

Сивцов Р.О.

Студент 2 курса магистратуры

Поволжский государственный университет телекоммуникаций и

информатики

Россия, г. Самара

ИССЛЕДОВАНИЕ ПРОБЛЕМ АВТОМАТИЗИРОВАННОГО ТЕСТИРОВАНИЯ

Аннотация: В работе рассмотрены ключевые проблемы автоматизированного тестирования программного обеспечения, включая длительное выполнение тестов, поддержку и обновление существующих тестов, а также ложные срабатывания и нестабильные тесты. Описаны основные причины возникновения этих проблем и предложены методы их решения.

Ключевые слова: Автоматизированное тестирование, Поддержка тестов, Технический долг, QA, DevOps.

Sivtsov R.

Second year graduate student

Povolzhskiy State University of Telecommunications and Informatics

Russia, Samara

RESEARCH ON THE PROBLEMS OF AUTOMATED TESTING

Annotation: The work analyzes the key problems of automated software testing, such as long test execution, support and updating of existing tests, and false positives and flaky tests. The main causes of these problems are described and methods of their solution are proposed.

Keywords: Automated testing, Test support, Technical debt, QA, DevOps.

Введение. Автоматическое тестирование является неотъемлемой частью процесса разработки программного обеспечения, позволяя существенно повысить качество продукта, снизить количество ошибок и ускорить выпуск новых версий.

Несмотря на значительные преимущества, автоматизированное тестирование сталкивается с рядом серьезных проблем, которые могут снижать его эффективность и приводить к нежелательным последствиям.

В ходе работы был проведен анализ литературы, в результате чего выделены следующие ключевые проблемы: длительное выполнение тестов, управление зависимостями и конфигурациями, поддержку и обновление существующих тестов, а также ложные срабатывания и нестабильные тесты.

Длительное выполнение тестов является одной из наиболее распространенных и существенных проблем автоматизированного тестирования. С увеличением объема кода и числа тестовых сценариев время, необходимое для выполнения всех тестов, может значительно возрастать.

Причины длительного выполнения тестов:

1. Рост проекта.
2. Тесты выполняют ресурсоёмкие задачи или обращаются к внешним сервисам.
3. Отсутствие параллелизации.
4. Некачественные или плохо оптимизированные тесты, дублирующиеся проверки.
5. Недостаточно мощная или плохо настроенная тестовая инфраструктура.

Методы решения проблемы:

1. Разделение тестов на группы.

2. Параллельное выполнение тестов.
3. Рефакторинг тестов для устранения избыточных проверок и повышения их эффективности.
4. Применение методик выборочного тестирования для проверки только тех частей кода, которые были изменены.
5. Обеспечение достаточных вычислительных ресурсов для выполнения тестов.
6. Настройка и оптимизация тестовой среды для обеспечения максимальной производительности.

Поддержка и обновление существующих тестов. Постоянное развитие проектов требует регулярного обновления тестов для соответствия новым требованиям и функционалу. Причины, приводящие к необходимости обновления тестовой базы:

1. Частые изменения в требованиях и функциях.
2. Увеличение числа тестов усложняет их поддержку.
3. Обновления внешних библиотек и сервисов.
4. Изменения в архитектуре и логике системы.
5. Накопление устаревших или неэффективных тестов.

Методы решения проблемы:

1. Использование инструментов и скриптов для автоматизации процесса обновления тестов.
2. Регулярный рефакторинг и оптимизация тестов для поддержания их актуальности.
3. Создание и поддержание качественной документации для тестов.
4. Выделение тестовых групп.

Ложные срабатывания (false positives) и нестабильные тесты (flaky tests) могут приводить к значительному времени, затрачиваемого на тестирование, ввиду необходимости перезапуска таких тестов.

Причины ложных срабатываний и нестабильностей в тестах:

1. Использование нестабильных или некорректных тестовых данных.
2. Асинхронные операции и проблемы с синхронизацией.
3. Зависимость от нестабильных внешних сервисов.
4. Неправильные предположения о состоянии системы или входных данных.

Методы решения проблемы:

1. Обеспечение консистентных и надежных тестовых данных.
2. Рефакторинг тестов для улучшения синхронизации, внедрение явных ожиданий и тайм-аутов для асинхронных операций.
3. Использование тестовых данных, заглушек или симуляторов для изоляции тестов.
4. Обеспечение стабильной и предсказуемой тестовой среды.
5. Внедрение систем мониторинга и логирования для отслеживания состояния тестов и выявления причин нестабильности.

Вывод. Реализация предложенных методов может существенно повысить надежность и эффективность процессов автоматизированного тестирования, что в конечном итоге способствует улучшению качества программного обеспечения. Однако, нужно учитывать потребности каждого отдельно взятого проекта. Универсальные решения не всегда могут быть применимы к конкретному проекту без адаптации под его уникальные особенности и требования. Это означает, что выбор подходящих инструментов и методов должен основываться на специфике проекта, его архитектуре, масштабе и динамике изменений.

Список литературы

1. Хамбл Д., Фарли Д. Непрерывное развертывание ПО: автоматизация процессов сборки, тестирования и внедрения новых версий программ [Текст]: учеб. пособие – М.: Диалектика-Вильямс, 2019. – 512 с.
2. Бланделл П., Милано Д. Т. Learning Android Application [Текст]: учеб. пособие – Бирмингем: Packt Publishing, 2015. — 276 с.
3. Маккаллох М., Берглунд Т. Building and Testing with Gradle [Текст]: учеб. пособие – Sebastopol: O'Reilly, 2011. — 168 с.