

## Тестирование, верификация и валидация – различия в понятиях

Несмотря на кажущуюся схожесть, термины «тестирование», «верификация» и «валидация» означают разные уровни проверки корректности работы программной системы. Дабы избежать дальнейшей путаницы, четко определим эти понятия (рис. 1.).

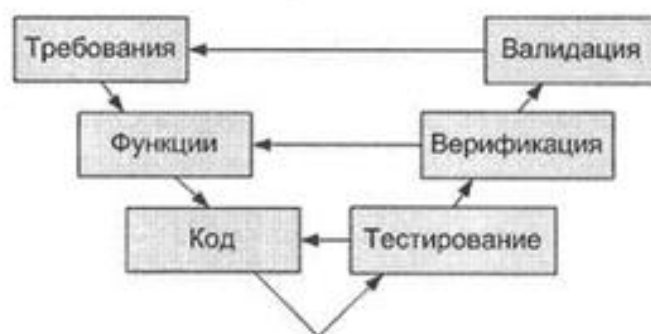


Рисунок 1. - Тестирование, верификация и валидация

**Тестирование программного обеспечения** – вид деятельности в процессе разработки, который связан с выполнением процедур, направленных на обнаружение (доказательство наличия) ошибок (несоответствий, неполноты, двусмысленностей и т.д.) в текущем определении разрабатываемой программной системы. Процесс тестирования относится в первую очередь к проверке корректности программной реализации системы, соответствия реализации требованиям, т.е. тестирование – это управляемое выполнение программы с целью обнаружения несоответствий ее поведения и требований.

**Верификация программного обеспечения** – более общее понятие, чем тестирование. Целью верификации является достижение гарантии того, что верифицируемый объект (требования или программный код) соответствует требованиям, реализован без непредусмотренных функций и удовлетворяет проектным спецификациям и стандартам. Процесс верификации включает в себя инспекции, тестирование кода, анализ результатов тестирования, формирование и анализ отчетов о проблемах. Таким образом, принято считать, что процесс тестирования является составной частью процесса верификации, такое же допущение сделано и в данном учебном курсе.

**Валидация программной системы** – целью этого процесса является доказательство того, что в результате разработки системы мы достигли тех целей, которые планировали достичь благодаря ее использованию. Иными словами, валидация – это проверка соответствия системы ожиданиям заказчика.

Если посмотреть на эти три процесса с точки зрения вопроса, на который они дают ответ, то тестирование отвечает на вопрос "Как это сделано?" или "Соответствует ли поведение разработанной программы требованиям?", верификация - "Что сделано?" или "Соответствует ли разработанная система требованиям?", а валидация - "Сделано ли то, что нужно?" или "Соответствует ли разработанная система ожиданиям заказчика?".

## Задачи и цели процесса верификации

Верификация – это процесс определения, выполняют ли программные средства и их компоненты требования, наложенные на них в последовательных этапах жизненного цикла разрабатываемой программной системы. Основная цель верификации состоит в подтверждении того, что программное обеспечение соответствует требованиям. Дополнительной целью является выявление и регистрация дефектов и ошибок, которые внесены во время разработки или модификации программы. Верификация является неотъемлемой частью работ при коллективной разработке программных систем. При этом в задачи верификации включается контроль результатов одних разработчиков при передаче их в качестве исходных данных другим разработчикам. Для повышения эффективности использования человеческих ресурсов при разработке, верификация должна быть тесно интегрирована с процессами проектирования, разработки и сопровождения программной системы. Заранее разграничим понятия верификации и отладки. Оба этих процесса направлены на уменьшение ошибок в конечном программном продукте, однако отладка – процесс, направленный на локализацию и устранение ошибок в системе, а верификация – процесс, направленный на демонстрацию наличия ошибок и условий их возникновения. Кроме того, верификация, в отличие от отладки – контролируемый и управляемый процесс. Верификация включает в себя анализ причин возникновения ошибок и последствий, которые вызовет их исправление, планирование процессов поиска ошибок и их исправления, оценку полученных результатов. Все это позволяет говорить о верификации, как о процессе обеспечения заранее заданного уровня качества создаваемой программной системы.

Сначала рассмотрим цели верификации. Основная цель процесса – доказательство того, что результат разработки соответствует предъявленным к нему требованиям. Обычно процесс верификации проводится сверху вниз, начиная от общих требований, заданных в техническом задании и/или спецификации на всю информационную систему до детальных требований на программные модули и их взаимодействие. В состав задач процесса входит последовательная проверка того, что в программной системе:

- общие требования к информационной системе, предназначенные для программной реализации, корректно переработаны в спецификацию требований высокого уровня к комплексу программ, удовлетворяющих исходным системным требованиям;
- требования высокого уровня правильно переработаны в архитектуру ПО и в спецификации требований к функциональным компонентам низкого уровня, которые удовлетворяют требованиям высокого уровня;
- спецификации требований к функциональным компонентам ПО, расположенным между компонентами высокого и низкого уровня, удовлетворяют требованиям более высокого уровня;
- архитектура ПО и требования к компонентам низкого уровня корректно переработаны в удовлетворяющие им исходные тексты программных и информационных модулей;
- исходные тексты программ и соответствующий им исполняемый код не содержат ошибок.

Кроме того, верификации на соответствие спецификации требований на конкретный проект программного средства подлежат требования к технологическому обеспечению жизненного цикла ПО, а также требования к эксплуатационной и технологической документации.

Цели верификации ПО достигаются посредством последовательного выполнения комбинации из инспекций проектной документации и анализа их результатов, разработки тестовых планов тестирования и тест-требований, тестовых сценариев и процедур и последующего выполнения этих процедур. Тестовые сценарии предназначены для проверки внутренней непротиворечивости и полноты реализации требований.

Выполнение тестовых процедур должно обеспечивать демонстрацию соответствия испытываемых программ исходным требованиям.

На выбор эффективных методов верификации и последовательность их применения в наибольшей степени влияют основные характеристики тестируемых объектов:

- класс комплекса программ, определяющийся глубиной связи его функционирования с реальным временем и случайными воздействиями из внешней среды, а также требования к качеству обработки информации и надежности функционирования;
- сложность или масштаб (объем, размеры) комплекса программ и его функциональных компонентов, являющихся конечными результатами разработки;
- преобладающие элементы в программах: осуществляющие вычисления сложных выражений и преобразования измеряемых величин или обрабатывающие логические и символьные данные для подготовки и отображения решений.