Тестирование документации и требований

(ЧАСТЬ 1)

Главное:

Что такое «требование»

Важность требований

Источники и пути выявления требований

Уровни и типы требований

Свойства качественных требований

Техники тестирования требований

Пример анализа и тестирования требований

Типичные ошибки при анализе и тестировании требований

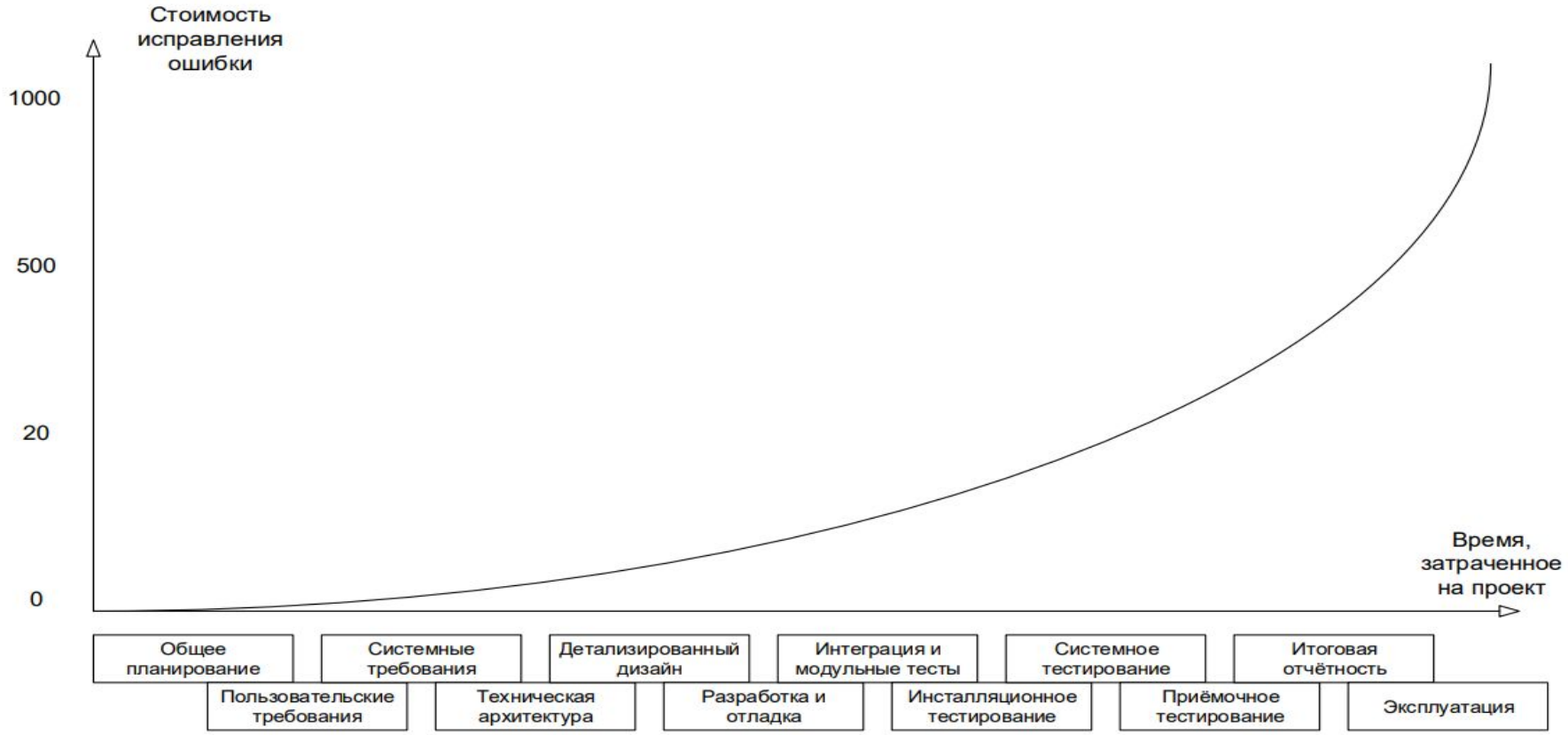
Что такое «требование»

Требование (requirement) — описание того, какие функции и с соблюдением каких условий должно выполнять приложение в процессе решения полезной для **пользователя** задачи.

«What is documentation testing in software testing».

[http://istqbexamcertification.com/what-is-documentation-testing/]

Важность требований



Стоимость исправления ошибки в зависимости от момента её обнаружения

Важность требований

Требования:

Позволяют понять, что и с соблюдением каких условий система должна делать.

Предоставляют возможность оценить масштаб изменений и управлять изменениями.

Являются основой для формирования плана проекта (в том числе плана тестирования).

Помогают предотвращать или разрешать конфликтные ситуации.

Упрощают расстановку приоритетов в наборе задач.

Позволяют объективно оценить степень прогресса в разработке проекта.

Покупки в магазине

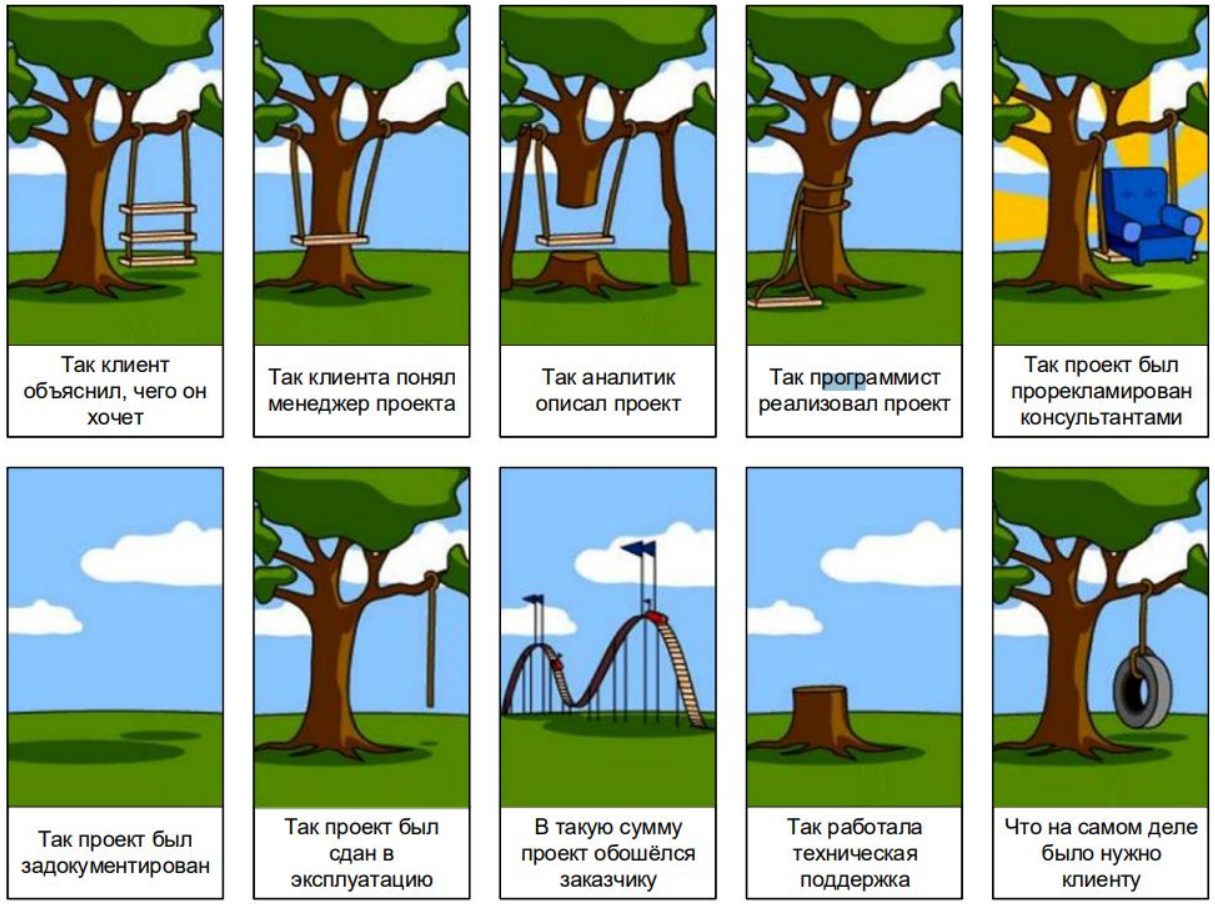
Задача :

Планирование покупок в магазине

Вопрос:

представьте, что ваш с друзьями бюджет ограничен, и в списке требований появляются приоритеты (что-то купить надо обязательно, что-то, если останутся деньги, и т.п.). Как это повлияет на риски, связанные с ошибками в списке?

Важность требований



Виды документации

Продуктная документация (product documentation, development documentation) используется проектной командой во время разработки и поддержки продукта.

Проектная документация (project documentation) включает в себя как продуктную документацию, так и некоторые дополнительные виды документации и используется не только на стадии разработки, но и на более ранних и поздних стадиях (например, на стадии внедрения и эксплуатации).

Product documentation

План проекта (project management plan54) и в том числе тестовый план (test plan).

Требования к программному продукту (product requirements document,

PRD) и функциональные спецификации (functional specifications document, FSD; software requirements specification, SRS).

Архитектуру и дизайн (architecture and design).

Тест-кейсы и наборы тест-кейсов (test cases, test suites).

Технические спецификации (technical specifications), такие как схемы баз данных, описания алгоритмов, интерфейсов и т.д.

Project documentation

Пользовательскую и сопроводительную документацию (user and accompanying documentation), такую как встроенная помощь, руководство по установке и использованию, лицензионные соглашения и т.д.

Маркетинговую документацию (market requirements document, MRD), которую представители разработчика или заказчика используют как на начальных этапах (для уточнения сути и концепции проекта), так и на финальных этапах развития проекта (для продвижения продукта на рынке).

Источники и пути выявления требований

Интервью	Работа с фокусными группами	Анкетирование
Семинары и мозговой штурм	Наблюдение	Прототипирование
Анализ документов	Моделирование процессов и взаимодействий	Самостоятельное описание

Уровни и типы требований

Уровни и типы требований

Форма представления, степень детализации и перечень полезных свойств требований зависят от уровней и типов требований.

Бизнес-требования (business requirements) выражают цель, ради которой разрабатывается продукт (зачем вообще он нужен, какая от него ожидается польза, как заказчик с его помощью будет получать прибыль). Результатом выявления требований на этом уровне является 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

Уровни и типы требований

Пользовательские требования (user requirements) описывают задачи, которые пользователь может выполнять с помощью разрабатываемой системы (реакцию системы на действия пользователя, сценарии работы пользователя). Поскольку здесь уже появляется описание поведения системы, требования этого уровня могут быть использованы для оценки объёма работ, стоимости проекта, времени разработки и т.д. Пользовательские требования оформляются в виде вариантов использования (use cases), пользовательских историй (user stories), пользовательских сценариев (user scenarios).

Уровни и типы требований

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.

Атрибуты качества (quality attributes) расширяют собой нефункциональные требования и на уровне пользовательских требований могут быть представлены в виде описания ключевых для проекта показателей качества (свойств продукта, не связанных с функциональностью, но являющихся важными для достижения целей создания продукта — производительность, масштабируемость, восстанавливаемость). Атрибутов качества очень много, но для любого проекта реально важными является лишь некоторое их подмножество.

Уровни и типы требований

Функциональные требования (functional requirements) описывают поведение системы, т.е. её действия (вычисления, преобразования, проверки, обработку и т. д.). В контексте проектирования функциональные требования в основном влияют на дизайн системы.

Нефункциональные требования (non-functional requirements) описывают свойства системы (удобство использования, безопасность, надёжность, расширяемость и т. д.), которыми она должна обладать при реализации своего поведения.

Уровни и типы требований

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

Требования к интерфейсам (external interfaces requirements) описывают особенности взаимодействия разрабатываемой системы с другими системами и операционной средой.

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.

Спецификация требований (software requirements specification, SRS) объединяет в себе описание всех требований уровня продукта и может представлять собой весьма объёмный документ (сотни и тысячи страниц)

Уровни и типы требований

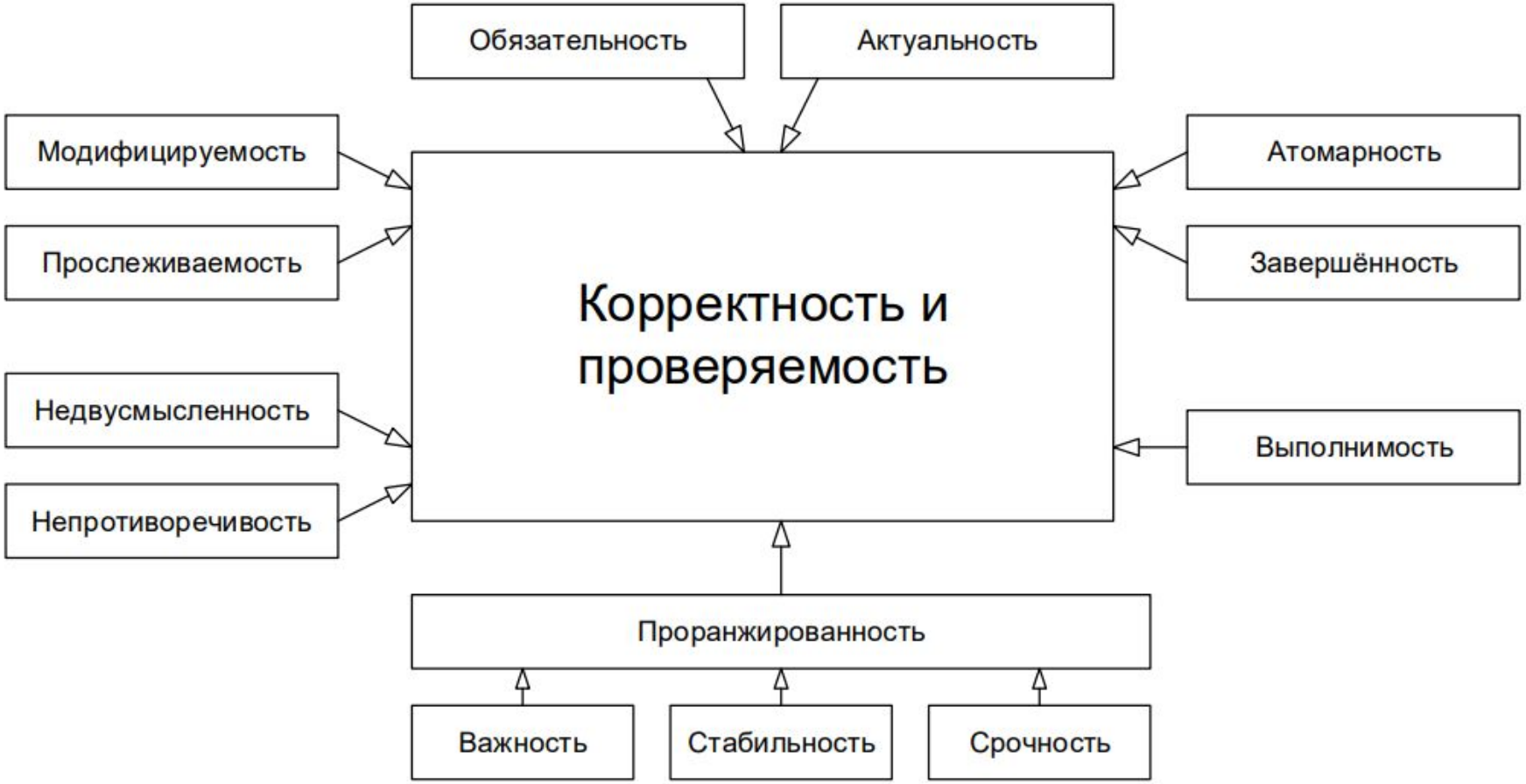
Ограничения (limitations, constraints) представляют собой факторы, ограничивающие выбор способов и средств (в том числе инструментов) реализации продукта.

Требования к интерфейсам (external interfaces requirements) описывают особенности взаимодействия разрабатываемой системы с другими системами и операционной средой.

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.

Спецификация требований (software requirements specification, SRS) объединяет в себе описание всех требований уровня продукта и может представлять собой весьма объёмный документ (сотни и тысячи страниц)

Свойства качественных требований



Свойства качественных требований

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».

Атомарность, единичность (atomicity). Требование является атомарным, если его нельзя разбить на отдельные требования без потери завершённости и оно описывает одну и только одну ситуацию.

Непротиворечивость, последовательность (consistency). Требование не должно содержать внутренних противоречий и противоречий другим требованиям и документам.

Недвусмысленность (unambiguousness , clearness). Требование должно быть описано без использования жаргона, неочевидных аббревиатур и расплывчатых формулировок, должно допускать только однозначное объективное понимание и быть атомарным в плане невозможности различной трактовки сочетания отдельных фраз.

Свойства качественных требований

Выполнимость (feasibility). Требование должно быть технологически выполнимым и реализуемым в рамках бюджета и сроков разработки проекта.

Обязательность, нужность (obligatoriness) 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.

Свойства качественных требований

Прослеживаемость (traceability). Прослеживаемость бывает вертикальной (vertical traceability) и горизонтальной (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.

Модифицируемость (modifiability). Это свойство характеризует простоту внесения изменений в отдельные требования и в набор требований. Можно говорить о наличии модифицируемости в том случае, если при доработке требований искомую информацию легко найти, а её изменение не приводит к нарушению иных описанных в этом перечне свойств

Свойства качественных требований

Проранжированность по важности, стабильности, срочности (ranked for importance, stability, priority). Важность характеризует зависимость успеха проекта от успеха реализации требования. Стабильность характеризует вероятность того, что в обозримом будущем в требование не будет внесено никаких изменений. Срочность определяет распределение во времени усилий проектной команды по реализации того или иного требования.

Корректность (correctness) и проверяемость (verifiability). Фактически эти свойства вытекают из соблюдения всех вышеперечисленных (или можно сказать, что они не выполняются, если нарушено хотя бы одно из вышеперечисленных). В дополнение можно отметить, что проверяемость подразумевает возможность создания объективного тест-кейса (тест-кейсов), однозначно показывающего, что требование реализовано верно и поведение приложения в точности соответствует требованию.