

Методы формализации требований

Ксения Шипина
Системный аналитик



Что изучим?

- 1 Как написать хорошее требование
- 2 Формализация требований
- 3 Приоритизация требований

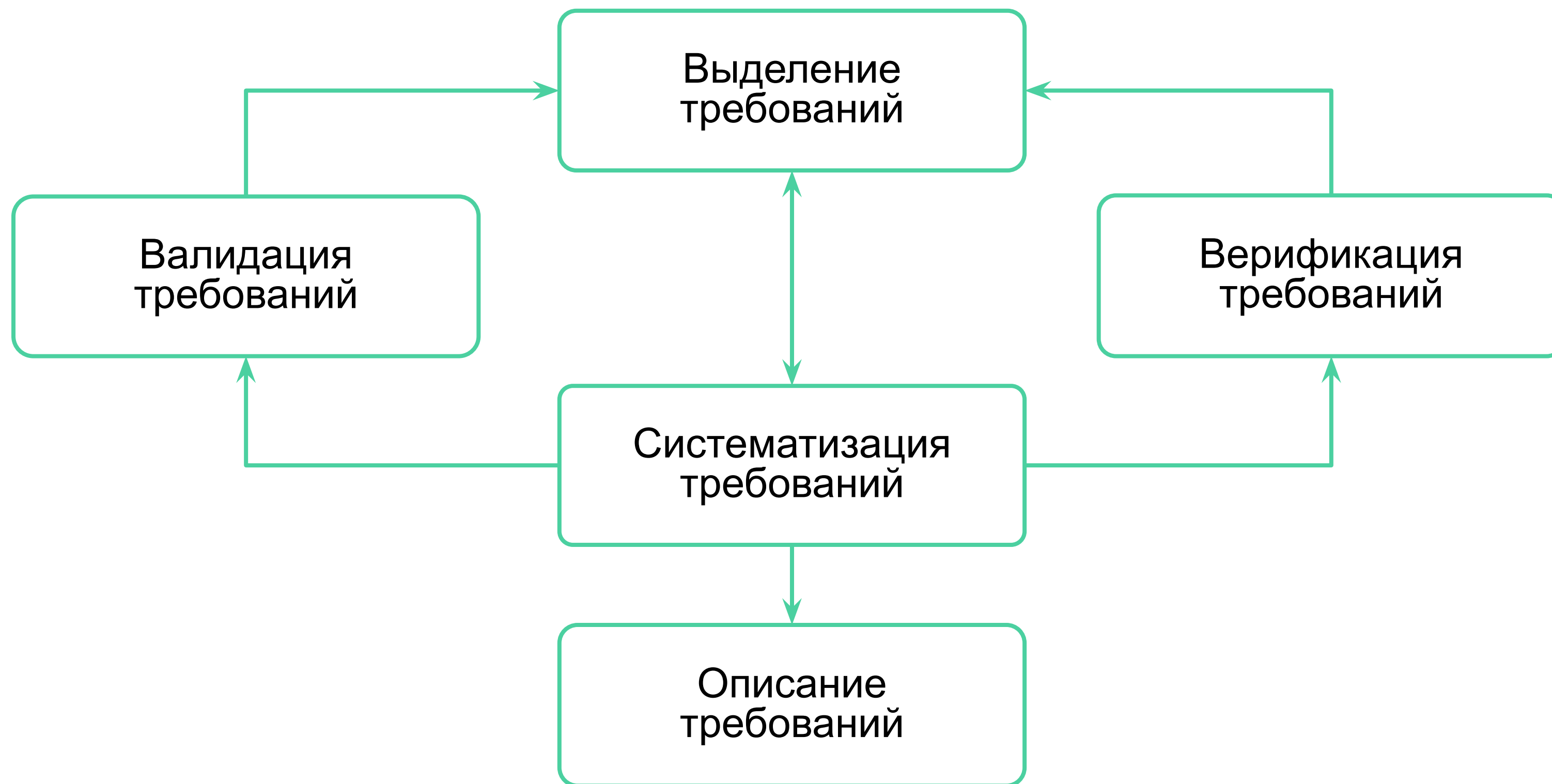


Как написать хорошее требование?

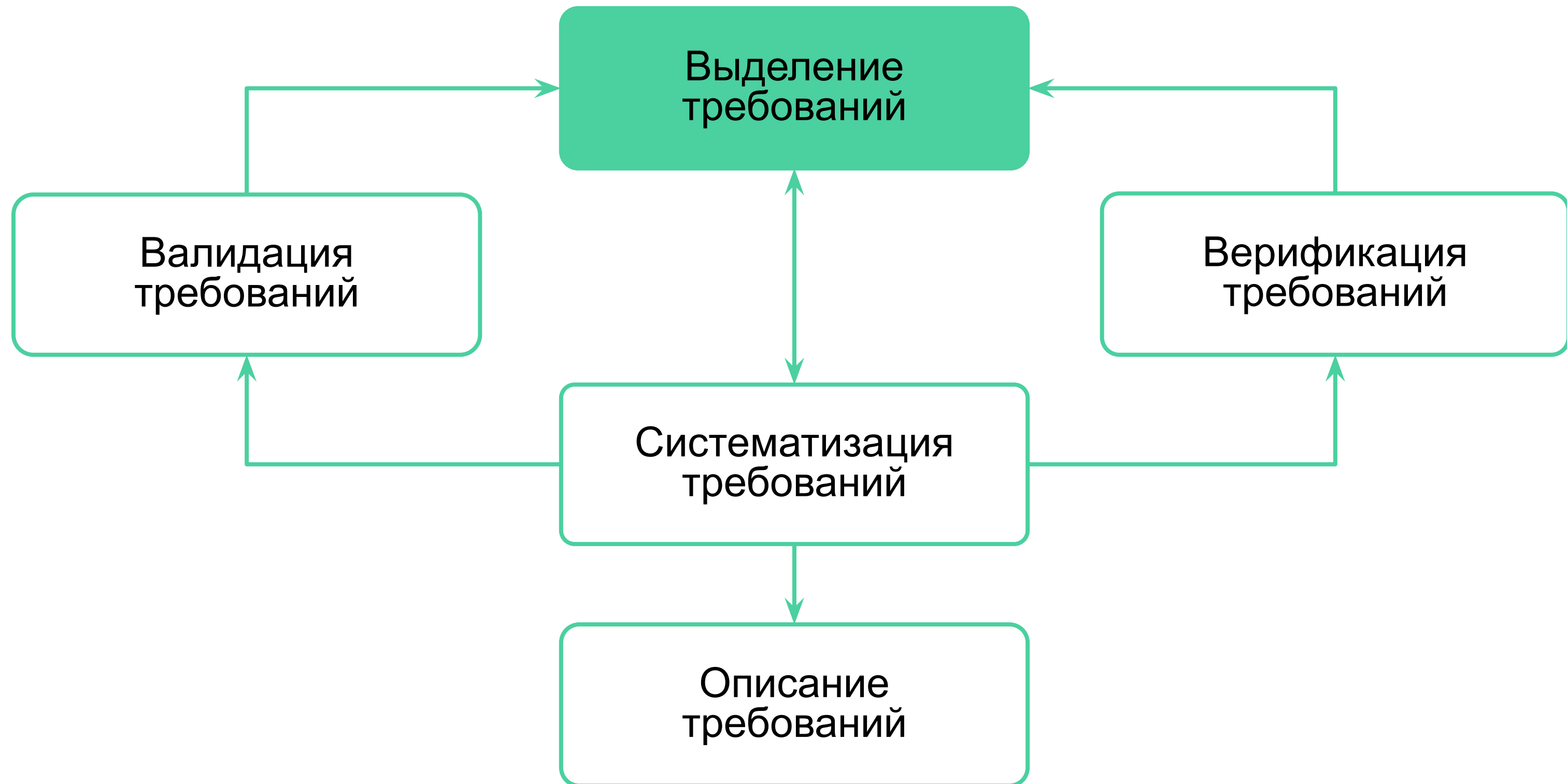
1



Цикл работы с требованиями — Ф-модель



Цикл работы с требованиями — Ф-модель



Выделение требований —

процесс выявления полного
списка требований, их
извлечение из всех
доступных
источников



3 этапа выделения требований



Определение источников требований



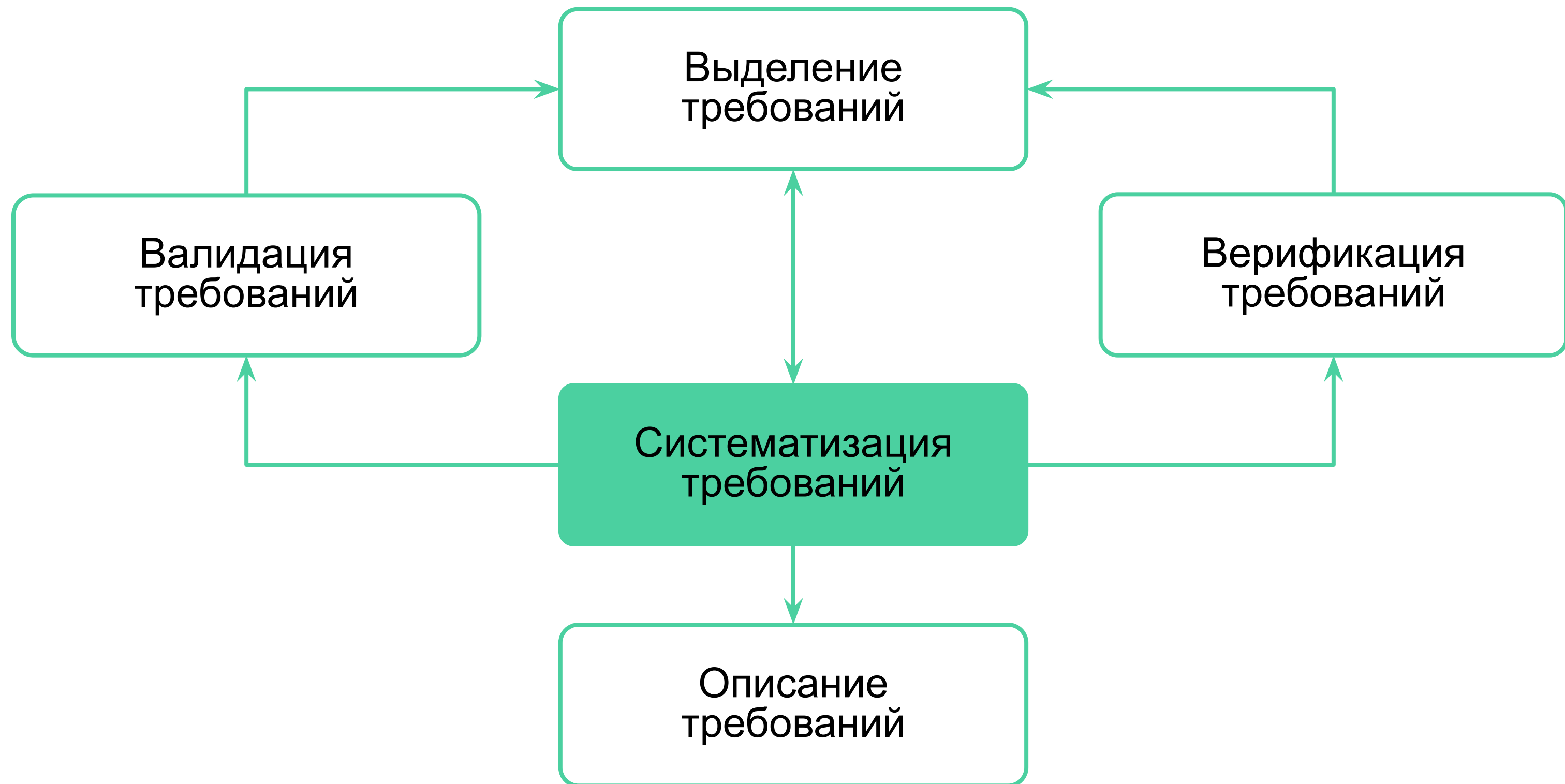
Извлечение требований



Согласование требований



Цикл работы с требованиями — Ф-модель

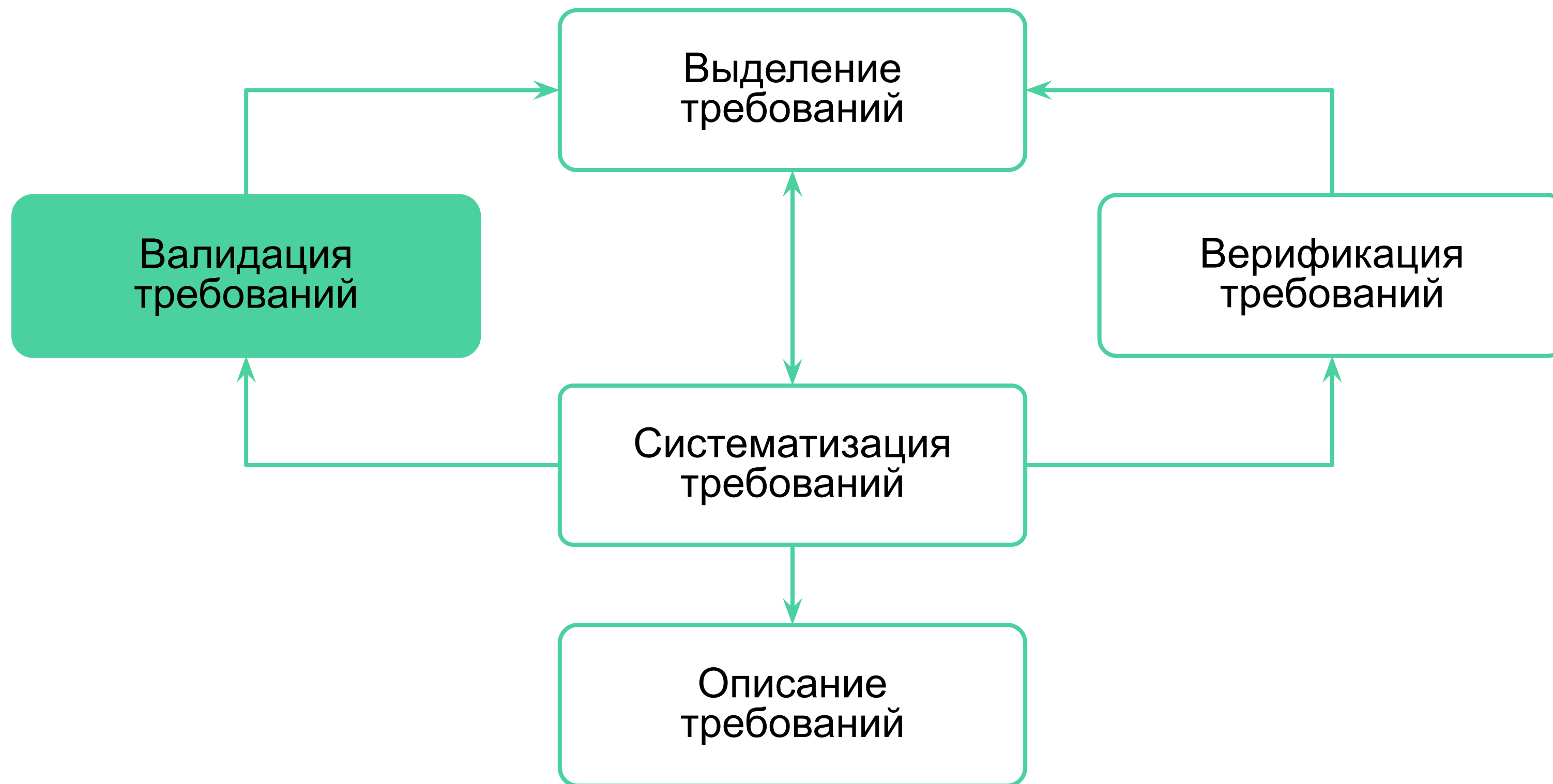


Систематизация требований —

построение целостного
набора, включающего все
выделенные требования,
определение всех
существенных
взаимоотношений
и связей между ними



Цикл работы с требованиями — Ф-модель

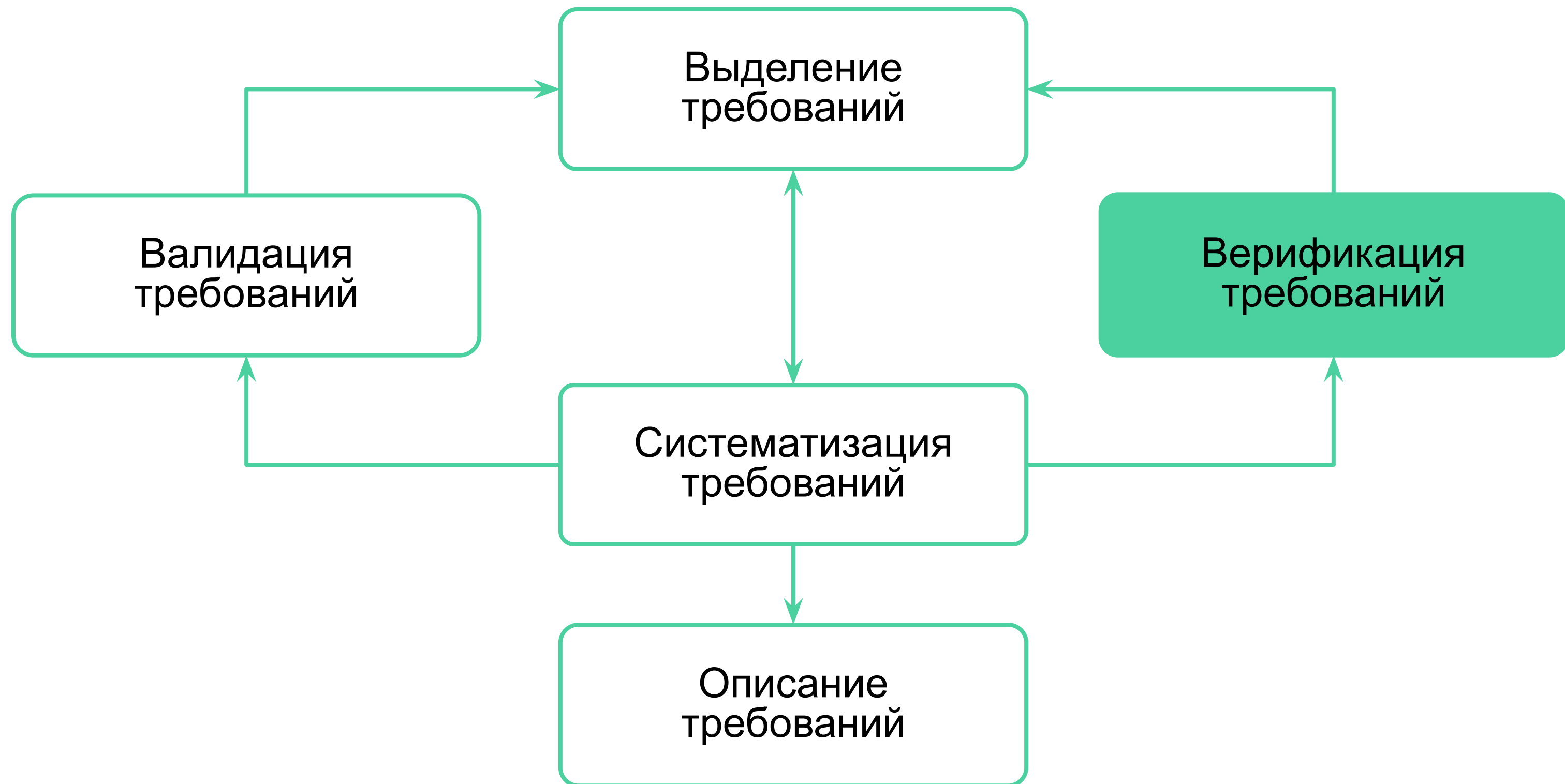


Валидация требований —

взаимодействие
с заинтересованными
лицами проекта для
проверки корректности
требований



Цикл работы с требованиями — Ф-модель



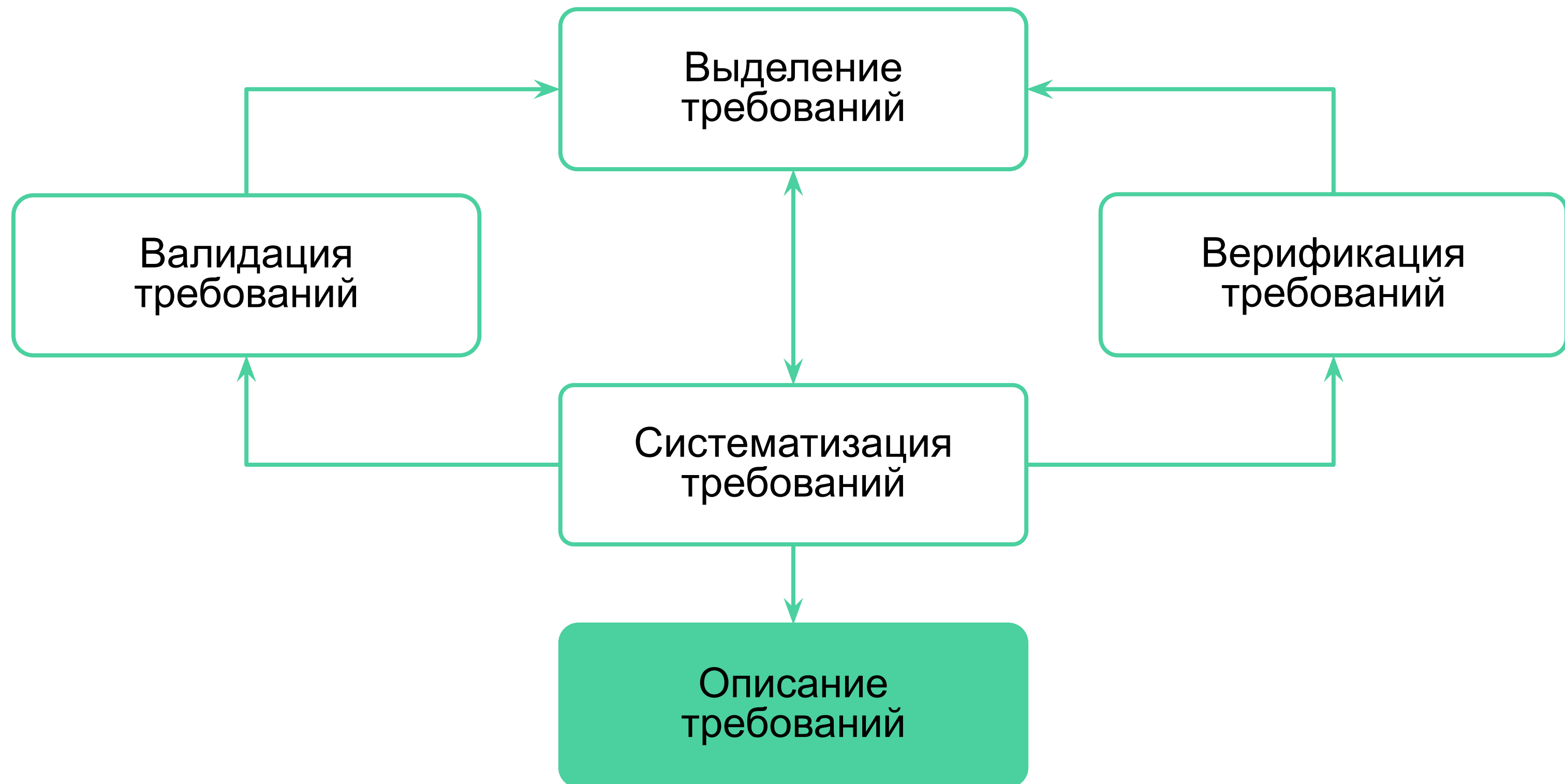
Верификация требований —

проверка внутренних свойств
набора требований: полноты,
ясности, корректности,
согласованности, необходимости
и осуществимости.

Найденные дефекты нужно
устранить в процессе
систематизации



Цикл работы с требованиями — Ф-модель



Описание требований —

процесс, позволяющий
сформулировать различные
представления требований
в виде документов или
моделей



Описание требований

ФТ001 <ФТ — функциональное требование, 001 — порядковый номер>

hO017 <h — высокий приоритет, O — ограничение, 017 — порядковый номер>

Стиль — официально-деловой.

Профессиональные термины нужно пояснять.



Формализация требований

2



Формализация требований —

приведение требований
к формальному виду для
исключения их разночтения
и неоднозначного восприятия



Унифицированные подходы к формализации

- Использование шаблонов
- Варианты использования
- Алгоритм преобразования требований в формальную модель



Использование шаблонов

В формальном подходе обычно описываются либо возможности/ограничения пользователя, либо функции/ограничения системы.

Возможности и Функции:

<Роль> должен иметь возможность <возможность>

<Система> должна <выполняемая функция>

Ограничения:

<Роль> не должен <действие> <документ/регламент, регулирующий ограничение>

<Система> должна <выполняемая функция> каждые <производительность> <единица измерения>



Использование шаблонов: возможности и функции

<Роль> должен иметь возможность <возможность>

Пример:

Администратор системы должен иметь возможность просматривать отзывы о вкусе и качестве блюд, которые были оставлены через форму обратной связи

<Система> должна <выполняемая функция>

Пример:

Система “Школьный портал” должен предоставить пользователю с ролью “Гость” возможность просматривать меню столовой на неделю



Использование шаблонов: возможности и функции

<Роль> должен иметь возможность <возможность>

Пример:

Администратор системы должен иметь возможность просматривать отзывы о вкусе и качестве блюд, которые оставили гости через форму обратной связи.

<Система> должна <выполняемая функция>

Пример:

Система «Школьный портал» должна предоставить пользователю с ролью «Гость» возможность просматривать меню столовой на неделю.



Использование шаблонов: возможности и функции

<Роль> должен иметь возможность <возможность>

Пример:

Администратор системы должен иметь возможность просматривать отзывы о вкусе и качестве блюд, которые оставили гости через форму обратной связи.

<Система> должна <выполняемая функция>

Пример:

Система «Школьный портал» должна предоставить пользователю с ролью «Гость» возможность просматривать меню столовой на неделю.



Использование шаблонов: ограничения

<Роль> не должен <действие> <документ, регламент, регулирующий ограничение>

Пример:

Администратор системы не должен иметь возможность удалять отзывы о качестве блюд, которые оставили гости через форму обратной связи.

<Система> должна <выполняемая функция> каждые <производительность>
<единица измерения>

Пример:

Система «Школьный портал» должна автоматически проверять поступление новых отзывов с формы обратной связи раз в 4 часа.



Использование шаблонов: ограничения

<Роль> не должен <действие> <документ, регламент, регулирующий ограничение>

Пример:

Администратор системы не должен иметь возможность удалять отзывы о качестве блюд, которые оставили гости через форму обратной связи

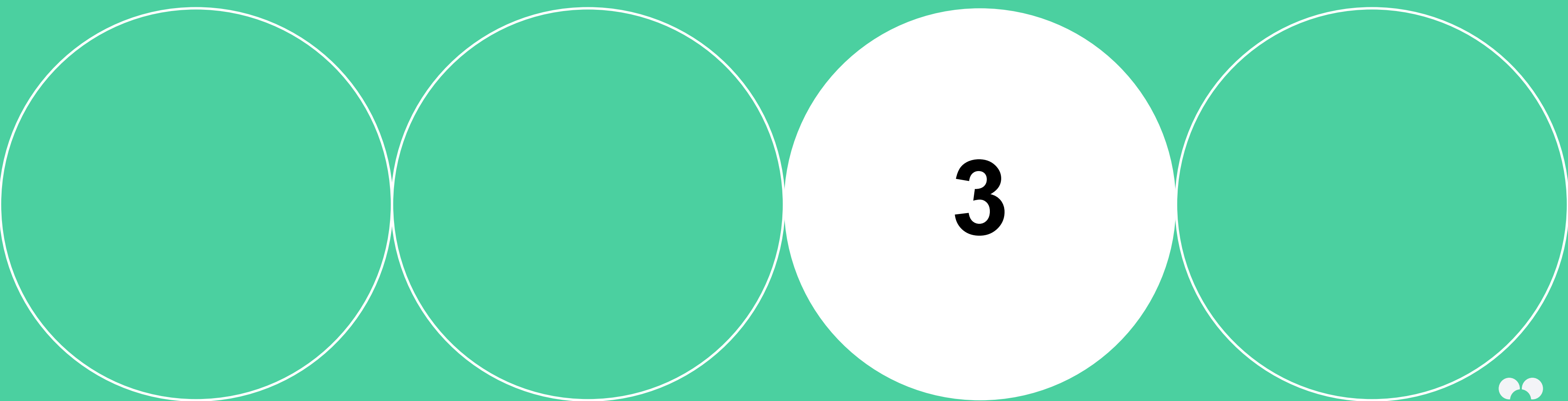
<Система> должна <выполняемая функция> каждые <производительность>
<единица измерения>

Пример:

Система «Школьный портал» должна автоматически проверять поступление новых
отзывов с формы обратной связи раз в 4 часа.



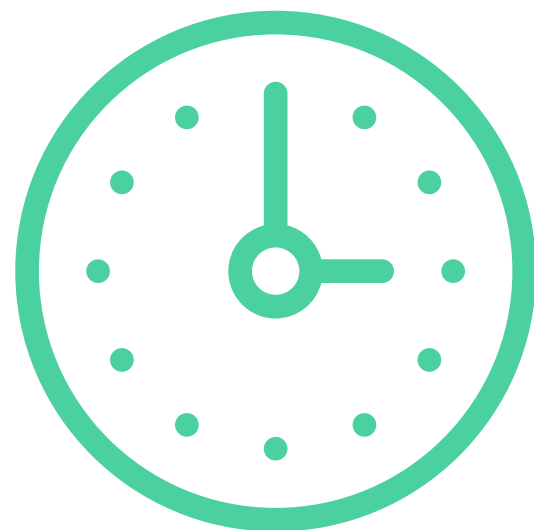
Приоритизация требований



Приоритизация требований



Ответственные —
product owner или
заказчик



Когда — перед стартом
проекта и перед
началом каждого
спринта



Как — принцип
MoSCoW



MoSCoW

Must have — мы обязаны иметь эту функциональность

Should have — нам следует иметь эту функциональность

Could have — было бы здорово иметь эту функциональность

Won't have — не будем это реализовывать в этот раз

или

Wish — «хотелка», хотели бы, но не уверены, что успеем/хватит ли ресурсов



Итоги занятия

1

Ф-модель: цикл работы над требованиями

2

Подходы к формализации требований:

- использование шаблонов
- варианты использования
- алгоритм преобразования требований в формальную модель

3

Шаблоны, которыми можно пользоваться для описания разных видов требований

4

Приоритизация требований и принцип MoSCoW



Практика

4

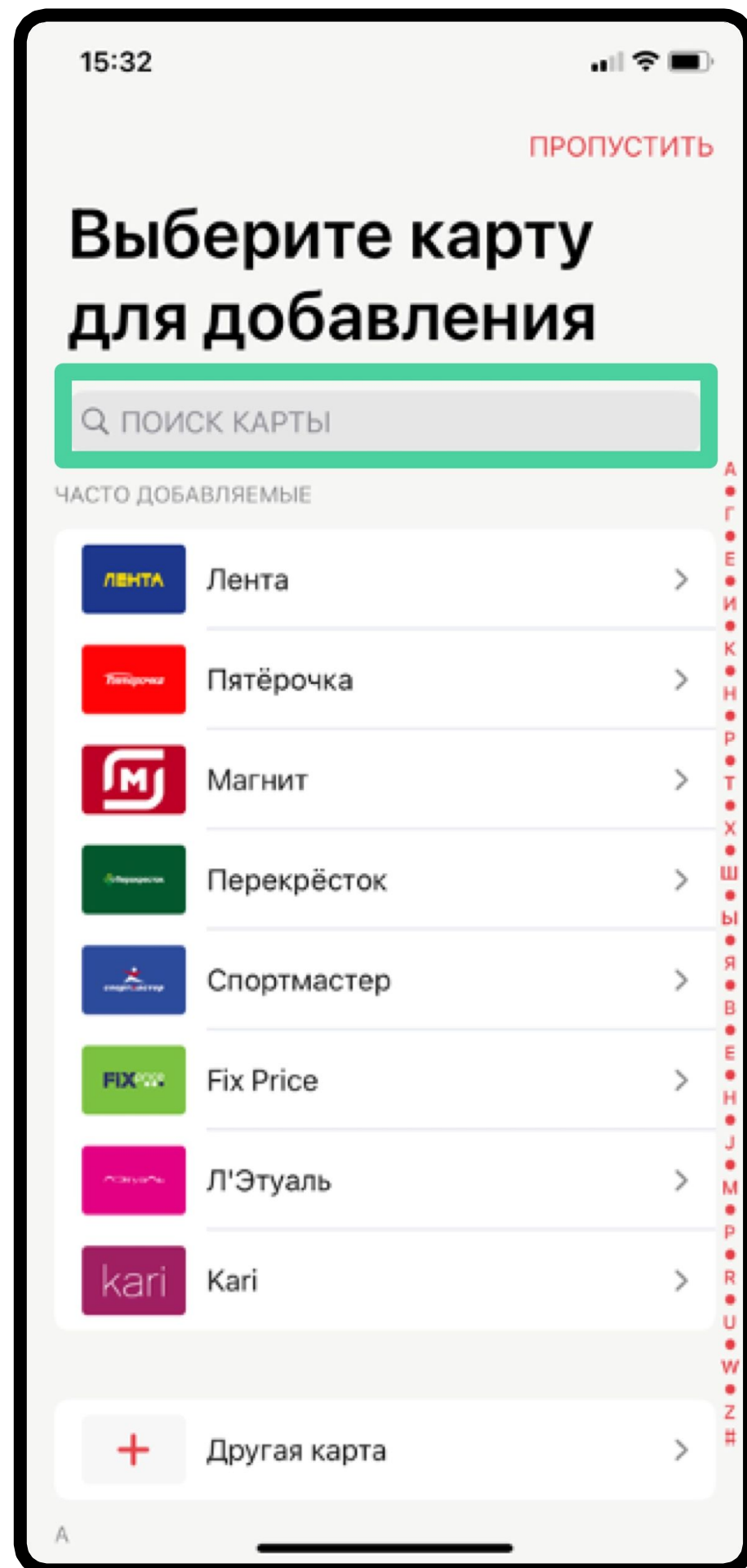


Практика

Задание:

На скриншотах уже реализованного приложения будут выделены функции, к которым нужно будет написать формализованное требование.





Функция: поиск карты по названию магазина

Требование:

Пользователь должен иметь возможность ввести название магазина, карту которого он хочет добавить



Функция: ввести номер карты вручную

Требование:

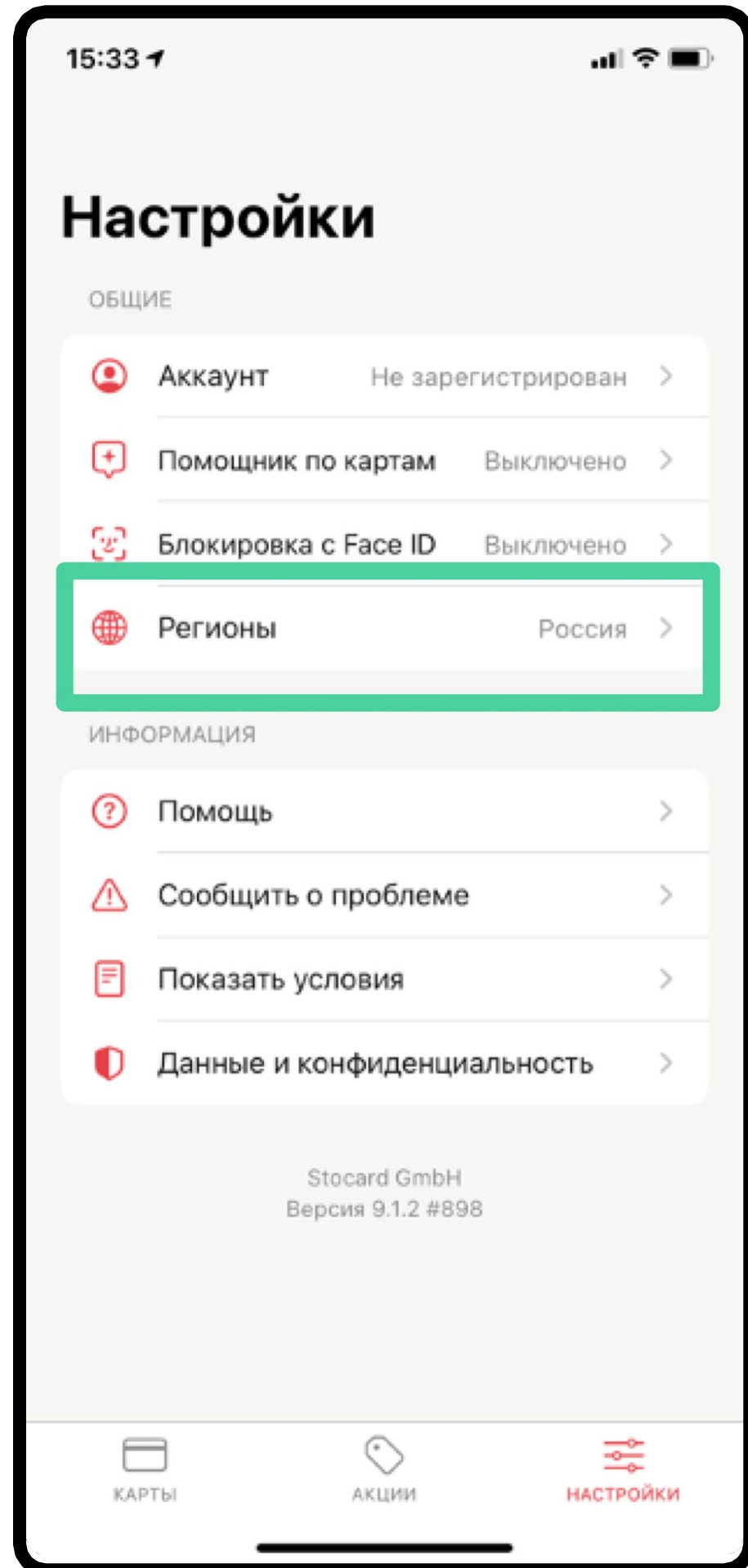
Система должна предоставлять возможность пользователю ввести номер карты вручную в случае, если система не смогла распознать номер карты автоматически



Функция: просмотр акций

Требование:

Пользователь должен иметь возможность просматривать рекомендованные ему акции в разделе «Акции»



Функция: определение региона пользователя

Требование:

Система должна автоматически определять местоположение пользователя для отображения корректного списка магазинов, карты которых можно добавить, и акций

Домашнее задание



Требования к новому продукту

Контекст:

Вы аналитик в команде, которая разрабатывает новое мобильное приложение:

- Если ваша фамилия начинается на **согласную**, и имя начинается на **гласную**, то ваша команда разрабатывает приложение "Будильник"
- Если ваша фамилия начинается на **согласную**, и имя начинается на **согласную**, то ваша команда разрабатывает приложение "Погода"
- Если ваша фамилия начинается на **гласную**, и имя начинается на **согласную** или **гласную**, то ваша команда разрабатывает приложение "Заметки"

Задание:

Вам необходимо формализовать и описать требования к разрабатываемому приложению по классификации Карла Вигерса по характеру



Требования к новому продукту

Условия:

- Необходимо описать не менее 6 требований функционального характера (минимум по 2 на функциональные и пользовательские требования) и не менее 2 требований нефункционального характера
- Требования не должны противоречить друг другу
- Требования должны обладать всеми свойствами требований, которые были рассмотрены на лекции "Требования к ПО"
- Требованиям должны быть присвоены уникальные идентификаторы

Формат сдачи работы:

- Google документ с правом комментирования/редактирования



Цель

Описать формализованные требования к заданному программному продукту

Задание

Составить google-документ с требованиями к продукту

Результат

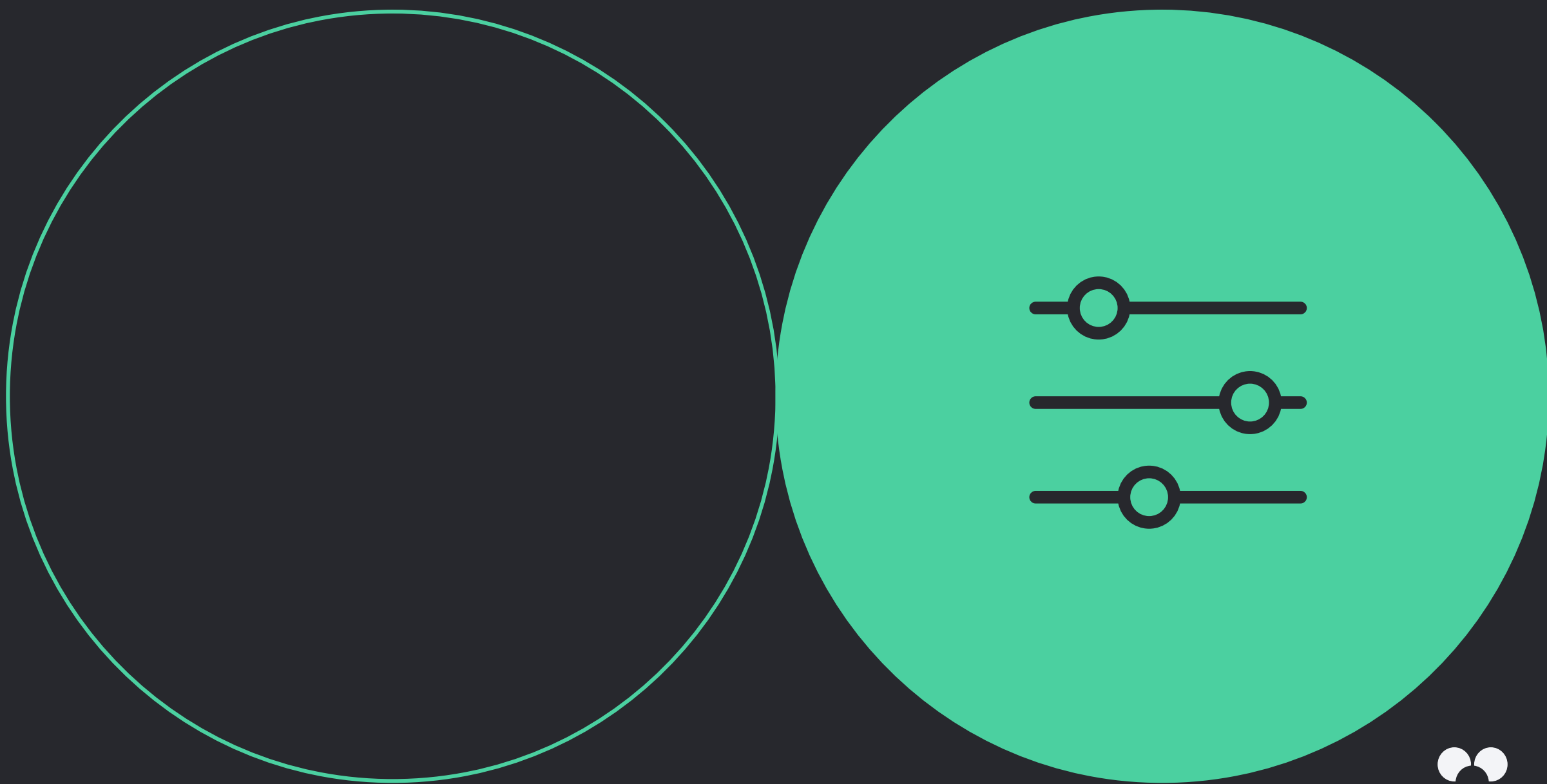
Ссылка на google-документ с возможностью комментировать

Инструменты

Google-документы



Дополнительные ресурсы



Что почитать, посмотреть, вдохновиться

1. Ю.Ю. Липко “Алгоритм формализации требований при разработке информационных систем”
2. В. В. Кулямин, Н. В. Пакулин, О. Л. Петренко, А. А. Сортов, А. В. Хорошилов "Формализация требований на практике"
3. Методическое пособие “Технологии разработки программного обеспечения” раздел “Формализация требований”



Методы формализации требований

Ксения Шипина
Системный аналитик

