

ТЕСТИРОВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Аннотация: Статья посвящена рассмотрению основных аспектов тестирования программного обеспечения. Приводятся определения места и роли контроля качества в современной программной инженерии, различия процессов тестирования, верификации и валидации.

Ключевые слова: тестирование, верификация, валидация, контроль качества, программное обеспечение.

Annotation: The article is devoted to the main aspects of software testing. The definitions of the place and role of quality control in modern software engineering, the differences in testing, verification and validation processes are given.

Key words: testing, verification, validation, quality control, software.

Технология тестирования программного обеспечения стала стремительно развиваться в девяностые года в Америке. Именно акцентирование на вопросах тестирования позволило резко увеличить производство программных средств самого широкого функционала. Возросшая конкуренция среди вендоров заставила разработчиков уделить повышенное внимание качеству производимого программного обеспечения. Желание расширить рынки потребовало использования чисто рыночных

механизмов, и, в первую очередь, повышения качества программной продукции. Цифровизация проникла во все сферы жизнедеятельности человека. Кроме повседневной жизни программное обеспечение применяется в гораздо более жизненно важных сферах, таких как медицина, ЖКХ, транспорт, авиация, обеспечение безопасности и многих других. Таким образом, качество программного обеспечения стало критичной характеристикой, требующей самого пристального внимания [1].

Осознание ситуации привело к тому, что большинство вендоров в разных странах стали инвестировать существенные средства в обеспечение и контроль качества программных продуктов. В структуре этих компаний появились соответствующие подразделения и специалисты (QA-специалисты, QC-инженеры, тестировщики). Начали стремительно развиваться новые технологии, обеспечивающие конкурентное преимущество производителям, стремящимся к повышению качества производимого программного обеспечения.

Сегодня тестирование программного обеспечения является неотъемлемой частью программной инженерии. Только с помощью грамотно спланированной верификации и валидации можно понять, правильно ли работает приложение, насколько его функционал и интерфейс отвечают зафиксированным требованиям. Системное выявление и оперативное исправление допущенных ошибок существенно уменьшает риски при разработке программного обеспечения и приводит к заметному снижению финансовых, ресурсных и временных производственных затрат. Если процесс тестирования программного обеспечения автоматизируется, это благоприятно сказывается на скорости и функциональности тестирования, что в свою очередь, также снижает себестоимость производства [2].

Отметим также иные аспекты, повышающие значение тестирования в современной программной инженерии [3]:

— Сложность программного обеспечения постоянно возрастает. Вероятность появления ошибок в сложных программных системах выше, чем в простых.

— Особенность процесса разработки приложений заключается в высокой степени его изменчивости. Внесение изменений также повышает вероятность появления ошибок. Причем не важно с чем эти изменения связаны: либо исправляются обнаруженные ошибки, либо расширяется функционал информационной системы.

— Дефекты тестирования через ухудшение качества программного продукта приводят к усилению недовольства его потребителей и заказчиков. Компания – производитель неизбежно будет нести финансовые потери.

— Тестирование позволяет не только обнаружить допущенные ошибки, но и выяснить причины их возникновения. Это обстоятельство позволяет повысить качество продукции.

Тестирование программного обеспечения – неотъемлемый этап жизненного цикла информационной системы, заключающийся в определении уровня соответствия ее реального и ожидаемого состояний (качества), включая функциональные возможности. Проверка реализуется на некотором конечном наборе тестов в соответствии с определенной стратегией [4]. Тестирование – технология контроля качества, включающая в себя осмысление предметной области, целей и задач программного обеспечения, планирование и разработку тестов, собственно выполнение тестирования и анализ результатов. Говорят о жизненном цикле тестирования (STLC), фиксирующем последовательность операций, гарантирующих заданный уровень качества программного обеспечения и его соответствие требованиям пользовательского соглашения (SLA).

Строго говоря, следует различать процедуры тестирования, верификации и валидации. Процесс тестирования в основном акцентирован на проверке корректности кодирования информационной системы. Для этого

осуществляется управляемое выполнение программы с целью выявления (доказательства наличия) ошибок (отклонений, неполноты, неоднозначности трактовок и др.) в актуальной на данный момент версии разрабатываемого приложения. Т.е., тестирование концептуально отвечает на вопрос «Как это сделано?». Понятие «верификация программного обеспечения» понимается шире. Ее цель заключается в обеспечении гарантии того, что верифицируемый объект соответствует требованиям, не включает незапланированных особенностей или функций, удовлетворяет заявленным проектным решениям и действующим стандартам. Таким образом, тестирование представляется как обособленный этап процесса верификации. Результатом верификации можно считать исчерпывающий ответ на вопрос «Что сделано?». Цель процесса валидации заключается в получении доказательств того, что разработанная система безусловно обеспечивает достижение поставленных целей. Здесь уже оценивается способность системы удовлетворить требования заказчика. Иными словами, на этапе валидации мы узнаем «Сделано ли то, что нужно?».

В идеале тестировщик должен понимать какие тесты покрывают конкретные верификационные (четко прописанные) требования, а какие – не столь явные валидационные (часто даже не обозначенные в техзадании) и соответственно разделять их по разным сборникам тестов.

На практике процесс тестирования упрощенно можно представить следующим образом. Специалист по тестированию получает требования к информационной системе, ее описание и начинает проектировать тесты. Здесь важно определиться, имеет ли смысл переходить на автоматическое тестирование. Создавая различные ситуации (наборы данных, условия их обработки), тестировщик отслеживает реакцию системы (ее внутреннее состояние). Таким образом, формируется представление о соответствии или несоответствии программного продукта требованиям. На основе этой информации планируются мероприятия по устранению ошибок в

информационной системе, либо изменяются требования к создаваемому программному продукту [5].

Тест (проверка) заключается в искусственном создании смоделированной определенным образом ситуации и описании наблюдений за реакцией системы с целью определения степени соответствия программного продукта заданным требованиям.

Сформулируем две основные задачи, которые тестировщик решает в процессе тестирования:

- управление ходом выполнения программы, генерация искусственных ситуаций, предназначенных для проверки качества приложения.
- фиксация реакций программы на изменение условий и сравнение полученных откликов с ожидаемыми.

В современных условиях задача тестирования ставится шире. Кроме непосредственного обнаружения программных ошибок, еще требуется выяснить их источники или причины, по которым они возникают. В результате такого подхода для разработчиков появляется возможность достичь максимальной эффективности производственного процесса.

В данном исследовании рассмотрены роль и место тестирования в жизненном цикле программного обеспечения.

Использованные источники:

1. Скотовиков А.Г., Скотовиков Н.А. Основные тренды развития цифровой экономики на современном этапе // Цифровые технологии в экономике и промышленности: сб. науч. тр. Санкт-Петербург, 2019. С. 55-63.
2. Ошероув, Рой. Искусство автономного тестирования с примерами на C# / Рой Ошероув. - М.: ДМК Пресс, 2016. - 360 с.
3. Бек, К. Экстремальное программирование. Разработка через тестирование / Кент Бек. – СПб.: Питер, 2017. – 224 с.
4. Керниган, Брайан. Практика программирования / Брайан Керниган, Роб

Пайк. - М.: Вильямс, 2015. - 288 с.

5. Куликов С. С. Тестирование программного обеспечения. Базовый курс / С.С. Куликов. — 3-е изд. — Минск: Четыре четверти, 2020. — 312 с.