

МЕТОДЫ ТЕСТИРОВАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Тестирование программного обеспечения – процесс исследования, испытания программного продукта с целью получения информации о качестве, имеющий две различные цели:

1) продемонстрировать разработчикам и заказчикам, что программа соответствует требованиям;

2) выявить ситуации, в которых поведение программы является неправильным, нежелательным или не соответствующим спецификации.

Качество исследуемого ПО можно определить как совокупную характеристику ПО с учётом надёжности, сопровождаемости, практичности, эффективности, мобильности, функциональности. Существует множество подходов к решению задачи тестирования ПО.

Обычно выделяют следующую классификацию видов тестирования:

1. По объекту тестирования:

- функциональное тестирование (functional testing);
- нагрузочное тестирование;
- тестирование производительности (performance/stress testing);
- тестирование стабильности (stability/load testing);
- тестирование удобства использования (usability testing);
- тестирование интерфейса пользователя (UI testing);
- тестирование безопасности (security testing);
- тестирование локализации (localization testing);
- тестирование совместимости (compatibility testing).

2. По знанию системы:

- тестирование чёрного ящика (black box);
- тестирование белого ящика (white box).

«Тестирование белого ящика» и «тестирование чёрного ящика» отличаются тем, имеет ли разработчик тестов доступ к исходному коду тестируемого ПО, или же тестирование выполняется через пользовательский интерфейс либо прикладной программный интерфейс, предоставленный тестируемым модулем.

При тестировании белого ящика разработчик теста имеет доступ к исходному коду программ и может писать код, который связан с библиотеками тестируемого ПО. Оно обеспечивает то, что компоненты конструкции работоспособны и устойчивы, до определённой степени. При тестировании белого ящика используются метрики покрытия кода. Покрытие кода показывает процент, насколько исходный код программы

был протестирован. Существует несколько различных способов измерения покрытия: покрытие операторов – каждая ли строка исходного кода была выполнена и протестирована; покрытие условий – каждая ли точка решения (вычисления истинно ли или ложно выражение) была выполнена и протестирована; покрытие путей – все ли возможные пути через заданную часть кода были выполнены и протестированы; покрытие функций – каждая ли функция программы была выполнена; покрытие вход/выход – все ли вызовы функций и возвраты из них были выполнены.

При тестировании чёрного ящика, тестирущик имеет доступ к ПО только через те же интерфейсы, что и заказчик или пользователь, либо через внешние интерфейсы, позволяющие другому компьютеру либо другому процессу подключиться к системе для тестирования. Как правило, тестирование чёрного ящика ведётся с использованием спецификаций или иных документов, описывающих требования к системе.

3. По степени автоматизированности:

- ручное тестирование (manual testing);
- автоматизированное тестирование (automated testing);
- полуавтоматизированное тестирование (semiautomated testing).

4. По степени изолированности компонентов:

- компонентное (модульное) тестирование (component/unit testing) – тестируется минимально возможный для тестирования компонент, например, отдельный класс или функция;
- интеграционное тестирование (integration testing) – тестируются интерфейсы между компонентами, подсистемами;
- системное тестирование (system/end-to-end testing) – тестируется интегрированная система на её соответствие требованиям.

5. По времени проведения тестирования:

- альфа тестирование (alpha testing) – имитация реальной работы с системой штатными разработчиками, либо реальная работа с системой потенциальными пользователями/заказчиком. Чаще всего альфа-тестирование проводится на ранней стадии разработки продукта, но в некоторых случаях может применяться для законченного продукта в качестве внутреннего приёмочного тестирования;
- тестирование при приёмке (smoke testing);
- тестирование новых функциональностей (new feature testing);
- регрессионное тестирование (regression testing) – собирательное название для всех видов тестирования программного обеспечения, направленных на обнаружение ошибок в уже протестированных участках исходного кода. Такие ошибки, когда после внесения изменений в программу перестает работать то, что должно было продолжать работать, называют регрессионными ошибками (regression bugs). Обычно

используемые методы регрессионного тестирования включают повторные прогоны предыдущих тестов, а также проверки, не попали ли регрессионные ошибки в очередную версию в результате слияния кода;

- тестирование при сдаче (acceptance testing);

- бета тестирование (beta testing) – в некоторых случаях выполняется распространение версии с ограничениями (по функциональности или времени работы) для некоторой группы лиц, с тем чтобы убедиться, что продукт содержит достаточно мало ошибок. Иногда бета-тестирование выполняется для того, чтобы получить обратную связь о продукте от его будущих пользователей.

6. По признаку позитивности сценариев:

- позитивное тестирование (positive testing) – это тестирование на данных или сценариях, которые соответствуют нормальному (штатному, ожидаемому) поведению тестируемой системы. Основной целью «позитивного» тестирования является проверка того, что при помощи системы можно делать то, для чего она создавалась;

- негативное тестирование (negative testing) – это тестирование на данных или сценариях, которые соответствуют нештатному поведению тестируемой системы – различные сообщения об ошибках, исключительные ситуации, «запредельные» состояния и т. п. Основной целью «негативного» тестирования является проверка устойчивости системы к воздействиям различного рода, валидация неверного набора данных, проверка обработки исключительных ситуаций (как в реализации самих программных алгоритмов, так и в логике бизнес-правил).

7. По степени подготовленности к тестированию:

- тестирование по документации (formal testing) – осуществляется на этапе разработки требований к программному продукту после создания функциональных спецификаций. Тестирование документации охватывает следующие виды документов: функциональные спецификации; спецификации по графическому интерфейсу пользователя; руководства пользователя и онлайн-справочные системы. В ходе тестирования документации проверяются следующие аспекты: логика и согласованность; полнота и ясность изложения; точность; структура документа; орфография, грамматика, правильность употребления слов и пунктуация; форматирование.

- Эд Хок (интуитивное) тестирование (ad hoc testing) – тестирование по тест-кейсам. Интуитивное тестирование, как правило, применяется тестировщиком в качестве теста приемки и/или теста сдачи.

Используя изложенные методы тестирования программного обеспечения, была протестирована программа calc.exe, выявлены ее ошибки и составлен отчет об ошибках.