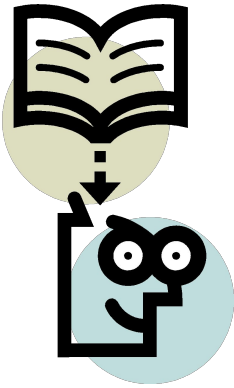




Инженерия программного  
обеспечения  
Дополнительная лекция

**Уровни и типы  
требований**

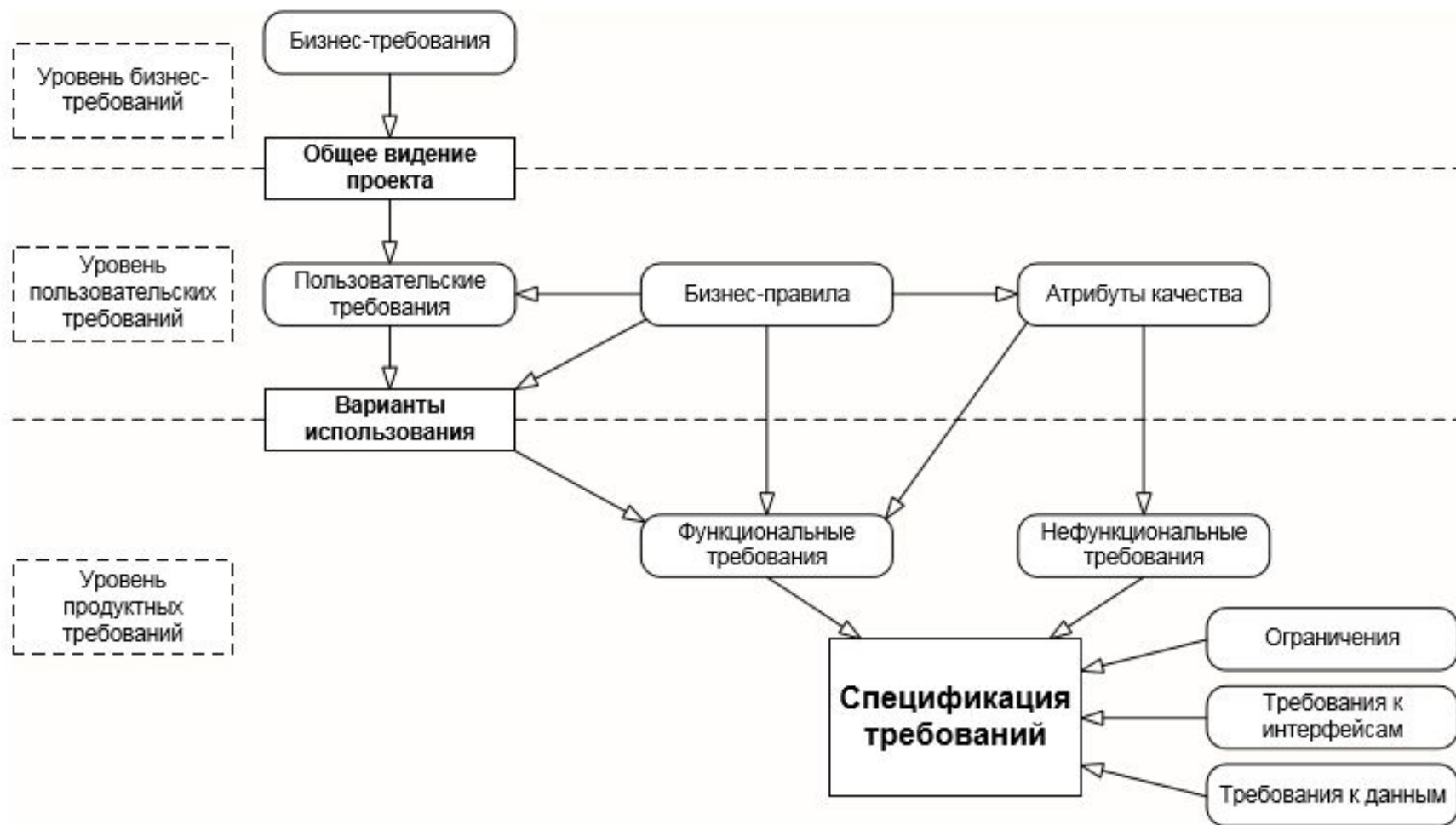
# Уровни и типы требований



**«SOFTWARE REQUIREMENTS ENGINEERING: WHAT, WHY, WHO, WHEN, AND HOW», LINDA WESTFALL**

[[HTTPS://CS.ANU.EDU.AU/COURSES/COMP3530/READINGS/THE\\_WHY\\_WHAT\\_WHO\\_WHEN\\_AND\\_HOW\\_OF\\_SOFTWARE\\_REQUIREMENTS.PDF](https://cs.anu.edu.au/courses/comp3530/readings/the_why_what_who_when_and_how_of_software_requirements.pdf)]

# Уровни и типы требований(2)



# Основные понятия(1)

## **Результат:**

**общее видение (vision and scope)**

— документ, который, как правило, представлен простым текстом и таблицами. Здесь нет детализации поведения системы и иных технических характеристик, но вполне могут быть определены приоритеты решаемых бизнес-задач, риски и т. п.

# Примеры:

- ❑ Нужен инструмент, в реальном времени отображающий наиболее выгодный курс покупки и продажи валюты.
- ❑ Необходимо в два-три раза повысить количество заявок, обрабатываемых одним оператором за смену.
- ❑ Нужно автоматизировать процесс выписки товарно-транспортных накладных на основе договоров.

# Основные понятия(2)

## **Результат:**

**варианты использования (use cases), пользовательские истории (user stories), пользовательские сценарии (user scenarios).**

**пользователя).**

# Примеры:

- ❑ При первом входе пользователя в систему должно отображаться лицензионное соглашение.
- ❑ Администратор должен иметь возможность просматривать список всех пользователей, работающих в данный момент в системе.
- ❑ При первом сохранении новой статьи система должна выдавать запрос на сохранение в виде черновика или публикацию

# Основные понятия(3)

**БИЗНЕС-ПРАВИЛА (BUSINESS RULES)** - описывают особенности принятых в предметной области (и/или непосредственно у заказчика) процессов, ограничений и иных правил. Эти правила могут относиться к бизнес-процессам, правилам работы сотрудников, нюансам работы ПО и т. д.



## Примеры:

- ❑ Никакой документ, просмотренный посетителями сайта хотя бы один раз, не может быть отредактирован или удалён.
- ❑ Публикация статьи возможна только после утверждения главным редактором.
- ❑ Подключение к системе извне офиса запрещено в нерабочее время.

# Основные понятия(4)

## **АТТРИБУТЫ КАЧЕСТВА (QUALITY ATTRIBUTES) –**

**расширяют собой нефункциональные требования и на уровне пользовательских требований могут быть представлены в виде описания ключевых для проекта показателей качества.**

**Атрибутов качества очень много!**

## Примеры:

- ❑ Максимальное время готовности системы к выполнению новой команды после отмены предыдущей не может превышать одну секунду.
- ❑ Внесённые в текст статьи изменения не должны быть утеряны при нарушении соединения между клиентом и сервером.
- ❑ Приложение должно поддерживать добавление произвольного количества неиероглифических языков интерфейса.

# Основные понятия(5)

**ФУНКЦИОНАЛЬНЫЕ  
ТРЕБОВАНИЯ (FUNCTIONAL  
REQUIREMENTS) - описывают  
поведение системы, т. е. её действия  
(вычисления, преобразования,  
проверки, обработку и т. д.)**

## Примеры:

- ❑ В процессе инсталляции приложение должно проверять остаток свободного места на целевом носителе.
- ❑ Система должна автоматически выполнять резервное копирование данных ежедневно в указанный момент времени.
- ❑ Электронный адрес пользователя, вводимый при регистрации, должен быть проверен на соответствие требованиям RF8.

**НЕФУНКЦИОНАЛЬНЫЕ ТРЕБОВАНИЯ (NON-FUNCTIONAL REQUIREMENTS)** описывают свойства системы(удобство использования, безопасность, надёжность, расширяемость и т. д.), которыми она должна обладать при реализации своего поведения.

## Примеры:

- ❑ При одновременной непрерывной работе с системой 1000 пользователей, минимальное время между возникновением сбоев должно быть более или равно 100 часов.
- ❑ Ни при каких условиях общий объём используемой приложением памяти не может превышать 2 ГБ.
- ❑ Размер шрифта для любой надписи на экране должен поддерживать настройку в диапазоне от 5 до 15 пунктов.

## **ОГРАНИЧЕНИЯ (LIMITATIONS, CONSTRAINTS)**

**представляют собой факторы,  
ограничивающие выбор способов и  
средств (в том числе инструментов)  
реализации продукта.**



## Примеры:

- ❑ Все элементы интерфейса должны отображаться без прокрутки при разрешениях экрана от 800x600 до 1920x1080.
- ❑ Не допускается использование Flash при реализации клиентской части приложения.
- ❑ Приложение должно сохранять способность реализовывать функции с уровнем важности «критический» при отсутствии у клиента поддержки JavaScript.

## **ОГРАНИЧЕНИЯ (LIMITATIONS, CONSTRAINTS)**

**представляют собой факторы,  
ограничивающие выбор способов и  
средств (в том числе инструментов)  
реализации продукта.**

## Примеры:

- ❑ Все элементы интерфейса должны отображаться без прокрутки при разрешениях экрана от 800x600 до 1920x1080.
- ❑ Не допускается использование Flash при реализации клиентской части приложения.
- ❑ Приложение должно сохранять способность реализовывать функции с уровнем важности «критический» при отсутствии у клиента поддержки JavaScript.

# Основные понятия(9)

**ТРЕБОВАНИЯ К ИНТЕРФЕЙСАМ  
(EXTERNAL INTERFACES REQUIREMENTS)  
описывают особенности взаимодействия  
разрабатываемой системы с другими  
системами и операционной средой.**

## Примеры:

- ❑ Обмен данными между клиентской и серверной частями приложения при осуществлении фоновых AJAX-запросов должен быть реализован в формате JSON.
- ❑ Протоколирование событий должно вестись в журнале событий операционной системы.

# Основные понятия(10)

## ТРЕБОВАНИЯ К ДАННЫМ (DATA REQUIREMENTS)

описывают структуры данных (и сами данные), являющиеся неотъемлемой частью разрабатываемой системы. Часто сюда относят описание базы данных и особенностей её использования.

## Примеры:

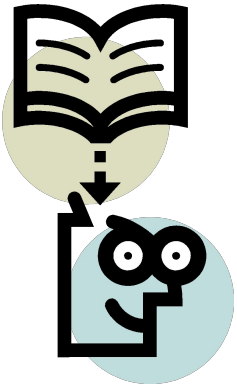
- ❑ Все данные системы, за исключением пользовательских документов, должны храниться в БД под управлением СУБД MySQL, пользовательские документы должны храниться в БД под управлением СУБД MongoDB.
- ❑ Для ускорения операций поиска по тексту статей и обзоров должны быть предусмотрены полнотекстовые индексы на соответствующих полях таблиц.

# Спецификация требований Software requirements specification

**объединяет в себе описание всех требований уровня продукта и может представлять собой весьма объёмный документ.**



# Принципы создания, организации и использования набора требований



**КАРЛА ВИГЕРСА**

**«РАЗРАБОТКА ТРЕБОВАНИЙ К ПРОГРАММНОМУ ОБЕСПЕЧЕНИЮ»**

**(«SOFTWARE REQUIREMENTS (3RD EDITION) (DEVELOPER BEST PRACTICES)», KARL WIEGERS, JOY BEATTY).**

# Вопросы

