МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«Гжельский государственный университет»** (ГГУ)

Колледж ГГУ

Специальность 09.02.07 Информационные системы и программирование

**Реферат**

**По дисциплине «Технология разработки программного обеспечения»**

**На тему «СТАНДАРТЫ, РЕГЛАМЕНТИРУЮЩИЕ РАБОТУ С ТРЕБОВАНИЯМИ»**

ВЫПОЛНИЛ:

Студент группы ИСП-О-18

Кукушкин А.А.

ПРОВЕРИЛА:

Прокуронова А. Ю.

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

пос. Электроизолятор,

2019 год

**СТАНДАРТИЗАЦИЯ В ПРОГРАММНОЙ ИНЖЕНЕРИИ**

В Толковом словаре по информатике В.И. Першикова и В.М. Савинкова понятие стандартизация определяется как принятие соглашения по спецификации, производству и использованию аппаратных и программных средств вычислительной техники; установление и применение стандартов, норм, правил и т.п.

**Стандартизация выполняет следующие функции:**

• упорядочивание объектов (продукции, работ, услуг, процессов), создаваемых людьми в разных странах;

• закрепление в нормативных документах оптимальных требований к упорядоченным объектам;

• установление правил применения этих нормативных документов.

На международном уровне стандартизация:

• обеспечивает взаимозаменяемость элементов сложной продукции;

• сближает уровень качества товаров, производимых в разных странах;

• содействует взаимообмену научно-технической информацией;

• содействует международной торговле.

• ускоряет научно-технический прогресс участников международных организаций. 

В процессе стандартизации вырабатываются нормы, правила, требования, характеристики, касающиеся объекта стандартизации, которые оформляются в виде нормативного документа.

Стандарт – это нормативный документ, разработанный на основе консенсуса, утвержденный признанным органом, направленный на достижение оптимальной степени упорядочения в определенной области. В стандарте устанавливаются для всеобщего и многократного использования

общие принципы, правила, характеристики рекомендательного характера, касающиеся различных видов деятельности или их результатов.

Таким образом, стандарты в разработке ПС важны по целому ряду причин. Основными из них являются:

1. Стандарты аккумулируют все лучшее из практической деятельности создания ПС и позволяют избежать повторения прошлых ошибок.

2. Стандарты предоставляют необходимую основу для процесса обеспечения качества: достаточно контролировать соблюдение стандартов.

3. Стандарты позволяют упорядочить процесс разработки, что делает разработку прозрачной и снижает затраты на обучение профессиональной деятельности при ротации кадров.

Среди всего многообразия стандартов принято выделять следующие основные типы стандартов:

Корпоративные стандарты разрабатываются крупными фирмами (корпорациями) с целью повышения качества своей продукции. Такие стандарты разрабатываются на основе собственного опыта и с учетом требований мировых стандартов. Корпоративные стандарты не сертифицируются, но являются обязательными для применения внутри корпорации.

Отраслевые стандарты действуют в пределах организаций некоторой отрасли (министерства). Например, СНИП – строительные нормы и правила. Разрабатываются с учетом требований мирового опыта и специфики отрасли. Являются, как правило, обязательными для отрасли. Подлежат сертификации.

Государственные стандарты (ГОСТы) принимаются государственными органами, в некоторых случаях имеют силу закона. Разрабатываются с учетом мирового опыта или на основе отраслевых стандартов. Могут иметь как рекомендательный, так и обязательный характер (стандарты безопасности). Для сертификации создаются государственные или лицензированные органы

сертификации.

Международные стандарты. Разрабатываются, как правило, специальными международными организациями на основе мирового опыта и лучших корпоративных стандартов. Имеют сугубо рекомендательный характер. Право сертификации получают организации (государственные и

частные), прошедшие лицензирование в международных организациях.

**Международные организации, разрабатывающие стандарты**

Международная организация по стандартизации (ИСО)

Сфера деятельности ИСО касается стандартизации во всех областях, кроме электротехники и электроники, относящихся к компетенции Международной электротехнической комиссии (МЭК). Некоторые виды работ выполняются совместными усилиями этих организаций. Кроме стандартизации ИСО занимается и проблемами сертификации.

Крупнейший партнер ИСО — Международная электротехническая комиссия (МЭК).

Объединенный технический комитет (JTC1) - Объединенный технический комитет предназначенный для формирования всеобъемлющей

системы базовых стандартов в области ИТ и их расширений для конкретных сфер деятельности.

**Национальные организации, разрабатывающие стандарты**

Государственный комитет РФ по стандартизации

Это федеральный орган исполнительной власти, осуществляющий межотраслевую координацию, а также функциональное регулирование в области стандартизации, метрологии и сертификации.

Американский национальный институт стандартов и технологий.

Разрабатывают федеральные стандарты авторитетные организации, аккредитованные Американским национальным институтом стандартов. Наиболее известные из них:

• Американское общество по контролю качества;

• Американское общество инженеров-механиков;

• Институт инженеров по электротехнике и электронике;

**РОССИЙСКИЕ И МЕЖДУНАРОДНЫЕ СТАНДАРТЫ ПРОГРАММНОЙ ИНЖЕНЕРИИ**

По мнению ряда специалистов, программная инженерия должна содержать аспекты программной разработки, управления программным обеспечением, организации и использования проектов. Другие включают в её состав вопросы разработки программного обеспечения,

проектирования, кодирования и тестирования вместе с использованием наилучших практических решений.



**СТАНДАРТЫ, РЕГЛАМЕНТИРУЮЩИЕ РАБОТУ С ТРЕБОВАНИЯМИ**

Среди основополагающих нормативных документов в области работы с требованиями можно выделить следующие.

1. Разработки IEEE:

• IEEE 1362 "Concept of Operations Document".

• IEEE 1233 "Guide for Developing System Requirements Specifications".

• IEEE Standard 830-1998, "IEEE Recommended Practice for Software Requirements Specifications"

• IEEE Standard Glossary of Software Engineering Terminology/IEEE Std 610.12-1990

• IEEE Guide to the Software Engineering Body of Knowledge (1) - SWEBOK®, 2004.

2. Отечественные ГОСТ:

• ГОСТ 34.601-90. Информационная технология. Автоматизированные системы. Стадии создания.

• ГОСТ 34.602-89. Информационная технология. Техническое задание на создание автоматизированной системы

• ГОСТ 19.201-78. Единая система программной документации. Техническое задание.

**ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ**

Артефакт — неотъемлемая часть результата и процесса выполнения программного проекта, реализованная в виде документации, программного кода (исходного или скомпилированного) или его части (например, модуля).

Программный проект — процесс реализации комплекса мероприятий, направленных на создание программного продукта.

Программный продукт — результат реализации программного проекта, обладающий заявленной функциональностью и потребительскими характеристиками.

Сопровождение программного обеспечения — комплекс работ, выполняемых после поставки программного продукта заказчику, направленный, главным образом, на исправление обнаруженных в

процессе эксплуатации дефектов и расширение возможностей программного обеспечения.

Тестирование — комплекс мероприятий, направленный на выявление дефектов в готовом программном обеспечении и/или его составляющих.

Требования к программному обеспечению — документ, отражающий, что должно делать разрабатываемое программное обеспечение.

|  |  |
| --- | --- |
| **Составляющие программного**  **продукта** | **Артефакты** |
| Требования | Спецификация требований к программному обеспечению |
| Программная архитектура | Проектная модель |
| Детальное проектирование | Исходный и объектный коды |
| Реализация |
| Тестирование | Тестовые процедуры и тестовые варианты |

Можно выделить две категории программных продуктов.

1. Программные продукты общего назначения (так называемое тиражное, «коробочное программное обеспечение») — автономные программные системы, созданные соответствующими компаниями для распространения на открытом рынке программных продуктов любому потребителю.

2. Программные продукты, созданные на заказ. Такие программные системы создаются по заказу конкретного потребителя или целевой категории; они весьма специфичны и обладают узконаправленной функциональностью.

**КЛАССИФИКАЦИЯ ПРОГРАММНЫХ ТРЕБОВАНИЙ.**

Пользовательские требования — описание на естественном языке (с использованием поясняющих диаграмм) функций, выполняемых системой, и ограничений, накладываемых на нее.

Системные требования — детальное описание системных функций и ограничений, которое иногда называют функциональной спецификацией. Она служит основой для заключения контракта между покупателем системы и разработчиками ПО.

Проектная системная спецификация — обобщенное описание структуры программной системы, которое будет основой для более детализированного проектирования системы и ее последующей реализации. Эта спецификация дополняет и детализирует спецификацию системных требований.

**Спецификация системных требований**

• Пользователь должен иметь возможность определять тип внешних файлов.

• Для каждого типа внешнего файла должно иметься соответствующее средство, применимое к этому типу файлов.

• Внешний файл каждого типа должен быть представлен соответствующей пиктограммой на дисплее пользователя.

• Пользователю должна быть предоставлена возможность самому определять пиктограмму для каждого типа внешних файлов.

• При выборе пользователем пиктограммы, представляющей внешний файл, к этому файлу должно быть применено средство, ассоциированное с внешними файлами данного типа.

**Функциональные требования.**

Эти требования описывают поведение системы и сервисы (функции), которые она выполняет, и зависят от типа разрабатываемой системы и от потребностей пользователей. Если функциональные требования оформлены как пользовательские, они, как правило, описывают систему в обобщенном виде. В противоположность этому функциональные требования, оформленные как системные, описывают систему максимально подробно, включая ее входные и выходные данные, исключения и т.д.

**Нефункциональные требования.**

Как следует из названия, нефункциональные требования не связаны непосредственно с функциями, выполняемыми системой. Они связаны с такими интеграционными свойствами системы, как надежность, время ответа или размер системы. Кроме того, нефункциональные требования могут определять ограничения на систему, например на пропускную способность устройств ввода-вывода или форматы данных, используемых в системном интерфейсе.

Все нефункциональные требования могут быть разбиты на три группы.

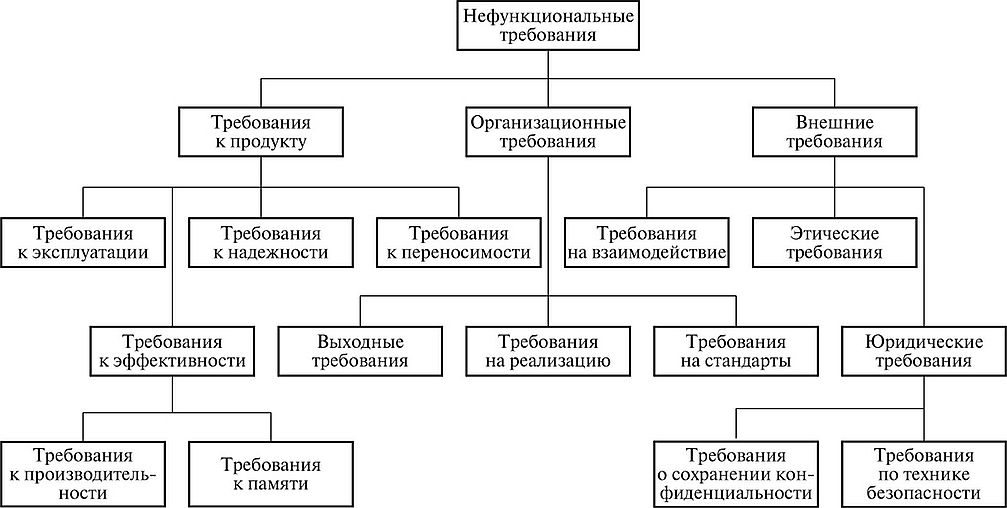
Требования к продукту. Описывают эксплуатационные свойства программного продукта.

Организационные требования отображают политику и организационные процедуры заказчика и разработчика ПО, включают стандарты разработки программного продукта, требования к реализации ПО (к языку программирования и методам проектирования), выходные требования,

которые определяют сроки изготовления программного продукта, и сопутствующую документацию.

Внешние требования учитывают факторы, внешние по отношению к разрабатываемой системе

и процессу ее разработки



**Таблица 1.2 Количественные показатели нефункциональных требований**

|  |  |
| --- | --- |
| **Показатель** | **Единицы измерения** |
| Скорость | Число выполненных транзакций в секунду; время реакции на действия пользователя; время обновления экрана |
| Размер | Килобайты; количество модулей памяти |
| Простота эксплуатации | Время обучения персонала; количество статей в справочной системе |
| Надежность | Средняя продолжительность времени между двумя  последовательными проявлениями ошибок в системе; вероятность  выхода системы из строя; коэффициент готовности системы |
| Устойчивость к сбоям | Время восстановления системы после сбоя; процент событий,  приводящих к сбоям; вероятность порчи данных при сбоях |
| Переносимость | Процент машинно-зависимых операторов; количество машинно-зависимых подсистем |

**Требования предметной области.** Эти требования отображают условия, в которых будет эксплуатироваться программная система.

**Пользовательские требования.** Пользовательские требования к системе должны описывать функциональные и нефункциональные системные требования так, чтобы они были понятны даже пользователю, не имеющему специальных технических знаний.

**Системные требования.** Системные требования представляют собой более детальное описание пользовательских требований и обычно служат основой для заключения контракта на разработку ПС.

**МЕТОДЫ РАБОТЫ С ТРЕБОВАНИЯМИ**

**РАЗРАБОТКА ТРЕБОВАНИЙ**

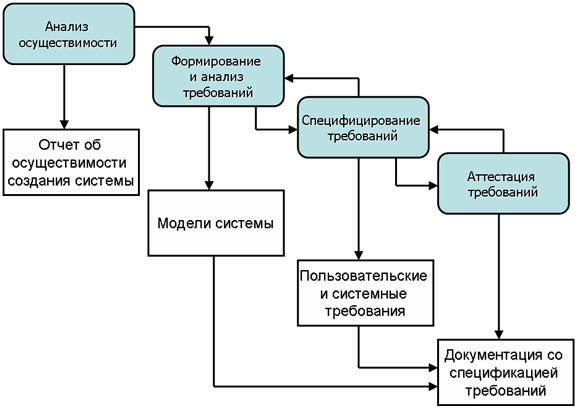
Разработка требований — это процесс, включающий в себя мероприятия, необходимые для создания и утверждения документа, содержащего спецификацию системных требований. Различают четыре основных этапа процесса разработки требований:

1. анализ технической осуществимости создания системы;

2. формирование и анализ требований;

3. специфицирование требований и создание соответствующей документации;

4. аттестация этих требований.



**Анализ осуществимости требований.**

Для новых программных систем процесс разработки требований должен начинаться с анализа

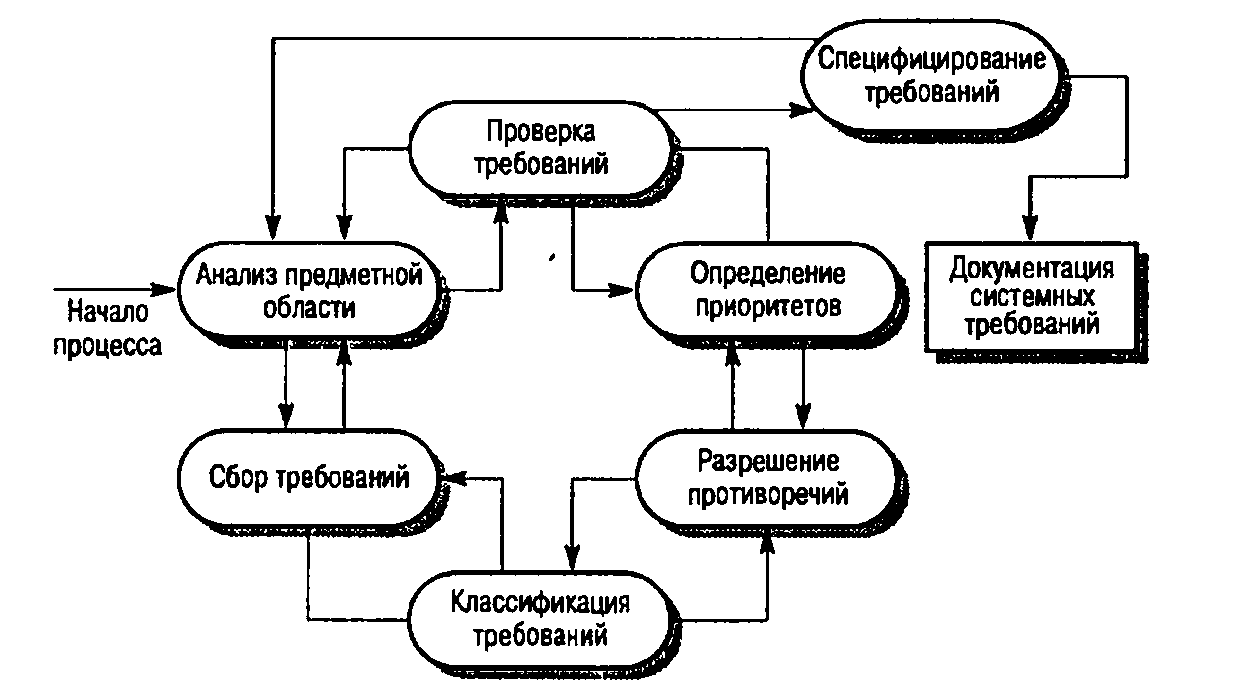
их осуществимости. Началом такого анализа является общее описание системы и ее назначения, а результатом анализа — отчет, в котором должна быть четкая рекомендация, продолжать или нет процесс разработки требований проектируемой системы.

**Извлечение требований.**

Вопросы сбора требований как с точки зрения организации процесса, так и определения источников, откуда поступают требования, являются первой стадией построения видения автоматизируемой проблемной области.

**Формирование и анализ требований.**

После выполнения анализа осуществимости и извлечения следующим этапом процесса разработки требований является формирование (определение) и анализ требований.



В общем случае процесс формирования и анализа требований состоит из нескольких этапов.

1. Анализ предметной области. Аналитики должны изучить предметную область, где будет эксплуатироваться система.

2. Сбор требований. Это процесс взаимодействия с лицами, формирующими требования. Во время этого процесса продолжается анализ предметной области.

3. Классификация требований. На этом этапе бесформенный набор требований преобразуется в логически связанные группы требований.

4. Разрешение противоречий. Без сомнения, требования многочисленных лиц, занятых в процессе формирования требований, будут противоречивыми. На этом этапе определяются и разрешаются противоречия такого рода.

5. Назначение приоритетов. В любом наборе требований одни из них будут более важны, чем другие. На этом этапе совместно с лицами, формирующими требования, определяется степень важности требований.

6. Проверка требований. На этом этапе определяется их полнота, последовательность и непротиворечивость.

Анализ требований включает в себя проверку требований по следующим показателям:

• корректность — отсутствие ошибок в формулировке требования;

• недвусмысленность — отсутствие неоднозначности в толковании требования; 27

• полнота набора требований — весь набор требований должен полностью описывать разрабатываемую систему;

• непротиворечивость набора требований — отсутствие во всей совокупности

требований, противоречащих друг другу;

• проверяемость — необходимо, чтобы каждое требование можно было в дальнейшем протестировать;

• трассируемость — наличие связи с требованиями, стоящими выше и ниже в общей иерархии (там, где указанные элементы иерархии существуют);

• понимаемость — требование должно быть описано на понятном для всех участников процесса разработки языке.

**СПЕЦИФИЦИРОВАНИЕ ТРЕБОВАНИЙ**

Итог разработки требований — задокументированное соглашение между клиентами и разработчиками о создаваемом продукте.

|  |  |
| --- | --- |
| **Система записи** | **Описание** |
| Структурированный  естественный язык | Использование стандартных форм и шаблонов для написания  спецификаций |
| Языки описания требований | Специальные структурированные языки (подобно языкам  программирования), где спецификация требований строится на  основе выбранной операционной модели системы |
| Графические нотации | Использование диаграмм и блок-схемы, дополненных  текстовыми пояснениями. Наиболее известные примеры такого описания: диаграммы структурного анализа и проектирования, метод описания вариантов использования |
| Математические нотации | Системы нотаций на основе математических концепций, таких  как теория конечных автоматов или теория множеств.  Формализованная однозначная запись системных требований. |