

# Формирование и анализ требований

# Проблемы определения требований

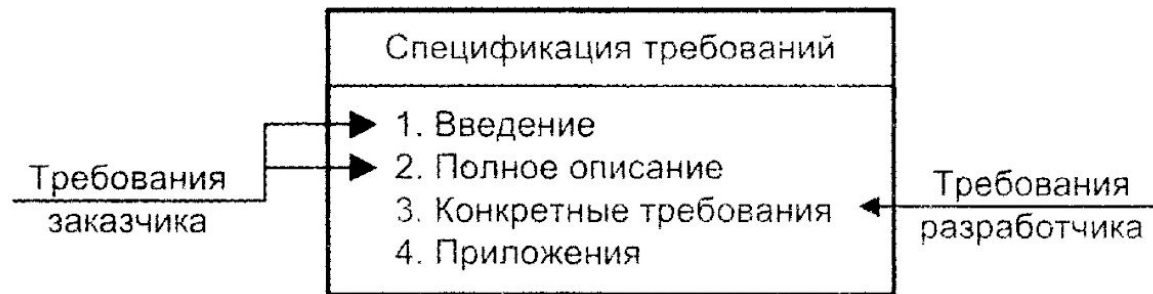
- ▶ Разработка требований – самая сложная часть проектирования ПО
- ▶ Требования пользователей постоянно меняются
- ▶ Неясность, двусмысленность и противоречивость требований
- ▶ Недостаточная определенность спецификаций
- ▶ Недостаточная представительность пользователи

# План раздела «Управление требованиями»

- ▶ Определение требований
- ▶ Разработка требований
  - Выявление требований
  - Анализ требований
- ▶ Документирование и организация требований
- ▶ Изменение требований
- ▶ Планирование и управление требованиями

# Свойства требований

- ▶ Корректность (correct)
- ▶ Однозначность (unambiguous)
- ▶ Полнота (complete)
- ▶ Непротиворечивость (consistent)
- ▶ Приоритезация (prioritized)
- ▶ Проверяемость (verifiable)
- ▶ Модифицируемость (modifiable)
- ▶ Отслеживаемость (traceable)



# 1. Требования

Требование по IEEE 1990:

- ▶ Условие или возможность, необходимые пользователю для решения его задач или достижения цели.
- ▶ Условие или возможность, которым должна отвечать или которыми должна обладать система или ее компонента, чтобы удовлетворить контракт, стандарт, спецификацию или иной формальный документ.
- ▶ Документированное представление условия или возможности, указанное в (1) или (2)

# Требования

- ▶ Виды требований:
  - Функциональные требования
    - Бизнес-требования
    - Пользовательские требования
  - Нефункциональные требования
    - Ограничения
    - Требования к качеству

# Функциональные требования

- ▶ Бизнес–требования
  - Формулируются заказчиками
  - Описывают цели, которые требуется достичь с данной системой
- ▶ Требования пользователей
  - Какие задачи можно решить с помощью системы
- ▶ Собственно функциональные требования
  - Определяются функциональность, которую необходимо реализовать



# Нефункциональные требования

- ▶ Характеристики качества
  - Требования к надежности
  - Требования к совместимости
  - Требования к эффективности
  - Требования к гибкости
  - Требования к эргономике
- ▶ Ограничения
  - Соответствия стандартам и правилам
  - Предопределенные архитектурные решения
  - Бюджет
  - Сроки

# Что не является требованиями?

- ▶ Детали архитектуры
- ▶ Детали реализации
- ▶ Сведения о планировании
- ▶ Сведения о тестировании
- ▶ Проектная информация:
  - Инфраструктура разработки
  - Процесс разработки
  - Команда разработки

## 2. Разработка требований

- ▶ Выявление требований
- ▶ Анализ требований
- ▶ Результат – спецификация требований

# Выявления требований

- ▶ **Заинтересованные лица**
  - Заказчики
  - Менеджеры
  - Пользователи
    - Операторы
    - Менеджеры
    - ...
  - Разработчики
  - Служба поддержки
  - Другие лица
- ▶ **ВАЖНО: заказчик ≠ пользователь**

# Выявления требований

## ▶ Планирование

- Цели выявления требований
- Стратегии и процессы выявления требований
- Результаты усилий по выявлению требований
- Оценки календарного плана и ресурсов
- Риски, связанные с выявлением требований

# Выявления требований

- ▶ Проблемы определения требований:
  - Ожидания пользователей
  - Умение оценить противоречивые требования
  - Недостаточные требования
  - Умение понять требования пользователей

- ▶ Способы выявления требований
  - Исследования
  - Интервью
  - Семинары
  - Создание прототипов
  - Создание вариантов использования (Use Case)
  
- ▶ Проблемы:
  - Сложности формулирования требований
  - Терминология
  - Используемые неявные допущения
  - Предвзятые решения

# Анализ требований

- ▶ Выявление требований – расходящийся процесс, цель которого собрать как можно больше данных
- ▶ Анализ требований – сходящийся процесс:
  - Уточняет данные
  - Структурирует информацию
  - Устанавливает приоритеты
- ▶ Результат анализа – спецификация требований



# Анализ требований

## ▶ Уточнение требований

- Каждое требование должно быть максимально полным
- Уточнение достигается путем повторных встреч с заинтересованными лицами
- Не должно появляться много новых требований – иначе следует вернуться к выявлению
- На этапе уточнения требования должны быть описаны *количественно*, а не качественно, как на этапе выявления

# Анализ требований

## ▶ Приоритеты

- Необходимо отсортировать требования по важности и срочности
- Должны участвовать все заинтересованные лица проекта:
  - Заказчики
  - Пользователи
  - Разработчики
- Все требования *не могут быть* основными
- Приоритеты могут изменяться по мере развития проекта

# Анализ требований

## ▶ Приоритеты

- Каждое требование относится к какой-либо качественной категории по важности:
  - Высокая, средняя, низкая
  - Обязан, должен бы, мог бы
  - Основной, полезный, желаемый
- Каждое требование относится к какой-либо качественной категории по срочности:
  - Прямо сейчас, чуть позже, когда-нибудь
  - Срочно, чуть позже, потом
- Сортируются по двум измерениям:

	<b>Важно</b>	<b>Не важно</b>
<b>Срочно</b>	Высший приоритет	Минимальный приоритет
<b>Не срочно</b>	Средний приоритет	Не стоит делать



### 3. Документирование и организация требований

- ▶ Как документировать разные требования?
- ▶ Требования пользователей
  - варианты использования
- ▶ Бизнес-требования
  - документ о представлении/границах проекта
- ▶ Функциональные требования
  - спецификация требований к ПО

# Организация требований

- ▶ Группирование требований
  - Требования объединяются в родственные группы
- ▶ Иерархическая структуризация требований
  - Подчинение
  - Уточнение

# Способы документирования требований

- ▶ Документы на естественном языке
- ▶ Графические модели
  - Диаграммы
  - Графы
  - Схемы
  - Потoki
- ▶ Формальные спецификации

# Типы документов

- ▶ Создаются все или некоторые из документов:
  - Состав и распределение работ
  - **Спецификация требований**
  - Концепция эксплуатации
  - Начальный план разработки ПО
  - Критерии принятия работ



# Состав и распределение работ

- ▶ Распределяет ответственности между заинтересованными сторонами проекта – задает правила игры:
  - Кто создает, что и когда
  - Кто тестирует, что, как и когда
  - Кто платит, за что и когда
  - Кто докладывает кому
  - Кто принимает/утверждает завершение работ или этапов
  - Кто, как и когда санкционирует изменения
  - И т.п.

# Концепция эксплуатации

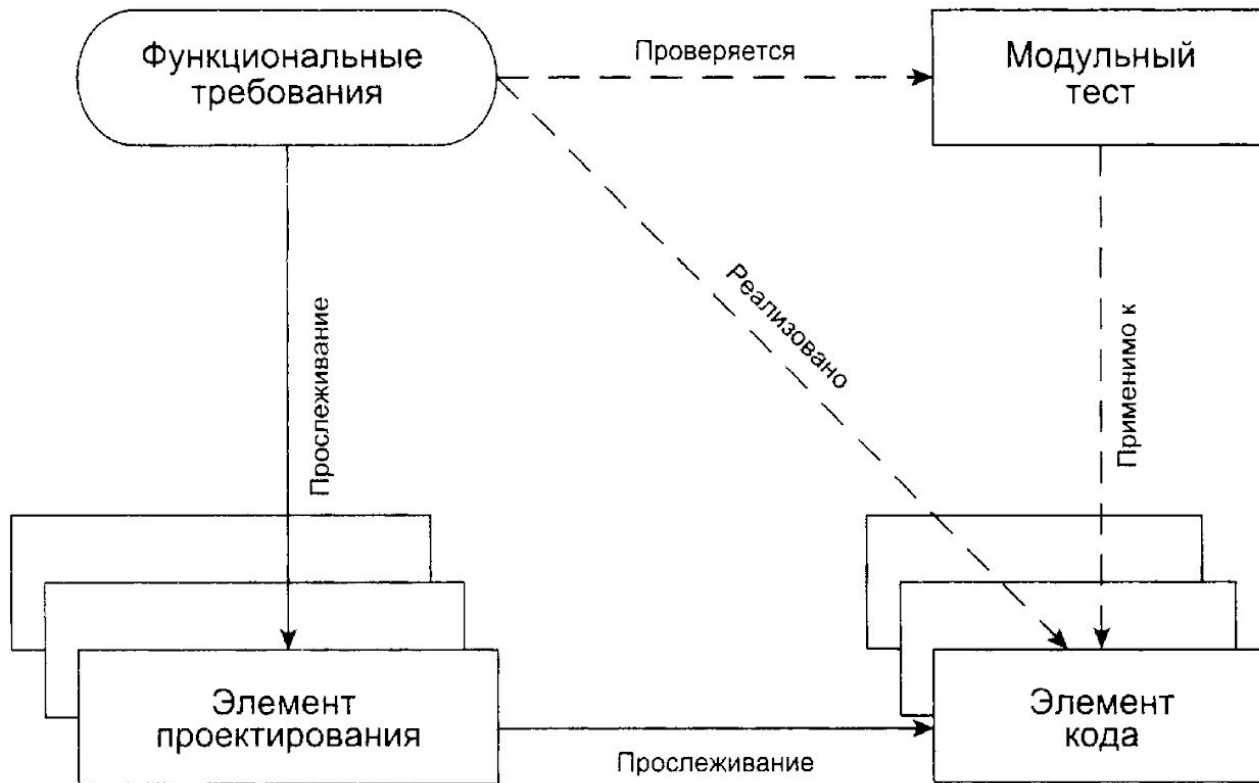
- ▶ Описание того, как система должна работать или будет использоваться
  - Какие функции будут использоваться и кем
  - Как эти функции будут использоваться
  - В каких условиях эти функции будут использоваться
  - Как будет происходить ввод/вывод данных
  - Как система взаимодействует с другими системами
- ▶ Этот документ задает основу для разработки вариантов использования

# Начальный план разработки ПО

- ▶ Высокоуровневый и приблизительный план разработки
- ▶ Задает
  - Основные документы
  - Точки принятия решений
  - Поставляемые артефакты
  - Этапы работ и контрольные точки
  - Графики платежей

# Спецификация требований

- ▶ Фундамент всего последующего планирования, проектирования, реализации проекта
- ▶ Основание для тестирования проекта
- ▶ Основание для документирования проекта
- ▶ Должна содержать ограничения проекта
- ▶ **НО:** не должна содержать деталей проектирования, реализации, тестирования и управления проектом
- ▶ Является исходным техническим соглашением между заказчиком и разработчиком



# Шаблоны спецификаций требований к ПО

- ▶ Существуют различные государственные, отраслевые и корпоративные стандарты
- ▶ Наиболее распространенные в России:
  - ГОСТ 19.201–78. ЕСПД. Техническое задание. Требования к содержанию и оформлению
  - ГОСТ 34.602–89 «Техническое задание на создание автоматизированной системы»
  - IEEE 830–1998 «Recommended Practice for Software Requirements Specification»
- ▶ Шаблон не должен являться догмой (если это не требование заказчика)
- ▶ Следует при необходимости модифицировать шаблон в соответствии с природой и потребностями проекта
- \* Полезный документ: IEEE Guide for Developing System Requirements Specifications

