

3. Республика Беларусь в зеркале социологии: сборник материалов социологических исследований / Информационно-аналитический центр при Администрации Президента Республики Беларусь. – Минск, 2018. – 180 с.

4. Титова, С. В. Геймификация в обучении иностранным языкам: психолого-дидактический и методический потенциал / С. В. Титова, К. В. Чиркизова // Педагогика и психология образования. – 2019. – № 1. – С. 135–151.

УДК 004.054

Ковалева Е. С., Толстова Н. С.

КЛАССИФИКАЦИЯ ВИДОВ ТЕСТИРОВАНИЯ И ПРОБЛЕМЫ АВТОМАТИЗИРОВАННОГО ТЕСТИРОВАНИЯ

Елена Сергеевна Ковалева

магистрант

kovalevaeleena18a@gmail.com

Наталья Сергеевна Толстова

кандидат педагогических наук

natalya.tolstova@rsvpu.ru

ФГАОУ ВО «Российский государственный профессионально-педагогический университет», Россия, Екатеринбург

CLASSIFICATION OF TEST TYPES AND PROBLEMS OF AUTOMATED TESTING

Elena Sergeevna Kovaleva

Natalya Sergeevna Tolstova

Russian State Vocation Pedagogical University, Russia, Yekaterinburg

Аннотация. В статье раскрыта классификация видов тестирования по видам и уровням. Дано определения каждому виду для большего понимания. Рассмотрена проблема автоматизированного тестирования.

Abstract. The article discloses the classification of types of testing by types and levels. Definitions are given for each species for greater understanding. The problem of automated testing is considered.

Ключевые слова: тестирование, виды тестирования, автоматизированного тестирования.

Keywords: testing, types of testing, automated testing.

На сегодняшний день тестирование является неотъемлемой частью нашей жизни и именно является актуальным рассмотреть и разобрать классификацию видов тестирования.

Согласно «Руководству к своду знаний по программной инженерии» (IEEE, SWEBOOK, 2004), тестирование — это проверка соответствия между реальным и ожидаемым поведением программы, осуществляемая на конечном наборе тестов, выбранном определенным образом [2].

У тестирования есть множество классификаций:

1. По критерию запуска программы — это статическое и динамическое тестирование. Статическое тестирование — это когда система не запущена, то есть это может быть тестирование документации или же проверка кода написание. Динамическое тестирование — это когда система запущена, то есть можно проверить процесс установки, загрузку файла.

2. По объекту тестирования — это функциональное и нефункциональное тестирование. Функциональное тестирование — это есть проверка только тех функций, какие требовал заказчик. Нефункциональное тестирование фокусируется на том, чтоб проверить все функции программы и на сколько она удобна и стабильно работает [1].

3. По уровням — это компонентное (модульное), интеграционное и системное. Компонентные или модульные тесты пишут разработчики, называемые автотесты. Такие тесты проверяют определенные элементы отдельно от зависимых. Интеграционные тесты направлены на проверку связей между элементами.

Системные тесты направлены на проверку функциональных и нефункциональных требований в системе в целом.

4. По степени подготовленности — это тестирование по документации, исследовательское тестирование и интуитивное тестирование. Тестирование документации предполагает начало тестирования еще до разработки продукта. В нем должна быть логика, полнота, функциональность, однозначность. Исследовательское тестирование — такие тесты не спланированы заранее, а создаются по ходу изучения функционала, в которых заложен ожидаемый результат. Интуитивное тестирование — такие тесты выполняются без подготовки к тестам, без определенных ожидаемых результатов, без проектирования сценариев [1].

5. По знанию тестирования требуемого объекта — это тестирование черного, серого и белого ящика. Тестирование «черного ящика» - проводится тестирование без доступа к коду. Тестирование «белого ящика» - позволяет тестирующему заглянуть в процессе тестирования внутрь кода, посмотреть методы, классы, можно посмотреть в сам код и найти ошибку внутри. Эта процедура требует определенных навыков и чаще всего их пишут разработчики. Тестирование «серого ящика» — часто используется в тестировании веб-приложений, когда есть возможность частично узнать, как обрабатывает код. Открыть консоль в браузере и посмотреть какие запросы уходят к серверу и какие ответы от сервера приходят и локализовать ошибку.

6. По степени автоматизации: ручное и автоматизированное. Ручное тестирование — это процесс тестирования вручную, то есть без каких-либо автоматизированных средств. Для этого тестирущик берет на себя роль конечного пользователя и проверяет программу, в соответствии с сценариями. Его задача — найти в программе ситуации, когда фактический результат будет отличаться от ожидаемого. Автоматизированное тестирование — это процесс, при котором все функции и шаги выполняются автоматически. Сначала разрабатывают ручные тесты, а потом уже их автоматизируют [1]. Для этого дела есть специальные люди, автоматизаторы, они пишут код, программируют, для того,

чтобы сценарии прогонялись программой. Это и есть понимание автоматизированного тестирования, чтобы компьютер запускал тесты, и они прогонялись сами.

Проблемы автоматизированного тестирования. Основной проблемой автоматизированного тестирования является то, что могут быть затрачены большие ресурсы на обновление самих тестов. Иногда следует перепроектировать код, для того, чтобы облегчит понимание работы программы. Для этого изменяют ее внутреннюю структуру, но при этом не затрагивают внешний вид программы. И после всех этих действий необходимо обновить и модульные тесты, а изменение кода тестов может занять столько же времени, сколько и изменение основного кода [3]. С другой стороны, при изменении интерфейса приложения необходимо заново переписать все тесты, которые связаны с обновленными окнами, что при большом количестве тестов может отнять значительные ресурсы.

Вторая проблема автоматизированного тестирования является гибкость. Хотя ручное тестирование не всегда настолько же точное, как автоматическое, ручные процессы позволяют тестерам быть более гибкими в операциях. Поскольку автоматическое тестирование полезно для повторяющихся сценариев, часто это означает, что каждый раз используется один и тот же код. Если же тестеры хотят поработать над чем-то конкретным, то разработка автоматического инструмента и последующий запуск теста отнимают много времени. В этом случае и легче, и быстрее сделать все вручную.

В краткосрочных проектах автоматизированное тестирование не выгодно. Автоматизация требует больших инвестиций и планирования - и то, и другое может быть чрезмерным для краткосрочных проектов. Автоматическое тестирование в таком случае не имеет смысла из-за стоимости. Цена автоматизации в таких проектах может быть слишком высока и для возврата инвестиций, и для конечной стоимости продукта. В таких случаях ручное тестирование будет в итоге дешевле и выгоднее [3].

Еще одна из проблем возникает, когда тестируется юзабилити (удобство использования, простота). Успех или провал приложения в значительной степени зависит от юзабилити. Однако это выходит далеко за рамки простого запуска автоматических тестов для проверки, что дефектов нет. У компьютеров нет собственного мышления — они выполняют то, что запрограммировано, и не могут дать такой обратной связи, как пользователь приложения. К примеру, автоматические тесты могут легко найти ошибки в коде, но не способны сказать, как пользователи взаимодействуют с конкретной особенностью приложения или удобна ли человеку навигация программы. Если пользователи находят, что какой-то инструмент в ПО работает не так, как им нужно, это может стать критическим фактором, примут ли это приложение. Исследовательскому тестированию и проверке юзабилити требуется человеческое восприятие, поскольку машины отлично справляются с повторами, но менее эффективны в исследовании и выводах.

Ручное тестирование помогает быстро выявить и устранить дефекты, которые возникают при взаимодействии пользователей с приложением. Автоматическое тестирование имеет множество преимуществ, но ручное тестирование в определенных обстоятельствах приносит больше пользы. Понимая, какой метод подходит лучше, тестировщики могут оптимизировать операции, увеличивать продуктивность и улучшать общее качество.

Список литературы

1. Классификация видов тестирования. – URL: <https://qa-academy.by/qaacademy/news/klassifikaciya-vidov-testirovaniya>.
2. Тестирование программного обеспечения – основные понятия и определения. – URL: <http://www.protesting.ru/testing>.
3. Автоматизированное тестирование. – URL: <https://gist.github.com/codedokode/a455bde7d0748c0a351a>.