

Голубова И.А. — рецензент *Киселёв Г.Д.*

УНК “Институт прикладного системного анализа” НТУУ “КПИ”, Киев, Украина

Способы тестирования приложений, написанных для операционной системы Android

Операционная система Google Android была выпущена в 2008 году, для мобильных устройств, по open-source-модели. За прошедшее время вышло много версий ОС (начиная с 1.5 “Cupcake” и до 4.0 “Ice Cream Sandwich”) что значительно усложнило написание и соответственно тестирование новых приложений. Для полноценной разработки программы необходимо оптимально использовать возможности среды разработки. Так, для того что бы провести функциональное тестирование, недостаточно просто запустить проект на виртуальной машине Android SDK [1]. Хотя бы потому, что сам процесс загрузки виртуальной машины занимает много времени, не говоря о том что, тестирование программного приложения для разных версий мобильного устройства, требует подгрузки разных виртуальных моделей, что значительно затягивает время доработки программы. Для ускорения процесса тестирования, а значит и написания конечного продукта существуют разные способы тестирования. Помимо Android testing API основанного на стиле JUnit 3 и не поддерживающего JUnit 4. Где тесты запускаются и работают на эмуляторе/девайсе, т.е. запуск медленный. Можно выделить Android независимые тесты в отдельный Java проект и запускать их на JVM компьютера, но это ситуацию кардинально не изменит. Для получения хороших показателей по скорости тестирования, рекомендуют использовать библиотеку Robolectric. В этом случае, тесты запускаются не на эмуляторе или устройстве, а непосредственно на JVM компьютера. Эта библиотека позволяет тестировать большую часть функциональности Android, при этом используя синтаксис JUnit4, хотя полнота эмуляции виртуальной машины, в таком случае, ниже чем у Android API. Также, полезным может оказаться, фреймворк Robotium, использующий синтаксис JUnit3, и создающий тест по типу «черного ящика». В этом случае тесты пишутся на Java, и во время тестирования в тестовый проект добавляется библиотека Robotium. За время существования Android было создано немало инструментов по построению тестов. Встречались и довольно необычные, так например, в Sikuli для задания последовательности действий по тестированию приложения необходимо использовать скриншоты [2]. Для того, что бы провести какое либо действие над объектом необходимо в скрипт вставить скриншот с изображением данного объекта. Не смотря на многообразие всевозможных эмуляторов, к сожалению, все они не полноценны, а значит- не способны исключить возможность платформенного или архитектурного бага. Подобные вещи можно отследить только проводя тесты на реальных устройствах, но, памятуя о том, что ОС развивалась очень быстро с большим количеством версий очевидно, что для полноценного тестирования понадобится масса устройств. В таких случаях, можно прибегнуть к услугам облачных сервисов, на которых за умеренную плату можно получить возможность использовать различные устройства с операционной системой Android для тестирования новой программы.

Литература. 1. Хашими С. Коматинени С. Маклин Д.. Разработка приложений для Android.- СПб.: Питер, 2012.-736с. 2. Тестирование Android приложений: <http://derevyanko.blogspot.com/2012/02/android.html>.