

Разработка и анализ требований

Инженерия требований (requirements engineering) –

**процессы формулирования,
документирования и поддержки
требований к ПО,**

+

**соответствующая область
программной инженерии**

Процессы разработки и управления требованиями по SWEBOK 2004-2014

Инженерия требований

Разработка требований

Выявление - Requirements Elicitation

Анализ - Requirements Analysis

Спецификация (документирование) -
Requirements Specification

Проверка и согласование -
Requirements Validation

Управление требованиями

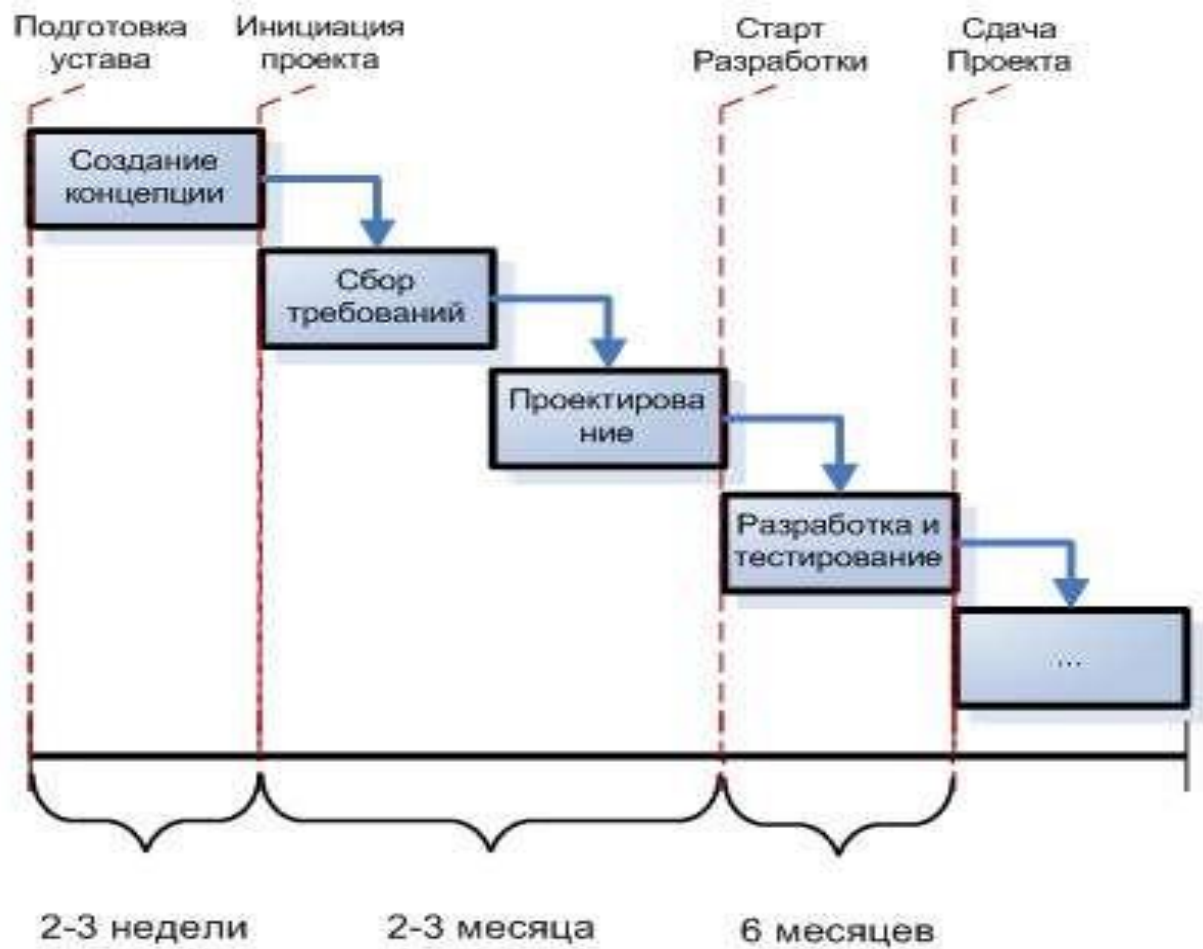
изменениями

версиями

статусами

трассировка

Место требований в жизненном цикле проекта, на примере каскадного и итеративного процессов



Rational Unified Process

Рабочие процессы

Стадии

Основные процессы

Бизнес-моделирование

Управление требованиями

Анализ и проектирование

Реализация

Тестирование

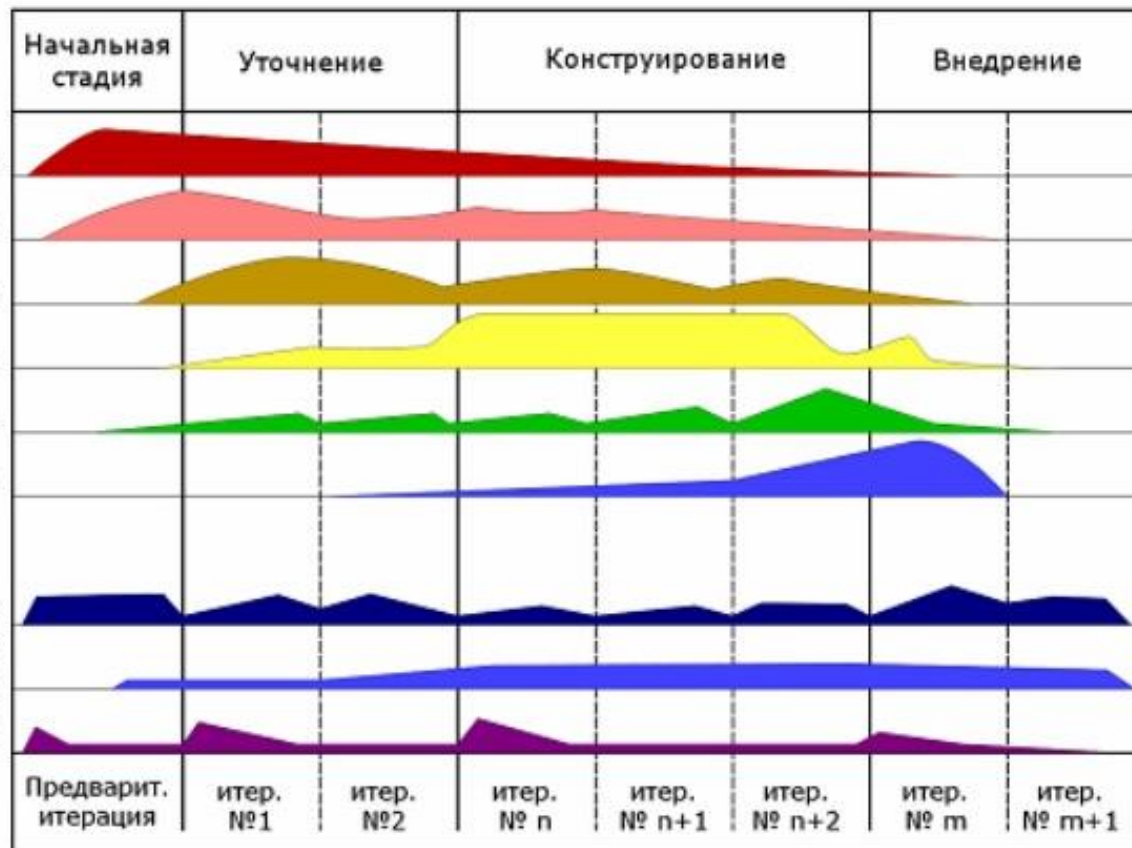
Развертывание

Поддерживающие процессы

Управление проектом

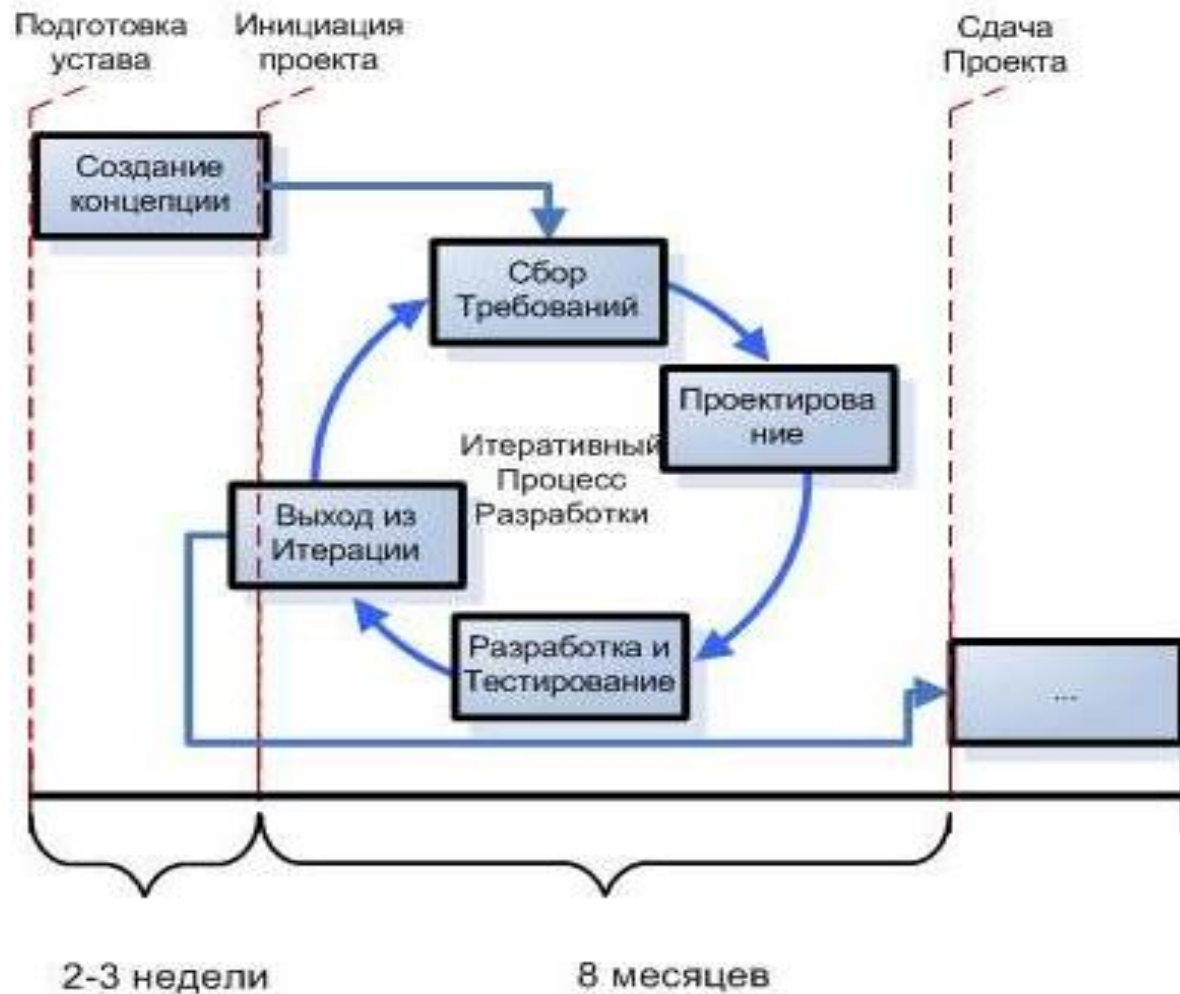
Управление конфигурацией
и изменениями

Создание инфраструктуры
(среда разработки)



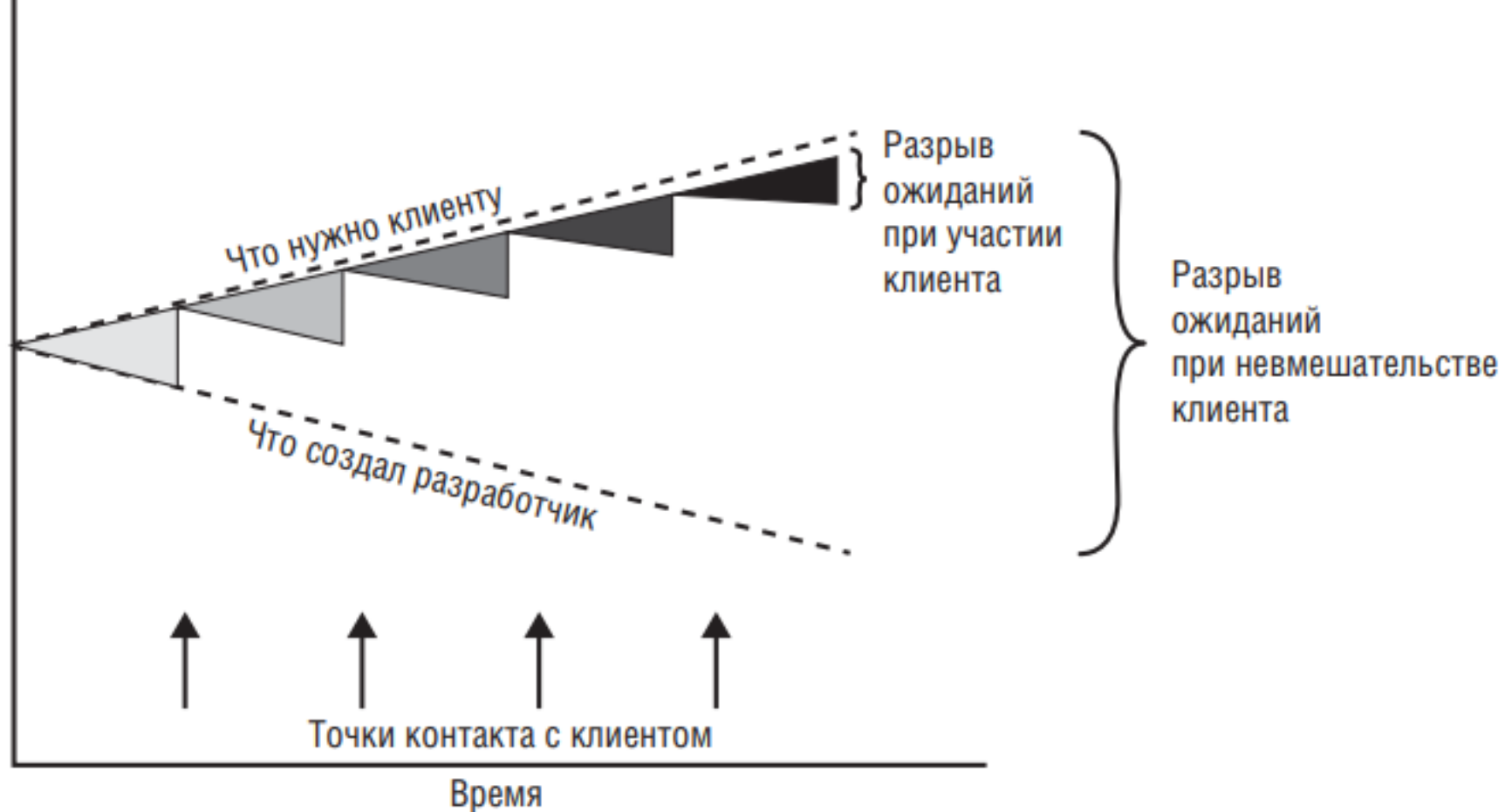
Итерации

Место требований в жизненном цикле проекта, на примере каскадного и итеративного процессов



Распределение работ с требованиями на протяжении жизненного цикла проекта в разных моделях разработки





**Частое эффективное
взаимодействие с клиентом
сокращает разрыв ожиданий**



Виды деятельности в процессе создания требований

0. Подготовить среду для работы
1. Определить лиц, ответственных за принятие решений по требованиям
2. Спланировать выявление требований
3. Определить бизнес-требования
4. Определить классы пользователей
5. Определить представителей пользователей

Однократно

6. Определить пользовательские требования
7. Определить приоритеты пользовательских требований
8. Документировать пользовательские требования
9. Вывести функциональные требования
10. Смоделировать требования
11. Определить атрибуты качества
12. Просмотр требований
13. Разработка прототипов
14. Разработка или расширение архитектуры
15. Распределение требований по компонентам
16. Разработка тестов на основе требований
17. Проверка пользовательских, функциональных, нефункциональных требований, моделей анализа и прототипов

N итераций

Выявление требований

**Мероприятия, действия, работы,
направленные на
поиск, извлечение, сбор и фиксирование «сырых»
неструктурированных требований**

Источники требований

- Заинтересованные лица
- Документированные сведения о предметной области
- Предметная область (специфика бизнеса заказчика, сегмент рынка и класс продуктового ПО, ...)
- Конкурентные системы будущего продукта, их маркетинговые материалы

Анализ заинтересованных лиц

Заинтересованное лицо (stakeholder) –

человек, группа или организация, которая

- а) активно задействована в проекте
- б) подвержена влиянию процесса или результата проекта
- в) может влиять на процесс или результат



Внешние

Внутренние

к проекту

Заинтересованные лица (stakeholders)

За пределами разрабатывающей организации

Прямой пользователь	Руководство компании	Консультант
Непрямой пользователь	Специалист по контрактам	Аудитор
Покупатель	Государственное учреждение	Орган сертификации
Специалисты по закупкам	Эксперт предметной области	Регулирующий орган
Юристы	Менеджер программ	Поставщик ПО
Подрядчик	Бета-тестировщик	Поставщик материалов
Субподрядчик	Широкая общественность	Венчурный предприниматель

Разрабатывающая организация

Менеджер разработки	Сбытовики	Куратор в руководстве компании
Отдел маркетинга	Специалист по установке	Управление проектом
Специалисты по эксплуатационной поддержке	Специалист по поддержке	Производство
Юристы	Менеджер программ	Специалисты по обучению
ИТ-архитектор	Эксперт по эргономике	Архитектор портфолио программ
Собственник компании	Эксперт предметной области	Специалисты по поддержке инфраструктуры

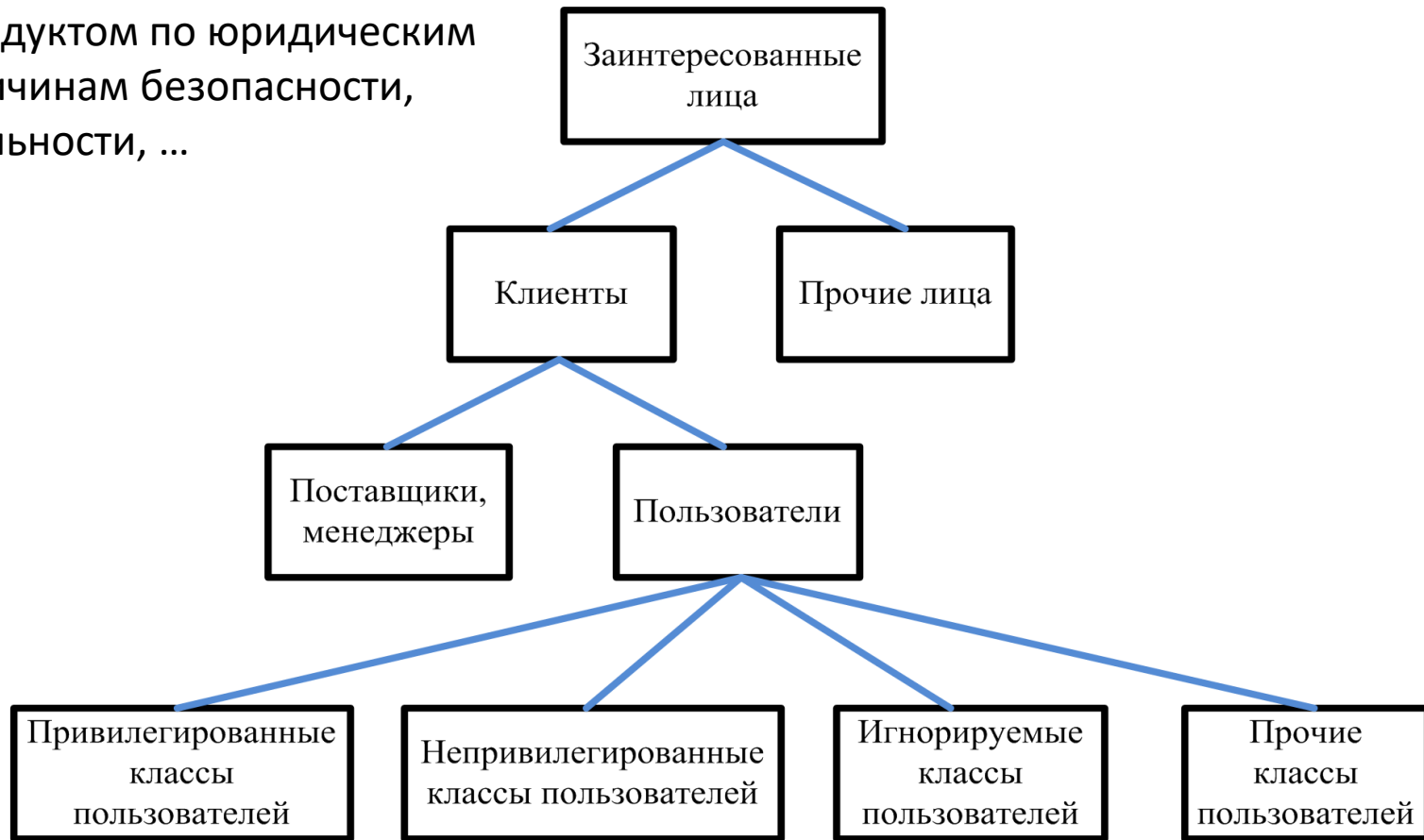
Команда проекта

Менеджер проекта	Тестировщик
Бизнес-аналитик	Менеджер продукта
Архитектор приложения	Специалисты по контролю качества
Дизайнер	Технический писатель
Разработчик	Администратор баз данных
Владелец продукта	Инженер аппаратного обеспечения
Специалист по моделированию данных	Аналитик инфраструктуры
Аналитик процессов	Архитектор бизнес-решений

Привилегированные классы формулируют требования, удовлетворение которых более всего соответствует достижению бизнес-целей проекта, имеют приоритет

- в очередности реализации требований
- в разрешении противоречий и конфликтов

Непривилегированные классы не должны работать с продуктом по юридическим причинам, причинам безопасности, конфиденциальности, ...



Важнейший источник требований

Клиент (customer) — человек или организация, получает от продукта прямую или косвенную выгоду

- запрашивает
- оплачивает
- выбирает
- определяет
- использует
- получает

результаты работы программного продукта

прямой пользователь, непрямой пользователь, куратор, специалист по закупкам, покупатель,...

Классифицировать и выбрать клиентов как источник требований

- **Определить классы пользователей и их характеристики**
по привилегиям доступа (рядовой, гость, администратор); по задачам, решаемым в бизнес-операциях; по используемым функциям ПО; по частоте использования; по опыту в предметной области и опыту работы с компьютером; по платформе (настольные ПК, ноутбуки, ...); по родному языку; по виду доступа к системе — прямой или косвенный
- **Выбрать сторонника продукта (product champion) в каждом классе** (принимает решения от их лица)
- **Проведение фокус-групп типичных пользователей**
(только опрос, без полномочий на принятие решений)

File Edit View Insert Format Table Tools Help

Heading 1 Verdana 12pt B I U Link

Геймеры.

Домохозяйка Нина

33 года, двое детей (один ходит в школу, другой в детский сад). Муж основную часть времени проводит на работе или отсыпается. Все время, не занятое домашним хозяйством проводит около домашнего компьютера (одноклассники, "правильный шоппинг", игры, развесистые ответы в социальных сетях, наработанные навыки работы в интернете). Следит за собой, регулярно наводит "марафет". Может включить, выключить, запустить браузер, и т.д. С компьютером познакомилась еще до замужества. Игровые предпочтения: простенькие стратегии, и т.д. Терпение: сумочку нужной расцветки, фасона и т.д. не хочет). Не против писать отзывы и сообщения об ошибках.

Игроделы

Алексей

28 лет, профи-программист flash. Вырос из кодировщика, не очень хорошо рисует, кроме механикой. Делает хорошие аркады, но тестировать не умеет, поэтому продукт выходит, и не всегда востребованный. Имеет хороший опыт работы в фирмах с высокой культурой разработки ПО. Игры любит, и считает что ими можно зарабатывать. Основную работу еще не бросил, но с

Офисная планктонина Георгий

25 лет, не женат, детей нет. С компьютером на вы, Тоха. Умеренно следит за собой. Играет преимущественно в стратегии. Игровые предпочтения

Тоха

40 лет, вольный художник и всегда им был.

До сих пор держит небольшую сеть (являясь субпровайдером), для ограниченного числа пользователей. Но из этого вытесняется крупными провайдерами.

Берется за любую ИТ-шную халтуру (протянуть сеть, настроить 1С, поднять шлюз и т.д.) лишь бы не работать на полную ставку. Имеет несколько постоянных клиентов, которые его знают лет 10-15 и подкидывают постоянные заказы.

Выбор редакции.

Тоха или Нина. Скорее всего Нина, т.к. Тоха может просто "забить" на критику.

В таблице (Таблица 3) приведен перечень лиц, заинтересованных в создании *ИС Аттестации студентов*. Для каждого заинтересованного лица указаны его интересы.

Таблица 3. Заинтересованные лица.

№	Наименование	Описание	Интересы
1	УМУ	Подразделение ВУЗа, ответственное за контроль и обеспечение учебного процесса	11, 12, 13, 14, 15, 16, 17
2	Деканат	Организационный центр по управлению работой факультета, возглавляемый деканом	11, 12, 13, 14, 15
3	Кафедра	Подразделение деканата, обеспечивающее управление учебным процессом по определённому направлению факультета.	11, 13, 14, 15, 18
4	Отдел кадров	Подразделение ВУЗа, занимающееся кадровой работой со студентами.	11, 13, 14
5	Преподаватель	Работник ВУЗа, ведущий занятия по какому-либо предмету	12, 15, 18, 19, 111
6	Студент	Учащийся ВУЗа	18, 110, 112

Аааааааааа, ну зачем ты так ☹ Нужно же и к трассировке подходить творчески ☹
 ЦМХО более понятно было когда у каждого ЗЛ в табличке были заданы свои нужды, ну ничего что они пересекаются

3. Проблемы Заказчика

В таблице (Таблица 4) приведен перечень проблем Заказчика *ИС Аттестации студентов*.

Таблица 4. Проблемы Заказчика.

№	Проблема	Интересы
P1	Несвоевременность, недостоверность и неактуальность получаемой	11

Приемы выявления требований

- **Работа с заказчиками (руководство, пользователи,...)**
 - Интервью
 - Совместные семинары
 - Мозговой штурм
 - Раздача опросных листов – анкетирование
- **Наблюдение за пользователями на рабочих местах**
- **Анализ документов**
- **Изучение уже работающих систем**
- **Повторное использование требований**

Собеседование (интервью)

Преимущества

- Опрашиваемое лицо может свободно и открыто отвечать на вопросы и почувствовать себя участником проекта
- Лицо, проводящее собеседование, может наблюдать за поведением опрашиваемого лица, изменять ход опроса, переформулировать или иначе построить вопросы во время собеседования

Недостатки

- Метод трудоемкий и дорогой, поэтому может оказаться не практичным
- Успех собеседования зависит от навыков общения лица, проводящего собеседование, и от желания опрашиваемых лиц участвовать в интервью

Некоторые правила проведения интервью

- Перед проведением интервью желательно познакомиться с документацией и рассматриваемым вопросам
- Все вопросы должны быть подготовлены и продуманны заранее
- Четко представлять цель интервью
- Включать в интервью ограниченное количество вопросов
- Малая длительность (от 1 до 2 часов)
- Не перед обедом и не поздно вечером (перед концом рабочего дня)
- Объяснить свою роль сотруднику подразделения перед началом интервью

Что нужно сказать сотруднику подразделения в начале интервью

- Почему проводится это интервью
- От кого получено разрешение его проводить
- Кто еще будет проинтервьюирован
- По какому принципу и кто выбирал интервьюируемых
- Как будет использована полученная информация
- Будет ли интервью анонимным
- Будет ли интервью отражено в отчете
- Какова будет обратная связь по итогам обработки результатов интервью
- Почему важно получить детальную и точную информацию в процессе интервью

Некоторые правила проведения интервью

- Не отвлекаться на посторонние комментарии по проблемам, связанным с обсуждаемым предметом
- Не пытаться завязать дружеские отношения
- Не указывать интервьюируемому на его затруднения при описании предметной области
- Давать интервьюируемому время подумать
- Отделить «мнения О..» от фактов
- Не иронизировать
- Концентрироваться на наиболее сложных аспекта предметной области
- Не пытаться показывать свои звания (эксперт не вы, а интервьюируемый)
- Не увеличивать длительность проведения интервью

Обработка полученной информации

- Перенос полученной информации в электронную форму
(некоторые руководители требуют полностью документировать результаты интервью)
- Проверка информации на корректность
 - устранение видимых нестыковок, противоречий, неоднозначностей, разночтений с ранее полученными и зафиксированными данными

Совещания

- Достоинства:
 - Все заинтересованные лица могут высказывать свое мнение
 - Формируется соглашение между заинтересованными лицами и разработчиками
 - Разрешаются политические вопросы, влияющие на успех проекта
 - Результат становится известным немедленно
- Недостатки:
 - Необходимость управления временем
 - Доминирующие позиции отдельных участников
 - Недостатки предложений от участников
 - Негативные комментарии, мелочное поведение и скрытая враждебность

Мозговой штурм и отбор идей

- это метод проведения собрания, при котором группа людей пытается найти решение специфической проблемы посредством накопления всех спонтанно возникающих идей

- Достоинства:
 - Поддерживает участие всех присутствующих
 - Позволяет участникам развивать идеи друг друга
 - Как правило, в результате получается множество возможных решений
 - Способствует свободному мышлению, не ограниченному обычными рамками
- Проблемы: те же, что и для совещания

Анкетирование

- Преимущества:
 - Люди могут заполнять и возвращать анкеты в удобное для них время
 - Люди склонны сообщать в ответах действительные факты, если анкетирование анонимное
 - Анкетирование – относительно недорогой способ сбора данных с участием большого числа людей
- Недостатки:
 - Не все могут согласиться ответить на вопросы, анкеты могут возвращаться незаполненными
 - Нет возможности пояснить или переформулировать не правильно понятые вопросы, наблюдать и анализировать реакцию респондента на отдельные вопросы
 - Предложения, стоящие за вопросами, оказывают влияние на ответы

Наблюдение

- Различают пассивное и активное наблюдение. При активном наблюдении аналитик работает, как участник команды, что позволяет улучшить понимание процессов.
- Через наблюдение, а возможно, и участие аналитики получают информацию о происходящих день за днем операциях из первых рук.

Во время наблюдения за работой системы часто возникают вопросы, которые никогда бы не появились, если бы аналитик только читал документы или разговаривал с экспертами.

- Недостатком этой стратегии является то, что наблюдатель, как и всякий "измерительный прибор", вносит помехи в результаты измерений: сотрудники организации, находясь "под колпаком" могут начать вести себя принципиально по-другому, чем обычно.

Типы раскадровок

- Пассивные:
 - Экранные копии
 - Выходные документы (отчеты)
- Активные
 - Демонстрация слайдов
 - Анимация
 - Имитация
- Интерактивные
 - «Живая» демонстрационная версия
 - Интерактивная презентация

Документы для исследования предметной области (бизнес-автоматизация)

1. Сводная информация о деятельности предприятия

- Информация об управленческой, финансово-экономической, производственной деятельности предприятия
- Штаты, положения о подразделениях, функциональные обязанности должностных лиц
- Сведения об учетной политике и отчетности

2. Регулярный документооборот предприятия

- Реестр входящей информации
- Реестр внутренней информации
- Реестр исходящей информации

3. Сведения об информационно-вычислительной инфраструктуре предприятия

4. Сведения об ответственных лицах

Признаки окончания процесса выявления

- пользователи не могут придумать новых Use case или User story *(обычно они называют их в порядке убывания значимости)*
- вновь предлагаемые требования имеют низкий приоритет
- пользователи предлагают новые Use case , но те не приводят к появлению новых функциональных требований *(новые предложения оказываются случаями других, уже рассмотренных UC)*
- пользователи повторно описывают уже обсужденные проблемы
- предлагаемые новые функции выходят за рамки проекта
- пользователи предлагают возможности, которые можно реализовать «когда-то позже», а не включить «в конкретный продукт, который мы сейчас обсуждаем»
- разработчики и тестировщики, задают очень мало вопросов по требованиям

Анализ требований

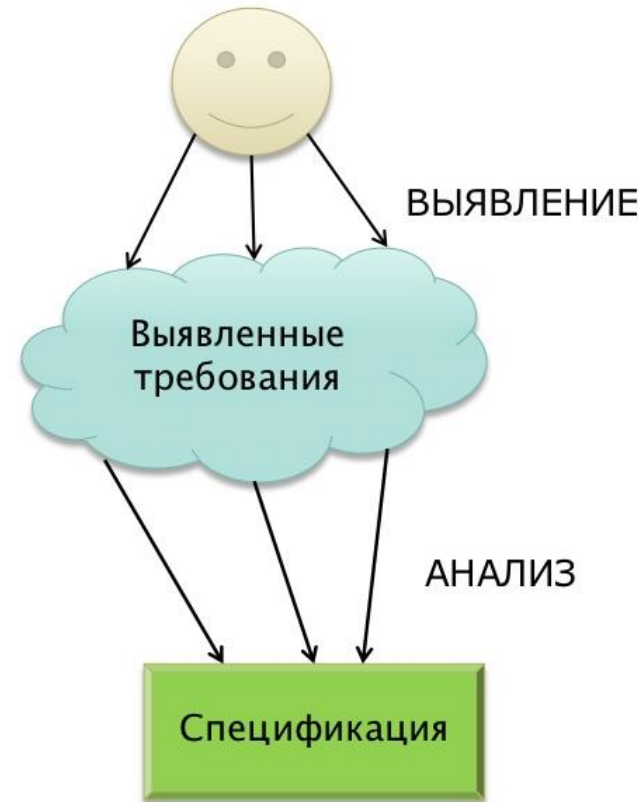
(в узком смысле)

- **Выявление** – расходящийся процесс, цель – собрать как можно больше данных

Результат выявления –
неструктурированный реестр требований

- **Анализ** – сходящийся процесс, цель – добиться качества требований.

Результат анализа – готовый к
документированию набор
структурированных данных о требованиях



Организация требований

Задачи, решаемые при анализе требований:

1. Структурировать информацию о требованиях
(систематизировать, выявить взаимосвязи)
 - Группирование
 - Иерархическая структуризация
(подчинение, уточнение)
2. Устранить неточности и явные противоречия
3. Уточнить недостающие данные
4. Установить приоритеты



Свойства требований

Обязательность (необходимость) — требование должно отражать возможность или характеристику, **действительно** необходимую пользователям, или вытекающую из других требований, ее отсутствие ведет к неполноценности решения, она не может быть проигнорирована.

Отслеживаемость — требование полностью или частично соответствует деловым нуждам как это заявлено заинтересованными лицами и задокументировано.

Есть возможность отследить соответствие бизнес-целям

Выполнимость (осуществимость)— требование должно быть выполнимым при заданных ограничениях.

Проверяется в процессе анализа осуществимости. Для нефункциональных требований проверяется возможность достижения указанных численных значений при существующих ограничениях

Проверяемость — существует конечный и разумный по стоимости процесс ручной или машинной проверки того, что ПО удовлетворяет этому требованию

Корректность — требование не содержит неверной, неточной информации и полностью соответствует документации

Согласованность (непротиворечивость, последовательность) системы требований — отдельные требования в системе требований не противоречат друг другу

Полнота (завершенность) — требование должно быть полностью определено в одном месте и содержать всю необходимую информацию для его реализации. Система требований не должна содержать невыявленных и не определенных требований. Причины неполноты описания следует явно объявлять.

Недвусмысленность (однозначность) — требование внутренне непротиворечиво и все работающие с ним должны понимать его одинаково (возможна одна и только одна его интерпретация).
Требование выражает факты, а не субъективные мнения.
Определение не содержит нечетких фраз, обращения к техническому жаргону, акронимам и другим скрытым формулировкам, не использует отрицательных и составных утверждений

Единичность — требование описывает одну и только одну вещь

Атомарность — требование нельзя разделить на более мелкие

Актуальность — не устарело с течением времени

Требования к требованиям

Отдельные требования должны быть

- необходимы
- четко и однозначно сформулированы
- непротиворечивы
- полны
- исполнимы в рамках проекта
- проверяемы
- трассируемы

Наборы требований должны быть

- полны
- согласованы
- отслеживаемы
- специфицированы в структурированный документ так, чтобы пользователь мог легко понять смысл каждого требования в контексте всего документа

Цель анализа и последующей спецификации – добиться этого.

Уточнение формулировок

Обеспечивает

четкость формулировок, однозначность, недвусмысленность:

Читатель требования должен интерпретировать его также как

- все другие читатели

- автор

Количественные, а не качественные оценки везде, где это возможно

внутреннюю полноту и непротиворечивость,

- описание лаконично, но содержит все необходимое для понимания

- описание не содержит логических противоречий

Достигается повторными контактами со стейкхолдерами.

Может быть дорого!

Уточнение формулировок

Пример 1:

- а) Если возможно, номера счетов следовало бы проверять по списку корпоративных счетов
- б) В момент ввода номера счета система должна отобразить сообщение об ошибке, если этого номера счета в основном корпоративном списке счетов

Пример 2:

- а) Диспетчер фоновых задач должен предоставлять сообщения о состоянии через регулярные интервалы, составляющие не менее 60 секунд.
- б) 1. Диспетчер фоновых задач (ДФЗ) должен отображать сообщения о состоянии задачи в *определенной области* пользовательского интерфейса.
 - 1.1. ДФЗ должен обновлять сообщения каждые 60 плюс/минус 5 секунд после запуска фоновой задачи
 - 1.2. Сообщения должны оставаться видимыми все время, пока работает фоновая задача.
 - 1.3. Если взаимодействие с фоновой задачей возможно, ДФЗ должен отображать процент выполнения фоновой задачи.
 - 1.4. По завершению фоновой задачи ДФЗ должен отобразить сообщение «Выполнено»

Полнота одиночных требований.

Явные и неявные требования

Явные требования – прямо указаны источником требований, в дальнейшем явным образом зафиксированы в спецификации, ТЗ,...

Неявные и подразумеваемые требования

Неявные (implied) – не упомянуты и не описаны явным образом в документации, но

- Прямо или косвенно следуют из других требований

Подразумеваемые (assumed) — то, что люди ожидают получить, считая само-собой разумеющимся

- Подчиняются известным научным законам
- Подчиняются жизненным реалиям

Явные и неявные требования

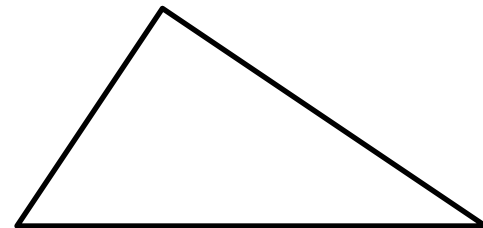
Требование: обеспечить расчет площади треугольника по трем сторонам

Входные данные:

3 вещественных числа

Выходные данные:

площадь треугольника, 1 вещественное число



Явные:

на вход подаются три вещественных числа,

на выходе – площадь, посчитанная по формуле Герона

Неявные: ?

№ теста	a	b	c	s	passed
1	3	4	5	6	yes
2	52	51	25	624	yes
3	101	63	34	NaN	no

Входные числа должны соответствовать длинам сторон существующего ТРЕУГОЛЬНИКА

Явные и неявные требования

Требование: регистрация пользователя в системе
с сохранением его домашнего адреса

Входные данные:
домашний адрес

Явные:
адрес содержит поля:
Улица (строка)
Дом (число)
Квартира (число)

Неявные: ?

- 1. Страна, Область, Населенный пункт,... - надо?**
Москва – столица РФ; Москва, штат Айдахо
- 2. Улица, проспект, проезд, переулок / Комплекс**
- 3. Номер дома с суффиксом, с дробью**
146, 8/12
- 4. Номер квартиры с суффиксом**
5а
- 5. Номер корпуса**
(в одном доме нумерация квартир в разных корпусах),
числовой/символьный

Какая система? Например:

- а) доставка пиццы –**
масштаб населенный
пункт
- б) доставка товаров**
интернет-магазина по
всему миру

Явные и неявные требования

Требование: обеспечить возможность посылать/получать личные сообщения в соц.сети, на форуме,...

Явные:

- Пользователь имеет возможность послать сообщение любому другому зарегистрированному пользователю системы
- Пользователь получает и может просматривать полученные личные сообщения

Неявные: ?

Никакой другой пользователь не должен увидеть (в своих входящих) сообщения, предназначенные не ему, а другому пользователю

Явные и неявные требования

Требование: система должна работать 24/7

Явные:

- Система должна штатно функционировать 24 часа каждый день
- Система должна иметь «горячий» резерв / систему восстановления на случай внештатных ситуаций

Неявные: ?

1. Система должна функционировать в любое время и в любой день, любого месяца, любого года, в том числе
 - при смене дня (в 23:59, 00:00 ч)
 - при смене месяца (28, 29, 30, 31 числа - !!! Платежи)
 - при смене года, тысячелетия, ...
 - при смене зимнего/летнего времени (если есть) , ...
2. Система должна работать без ребута – все функции выполняются вне зависимости от длительности работы
 - без переполнения RAM, HDD, ...
 - без накопления ошибок

Явные и неявные требования

Требование: вход в систему оплаты ЖКХ по номеру договора, номеру счетчика, сумме последнего платежа

Явные:

Пользователь указывает

номер договора (целое),
номер счетчика (целое),
сумму последнего платежа (вещественное)

Неявные: ?

Все входные компоненты должны быть предварительно зафиксированы в БД

? я поставил новый счетчик / заключил новый договор

? я сделал платеж только что (минуту назад), он уже зафиксирован?

? я не помню сумму последнего платежа

? я сделал несколько платежей одновременно, какой из них указать?

...

Техники анализа требований

- Проверить пограничные значения.
Наборы требований со сложной булевой логикой («И», «ИЛИ» и «НЕ») часто неполны, для комбинации логических условий не определено соответствующее функциональное требование.

Пример. Условие «Иначе» часто упускают.

Пример: «Если стоимость заказа меньше 100 долларов, стоимость доставки составит 5,95 долларов»,

«Если стоимость заказа превышает 100 долларов, стоимость доставки составляет 5% от общей стоимости заказа».

Ровно 100 долларов?

Устранение противоречий

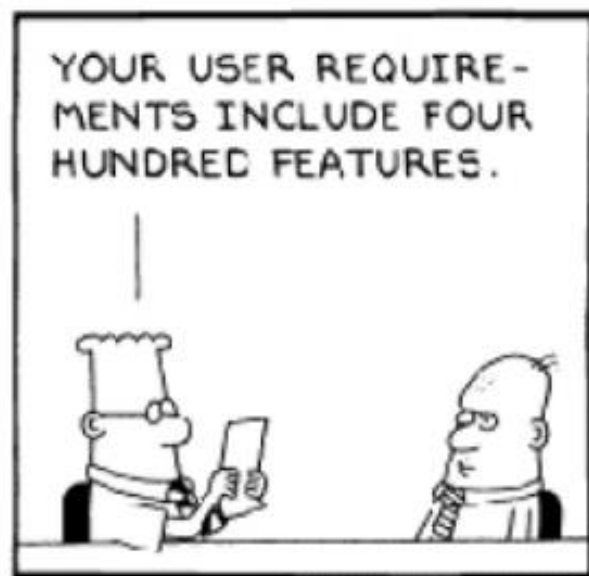
1. Противоречия в требованиях одного источника устраняются в процессе уточнения формулировок

2. Противоречивые требования разных источников требуют разрешения **конфликта интересов**



Анализ требований

- ▶ Все ли требования собраны?



Сортировка, расстановка приоритетов

Все требования не могут быть основными => они сортируются

а) по важности

б) по срочности

Участвуют

- заказчики (*! имеют преимущество*)
- пользователи
- разработчики

Приоритеты могут меняться в ходе реализации проекта

Приоритеты требований

Качественные категории важности (до 10 категорий)

- Высокая, средняя, низкая
- Обязан, должен бы, мог бы
- Основной, полезный, желаемый

Качественные категории срочности

- Срочно, чуть позже, потом
- Позавчера, прямо сейчас, чуть позже, когда-нибудь

Сортировка по двум измерениям

	Важно	Не важно
Срочно		
Не срочно		

Категоризация и именование требований

Каждое выявленное требование

- относится (классифицируется) к одной из категорий
- именуется уникальным идентификатором (обычно с информацией о категории)
- помещается в электронный реестр (БД, CASE-средство, Wiki, Excel, Google-таблицу,...)



Атрибуты требований

- позволяют не перегружать формулировки требований излишними деталями
- обеспечивают управляемость на всех этапах работы с требованиями

[SH234] Система управления скорой помощью должна быть способна принимать до 100 ста вызовов одновременно

Автор:	R. Thomas
Приоритет:	Обязательное
Релиз:	1
Статус рецензирования:	Одобрено
Возможность проверки:	Да
Способ проверки:	Симуляция, затем системные тесты

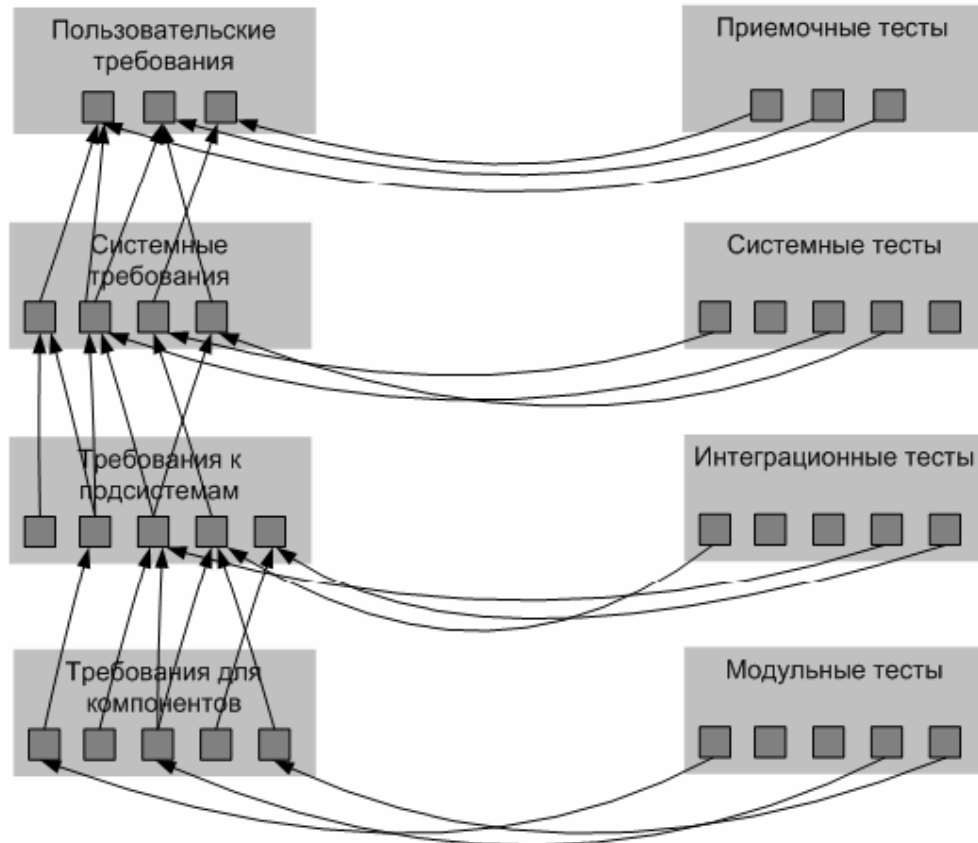
Атрибуты требований

- Идентификатор
- Автор
- Источник
- Владелец
- Класс (уровень, основной тип, подтип, качественный/количественный,...)
- Приоритет (высокий, средний, низкий)
- Важность (высокая, средняя, ...; шкала 1-10)
- Сложность реализации (высокая, средняя, низкая)
- Статус (предложено, одобрено, утверждено, реализовано, верифицировано)
- Возможность и процедура проверки
- Стоимость (высокая, средняя, низкая или числовое значение)
- Стабильность (высокая, средняя, низкая)
- Исполнитель

Техники анализа требований

- Переходить от требований высокого уровня к более конкретным (бизнес-требования -> пользовательские -> функциональные ->...)
Фиксировать связи между требованиями (*на каких функциональных требованиях основаны требования к системе; чему отвечают пользовательские требования, списки откликов на события,...*)

Связи между требованиями



Техники анализа требований

- Использовать разные формы представления требований
 - текст, структурированные текстовые шаблоны
 - таблицы
 - графические формы (диаграммы, деревья, сети, графы,...)
- Проверить, что все классы пользователей дали информацию и что у каждого пользовательского требования есть по крайней мере один класс пользователей, который выиграет от реализации этого требования
(проверка на избыточность и неполноту, на соотв. таблице)
- Отслеживать список стандартных функциональных областей:
 - ведение журнала ошибок,
 - архивирование и восстановление,
 - безопасность доступа,
 - отчетность,
 - печать,
 - возможности предварительного просмотра и конфигурирования пользовательских параметров

Периодически сверять контрольный список с уже определенными функциями на предмет выявления пробелов

Техники анализа требований

Дополнительное моделирование и исследование

Пример. Создать модель данных. Проверить, что все сущности данных имеют функциональность для их создания, чтения из внешних источников, обновления текущих значений, удаления. CRUD (Create, Read, Update, Delete — создание, чтение, обновление, удаление).

Спецификации требований

Способы документирования (представления) требований

1. Документация на основе стандартных шаблонов, с использованием четко структурированных естественных языковых конструкций
2. Графические модели, иллюстрирующие процессы преобразования, состояния системы и их изменения, отношения данных, а также логические потоки и т. п. (могут входить в состав документов из п.1)
3. Формальные спецификации, где требования определены с помощью математически точных, формальных логических языков (например, язык PDL)

ГОСТ 19.201-78 ЕСПД.
Техническое задание.
Требования к содержанию и
оформлению.

МЕЖГОСУДАРСТВЕННЫЙ СТАНДАРТ

1. ОБЩИЕ ПОЛОЖЕНИЯ

1.1. Техническое задание (ТЗ) оформляют в соответствии с *ГОСТ 19.106-78 (Требования к программным документам, выполненным печатным способом)* на листах формата 11 и 12 по *ГОСТ 2.301-68 (Форматы)*, как правило, без заполнения полей листа. Номера листов (страниц) проставляют в верхней части листа над текстом.

1.2. Лист утверждения и титульный лист оформляют в соответствии с *ГОСТ 19.104-78 (Основные надписи)*. Информационную часть (аннотацию и содержание), лист регистрации изменений допускается в документ не включать.

1.3. Для внесения изменений или дополнений в ТЗ на последующих стадиях разработки программы или программного изделия выпускают дополнение к нему. Согласование и утверждение дополнения к ТЗ проводят в том же порядке, который установлен для ТЗ.

1. ОБЩИЕ ПОЛОЖЕНИЯ

1.4. ТЗ должно содержать следующие разделы:

- введение;
- основания для разработки;
- назначение разработки;
- требования к программе или программному изделию;
- требования к программной документации;
- технико-экономические показатели;
- стадии и этапы разработки;
- порядок контроля и приемки;
- в техническое задание допускается включать приложения.

В зависимости от особенностей программы или программного изделия допускается уточнять содержание разделов, вводить новые разделы или объединять отдельные из них.

2. СОДЕРЖАНИЕ РАЗДЕЛОВ

2.1. В разделе "[Введение](#)" указывают наименование, краткую характеристику области применения программы или программного изделия и объекта, в котором используют программу или программное изделие.

2.2. В разделе "[Основание для разработки](#)" должны быть указаны:

- документ (ы), на основании которых ведется разработка;
- организация, утвердившая этот документ, и дата его утверждения;
- наименование и (или) условное обозначение темы разработки.

2.3. В разделе "[Назначение разработки](#)" должно быть указано функциональное и эксплуатационное назначение программы или программного изделия.

2. СОДЕРЖАНИЕ РАЗДЕЛОВ

2.4. Раздел "Требования к программе или программному изделию" должен содержать следующие подразделы:

- требования к функциональным характеристикам;
- требования к надежности;
- условия эксплуатации;
- требования к составу и параметрам технических средств;
- требования к информационной и программной совместимости;
- требования к маркировке и упаковке;
- требования к транспортированию и хранению;
- специальные требования.

2. СОДЕРЖАНИЕ РАЗДЕЛОВ

2.4.1. В подразделе "Требования к функциональным характеристикам" должны быть указаны требования к составу выполняемых функций, организации входных и выходных данных, временным характеристикам и т.п.

2.4.2. В подразделе "Требования к надежности" должны быть указаны требования к обеспечению надежного функционирования (обеспечение устойчивого функционирования, контроль входной и выходной информации, время восстановления после отказа и т.п.).

2. СОДЕРЖАНИЕ РАЗДЕЛОВ

2.4.3. В подразделе "[Условия эксплуатации](#)" должны быть указаны условия эксплуатации (температура окружающего воздуха, относительная влажность и т.п. для выбранных типов носителей данных), при которых должны обеспечиваться заданные характеристики, вид обслуживания, необходимое количество и квалификация персонала.

2.4.4. В подразделе "[Требования к составу и параметрам технических средств](#)" указывают необходимый состав технических средств с указанием их основных технических характеристик.

2. СОДЕРЖАНИЕ РАЗДЕЛОВ

2.4.5 В подразделе "Требования к информационной и программной совместимости" должны быть указаны требования к информационным структурам на входе и выходе и методам решения, исходным кодам, языкам программирования и программным средствам, используемым программой. При необходимости должна обеспечиваться защита информации и программ.

2.4.6. В подразделе "Требования к маркировке и упаковке" в общем случае указывают требования к маркировке программного изделия, варианты и способы упаковки.

2. СОДЕРЖАНИЕ РАЗДЕЛОВ

2.4.7. В подразделе "Требования к транспортированию и хранению" должны быть указаны для программного изделия условия транспортирования, места хранения, условия хранения, условия складирования, сроки хранения в различных условиях.

2.5а. В разделе "Требования к программной документации" должны быть указаны предварительный состав программной документации и, при необходимости, специальные требования к ней.

2. СОДЕРЖАНИЕ РАЗДЕЛОВ

2.5. В разделе "Технико-экономические показатели" должны быть указаны: ориентировочная экономическая эффективность, предполагаемая годовая потребность, экономические преимущества разработки по сравнению с лучшими отечественными и зарубежными образцами или аналогами.

2.6. В разделе "Стадии и этапы разработки" устанавливают необходимые стадии разработки, этапы и содержание работ (перечень программных документов, которые должны быть разработаны, согласованы и утверждены), а также, как правило, сроки разработки и определяют исполнителей.

2. СОДЕРЖАНИЕ РАЗДЕЛОВ

2.7. В разделе "[Порядок контроля и приемки](#)" должны быть указаны виды испытаний и общие требования к приемке работы.

2.8. В приложениях к ТЗ, при необходимости, приводят: перечень научно-исследовательских и других работ, обосновывающих разработку; схемы алгоритмов, таблицы, описания, обоснования, расчеты и другие документы, которые могут быть использованы при разработке; другие источники разработки.