

ПОДХОДЫ К АВТОМАТИЗАЦИИ МОБИЛЬНЫХ ПРИЛОЖЕНИЙ

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Мищенко А. В., Мельников А. В.

Пискун Г. А. – канд. техн. наук, доцент

Тестирование мобильных приложений имеет свою специфику и во многих аспектах кардинально отличается от тестирования веб-приложений. Соответственно будут различаться и подходы, которые следует использовать для автоматизации тестирования. В данной статье рассматриваются основные инструменты и подходы для автоматизации тестирования мобильных приложений.

Растущая функциональность и частые обновления мобильных приложений не всегда позволяют ручному тестировщику покрыть все тест-кейсы за короткий промежуток времени. Преимущества автоматизации тестирования мобильных приложений во многом совпадают с плюсами автоматизированного тестирования веб-приложений [1]:

- 1) Быстрое получение результатов;
- 2) Исключение человеческого фактора;
- 3) Параллельное тестирование на множестве устройств;
- 4) Возможность покрыть множество локализаций.

Несмотря на все преимущества, так же как и при автоматизации тестирования веб-приложений, у автоматизации тестирования мобильных приложений также есть свои особенности и минусы:

1) Автотесты перестают работать с обновлением операционной системы. Поэтому после каждого обновления ОС придется ждать, пока обновятся инструменты для разработки автоматических тестов.

2) Сложность в написании тестов. На написание одного автотеста для веб-приложения в среднем уходит 8 часов, а мобильного – около 20 часов.

3) Нестабильная работа многих инструментов. Для мобильных платформ существует множество отличных друг от друга UI-компонентов. В силу многообразия разработать один инструмент, способный работать с любыми типами UI, либо физически невозможно, либо его использование является чрезвычайно трудоемким, технически сложным и, как следствие, экономически невыгодным [1].

- 4) Требования к iOS: все *ipa*-файлы должны быть подписаны разработчиками.

При автоматизацией тестирования веб-приложений *Selenium WebDriver* – самый популярный инструмент на рынке, который подойдет для большинства сценариев тестирования. При автоматизации тестирования мобильных приложений однозначного ответа при выборе инструмента нет.

Однако главный вопрос касается не выбора инструмента, а подхода. Именно выбранный подход предопределяет инструмент для последующей разработки автотестов.

При автоматизации тестирования мобильных приложений можно использовать один из следующих подходов [1]:

- 1) *Record and Play*;
- 2) *Screen Object*

Первый способ прост и сводится к записи всех действий пользователя (тестировщика) в приложении. После записи действий инструмент генерирует понятный для него код и создает автотесты.

Плюсы: быстрая реализация, не требуется знаний программирования;

Минус: малейшие изменения в приложении потребуют создать новый автотест [1].

Как следствие, данный подход, как и аналогичный подход при тестировании веб-приложений является сложным для масштабирования и расширения. Может быть использован, только в ситуациях, когда тестовое приложение стабильно, редко обновляется и тестируется.

Второй подход – *Screen Object* – паттерн, аналогичный *Page Object* [2] предназначенный для организации архитектуры автотестов в виде взаимодействия экранов приложения.

Screen Object моделирует экраны (или страницы) тестируемого приложения в качестве объектов в коде. В результате мы получаем набор классов, каждый из которых отвечает за работу с отдельным экраном приложения. Данный подход обладает рядом преимуществ по сравнению с обычной записью действий пользователя [1]:

- 1) переиспользуемость кода;
- 2) надежность кода, небольшая чувствительность к изменениям в приложении;
- 3) понятная структура.

Для полного и качественного тестирования продукта рекомендуется использовать второй подход.

Кроме выбора подхода, также необходимо решить, какой инструмент будет использоваться для взаимодействия кода автотестов и мобильных устройств. Самым популярным инструментом для данных целей является *Appium*

Appium — это бесплатный кроссплатформенный инструмент с открытым исходным кодом, который помогает автоматизировать как для *Android*, так и для *iOS*. Является одним из самых широко используемых инструментов для создания автоматических тестов для смартфонов и планшетов.

Несомненными преимуществами *Appium* является простота в использовании, а также поддержка многих языков программирования: *Java, Ruby, Python, C#, PHP*.

Важным этапом при тестировании мобильных приложений является выбор тестировать на реальном устройстве или же использовать эмулятор. Стоит отметить, что работа с эмулятором создает дополнительную нагрузку на машину, что замедляет выполнение теста.

К тому же результаты тестов на эмуляторах не всегда отвечают действительности. Нередко случается, что на эмуляторе все тесты пройдены, а при запуске на реальном устройстве, система блокирует работу из соображений безопасности, и тест не будет пройден.

Список использованных источников:

1. Автоматизация тестирования мобильных приложений с использованием *Appium* [Электрон. ресурс] – Электрон. дан. Режим доступа: <https://www.a1qa.ru/blog/avtomatizatsiya-testirovaniya-mobilnyh-prilozhenij-s-appium/>
2. Фреймворк автоматизации тестирования пользовательского интерфейса : материалы 54 науч. конф. аспирантов, магистрантов и студентов, Минск, 23-27 апреля 2018 г. / Белорус. гос. ун-т информатики и радиоэлектроники ; редкол.: А.Л. аднёнков [и др.]. – Минск : БГУИР, 2018. – 79 с.