

## **Характеристики качества программного обеспечения и методы их оценки**

*Лозинин А.И.,*

*Шубинский И.Б.*

Разработка программного обеспечения – это деятельность, которая в настоящее время является одной из самых дорогостоящих.

Любые нарушения в технологическом процессе его создания могут привести к нежелательным результатам:

- удорожание программного продукта из-за увеличения сроков его разработки;
- из-за ошибок, не выявленных при тестировании:
- как минимум – снижение производительности программного продукта;
- как максимум – снижение безопасности систем, критичных по безопасности;
- ошибки, непонятные сообщения, недружественный интерфейс и небрежное документирование создают неудобства для пользователей, что приводит их к выбору более качественного продукта конкурента.

Соответствие требованиям функциональной безопасности, надежности и качества программных средств должно быть оценено в результате испытаний в процессе жизненного цикла, сертификационных испытаний и экспертизы.

Проведение любых видов испытаний программных средств железнодорожного транспорта осуществляется в соответствии с типовой методикой испытаний. В методике устанавливаются правила отбора образцов программных средств, общие подходы к определению состава документации, к определяемым характеристикам и метрикам качества программного обеспечения, последовательности и видам испытаний, рекомендации по тестированию (планы, протоколы, методы), к методам обработки результатов испытаний, а также требования к персоналу, проводящему испытания, и распределение ответственности за обеспечение и проведение испытаний. В методике приводятся также формы документов, таблицы показателей качества, условия проведения испытаний и др.

При проведении сертификационных испытаний, на основании требований к программному обеспечению и типовой методики испытаний, испытателем разрабатывается Программа и методика сертификационных испытаний.

Что является объектом испытаний?

На испытания представляется образец программного средства с сопроводительными документами, а также документация требований, материалы разработки, документы и материалы по испытаниям и эксплуатационная документация.

Отбор и идентификацию образцов производит Регистр сертификации на федеральном железнодорожном транспорте совместно с испытательным центром и представителем заявителя в соответствии с разработанной для этих

целей методикой отбора, идентификации и хранения образцов.

При приемке образцов контролируется состояние упаковки, наличие этикеток или надписей на упаковке, наличие комплекта поставки.

Идентификация образцов производится после контроля внешнего вида, маркировки, соответствия комплекта сопроводительной документации.

Образцы, не соответствующие требованиям идентификации, возвращаются заявителю.

Виды и последовательность проведения испытаний, определяемые характеристики

Испытания на функциональную безопасность

Определение угроз безопасности в существующей системе, их возможных причин и последствий, а также рекомендация действий по снижению вероятности их возникновения производится экспертами в начале проведения испытаний. Наличие угроз безопасности в системе (программном обеспечении) делает дальнейший анализ невозможным.

Эксперты рассматривают как функциональные аспекты существующей системы, так и то, как система должна работать на практике (включая человеческую деятельность и обслуживание).

Выявление потенциальных опасностей и определение для каждой части системы с различной степенью опасности условий применения (ни одной ошибки, допустимость ошибки и ее критичность, количество допустимых ошибок и др.).

Каждое условие применения или вид отказа рассматриваются с точки зрения их осуществимости, того, как они могут возникнуть, и возможных последствий (являются ли они опасностями), того, как их можно исключить, и оправдывают ли методы их исключения необходимые для их проведения издержки.

Анализ опасностей (вероятностная или количественная оценка риска) осуществляется для более подробного рассмотрения основных опасностей.

Анализ отказов по общей причине

Выявление потенциальных отказов в многокомпонентных системах или подсистемах, в которых присущее им преимущество избыточности может быть подорвано из-за одновременного возникновения одинаковых отказов во многих их частях.

Анализ отказов по общей причине должен обследовать систему на возможность появления таких потенциальных отказов по общей причине.

Используются следующие методы анализа отказов по общей причине: общий контроль качества; проверка архитектуры системы; верификация, тестирование и анализ реальных инцидентов с обращением к опыту эксплуатации аналогичных систем.

Сфера действия анализа распространяется за пределы аппаратных средств. Если даже для различных каналов избыточной системы используется разное программное обеспечение, в нём могут использоваться некоторые

общие подходы, что может привести к отказам по общей причине, например, дефекты общей спецификации.

Если отказы по общей причине происходят не точно в одно и то же время, можно предпринять меры предосторожности путём сравнения разных каналов, и это может позволить выявить отказ до того, как он станет общим для всех каналов. Анализ отказов по общей причине должен учитывать и этот метод.

#### Определение надежности программного обеспечения

Создается граф системы. Этот граф представляет статус системы в отношении её отказов (состояние отказа представляется узлами графа). Рёбрам графа между узлами, которые представляют случаи отказа или ремонта, присваиваются соответствующие весовые показатели интенсивности отказов или ремонтов. Предполагается, что изменение состояния  $N$  на последующее состояние  $N + 1$  не зависит от предыдущего состояния  $N - 1$ . Надо иметь в виду, что случаи отказов, состояния и интенсивности могут быть подробно представлены таким образом, что получается точное описание системы, например, выявленные и не выявленные отказы, проявление ещё больших отказов, и т.п.

Метод Маркова удобен для моделирования многокомпонентных систем, в которых уровень избыточности изменяется по времени из-за отказов и ремонта компонентов.

В сложных случаях, результаты могут быть подсчитаны путём моделирования графа на компьютере.

Необходимо смоделировать в виде схемы набор событий, которые должны происходить и условий, которые должны выполняться для успешного функционирования системы или выполнения задачи.

Целью анализа является представление маршрута, состоящего из блоков, линий и логических узлов. Маршрут начинается с одной стороны схемы и проходит через блоки и узлы до другой стороны схемы.

Блок представляет условие или событие, и маршрут может проходить через него, если условие правильное или событие имело место.

Когда маршрут подходит к узлу, он продолжается, если выполняется логика этого узла. Если он достигает вершины, он может продолжаться по всем выходящим линиям. Если существует по меньшей мере один проходящий через схему успешный маршрут, объект анализа функционирует нормально.

Надежность оценивают на основе характеристик многократных отказов и процессов восстановления программного обеспечения при его функционировании. Процесс восстановления определяют его процентной вероятностью за некоторое время, плотностью распределения времени восстановления и средним временем восстановления.

Объединение характеристик отказов и процессов восстановления образует критерии: наработка на отказ и коэффициент готовности. Значение коэффициента готовности соответствует доле времени полезной работы программного обеспечения на достаточно большом интервале, содержащем отказы и восстановления.

Расчет надежности программного обеспечения осуществляется по Методике расчета надежности программного обеспечения с помощью инструментальных программных средств.

Качество программного обеспечения

Характеристики качества программного обеспечения должны быть четко и однозначно изложены в техническом задании и (или) спецификациях требований.

Модель качества

Модель качества программного обеспечения содержит три основных компонента: структуру характеристик качества, метрики, критерии.

Характеристики качества программного обеспечения в типовой методике представлены иерархической структурой, которая содержит комплексные и единичные показатели (оценочные элементы).

На всех уровнях показателей для метрик характеристик качества принята единая шкала оценки от 0,1 до 1 (1 - высшее качество).

Для рейтинговых оценок, при сопоставлении нескольких программных продуктов, служит другая шкала - от 0,1 до 10.

Метрики качества для единичных показателей определяют в зависимости от метода измерения (оценки).

Метрики комплексных показателей определяют на основе связанных с ними показателей нижестоящего уровня с использованием вычислительных методов или специально разработанных правил.

Для характеристик качества, при вычислении средневзвешенных значений комплексных показателей и интегральной оценки качества, должны быть установлены коэффициенты значимости, сумма которых должна быть равна единице.

Правила принятия решений о качестве программного обеспечения определяют критерии, выработанные на основе требований и результатов измерения (оценки) показателей качества. Требования к качеству задаются базовыми (эталонными) значениями.

Типовая номенклатура показателей качества программного обеспечения

Иерархическая структура комплексных показателей качества программного обеспечения (с соответствующей нумерацией) представлена в таблице 1.

Вариант оценки значимости коэффициентов приводится в качестве примера.

Дальнейшая детализация комплексных показателей зависит от приложения, оцениваемого элемента конфигурации и задач оценки. Типовой вариант детализации в методике приводится в таблицах оценочных элементов по показателям качества.

**Таблица 1. Типовая структура показателей качества**

Код	Показатель	Значимость
1	<b>Функциональные возможности (Functionality)</b>	0,3
1.1	Пригодность (Suitability)	0,3
1.2	Правильность (Accuracy)	0,2
1.3	Способность к взаимодействию (Interoperability)	0,1
1.5	Защищенность (Security)	0,2
1.4	Согласованность (Compliance)	0,2
2	<b>Надежность (Reliability)</b>	0,3
2.1	Завершенность (Maturity)	0,4
2.2	Устойчивость к ошибкам (Fault tolerance)	0,3
2.3	Восстанавливаемость (Recoverability)	0,2
2.4	Согласованность (Compliance)	0,1
3	<b>Практичность (Usability)</b>	0,1
3.1	Понятность (Understandability)	0,2
3.2	Обучаемость (Lernability)	0,2
3.3	Простота использования (Operability)	0,3
3.4	Привлекательность (Attractiveness)	0,2
3.5	Согласованность (Compliance)	0,1
4	<b>Эффективность (Efficiency)</b>	0,1
4.1	Временная эффективность (Time behavior)	0,5
4.2	Ресурсоемкость (Resource behavior)	0,3
4.3	Согласованность (Compliance)	0,2
5	<b>Сопровождаемость (Maintainability)</b>	0,1
5.1	Анализируемость (Analysability)	0,2
5.2	Изменяемость (Changeability)	0,2
5.3	Стабильность (Stability)	0,2
5.4	Тестируемость (Testability)	0,3
5.5	Согласованность (Compliance)	0,1
6	<b>Мобильность (Portability)</b>	0,1
6.1	Адаптируемость (Adaptability)	0,2
6.2	Простота установки (Installability)	0,3
6.3	Сосуществование (Co-existence)	0,2
6.4	Взаимозаменяемость (Replaceability)	0,2

Код	Показатель	Значимость
6.5	Согласованность (Compliance)	0,1

#### Метрики показателей качества

Для оценки характеристик качества должны быть установлены метрики показателей. Для простоты расчета комплексных показателей и интегральной оценки качества, все метрики должны иметь одну и ту же область значений, соответствующую выбранной шкале:  $[0,1 \div 1]$  или  $[0,1 \div 10]$ .

В метриках применяют различные методы определения значений показателей: измерительный, регистрационный, органолептический, расчетный, экспертный, социологический, а также их сочетания по установленным правилам. При определении метрик следует руководствоваться принципами реализуемости, объективности и точности оценки метрик.

**Измерительный** метод основан на получении информации с использованием инструментальных средств.

**Регистрационный** метод основан на получении информации во время испытаний или функционирования программного обеспечения, когда регистрируют или подсчитывают определенные события (время и число сбоев или отказов, время передачи управления другим модулям, время начала и окончания работы).

**Органолептический** метод основан на использовании информации, полученной в результате анализа восприятия органов чувств (зрения, слуха) для определения показателей удобства применения.

**Расчетный** метод основан на использовании теоретических и эмпирических зависимостей (на ранних стадиях разработки), статистических данных, накапливаемых при испытаниях, эксплуатации и сопровождении программного обеспечения. При помощи расчетного метода определяют длительность вычислений, время реакции, показатели надежности, необходимые ресурсы.

**Экспертный** метод основан на определении значений показателей качества ПО экспертами, компетентными в решении данной задачи, на базе их опыта и интуиции.

Экспертный метод применяют в тех случаях, когда задача не может быть решена никаким другим из существующих способов или другие методы являются значительно более трудоемкими.

**Социологические** методы основаны на обработке специальных анкет - вопросников.

В таблице 2 представлены общие контрольные вопросы для испытаний и оценки качества программного обеспечения.

В качестве исходных данных для определения значений единичных показателей используют тексты программ, документацию и результаты тестовых прогонов программного обеспечения.

Таблица 2. Контрольные вопросы для оценки качества

Контрольные вопросы	Показатель качества	Код
Соответствует ли реализация функций программного обеспечения задачам пользователя? Насколько полно автоматизированы задачи пользователя?	Пригодность	1.1
Насколько функционирование программного обеспечения и получаемые результаты (число десятичных знаков, округление) соответствуют требованиям приложения?	Правильность	1.2
Насколько легко и эффективно осуществляется взаимодействие с другим программным обеспечением в среде пользователя?	Способность к взаимодействию	1.3
Насколько программное обеспечение соответствует деловой практике (терминологии, стандартным формам документов, логике решения задач)?	Согласованность	1.4
Обеспечивает ли программное обеспечение средства санкционирования доступа и выполняет ли требования приложения?	Защищенность	1.5
Функционирует ли система надежно в соответствии с требованиями поддержки приложения и технологичности, включая управление аномалиями (с оценкой средств управления аномалиями: определение ошибочных ситуаций системы и условий, требующих специальной обработки для подтверждения целостности системы; особенности восстановления и работы в условиях неполной работоспособности)?	Надежность	2
Какие требуются усилия для изучения программного обеспечения, подготовки входных и интерпретации выходных данных?  Применимо ли программное обеспечение в заданной операционной и поддерживающей среде?	Практичность	3
Функционирует ли система эффективно, минимизируя издержки, с минимальным временем отклика и максимальной производительностью системы (с оценкой использования данных, оценкой эффективности по памяти, оценкой выполнения итераций и проверкой требований технологичности)?	Эффективность	4
Какие усилия требуются для локализации ошибки?	Анализируемость	5.1
Насколько легко исправлять ошибки и устранять недостатки? Насколько легко расширять возможности или технологию путем развития существующих функций или добавления новых функций или данных?	Изменяемость	5.2

Контрольные вопросы	Показатель качества	Код
Тестируемо ли программное обеспечение и допускает ли детальную верификацию, проверку функциональных возможностей и операционную оценку путем тестирования, измерения, наблюдения и анализа? Насколько легко его тестировать?	Тестируемость	5.4
Насколько легко адаптировать программное обеспечение к конкретным условиям эксплуатации предназначенными для этого средствами?	Адаптируемость	6
Насколько легко переносить программное обеспечение для использования в другой среде (конфигурация КТС и/или среда программной системы)?	Мобильность	6.1
Насколько программное обеспечение соответствует стандартам или соглашениям, относящимся к мобильности?	Соответствие	6.2

Единичные показатели (оценочные элементы) в методике представлены в соответствующих таблицах с указанием методов определения значений метрик.

Оценочные элементы документации также сведены в таблицу.

Типовая модель оценивания программного обеспечения

Типовая структура характеристик и атрибутов качества для оценивания исполняемого программного обеспечения, применяемые метрики, соответствующие виды испытаний и методы измерений в методике представляются в таблицах.

Регистрация означает протоколирование выявляемых в процессе оценивания событий или проявлений ошибок (недостатков, нарушений требований).

Целесообразно вести каталог ошибок программного обеспечения и индикаторов качества, примерный и далеко не полный, их перечень представлен в таблице 3.

В качестве индикаторов в данном примере используют виды возможных ошибок, которые регистрируют в процессе испытаний.

Опыт использования таких индикаторов качества аккумулируют в каталоге дефектов продуктов программного обеспечения. В процессе накопления опыта испытаний в каталог вносят изменения.

**Таблица 3. Виды ошибок**

<b>Вид ошибки</b>
Ошибки Функциональных возможностей
<b>Ошибки Пригодности:</b> невыполнение функций, указанных в спецификации требований
Ошибки Правильности
Несоответствие программной документации согласно спецификации
Недостатки документации
Отсутствие файла комплекта поставки программного обеспечения
Отсутствие функции программного обеспечения
Отсутствие элемента интерфейса с пользователем
Противоречие в выполнении функций (по сравнению с техническим заданием и/или документацией)
Противоречие в выполнении алгоритмов
Ошибка в вычислениях
Противоречие в настройке программного обеспечения
Ошибки Взаимодействия
Ошибки при обращении к другому программному обеспечению
Ошибки при взаимодействии с программным обеспечением обмена информацией по телекоммуникационной сети
Ошибки при использовании данных другого программного обеспечения
Несовместимость форматов файлов и данных
Ошибки Защищенности
Недостатки средств управления доступом
Недостатки средств регистрации и учета
Недостатки защиты информации от искажений
Несоответствие требованиям средств криптографии
Недостатки защиты от воздействия компьютерных вирусов
Ошибки Согласованности
Несоответствие комплектности, содержания и оформления документации требованиям стандартов
Несоответствие форм входных и выходных документов стандартам области применения
Отсутствие лицензии на используемое покупное программное обеспечение
Ошибки Надежности
Ошибки Завершенности

<b>Вид ошибки</b>
Сбои и отказы при работе системы
Сбои и отказы при работе программного обеспечения
<b>Недостатки Устойчивости к ошибкам</b>
Недостатки средств контроля работоспособности и диагностирования программных и технических средств
Неполнота обработки ошибочных ситуаций
Отсутствие диагностического сообщения в случае сбоя или отказа
Неполнота контроля корректности, полноты и непротиворечивости входных, выходных данных и баз данных
Неполнота контроля непротиворечивости входных данных и баз данных
Отсутствие возможности функционирования в сокращенном объеме в случае ошибок или помех
<b>Ошибки Восстанавливаемости</b>
Отсутствие требований по восстановлению вычислительного процесса в случае сбоя операционной системы и оборудования
Отсутствие требований по восстановлению данных при отказах операционной системы и оборудования
Ошибка восстановления процесса в случае сбоев оборудования
Потери данных при отказах операционной системы и оборудования
Ошибка восстановления данных в случае их искажений или разрушения
Ошибка восстановления предшествующего состояния системы после повторного запуска программного обеспечения (возможность повторного старта с точки останова)
Несоответствие требованиям стандартов, соглашений, законов или других предписаний, связанных с надежностью
<b>Недостатки Практичности</b>
<b>Недостатки Понятности</b>
Недостатки, выявленные при экспертизе документации
Затруднения в понимании входных и выходных данных
Недостатки в средствах обеспечения помощи пользователю при затруднениях средствами функции подсказки
Противоречия в реализации интерфейса с пользователем
Противоречия в диагностике системы
<b>Недостатки Обучаемости</b>
Затруднения при освоении программного обеспечения по документации
Затруднения при освоении программного обеспечения на контрольном

<b>Вид ошибки</b>
примере
<b>Затруднения при использовании</b>
Затруднения при установке программного обеспечения
Затруднения при загрузке и запуске программного обеспечения
Затруднения при управлении решением задач
Неудобство ввода данных
Недостатки управления степенью подробности запрашиваемой информации
Отсутствие значений по умолчанию
Затруднения восприятия выходных данных
Отсутствие диагностики ошибок пользователя
Наличие непонятных сообщений
Наличие непонятных состояний или поведения системы
Недостатки предотвращения ошибок пользователя
Отсутствие средств диагностики некорректных и аварийных ситуаций
Отсутствие необходимой помощи пользователю при затруднениях средствами функции подсказки
Недостаточность полученной информации для продолжения работы
Недостатки обеспечения выполнения программой разработанных рабочих процедур
Не нравится пользователю
Противоречия в реализации интерфейса с пользователем
Противоречия в диагностике системы
Недостатки Эффективности
Несоответствие требованиям по времени установки программного обеспечения
Несоответствие требованиям по времени реакции и ответов, обработки данных и т.п.
Несоответствие требованиям по времени рестарта
Замечания по трудоемкости подготовки входных и обработке выходных данных
Замечания по затратам других видов ресурсов
Несоответствие требованиям стандартов или соглашений, связанных с эффективностью
Недостатки Сопровождаемости
Недостаточно высокий уровень используемого языка

<b>Вид ошибки</b>
Отсутствие схемы иерархии модулей программного обеспечения
Низкая читабельность кода
Недостаточно комментариев
Отсутствие системного подхода при выделении модулей
Слишком большие модули
Недостатки средств диагностики некорректных и аварийных ситуаций
Недостаточность средств регистрации процесса выполнения
Высокая трудоемкость внесения изменений
Низкий уровень параметризации
Отсутствие регистрации изменений
Низкая локальность модификации (большое число модулей и объем изменений)
Вторичные дефекты после модификации
Недостаточность требований к тестированию программного обеспечения
Отсутствие описания методов испытаний программного обеспечения
Невозможность разработки объективного и наглядного теста для определения того, что требование выполняется
Невозможность использования инструментальных средств тестирования
Несоответствие требованиям стандартов или соглашений, связанных с сопровождаемостью
Несоответствие требованиям к исходным текстам программ
<b>Ошибки Адаптируемости</b>
Несоответствие требованиям по адаптируемости
Выявленные ограничения по адаптации на возможных структурах информации для настройки
Неоправданно высокие требования к уровню квалификации для установки программного обеспечения
Высокая трудоемкость подготовки к установке программного обеспечения
Низкий уровень автоматизации установки программного обеспечения
Неоправданно высокие затраты времени на установку программного обеспечения
Неспособность программного обеспечения работать одновременно с другим независимым программным обеспечением, разделяя общие ресурсы
Неспособность программного обеспечения работать вместо другого указанного программного обеспечения в среде заменяемого
Несоответствие требованиям стандартов или соглашений, связанных с

<b>Вид ошибки</b>
мобильностью: <ul style="list-style-type: none"><li>● по использованию стандартных протоколов связи;</li><li>● по использованию стандартных интерфейсных подпрограмм</li></ul>

Контрольные вопросы для аудита проекта

Для аудита проекта необходимо достоверно установить, что выполняются условия<sup>1</sup>:

разработанные программные продукты (такие, как элемент программного обеспечения) отражают проектную документацию;

требования к приемочному анализу и испытаниям (тестированию), предписанные документацией, адекватны для приемки программных продуктов;

тестовые данные соответствуют их спецификации;

программные продукты успешно прошли испытания и соответствуют своим спецификациям;

протоколы по испытаниям правильны, и проблемы расхождений между фактическими и ожидаемыми результатами решены;

документация пользователя соответствует установленным стандартам;

работы проведены в соответствии с применимыми требованиями, планами и договором;

затраты и расписания придерживаются утвержденных планов.

Условия проведения испытаний

В условиях проведения испытаний указывается, каким образом должны проводиться испытания на арендованном компьютерном оборудовании (например, категорически не допускается безвозмездное использование оборудования, площадей, трудовых и иных ресурсов заказчика), условия проведения части испытаний субподрядными организациями и как должны проводиться испытания в полевых условиях (диспетчерский пункт, локомотив, контейнерная площадка и др.).

Методика испытаний программной документации

Экспертиза документации проводится в несколько шагов.

1) Проверка комплектности документации на соответствие требованиям стандартов ЕСПД, а также полноты комплекта документации в соответствии со спецификацией. Критерий соответствия техническим требованиям - наличие каждого из обязательных документов, указанных в стандарте ГОСТ 19.101 и каждого документа, указанного в спецификации.

2) Экспертиза по каждому документу в отдельности на соответствие требованиям стандартов ЕСПД с регистрацией замечаний по характеристикам

---

<sup>1</sup> в зависимости от целей аудита отбор условий осуществляют из приведенного списка

оценочных элементов документации.

На каждый документ заполняют форму, аналогичную таблице оценочных элементов документации, с указанием вида документа. Результаты оценки проставляют в таблице (число в интервале  $[0,1 \div 1]$ ). Общую оценку документа проставляют исходя из оценок по единичным характеристикам.

3) Интегральную оценку каждой характеристики определяют с учетом важности документов исходя из области применения программного обеспечения. Интегральная оценка полноты документации должна учитывать наличие всей информации, необходимой для эксплуатации программного обеспечения, в представленном комплекте документов.

Программная документация, в общем виде, должна удовлетворять следующим требованиям:

- соответствие требованиям стандартов единой системы программной документации;
- понятность документации;
- полнота документации;
- возможность освоения программного обеспечения по документации;
- возможность освоения программного обеспечения на контрольном примере;
- легкость установки и запуска программного обеспечения;
- понятность входных и выходных данных;
- наличие описания структуры функций программного обеспечения;
- соответствие функций программного обеспечения требованиям технического заданию;
- наличие описания функций программного обеспечения;
- отсутствие противоречий в реализации интерфейса с пользователем;
- отсутствие противоречий в диагностике системы;
- обеспечение помощи пользователю при затруднениях средствами функции подсказки;
- достаточность документации для ввода программного обеспечения в эксплуатацию;
- правильность документации;
- приемлемость уровня технического исполнения документации;
- наличие краткой аннотации программного обеспечения;
- наличие описания решаемых задач;
- наличие описания ограничений по применению;
- наличие описания алгоритмов;
- наличие описания пользовательских интерфейсов;
- наличие описания входных и выходных данных;

- наличие описания диагностических сообщений;
- наличие описания характеристик программного обеспечения;
- наличие описания оборудования, необходимого для функционирования программного обеспечения;
- наличие описания программной среды функционирования программного обеспечения и других используемых программ;
- наличие оглавления;
- соответствие оглавления содержанию документации;
- грамматическая правильность изложения в документации;
- отсутствие противоречий;
- отсутствие неправильных ссылок;
- ясность формулировок и описаний;
- отсутствие неоднозначных формулировок и описаний;
- правильность использования терминов;
- краткость, отсутствие лишней детализации;
- единство формулировок;
- единство обозначений;
- отсутствие ненужных повторений;
- наличие нужных объяснений;
- приемлемость уровня стиля изложения;
- ясность логической структуры;
- наличие перекрестных ссылок;
- наличие непрерывной нумерации страниц документов;
- отсутствие незаконченных разделов, абзацев, предложений;
- наличие всех рисунков, формул, таблиц;
- наличие всех строк и примечаний;
- логический порядок частей внутри главы и другие требования.

### Тестирование программного обеспечения

Тестирование программного обеспечения в процессе испытаний проводится с целью получения данных для оценки его динамических характеристик.

#### Разработка плана тестирования

Разработка плана тестирования проводится на основе анализа программной документации, в частности, характеристик качества.

План тестирования - это документированный, детализированный и структурированный процесс с точным описанием, как, в какой среде и при

каких условиях должно тестироваться прикладное программное обеспечение.

Кроме того, план должен описывать общую концепцию и методологию тестирования, включая проявление, частоту и методы описания обнаруженных ошибок, а также график тестирования.

Для каждого тестируемого элемента необходимо проанализировать:

его прикладное значение (как воспринимает этот элемент пользователь и насколько он важен для успеха программного обеспечения);

сложность (высокая сложность может оказаться дефектом);

архитектуру (внесение изменений может оказаться затруднительным).

Сразу после завершения тестирования нужно проанализировать весь процесс и внести поправки для последующих проектов.

Протоколы тестирования

В процессе тестирования необходимо детально документировать выявленные проблемы так, чтобы тесты можно было повторить. Протоколы тестирования должны включать следующую информацию:

среда тестирования;

начальные условия каждого теста;

каждое действие, предшествующее тесту;

критерий успеха для каждого теста.

Протоколирование ошибки должно содержать пошаговое описание действий, вызвавших ее проявление и предшествующих событий, чтобы ее можно было повторить и повторно протестировать после исправления. Рекомендуется выдерживать доброжелательный и конструктивный стиль протокола, что стимулирует партнерские отношения с разработчиками для достижения общих целей. По обнаружении и осмыслении дефектов или отказов в процессе тестирования, в рабочий журнал рекомендуется включать предложения по их устранению или усовершенствованию программного обеспечения (они не включаются в обобщенные рабочий журнал испытаний, но будут полезны разработчикам).

Под ошибкой в широком смысле понимается неправильность, погрешность или неумышленное, невольное искажение объекта или процесса. При этом подразумевается, что известно правильное или неискаженное эталонное состояние, к которому относится ошибка. Исходя из специфики программного обеспечения, в связи с неполной определенностью и неполной корректностью эталонов, в состав ошибок включаются и ситуации, когда программное обеспечение соответствует формализованным эталонам, однако нарушаются некоторые правила здравого смысла, формально не предусмотренные эталонами. В соответствии с этим считается, что если программное обеспечение не выполняет того, что пользователь от нее ожидает, то в ней имеется ошибка. В результате наличие ошибки становится функцией как самого программного обеспечения, так и неформализованных требований его пользователей.

Методы тестирования программных средств

При проведении испытаний для оценки качества программного обеспечения проводится динамическое тестирование на основе стратегии

"черного ящика". Выделяются следующие виды тестов:

Стохастическое тестирование - на случайных наборах тестовых данных.

Детерминированное тестирование - выполнение программы на ЭВМ с использованием специальным образом подобранных наборов тестовых данных (функциональное, по входу-выходу). Контролируется каждая комбинация исходных данных и соответствующие результаты, а также каждое утверждение в спецификации тестируемой программы.

Тестирование в реальном времени - обработка исходных данных с учетом времени их поступления, длительности и приоритетности обработки, динамики использования ресурсов и взаимодействия с другими программами.

Категории тестов

Испытания на соответствие программного обеспечения документации требований и/или программной документации:

полноты решения функциональных задач при типовых исходных данных;  
вычислений и преобразования данных;

функционирования программного обеспечения в критических ситуациях по условиям и логике решения задач;

определение надежности, в частности, эффективности защиты от искажений входных данных и управляющих воздействий, а также оценки эффективности защиты от аппаратных сбоев и не выявленных ошибок.

Оценка удобства установки, подготовки рабочей версии, эксплуатации, интерфейса пользователя.

Проектирование тестовых наборов данных

Тесты представляют собой частные реализации взаимосвязей исходных и результирующих данных. Описание предполагаемых значений результатов тестовых прогонов должно быть необходимой частью тестового набора данных.

Тестовые наборы выбираются на основе анализа входных функциональных спецификаций. Стратегия "черного ящика" реализуется методами эквивалентного разбиения, анализа граничных значений и функциональных диаграмм. Каждый из этих методов должен быть подробно описан в типовой методике испытаний.

Регистрация полученных данных о корректности и надежности программного обеспечения

Для оценки корректности нужны данные о выявленных ошибках, для оценки надежности необходимо знать характеристики многократных отказов и восстановлений в процессе функционирования программного обеспечения.

По результатам тестирования составляют протокол обнаруженных ошибок, а также протокол отказов и восстановлений.

Для определения надежности, данные об отказах и интервалах между отказами должны быть получены при испытании программного обеспечения в реальных условиях функционирования. Можно также использовать данные опытной эксплуатации, полученные методом анкетирования или регистрации

обращений пользователей (первое предпочтительно).

Методика обработки результатов испытаний

Обработка рабочего журнала

По результатам испытаний составляют перечень ошибок для единичных характеристик:

обобщенные статистические данные для повторяющихся ошибок, где это возможно и целесообразно;

сводная итоговая таблица ошибок, где полученные данные об ошибках группируются в соответствии с их взаимосвязями с единичными характеристиками.

Исходными данными для расчетов являются:

результаты регистрации времени реакции, выполнения функций, задач, запросов;

сводная итоговая таблица ошибок, где полученные данные об ошибках группируют в соответствии с их взаимосвязями с единичными показателями надежности.

Оценка единичных показателей и соответствия требованиям

Оценка единичных характеристик качества и соответствия требованиям проводится группой экспертов коллегиально.

По каждой единичной характеристике экспертным путем оценивают значимость совокупности выявленных ошибок для решения прикладных задач:

0 – незначительная (применяется и для случая отсутствия ошибок);

1 – компенсируемая (технологически, методически или организационно);

2 – существенная, но не приводящая к отказу при выполнении критичных функций управления движением или грубым ошибкам пользователей;

3 – фатальная, приводящая к отказу при выполнении критичных функций или фатальным ошибкам пользователей.

На следующем шаге в итоговую таблицу ошибок заносят полученные оценки значимости, а в таблицу результатов испытаний (для единичных характеристик), аналогичной таблице 1 - оценки результатов испытаний, выполнения требований и соответствия нормативно-технической документации. Оценка характеристики - число в интервале  $[0, 1 \div 1]$ , а оценки выполнения требований и соответствия нормативно-технической документации получают по правилам, представленным в таблице оценочных элементов показателя «Практичность», с учетом **критерия принятия решения о соответствии** нормативно-техническим документам:

единичная характеристика получает оценку «Соответствует», если выявленные недостатки не приводят к отказу при выполнении критичных функций или грубым ошибкам пользователя, препятствующим выполнению работы.

**Таблица. Правила оценки по единичным характеристикам**

Значимость/Код		Оценка выполнения требований	Оценка соответствия
Незначительная	0	Выполняются	Соответствует
Компенсируемая	1	Выполняются	Соответствует
Существенная	2	Выполняются неполно	Соответствует
Фатальная	3	Не выполняются	Не соответствует

#### Оценка комплексных и интегральных показателей

Оценка комплексных характеристик проводится группой экспертов коллегиально. Исходными данными для оценки являются результаты оценки единичных характеристик качества и соответствия требованиям. Оценку (число в интервале  $[0,1 \div 1]$ ) проставляют в правой колонке соответствующей строки формы, частично заполненной на предыдущем этапе.

В случае если при обсуждении эксперты смогут установить весовые коэффициенты, комплексные показатели получают расчетным методом весовой свертки. В противном случае, оценку (число в интервале  $[0,1 \div 1]$ ) выставляют в результате обсуждения.

Расчет интегрального показателя качества выполняют по формуле весовой свертки после уточнения группой экспертов весовых коэффициентов (значимости) комплексных характеристик качества.

Правила оценки соответствия нормативно-техническим документам по комплексным и интегральному показателям:

если хотя бы одна единичная характеристика данного комплексного показателя имеет оценку «Не соответствует», такую же оценку получает комплексный показатель;

если хотя бы один комплексный показатель имеет оценку «Не соответствует», такую же оценку получает интегральный показатель.

Полученные оценки заносят в итоговую таблицу результатов испытаний.

По результатам выполнения данного и последующих этапов испытаний оформляется протокол с подписями всех участвовавших экспертов.

#### Требования к персоналу испытательного центра

Постоянный штатный персонал испытательного центра должен обладать опытом работы, компетенцией и квалификацией, обеспечивающей полноценное его функционирование.

Персонал должен быть аттестован в установленном порядке на право проведения испытаний в области аккредитации.

Персонал, временно привлекаемый к работам испытательного центра на основе трудовых соглашений, договоров и контрактов, подбирается по принципам отсутствия заинтересованности в получении определенного результата выполняемых работ.

Персонал испытательного центра должен постоянно повышать свои

знания и совершенствовать навыки, необходимые для качественного выполнения возложенных на него должностных обязанностей. Испытательный должен вести график повышения квалификации и аттестации штатного персонала.

Распределение ответственности за обеспечение и проведение испытаний

Все сотрудники должны осуществлять свою деятельность в соответствии с должностными инструкциями. Специалисты испытательного центра заполняют декларацию и руководствуются в своей работе методиками испытаний программного обеспечения, руководством по качеству, должностными инструкциями и требованиями, предъявляемыми к специалистам.

Документацией, определяющей ответственность сотрудников испытательного центра, являются:

- нормативно-технические документы;
- должностные инструкции сотрудников, в которых устанавливаются их функции, обязанности, права и требования к качеству выполняемых работ;
- Положение о ведущем испытания.