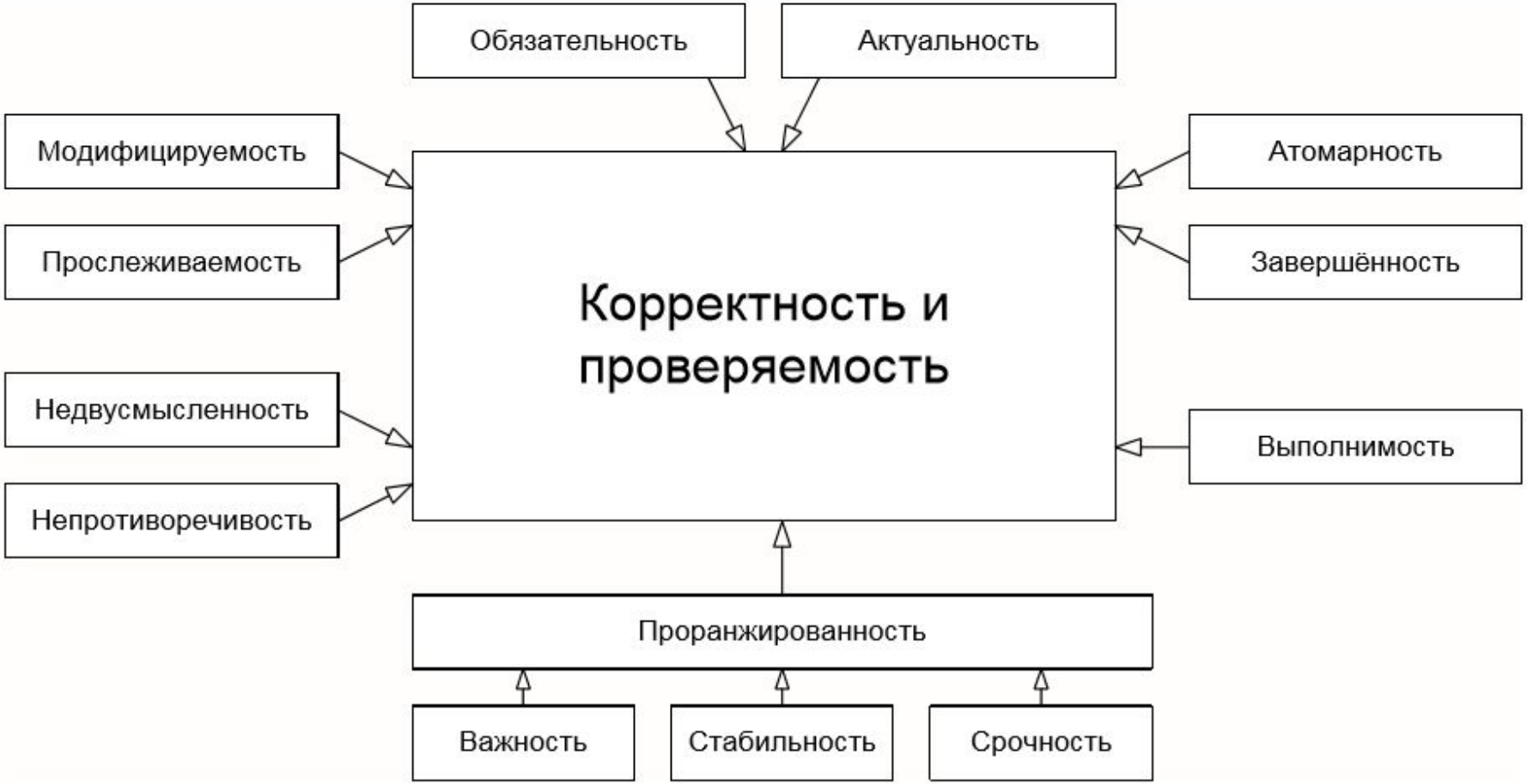
Кафедра компьютерных систем и сетей

Инженерия программного обеспечения

Дополнительная лекция

Свойства требований

Свойства качественного требования



Основные понятия(1)

ЗАВЕРШЁННОСТЬ (COMPLETENESS) Требование является полным и законченным с точки зрения представления в нём всей необходимой информации, ничто не пропущено по соображениям «это и так всем понятно».

Типичные проблемы с завершённостью:

- Отсутствуют нефункциональные составляющие требования или ссылки на соответствующие нефункциональные требования

  например: *«пароли должны храниться в зашифрованном виде»*
- Указана лишь часть некоторого перечисления например: «экспорт осуществляется в форматы PDF, PNG и т. д.»
- Приведённые ссылки неоднозначны например: «см. выше»

Основные понятия(2)

АТОМАРНОСТЬ, ЕДИНИЧНОСТЬ (ATOMICITY)

Требование является атомарным, если его нельзя разбить на отдельные требования <u>без потери</u> завершённости и оно описывает <u>одну и только одну</u> ситуацию.

Типичные проблемы с атомарностью:

- В одном требовании, фактически, содержится несколько независимых например: *«кнопка “Restart” не должна отображаться при остановленном сервисе, окно “Log” должно вмещать не менее 20-ти записей о последних действиях пользователя»*
- Требование допускает разночтение в силу грамматических особенностей языка (

  например: *«если пользователь подтверждает заказ и редактирует заказ или откладывает заказ, должен выдаваться запрос на оплату»*

  ! влечёт за собой возникновение противоречивости.
- В одном требовании объединено описание нескольких независимых ситуаций

  например: *«когда пользователь входит в систему, ему должно отображаться приветствие; когда пользователь вошёл в систему, должно отображаться имя пользователя; когда пользователь выходит из системы, должно отображаться прощание»*

Основные понятия(3)

НЕПРОТИВОРЕЧИВОСТЬ, ПОСЛЕДОВАТЕЛЬНОСТЬ (CONSISTENCY).

Требование не должно содержать внутренних противоречий и противоречий <u>другим</u> требованиям и документам.

Типичные проблемы с непротиворечивостью:

- Противоречия внутри одного требования

  например: *«после успешного входа в систему пользователя, не имеющего права входить в систему…»*
- Противоречия между двумя и более требованиями, между таблицей и текстом, рисунком и текстом, требованием и прототипом и т. д.

  например: *«712.a Кнопка “Close” всегда должна быть красной» и «36452.x Кнопка “Close” всегда должна быть синей»*
- Использование неверной терминологии или использование разных терминов для обозначения одного и того же объекта или явления

  например: *«в случае, если разрешение окна составляет менее 800x600…»*

Основные понятия(4)

НЕДВУСМЫСЛЕННОСТЬ (UNAMBIGUOUSNESS, CLEARNESS).

Требование описано без использования жаргона, неочевидных аббревиатур и расплывчатых формулировок и допускает только <u>однозначное</u> объективное понимание.

Требование атомарно в плане <u>невозможности различной трактовки</u> сочетания отдельных фраз.

Типичные проблемы с недвусмысленностью:

- Использование терминов или фраз, допускающих субъективное толкование (

  например: *«приложение должно поддерживать передачу больших объёмов данных» — насколько «больших»?*

«В случае необходимости оптимизации передачи больших файлов система должна эффективно использовать минимум оперативной памяти, если это возможно».

Типичные проблемы с недвусмысленностью:

- Использование неочевидных или двусмысленных аббревиатур без расшифровки

  например: *«доступ к ФС осуществляется посредством системы прозрачного шифрования»* и *«ФС предоставляет возможность фиксировать сообщения в их текущем состоянии с хранением истории всех изменений»*

- Формулировка требований из соображений, что нечто должно быть всем очевидно

  например: «*Система конвертирует входной файл из формата PDF в выходной файл формата PNG»* — и при этом автор считает совершенно очевидным, что имена файлов система получает из командной строки, а многостраничный PDF конвертируется в несколько PNG-файлов, к именам которых добавляется «page-1», «page-2» и т. д.). Эта проблема перекликается с нарушением корректности.

Основные понятия(5)

ВЫПОЛНИМОСТЬ (FEASIBILITY).

Требование технологически выполнимо и может быть реализовано в рамках бюджета и сроков разработки проекта.

Типичные проблемы с выполнимостью:

- Так называемое «озолочение» (gold plating) — требования, которые крайне долго и/или дорого реализуются и при этом практически бесполезны для конечных пользователей например: «*настройка параметров для подключения к базе данных должна поддерживать распознавание символов из жестов, полученных с устройств трёхмерного ввода*»
- Технически нереализуемые на современном уровне развития технологий требования

  например: *«анализ договоров должен выполняться с применением искусственного интеллекта, который будет выносить однозначное корректное заключение о степени выгоды от заключения договора»*
- В принципе нереализуемые требования

  например: *«система поиска должна заранее предусматривать все возможные варианты поисковых запросов и кэшировать их результаты»*

Основные понятия(6)

ОБЯЗАТЕЛЬНОСТЬ, НУЖНОСТЬ (OBLIGATION) И АКТУАЛЬНОСТЬ (UP-TO-DATE).

Если требование не является обязательным к реализации, оно должно быть просто исключено из набора требований.

Если требование нужное, но «не очень важное», для указания этого факта используется указание приоритета.

Также исключены (или переработаны) должны быть требования, утратившие актуальность.

Типичные проблемы с обязательностью и актуальностью:

- Требование было добавлено «на всякий случай», хотя реальной потребности в нём не было и нет.
- Требованию выставлены неверные значения приоритета по критериям важности и/или срочности.
- Требование устарело, но не было переработано или удалено.

Основные понятия(7)

ПРОСЛЕЖИВАЕМОСТЬ (TRACEABILITY)

- **вертикальная (vertical traceability)**
- **горизонтальная (horizontal traceability)**

Вертикальная позволяет соотносить между собой требования на различных уровнях требований.

Горизонтальная позволяет соотносить требование с тест-планом, тест-кейсами, архитектурными решениями и т. д.

Типичные проблемы с обязательностью и актуальностью:

- Требования не пронумерованы, не структурированы, не имеют оглавления, не имеют работающих перекрёстных ссылок.
- При разработке требований не были использованы инструменты и техники управления требованиями.
- Набор требований неполный, носит обрывочный характер с явными «пробелами».

Оновные понятия(8)

МОДИФИЦИРУЕМОСТЬ (MODIFI ABILITY)

Это свойство характеризует простоту внесения изменений в отдельные требования и в набор требований.

Типичные проблемы с модифицируемостью:

- Требования неатомарны и непрослеживаемы, а потому их изменение с высокой вероятностью порождает противоречивость. =))
- Требования изначально противоречивы. В такой ситуации внесение изменений (не связанных с устранением противоречивости) только усугубляет ситуацию, увеличивая противоречивость и снижая прослеживаемость.
- Требования представлены в неудобной для обработки форме

  например, не использованы инструменты управления требованиями, и в итоге команде приходится работать с десятками огромных текстовых документов

Оновные понятия(9)

ПРОРАНЖИРОВАННОСТЬ ПО ВАЖНОСТИ, СТАБИЛЬНОСТИ, СРОЧНОСТИ

Срочность определяет распределение во времени усилий проектной команды по реализации того или иного требования.

изменения реализации требования.

Типичные проблемы:

- Проблемы с проранжированностью по важности повышают риск неверного распределения усилий проектной команды, направления усилий на второстепенные задачи и конечного провала проекта из-за неспособности продукта выполнять ключевые задачи с соблюдением ключевых условий.
- Проблемы с проранжированностью по стабильности повышают риск выполнения бессмысленной работы по совершенствованию, реализации и тестированию требований, которые в самое ближайшее время могут претерпеть кардинальные изменения (вплоть до полной утраты актуальности).
- Проблемы с проранжированностью по срочности повышают риск нарушения желаемой заказчиком последовательности реализации функциональности и ввода этой функциональности в эксплуатацию.

Оновные понятия(9)

КОРРЕКТНОСТЬ (CORRECTNESS) И ПРОВЕРЯЕМОСТЬ (VERIFI ABILITY) Проверяемость подразумевает возможность создания объективного тест-кейса (тест-кейсов), однозначно показывающего, что требование реализовано верно и поведение приложения в точности соответствует требованию

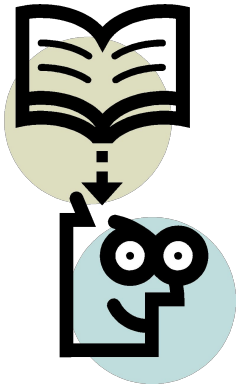
Типичные проблемы:

- опечатки (особенно **опасны опечатки в аббревиатурах**, превращающие одну осмысленную аббревиатуру в другую также осмысленную, но не имеющую отношения к некоему контексту) !

  !опечатки крайне сложно заметить

- наличие неаргументированных требований к дизайну и архитектуре;
- плохое оформление текста и сопутствующей графической информации, грамматические, пунктуационные и иные ошибки в тексте;
- неверный уровень детализации (например, слишком глубокая детализация требования на уровне бизнес-требований или недостаточная детализация на уровне требований к продукту);
- <u>требования к пользователю</u>, а не к приложению

  например: *«пользователь должен быть в состоянии отправить сообщение» =))))*



«Writing Good Requirements — The Big Ten Rules»

Вопросы