



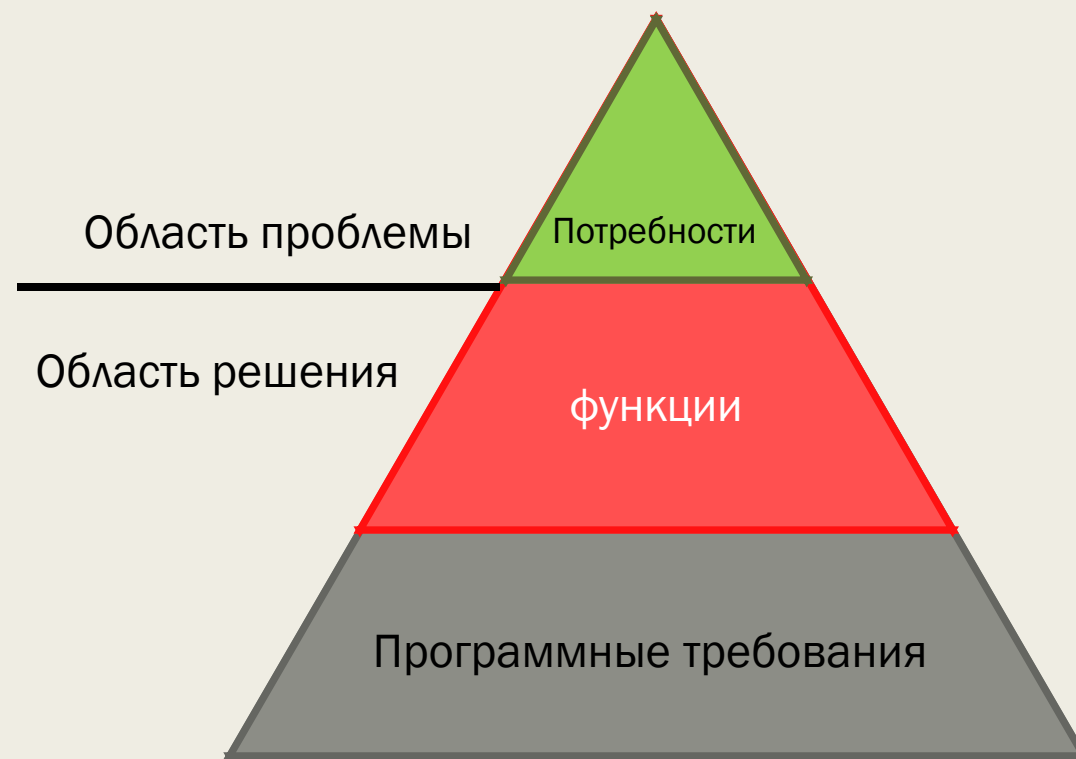
ЧТО ТАКОЕ ПО, КЛАССИФИКАЦИЯ ТРЕБОВАНИЙ, УРОВНИ ТРЕБОВАНИЙ

Митин Кирилл 20П-3



Для удобства работы с требованиями их обычно классифицируют по уровням и по видам. Для каждого уровня и для каждого вида требований существуют свои методы анализа и разработки. Давайте для начала разберёмся с делением требований по уровням.

На этой картинке представлена пирамидка из книги Леффингуэлла и Уидрига «Принципы работы с требованиями к программному обеспечению», которая уже упоминалась в предыдущей главе. Она описывает три уровня требований.



Любой продукт создается для решения каких-то проблем или создания каких-то новых возможностей. В книге предлагается, прежде всего, разделить область проблемы, для решения которой предназначен продукт, от области решения — то есть от того, как он решает эту проблему.

В области решаемой проблемы лежит первый уровень требований: потребности людей, для которых создаётся продукт. Это как непосредственные пользователи продукта, так и другие люди, в интересах которых он будет использоваться (например, владелец компании, приобретающей программу для использования своими сотрудниками).



После того, как мы выявили и описали потребности этих людей, мы можем определить, какие функции нужно реализовать в продукте, чтобы эти потребности удовлетворить. Перечень функций, которые должен выполнять продукт, представляет второй уровень требований. Этот уровень уже относится к области решения.

Наконец, определившись с перечнем функций, мы можем более детально описать, как они должны быть реализованы в продукте. Это третий уровень требований, которые в книге называются программными требованиями.

Основные термины программной инженерии (software engineering):

ПРОГРАММА - описание на языке программирования алгоритма решения задачи обработки информации.

ПРОГРАММИРОВАНИЕ - процесс создания программ.

АЛГОРИТМ – совокупность действий для решения задачи, точно предписывающая, как и в какой последовательности получить результат, однозначно определяемый исходными данными.

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ (ПРОГРАММНЫЙ КОМПЛЕКС) - комплекс взаимодействующих программ, описаний и инструкций, обеспечивающих автоматическое функционирование ЭВМ.

ПРОГРАММНЫЙ ПРОДУКТ - протестированное и сопровождаемое документацией программное обеспечение, которое могут использовать не только авторы.

ПРОЕКТ - объединение действий разработчиков, ориентированное на создание программного продукта. Действия ориентированы именно на продукт, а не на отдельный процесс (проектирование, тестирование и т. п.)

ПРОЦЕСС - набор взаимосвязанных действий, которые преобразуют исходные данные в выходные результаты для достижения определенных целей.

ЖИЗНЕННЫЙ ЦИКЛ – период времени между возникновением потребности в разработке программного средства и окончанием его применения.

ИНСТРУМЕНТАЛЬНЫЕ СРЕДСТВА - средства автоматизации технологических процессов.

ТЕХНОЛОГИЯ РАЗРАБОТКИ - образующая систему совокупность технологических процессов и инструментальных средств, ведущих к созданию или развитию программного средства.

ПРОГРАММНАЯ ИНЖЕНЕРИЯ — инженерная дисциплина, которая связана со всеми аспектами производства ПО от начальных стадий создания спецификации до поддержки системы после сдачи в эксплуатацию.

Стандартизация и обеспечение качества разработки ПП

- К настоящему времени разработано много *государственных, ведомственных, отраслевых и международных стандартов*, регламентирующих разработку программного обеспечения
- Стандарты отличаются друг от друга и содержат различные методы оценки качества.

Основные стандарты программной инженерии

- **ISO 12207:1995** Процессы жизненного цикла программных средств
- **ГОСТ 19.102-77** Единая система программной документации. Стадии разработки
- **ГОСТ 28195-89** Оценка качества программных средств. Общие положения
- **ISO 15504:1-9:1998** Оценка (аттестация) процессов жизненного цикла программных средств
- **ISO 9000** Системы менеджмента качества
- **CMM**: Модель зрелости процессов.

Внутрифирменные (внутрикорпоративные) стандарты

- регламентируют порядок оформления документации, приказов и технической литературы внутри компании, пользовательский интерфейс разрабатываемых приложений (например, запрет на использование некоторых элементов интерфейса), стиль программирования, спецификацию модулей, имена используемых переменных, таблиц баз данных (БД). Имеют узкую сферу полномочий (одна или несколько фирм), но играют большую роль, т.к. они абсолютно конкретны.

К О Н Е Ц