

УДК 004.415.53.

АНАЛИЗ ВИДОВ ОСНОВНЫХ ВИДЫ 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. В зависимости от целей тестирования, типа объекта тестирования, особенностей выбранной модели разработки программного обеспечения тестирование можно разделить на несколько основополагающих видов. В статье рассмотрены основные виды тестирования, их характеристики и случаи применения

Ключевые слова: тестирование программного обеспечения, метод черного ящика, метод белого ящика, функциональное тестирование, нефункциональное тестирование, тестирование связанное с изменениями

Тестирование программного обеспечения является одним из главных процессов обеспечения качества разрабатываемого продукта. В зависимости от целей тестирования, типа объекта тестирования, особенностей выбранной модели разработки программного обеспечения тестирование можно разделить на несколько основополагающих видов.

Рассмотрим следующие виды тестирования:

1. функциональное тестирование;
2. нефункциональное тестирование;
3. тестирование белого ящика;
4. тестирование связанное с изменениями.

Рассмотрим более подробно функциональное тестирование. Прежде чем начать данный вид тестирования, необходимо оценить функции, которые выполняет разрабатываемый программный продукт, на основе документации и недокументированных требований. Важно понимать, что функции - это то, что должна делать система, ключевые преобразования данных, выводимые результаты, которые необходимы для выполнения поставленных бизнес-требований. Функциональные тесты должны выполняться на всех уровнях тестирования. Данный вид тестирования учитывает поведение разрабатываемого программного продукта, выполняемые

им преобразования и модификации входных данных. Методы «черного ящика» могут использоваться в функциональном тестировании для получения условий тестирования и тест-кейсов для функциональности компонента или системы. Покрытие функционального тестирования оценивается процентом протестированных функциональных элементов программного обеспечения.

Рассмотрим основные функциональные тесты:

- для тестирования компонентов;
- для тестирования интеграции компонентов;
- для тестирования;
- для тестирования системной интеграции;
- для приемочного тестирования.

В отличие от функционального, нефункциональное тестирование призвано оценить характеристики системы и программного обеспечения: удобство его использования для конечного пользователя, эффективность работы и безопасность данных. Нефункциональное тестирование дает характеристики того, насколько хорошо система выполняет поставленные цели и задачи. Данный вид тестирования может и должно проводиться на каждом уровне тестирования и как можно раньше. Методы «черного ящика» в нефункциональном тестировании могут использоваться для получения условий тестирования и тест-кейсов (например, анализ граничных значений можно использовать для определения условий стресс нагрузки).

Покрытие нефункционального тестирования оценивается процентом протестированных нефункциональных элементов разрабатываемого программного обеспечения (например, используя прослеживаемость между тестами и поддерживаемыми устройствами для мобильного приложения, можно рассчитать процент устройств, для которых было выполнено тестирование на совместимость, потенциально выявляя пробелы в покрытии).

Рассмотрим основные нефункциональные тесты:

- для тестирования компонентов разрабатываемого программного продукта - тесты производительности предназначены для оценки количества ЦПУ, необходимых для выполнения комплексного расчета общего процента;
- для тестирования интеграции компонентов - тесты безопасности предназначены для уязвимостей переполнения буфера из-за данных, передаваемых из пользовательского интерфейса в бизнес-логику;
- для тестирования системы - тесты на переносимость предназначены для проверки того, работает ли «интернет страница» (web view) на всех поддерживаемых браузерах и

мобильных устройствах;

- для тестирования системной интеграции - тесты надежности предназначены для оценки надежности системы, оценки корректной работы микросервиса (например, расчет показателей, в случае, когда микросервис перестает отвечать из-за высокого количества запросов);
- для приемочного тестирования – тесты, направленные на оценку удобства использования предназначены для оценки доступности интерфейса при осуществлении работ конечными пользователями.

Тестирование белого ящика основывается на внутренней структуре (код, архитектура, рабочие потоки и / или потоки данных в системе) или реализации тестируемой системы.

Тщательность тестирования белого ящика может быть измерена через структурное покрытие. Структурное покрытие - это степень, в которой определенный тип структурного элемента был испытан тестами, и выражается в процентах от типа охватываемого элемента. Количество проверяемого кода оценивается «покрытием кода» (code coverage).

Рассмотрим основные тесты белого ящика:

- для тестирования компонентов;
- для тестирования интеграции компонентов;
- для тестирования системы;
- для тестирования системной интеграции;
- для приемочного тестирования.

Рассмотрим виды тестирования тестирование, связанные с изменениями в разрабатываемом программном продукте. Выделяют:

- confirmation testing (подтверждающее тестирование). Данный вид тестирования направлен на подтверждение того, что дефект исправлен, осуществляет проверку функционала заблокированного дефектом и проверку функционала, который отсутствовал из-за дефекта;
- регрессионное тестирование. Данный вид тестирования направлен на подтверждение того, что ранее рабочий функционал не поврежден изменениями.

Рассмотрим основные тесты, связанные с изменениями:

- для тестирования компонентов;
- для тестирования интеграции компонентов;
- для системного тестирования;
- для тестирования системной интеграции;
- при приемочном тестировании все ранее не пройденные испытания повторно

выполняются после устранения дефекта, обнаруженного при приемочном тестировании.

Список литературы:

1. Гленфорд Майерс, Том Баджетт, Кори Сандлер. Искусство тестирования программ, 3-е издание = The Art of Software Testing, 3rd Edition. — М.: «Диалектика», 2012. — 272 с. — ISBN 978-5-8459-1796-6. Архивная копия от 19 июля 2012 на Wayback Machine
2. Бейзер Б. Тестирование чёрного ящика. Технологии функционального тестирования программного обеспечения и систем. — СПб.: Питер, 2004. — 320 с. — ISBN 5-94723-698-2.