

ФУНКЦИОНАЛЬНЫЕ ТРЕБОВАНИЯ К ПРОГРАММНОМУ ОБЕСПЕЧЕНИЮ

Моркель Стефан ПО-32з



Требования могут выражаться в виде текстовых утверждений и графических моделей.

В классическом техническом подходе совокупность требований используется на стадии проектирования ПО. Требования также используются в процессе проверки ПО, так как тесты основываются на определённых требованиях.

Этапу разработки требований может предшествовать технико-экономическое обоснование или концептуальная фаза анализа проекта. Фаза разработки требований может быть разбита на выявление требований (сбор, понимание, рассмотрение и выяснение потребностей заинтересованных лиц), анализ (проверка целостности и законченности), спецификация (документирование требований) и проверка правильности.

Требования к программному обеспечению

1. Требование – это не функция системы, а описание задачи или проблемы, которую хочет решить конкретный человек.
2. Так как требования это желания нескольких людей, то анализ требований начинается с выявления лиц, чьи желания система должна учитывать.
3. Требования, это ожидание, то есть то, чего еще нет. Непостоянство будущего обусловлено самой природой, и желания человека постоянно подстраиваются под изменения. Причем речь идет не о неделях и месяцах, требования будут разными в зависимости от того, в какое время дня задать вопросы.

Функциональный характер — требования к поведению системы

Бизнес-требования

Пользовательские требования

Функциональные требования

Нефункциональный характер — требования к характеру поведения системы

Характеристика	Объяснение
Единичность	Требование описывает одну и только одну вещь.
Завершённость	Требование полностью определено в одном месте и вся необходимая информация присутствует.
Последовательность	Требование не противоречит другим требованиям и полностью соответствует внешней документации.
Атомарность	Требование «атомарно». То есть оно не может быть разбито на ряд более детальных требований без потери завершённости.
Отслеживаемость	Требование полностью или частично соответствует деловым нуждам как заявлено заинтересованными лицами и документировано.
Актуальность	Требование не стало устаревшим с течением времени.
Выполнимость	Требование может быть реализовано в пределах проекта.
Недвусмысленность	Требование кратко определено без обращения к техническому жаргону, акронимам и другим скрытым формулировкам. Оно выражает объективные факты, не субъективные мнения. Возможна одна и только одна интерпретация. Определение не содержит нечётких фраз. Использование отрицательных утверждений и составных утверждений запрещено.
Обязательность	Требование представляет определённую заинтересованным лицом характеристику, отсутствие которой приведёт к неполноценности решения, которая не может быть проигнорирована. Необязательное требование — противоречие самому понятию требования.
Проверяемость	Реализованность требования может быть определена через один из четырёх возможных методов: осмотр, демонстрация, тест или анализ.

Методы выявления требований

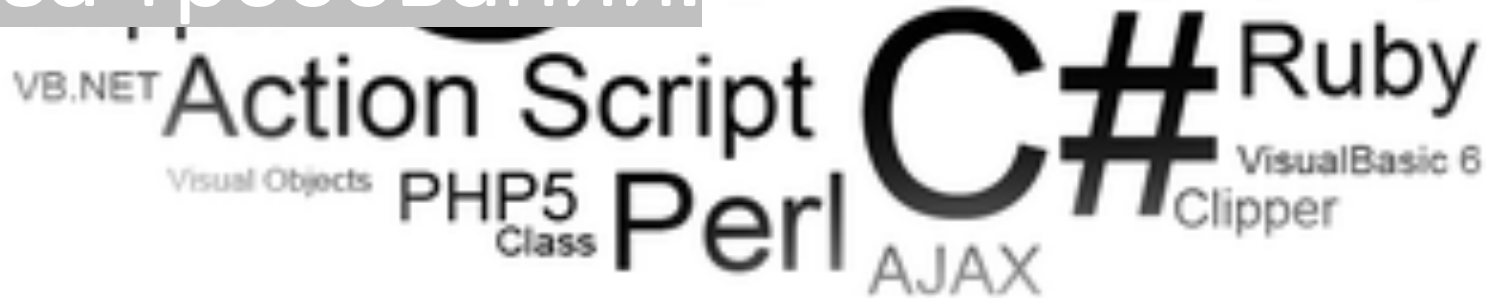
- Интервью, опросы, анкетирование
- [Мозговой штурм](#), семинар
- Наблюдение за производственной деятельностью, «фотографирование» рабочего дня
- Анализ нормативной документации
- Анализ моделей деятельности
- Анализ конкурентных продуктов
- Анализ статистики использования предыдущих версий системы

Анализ требований [[править](#) | [править код](#)]

При разработке требований часто возникают проблемы двусмысленности, неполноты, и несогласованности отдельных требований. Устранение этих проблем на этапе разработки требований стоит на несколько порядков меньше, чем устранение этих же проблем на поздних стадиях разработки. Для решения и устранения этих проблем существует процесс разработки требований.

При разработке требований существует технический компромисс между слишком неопределёнными требованиями и требованиями столь детализированными, что они: требуют много времени для разработки, иногда даже рискуют устареть к концу разработки; ограничивают возможные способы реализации; являются слишком дорогостоящими.

Требования к программному обеспечению – совокупность утверждений относительно атрибутов, свойств или качеств программной системы, подлежащей реализации. Создаются в процессе разработки требований к программному обеспечению, в результате анализа требований.



A word cloud of programming languages and technologies. The most prominent words are 'C#' and 'Perl'. Other visible words include 'Action Script', 'Ruby', 'PHP5', 'AJAX', 'Visual Objects', 'Visual Basic 6', 'Visual FoxPro 6', 'VB.NET', 'Class', and 'Clanion'.

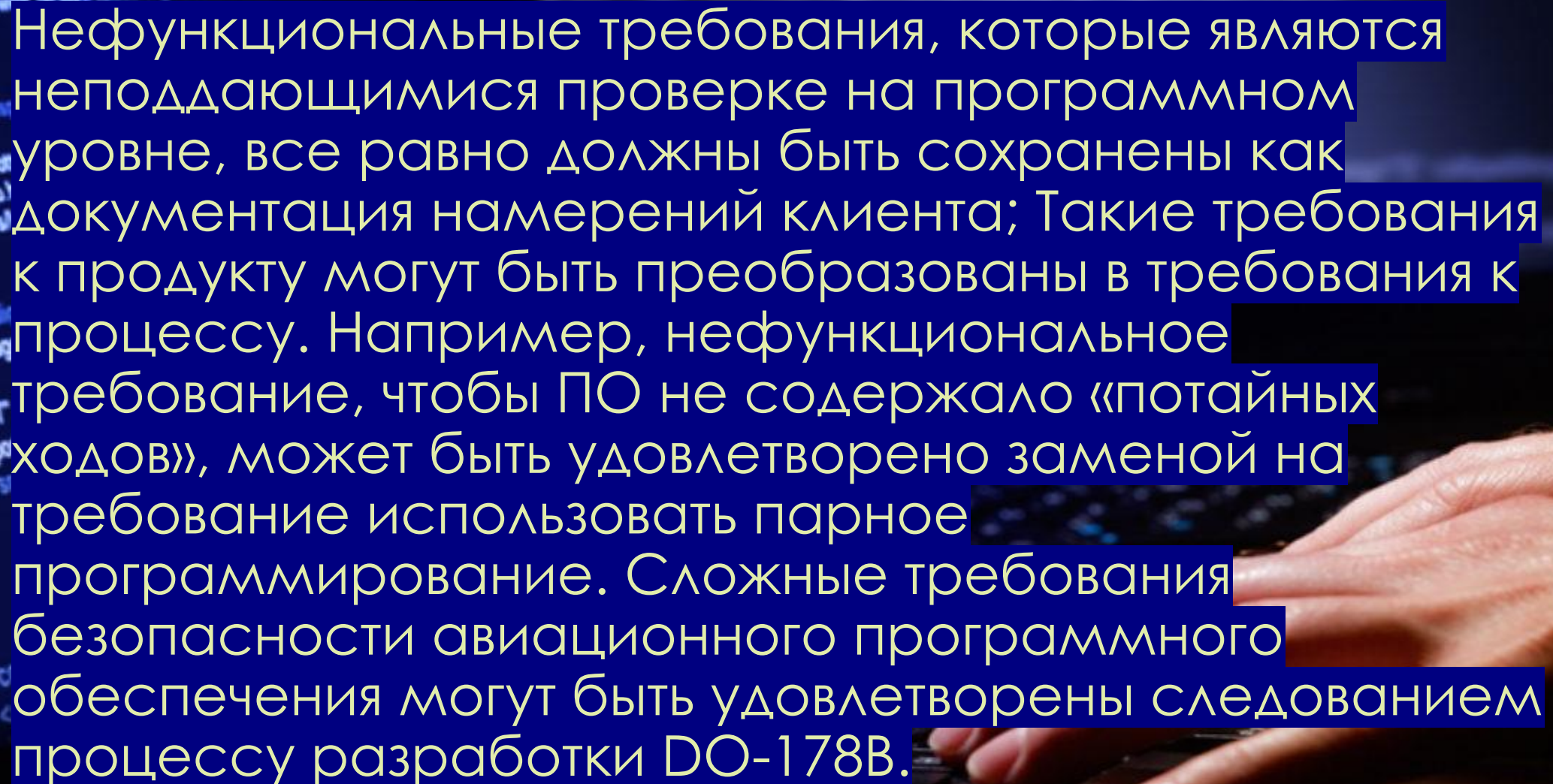
Требования могут выражаться в виде текстовых утверждений и графических моделей.

В классическом техническом подходе совокупность требований используется на стадии проектирования ПО. Требования также используются в процессе проверки ПО, так как тесты основываются на определенных требованиях.

Этапу разработки требований, возможно, предшествовало технико-экономическое обоснование, или концептуальная фаза анализа проекта. Фаза разработки требований может быть разбита на выявление требований (сбор, понимание, рассмотрение и выяснение потребностей заинтересованных лиц), анализ (проверка целостности и законченности), спецификация (документирование требований) и проверка правильности.

Проверка требований

Все требования должны быть поддающимися проверке. Наиболее общепринятая методика проверки – тесты. Если проверка тестами невозможна, тогда должен использоваться другой метод проверки (анализ, демонстрация, осмотр или обзор дизайна). Определённые требования, по своей сути, не являются поддающимися проверке. Они включают требования, которые говорят, что система *никогда* не должна или *всегда* должна показывать специфическое свойство. Надлежащее тестирование этих требований потребовало бы бесконечного цикла тестирования. Такие требования должны быть переопределены так, чтобы они стали поддающимися проверке. Как указано выше, все требования должны быть поддающимися проверке.



Нефункциональные требования, которые являются неподдающимися проверке на программном уровне, все равно должны быть сохранены как документация намерений клиента; Такие требования к продукту могут быть преобразованы в требования к процессу. Например, нефункциональное требование, чтобы ПО не содержало «потайных ходов», может быть удовлетворено заменой на требование использовать парное программирование. Сложные требования безопасности авиационного программного обеспечения могут быть удовлетворены следованием процессу разработки DO-178B.

Анализ требований

Требования склонны к проблемам двусмысленности, неполноты, и несогласованности. Их устранение на этапе разработки требований стоит на несколько порядков меньше, чем устранение этих те же проблемы на поздних стадиях разработки. Анализ требований направлен на решение данных проблем.

Существует технический компромисс между слишком неопределёнными требованиями и требованиями столь детализированными что они:

1. требуют много времени для разработки, иногда даже рискуют устареть к концу разработки
2. ограничивают возможные способы реализации
3. являются слишком дорогостоящими