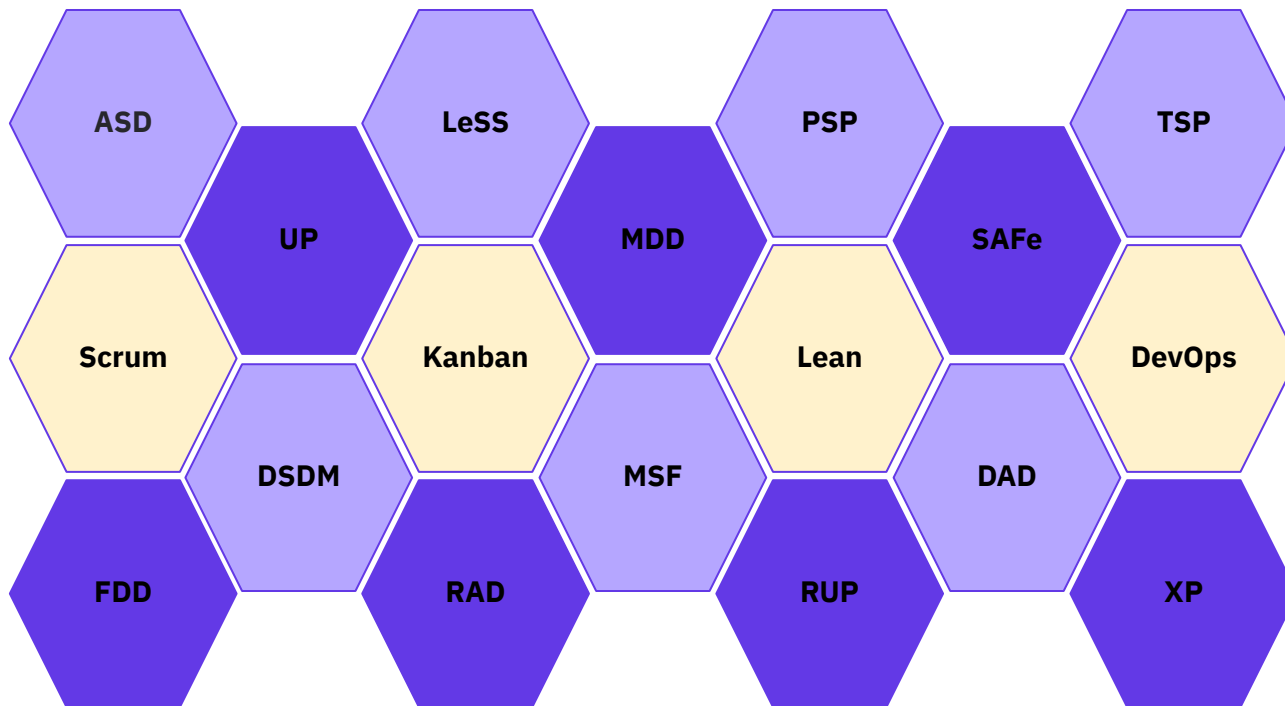
1989  
Появление  
Internet

1995-2001  
Бум доткомов

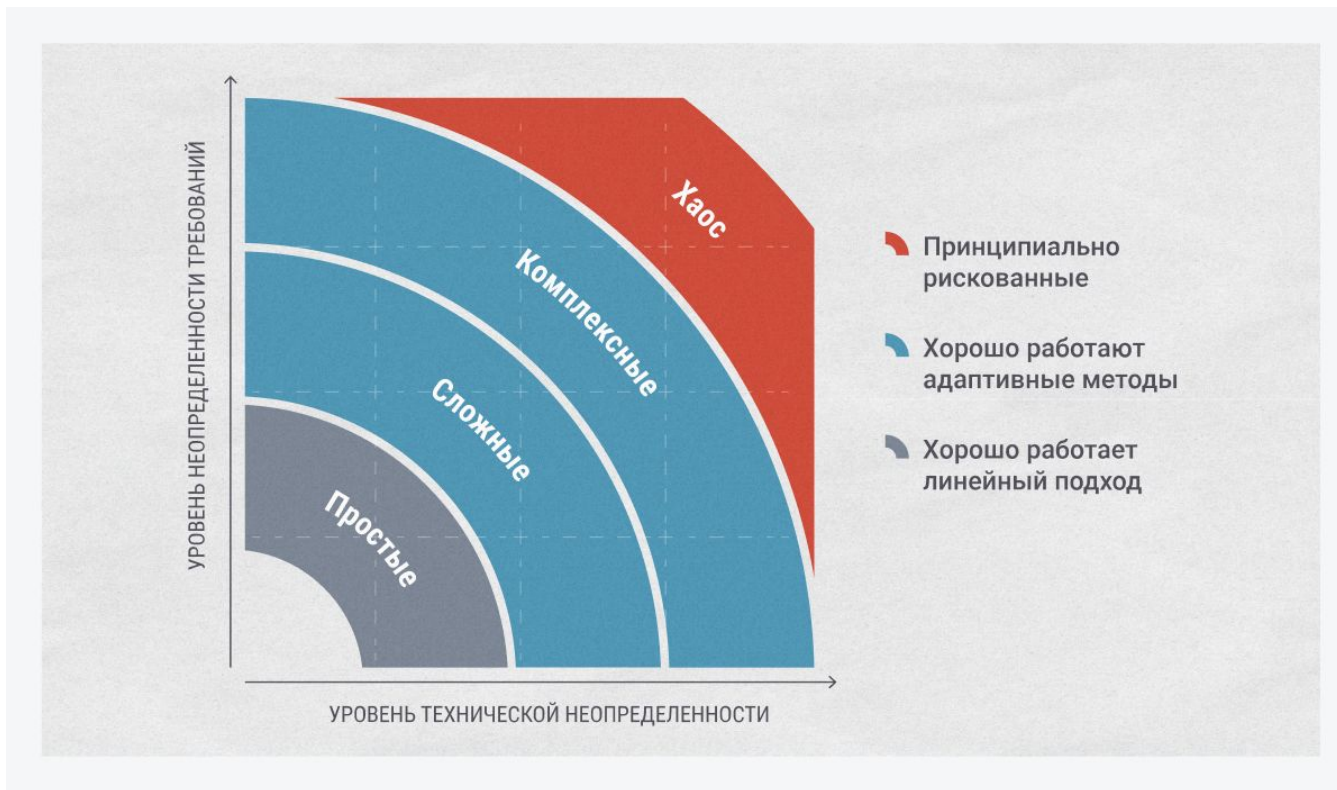
Этот период характерен...

... низкой степенью  
упорядоченности  
и предсказуемости в работе

# Гибкие методы



# Матрица Стейси



# Адаптивный жизненный цикл

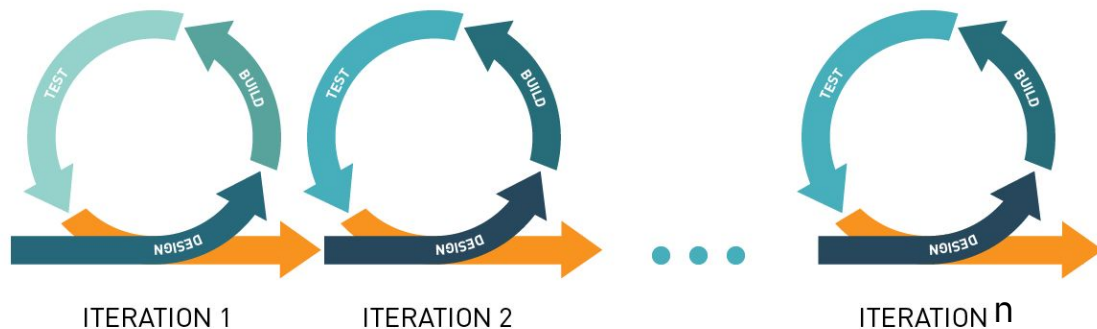
от англ. “adapt” - “приспосабливаться”

Отличительная особенность такого цикла заключается в том, что содержание, а вместе с ним и стоимость, проекта **уточняется постепенно** в ходе реализации, а заказчик оценивает работу по **промежуточным** результатам и оперативно вносит изменения и замечания.

# Итеративная модель

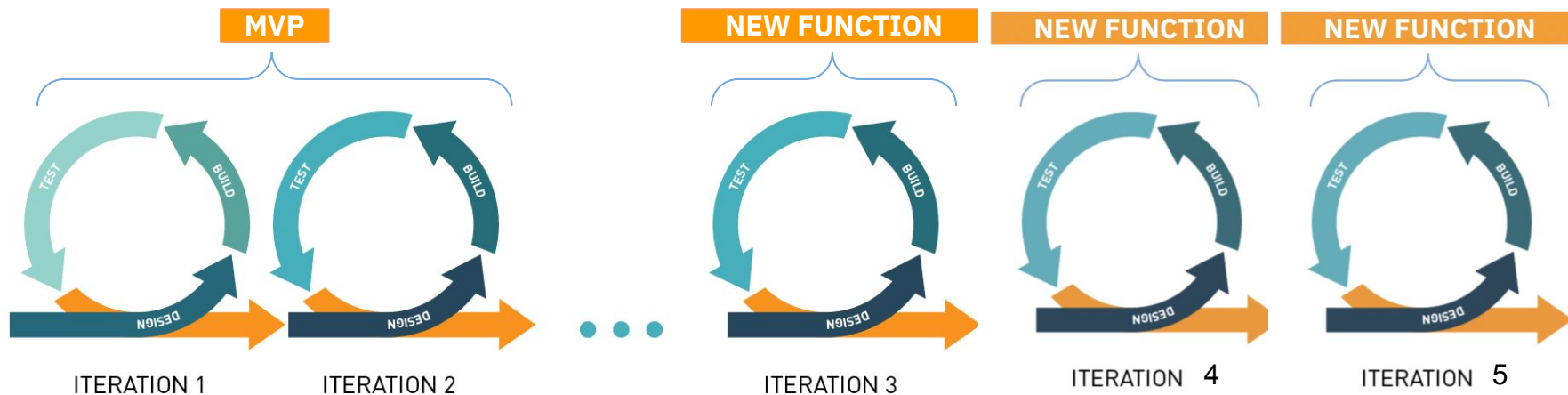
Разработка идет по циклам. В ходе каждой итерации осуществляется планирование, разработка, тестирование и анализ результата.

На основе анализа полученных результатов происходит корректировка последующих этапов работы.



# Инкрементная модель

Разработка идет по циклам так же, как и в итеративной модели. Каждый цикл завершается созданием некоего готового результата, несущего определенную ценность для клиента.



# Гибридная модель

от англ. “hybrid” - “смешанный”

Комбинация из каскадной и итеративной модели для управления разными частями проекта. Используется в сложных проектах, разделенных на подпроекты с разными жизненными циклами.



# Спасибо за внимание

