

УДК 004.415.53.

## ОСНОВНЫЕ МЕТОДИКИ И ХАРАКТЕРИСТИКИ ПРОВЕДЕНИЯ ТЕСТИРОВАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

**Борисов Евгений Алексеевич**

магистрант 2 курса, факультет «Политехнический институт»

**Теплов Александр Владимирович**

магистрант 2 курса, факультет «Политехнический институт»

научный руководитель: **Бодров Андрей Сергеевич**

кандидат технических наук, доцент

Орловский государственный университет имени И.С. Тургенева, г. Орел

**Аннотация.** Тестирование программного обеспечения является одним из основных процессов поддержания качества информационных продуктов. Существуют выработанные методики проведения качественного тестирования программного продукта. В статье рассмотрены основные методики проведения тестирования, их характеристики и случаи применения

**Ключевые слова:** тестирование программного обеспечения, метод черного ящика, метод белого ящика

Тестирование программного обеспечения является одним из основных процессов поддержания качества информационных продуктов. Существуют выработанные методики проведения качественного тестирования программного продукта, но прежде чем переходить непосредственно к методам тестирования рассмотрим принципы выбора техники тестирования.

При выборе техники тестирования учитываются следующие факторы:

- тип компонента или системы;
- компонент или сложность системы;
- нормативные стандарты;
- требования заказчика или контракты (договоры);
- уровни риска;
- типы риска;
- доступная документация;
- знания и навыки персонала;
- доступные инструменты;
- время и бюджет;
- модель жизненного цикла разработки программного обеспечения;

- ожидаемое использование программного обеспечения;
- типы дефектов, ожидаемых в компоненте или системе.

В зависимости от указанных факторов выбирают одну из техник тестирования. Рассмотрим классификацию техник тестирования. Техники тестирования делят на следующие методы:

- методы черного ящика (Black-box);
- методы белого ящика (White-box);
- техника тестирования основанная на опыте (Experience-based)[1].

Ознакомимся подробнее с каждой из перечисленных техник тестирования.

Методы черного ящика (Black-box) - это поведенческие или основанные на поведении методы, которые берут за основу анализ соответствующей тестовой базы (например, документов формальных требований, спецификаций, вариантов использования, пользовательских историй или бизнес-процессов). Данные методы применимы как для функционального, так и нефункционального тестирования. Они концентрируются на «входных» и «выходных» значениях параметров тестируемой системы.

Методы эквивалентного разбиения (построения классов эквивалентности) делят данные на разделы таким образом, что все данные должны обрабатываться одинаково.

В данном методе выделяют следующие значения:

- допустимые (валидные) значения;
- недопустимые (невалидные) значения.

Метод анализа граничных значений (Boundary Value Analysis) является расширением эквивалентного разбиения, но может использоваться только при упорядочении раздела, состоящего из числовых или последовательных данных. Минимальные и максимальные значения (или первое и последнее значения) раздела являются его граничными значениями. Граничное покрытие вычисляется, как отношение числа протестированных граничных значений к общему количеству граничных значений.

Метод составления таблиц решений тестирования является одним из методов комбинаторного тестирования, для определения к каким результатам приводят разные комбинации условий. Таблицы решений - хороший способ записать сложные бизнес-правила, которые должна реализовать система. При создании таблиц решений тестировщик определяет условия (input) и результирующие действия (output) системы, которые формируют строки таблицы, обычно с условиями сверху и действиями внизу.

Метод тестирования переходного состояния основан на том, что компоненты или системы могут по-разному реагировать на событие в зависимости от текущих условий или

предыдущей истории (например, событий, которые произошли после инициализации системы).

Метод тестирования вариантов использования основан на том, что тесты могут быть получены из вариантов использования, которые представляют собой особый способ проектирования взаимодействий с программными элементами, включая требования к программным функциям, представленным вариантами использования. Варианты использования связаны с действующими лицами (пользователи, внешнее оборудование или другие компоненты или системы) и субъектами (компонент или система, к которой применяется сценарий использования).

Методы белого ящика (White-box) являются структурными методами или основанные на структуре тестируемого объекта. Такие методы обычно выстраиваются на анализе архитектуры, детального проектирования, внутренней структуры или кода тестового объекта, и концентрируются на структуре и обработке данных внутри объекта испытания.

Методы основанные на опыте (Experience-based) используют опыт разработчиков, тестировщиков и пользователей; для разработки, реализации и выполнения тестов. Данные методы часто сочетаются с методами белого и черного ящика.

Метод угадывание ошибки (error guessing) - это метод, используемый для прогнозирования возникновения ошибок, дефектов и отказов, основанный на знаниях тестировщика. В этом случае учитываются знания о том:

- как приложение работало в прошлом;
- какие типы ошибок, в каком функционале имеют тенденцию делать разработчики; сбои, которые произошли в других приложениях [2].

Рассмотрим метод тестирования на основе чек-листа (checklist-based). Тестировщики занимаются дизайном, имплементацией и выполняют тесты для покрытия условий тестирования из чек-листа. В рамках анализа тестировщики создают новый чек-лист, расширяют существующий, или используют старый чек-лист без изменений. Чек-листы могут быть построены на основе опыта, знаний о том, что важно для пользователя, или понимания того, почему и как происходит сбой в работе программного обеспечения. Чек-листы могут быть созданы для поддержки различных типов тестов, включая функциональное и нефункциональное тестирование. В отсутствие подробных тест кейсов, тестирование на основе чек-листов может дать рекомендации и определенную степень согласованности. Поскольку это списки высокого уровня, возможна некоторая изменчивость в реальном тестировании, что может привести к потенциально большему покрытию, но меньшей повторяемости.

### **Список литературы:**

1. Синицын С. В., Налютин Н. Ю. Верификация программного обеспечения. — М.: БИНОМ, 2008. — 368 с. — ISBN 978-5-94774-825-3.
2. Канер Кем, Фолк Джек, Нгуен Енг Кек. Тестирование программного обеспечения. Фундаментальные концепции менеджмента бизнес-приложений. — Киев: ДиаСофт, 2001. — 544 с. — ISBN 9667393879.