

ВНЕШНЕЕ ОПИСАНИЕ ПРОГРАММНОГО СРЕДСТВА

Назначение и роль в обеспечении качества программного средства. Определение требований к программному средству.

(продолжение)

Таким образом на основании анализа требований и в соответствии с ГОСТ Р ИСО/МЭК 12207 разработчик должен установить и документально оформить следующие требования к программным средствам:

- функциональные и технические требования, включая производительность, физические характеристики и окружающие условия, под которые должно быть создано ПО;
- требования к внешним интерфейсам ПО;
- квалификационные требования;
- требования безопасности;
- требования защиты;
- эргономические требования;
- требования к определению данных и базе данных;
- требования по вводу в действие и приемке поставляемого ПО на объектах эксплуатации и сопровождения;
- требования к документации пользователя;
- требования к эксплуатации ПО пользователем;
- требования к обслуживанию пользователя.

III. Специфицирование требований и создание соответствующей документации.

Результатом анализа требований является официальный документ для разработчиков ПО, который в соответствии с ГОСТ 19.102 — 77 называют техническим заданием или спецификацией требований к программному обеспечению.

Он содержит совокупность формализованных требований к ПО и используется как основа для разработки программной системы.

Состав и содержание требований к АС в соответствии со стандартом ГОСТ 34.602 — 89.

1. **Требования к системе в целом:** в них включаются требования к структуре и функционированию системы; требования к численности и квалификации персонала системы и режиму его работы; показатели назначения; требования к надежности; и многие другие требования.
2. **Требования к функциям (задачам),** выполняемым системой содержат:
 - по каждой подсистеме перечень функций и задач (в том числе обеспечивающих взаимодействие частей системы), подлежащих автоматизации;
 - при создании системы в две или более очереди — перечень функциональных подсистем, отдельных функций или задач, вводимых в действие в первой и последующих очередях;

- требования к качеству реализации каждой функции (задачи или комплекса задач), форме представления выходной информации, необходимой точности и времени выполнения, а также одновременности выполнения группы функций и достоверности выдачи результатов;

- перечень и критерии отказов для каждой функции, по которой задаются требования по надежности.

3. К требованиям к видам обеспечения относятся:

требования к математическому, информационному, лингвистическому, программному, техническому, метрологическому, организационному, методическому и другим видам обеспечения системы.

Для документирования требований, анализа системы и ее модификации могут использоваться модели системы. Модель является абстракцией системы, опускает детали и легче поддается анализу, чем любое другое представление этой системы. Она является связующим звеном между процессом анализа исходной задачи и процессом проектирования системы. Модели, как правило, используются в качестве иллюстративного материала для представления системы в различных аспектах:

Модели, как правило, используются в качестве иллюстративного материала для представления системы в различных аспектах:

1. Внешнее представление, когда моделируется окружение или рабочая среда системы (модели системного окружения)
2. Описание поведения системы, когда моделируется ее поведение модели системного окружения.(поведенческие модели)
3. Описание структуры системы, когда моделируется системная архитектура или структуры данных, обрабатываемых системой (модели данных, объектные модели, модели наследования).

Объектное моделирование соединяет в себе поведенческое и структурное моделирование. Такие структурные методы, как структурный анализ систем и объектно-ориентированный анализ, обеспечивают основу для детального моделирования системы как части процесса формулировки и анализа требований.

Для поддержки структурных методов существуют различные CASE-средства, представляющие собой пакет программных средств, который поддерживает отдельные этапы процесса разработки ПО.

В этот пакет могут входить:

- редакторы диаграмм потоков данных;
- средства проектирования, анализа, проверки ПО;
- средства генерирования отчетов;
- средства создания форм.

Для разработки диаграмм, поясняющих требования, может использоваться унифицированный язык моделирования (Unified Modeling Language, UML). Кроме того, могут применяться специализированные диаграммы, такие как диаграммы модели сущность-связь {entity-relationship diagram, ERD) и диаграммы потоков данных (Data Flow Diagrams, DFD). Для формализованного описания системных требований может использоваться язык описания программ (Program Description Language, PDL). В качестве PDL может использоваться любой универсальный язык программирования.

IV. Аттестация требований и процесс управления изменениями системных требований

Аттестация должна подтвердить, что разработанные на предыдущих этапах требования действительно определяют систему, необходимую заказчику. Проверка требований важна, так как ошибки в спецификации требований могут привести к последующей переделке системы, а следовательно, к большим материальным и временным затратам.

В соответствии с ГОСТ Р ИСО/МЭК 12207 — 99 разработчик должен оценить требования по следующим критериям (при этом результаты оценок должны быть документально оформлены).

- учет требований к системе и проекту системы;
- внешняя согласованность с требованиями к системе;
- внутренняя согласованность требований к ПО между собой;
- тестируемость требований;
- выполнимость программного проекта;
- возможность эксплуатации и сопровождения.

После успешного проведения оценки должно быть документально зафиксировано состояние требований к ПО. Существует ряд методов аттестации требований, которые можно использовать совместно или каждый в отдельности:

1. Обзор требований — это процесс просмотра системной спецификации для нахождения неточных описаний и ошибок. Обзор требований может быть неформальным и формальным. **Неформальный обзор** — это простое обсуждение требований с большим количеством лиц, участвующих в их формировании.

При **формальном обзоре** группа разработчиков должна обосновать причину включения каждого требования в спецификацию. При этом проверяется непротиворечивость требований и их полнота.

2. Прототипирование. Прототип системы может создаваться для уточнения неясных требований. Прототип предоставляется конечным пользователям и заказчику, которые могут провести с ним эксперименты, позволяющие уточнить требования.

3. Генерация тестовых сценариев. Тесты для требований разрабатываются как часть процесса аттестации, что позволяет обнаружить проблемы в спецификации. Если такие тесты сложно или невозможно разработать, то обычно это означает, что требования трудно выполнить, и поэтому необходимо их пересмотреть.