

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г.ЧЕРНЫШЕВСКОГО»

Кафедра информатики и программирования

**Разработка программного клиента для облачной платформы
тестирования мобильных приложений**

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студента 4 курса 441 группы
направления 02.03.03 Математическое обеспечение и администрирование
информационных систем
факультета компьютерных наук и информационных технологий
Тучина Степана Кирилловича

Научный руководитель:

доцент кафедры ИиП, к. ф.-м. н. _____ К.П. Савина

Зав. кафедрой ИиП,

к. ф.-м. н., доцент _____ М.В. Огнева

Саратов 2020

ВВЕДЕНИЕ

В настоящее время многие компании используют не только обычные веб-сайты для привлечения конечного пользователя, но и приложения для смартфонов.

Приложения можно тестировать ручным и автоматизированным способом (на реальных устройствах или на эмуляторе).

Платформа Headspin предоставляет возможность удаленного тестирования программного обеспечения на реальных устройствах.

Любой программный компонент должен быть протестирован, чтобы конечный продукт был рабочим. Платформа Headspin решает проблему эмуляторов, которые позволяют быстро и гибко тестировать приложение, но не дают полного соответствия с реальным устройством, что в дальнейшем может привести к ошибкам [1]. Для автоматизации действий на смартфонах, подключенных к Headspin, зачастую используется библиотека Appium. Ввиду того, что Appium является кроссплатформенной библиотекой (поддерживает разнообразный ряд устройств на базе систем IOS, Android и Windows), она не фокусируется на отдельных функциях, тем самым ограничивая возможности взаимодействия с реальными устройствами [2], [6]. Программный интерфейс Headspin предоставляет возможность запуска отладочных команд для автоматизации управления смартфоном, что позволяет воспользоваться всеми возможными функциями устройств. Для взаимодействия с данным интерфейсом нужен клиент со своим списком отладочных команд [1], [7]. Android Debug Bridge является утилитой, которая предоставляет полное взаимодействие с устройствами Android.

Целью выпускной квалификационной работы является создание клиента к программному интерфейсу (API) средства отладки Android Debug Bridge для удаленного мобильного тестирования. Поставленная цель определила следующие задачи:

1. изучить библиотеку Appium, предназначенную для мобильного тестирования путем автоматизированного выполнения заданного

- набора команд на системах IOS, Android и Windows;
2. изучить возможности средства отладки Android Debug Bridge (ADB), являющегося командной утилитой для управления данными и функциями на платформе Android;
 3. изучить возможности программного интерфейса облачного сервиса тестирования Headspin, построенного на базе библиотеки Appium, для взаимодействия с удаленным устройством;
 4. реализовать программный клиент ADB-команд для отладки Android-устройств на базе API-сервиса удаленного тестирования Headspin;
 5. реализовать базовые тесты для проверки созданного API-клиента.

Методологические основы разработки программного клиента для облачной платформы тестирования мобильных приложений представлены в работах Маркуса Полса, Дэвида Гриффитса и Дона Гриффитса по разработке программных клиентов и работы с Android устройствами [3], [5].

Практическая значимость бакалаврской работы. В ходе выполнения практической части бакалаврской работы был создан программный клиент, который расширяет функционал библиотеки Appium, используемой в автоматизированном мобильном тестировании. Практическая ценность данного клиента состоит в том, что он предоставляет расширенное взаимодействие с устройствами системы Android. Данные функции требуются в производственной разработке.

Структура и объём работы. Бакалаврская работа состоит из введения, 11 разделов, заключения, списка использованных источников и 5 приложений. Общий объем работы – 56 страниц, из них 41 страница – основное содержание, включая 12 рисунков и 4 таблицы, список использованных источников информации – 25 наименований.

КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

Первый раздел «Программный интерфейс взаимодействия и его архитектура» описывает взаимодействие с сервисами в интернете через программный интерфейс на основе архитектуры REST, которая структурирует данное взаимодействие.

Программный интерфейс (Application Programming Interface, API) – это библиотека, содержащая набор функций, который показывает, как взаимодействовать с тем или иным сервисом в интернете.

Зачастую взаимодействие с сервисами происходит на основе архитектуры передачи состояния представления объекта (Representational State Transfer, REST), которая имеет принцип – каждый уникальный адрес является представлением некоторого объекта [1].

При взаимодействии пересылаются файлы состояния – JSON, XML или другие. Также ответ от сервисов сопровождается статусом, который означает состояние запроса.

Все сервисы используют API для взаимодействия. Библиотека Appium и платформа Headspin, применяемые в работе, также обращаются к API.

Второй раздел «Инструмент автоматизации тестирования Appium» посвящен изучению библиотеки автоматизированного тестирования Appium.

Данная библиотека взаимодействует с мобильными устройствами на основе архитектуры клиент-сервер. С помощью библиотеки можно управлять экраном, файловой системой и другими основными компонентами устройств. Взаимодействие происходит на основе сессий, которую открывает клиент при первом подключении. Главным компонентом является класс DesiredCapabilities, с помощью которого передаются параметры подключения к устройству.

Плюсом данной библиотеки является кроссплатформенность (библиотека Appium поддерживает разные системы IOS, Android и Windows).

Основным минусом данной библиотеки считается отсутствие более специфических функций, требуемых в реальной разработке, так как

библиотека Appium больше нацелена на взаимодействие посредством экрана [2], [6].

Третий раздел «Средство отладки Android устройств Android Debug Bridge» направлен на изучение средства отладки ADB, который предоставляет больше возможностей, чем библиотека Appium. Данная утилита устанавливает взаимодействие также на основе архитектуры клиент-сервер, но может устанавливать связь только с устройствами системы Android.

Плюсом данных утилит является их гибкость. Каждая команда имеет ряд ключей, которые выполняют определенные свойства, расширяющие обычный функционал. Appium не поддерживает данные ключи и оставляет команды базовыми.

Данное средство отладки лежит в основе всех вышестоящих библиотек и платформ, которые позволяют выполнять команды ADB удаленно с помощью облачно-ориентированных платформ. Внутри библиотеки Appium реализован программный клиент, который передает команды ADB серверу Appium в виде JSON формата и далее исполняет их на устройстве [7].

Четвертый раздел «Облачно-ориентированные платформы» дает представление об облачно-ориентированных платформах.

Облачно-ориентированные платформы – это платформы, которые предоставляют доступ к приложениям, услугам или ресурсам по запросу через интернет с серверов поставщика удаленных вычислений.

Облачные платформы уменьшают время на создание инфраструктуры и предоставляют удаленное использование сервисов.

Основными облачными системами являются:

- IaaS (инфраструктура как услуга) – поставщик предоставляет только вычислительные мощности.
- PaaS (платформа как услуга) – поставщик полностью обеспечивает инфраструктуру.

- SaaS (программное обеспечение как услуга) – поставщик предоставляет уже готовое приложение.

Пятый раздел «Обзор сервисов облачного тестирования» содержит обзор существующих облачных платформ тестирования.

- Сервис Open Smartphone Test Platform предоставляет только для создания инфраструктуры тестирования, но при этом является бесплатным.
- Сервис Headspin предоставляет инфраструктуру и является временно бесплатным.
- Сервис Amazon Web Services (AWS) Device Farm является платным, но при этом предоставляет интеграцию с другими сервисами платформы.

Плюсом всех этих платформ является удаленное тестирование.

Приведенные платформы построены на основе библиотеки Appium.

Шестой раздел «Облачный сервис тестирования Headspin» содержит подробное описание платформы облачного тестирования. Платформа Headspin предоставляет возможность удаленного тестирования на реальных устройствах, что позволяет избежать ошибок, которые могут быть пропущены на эмуляторах.

Платформа Headspin имеет свой программный интерфейс для взаимодействия удаленно [8].

Седьмой раздел «Библиотека инверсии контроля и внедрения зависимостей Spring» посвящен изучению функционала библиотеки Spring.

Данная библиотека использует специальные архитектурные подходы, которые позволяют уменьшить количество кода и увеличить читабельность, что ускоряет скорость разработки программного обеспечения.

Данная библиотека использует подходы: инверсия управления и внедрение зависимостей.

Инверсия управления – это принцип в разработке программного обеспечения, посредством которого управление объектами или частями программы переносится в контейнер используемой библиотеки [4].

Внедрение зависимостей – это шаблон для реализации инверсии управления, в котором объекты автоматически внедряются в код, где они требуются.

В восьмом разделе «HTTP клиент архитектуры REST» рассмотрены разные варианты взаимодействия в сети с сервисами.

Первым вариантом является командная утилита с URL, которая хорошо подходит для проверки доступности сервиса.

Вторым вариантом является графическое программное обеспечение Postman, который хорошо подходит для ручного тестирования.

Третьим вариантом является Java библиотека Retrofit, которая позволяет писать тесты более гибко и предоставляет надежность их исполнения [5].

В девятом разделе «Программный интерфейс Headspin» описан программный интерфейс облачного сервиса тестирования Headspin. В разделе рассмотрен ответ сервиса и основные адреса взаимодействия.

Также описаны основные этапы взаимодействия с платформой Headspin:

- инициализация клиента;
- составление тела, заголовков и атрибутов запросов;
- отправка запроса по адресу сервера Headspin;
- ожидание ответа от сервиса и его приведение к нужной модели;
- проверка модели и ее полей на соответствие.

Десятый раздел «Реализация клиента к программному интерфейсу Headspin» содержит описание реализации клиента к программному интерфейсу платформы облачного тестирования Headspin.

В разделе описаны этапы создания клиента для сервиса и архитектурный подход создания приложения.

Основными этапами реализации клиента являются:

1. Инициализация драйвера с помощью класса DesiredCapabilities;
2. Создание клиента OkHttp для передачи запросов;

3. Создание интерфейса взаимодействия Retrofit клиента на основе программного интерфейса платформы;
4. Разработка прослойки клиента для обработки ответов от платформы;
5. Реализация методов, которые будут использовать функции клиента.

Одиннадцатый раздел «Реализация тестов с помощью клиента к платформе Headspin» посвящен описанию реализации тестов на основе программного клиента платформы Headspin. Показан результат выполнения тестов и способ генерации отчёта. В разделе описаны этапы, которые выполняет каждый тест.

ЗАКЛЮЧЕНИЕ

В ходе выпускной квалификационной работы были изучены средства автоматизации мобильного тестирования с помощью библиотеки Appium и возможности средства отладки Android Debug Bridge.

Проанализирован программный интерфейс сервиса облачного тестирования Headspin, который построен на базе библиотеки Appium и предоставляет расширенный набор возможностей. Данная библиотека имеет большой функционал и является кроссплатформенной (поддерживает множество устройств на базе систем IOS, Android и Windows).

Однако недостатками библиотеки Appium являются отсутствие у нее специфических функций, которые необходимы на этапах реальной разработки, зависимость от версии продукта, то есть при переходе на новую версию операционной системы некоторые функции могут не работать. Основными функциями в библиотеке Appium являются взаимодействия с экраном, тем самым библиотека имеет только базовые функции по работе с файлами и другими основными компонентами.

В результате выпускной квалификационной работы был реализован клиент для расширения команд библиотеки Appium. Реализованный программный клиент позволяет улучшить работу с файлами, так как приложения зачастую производят файлы, которые должны быть проверены на этапе тестирования. Также клиент предоставляет возможность подключения к определенному Wi-Fi и дает возможность проверки настроек всех устройств, подключенных к сервису Headspin.

После разработки данного клиента был запущен ряд тестов для проверки функционала реализованного клиента. Отчет показал, что тесты прошли проверку, а разработанный клиент может быть расширен путем добавления новых команд с помощью новых методов.

Основные источники информации:

1. Кеннет Ланге, The little book on REST services. – Copenhagen, 2016. – 31 с.
2. Манодж Ханс, Appium Essentials. – Packt Publishing Ltd, 2015. – 275 с.
3. Дэвид Гриффитс, Дон Гриффитс, Head First. Программирование для Android. 2-е изд. – СПб.: Питер, 2018. – 912 с.
4. Крейг Уоллс. Spring in Action. 5-е изд. – Manning Publications Co, 2019. – 521 с.
5. Маркус Полс, Retrofit: Love Working with APIs on Android. – Future Studio, 2016. – 80 с.
6. Grid Dynamics QA Meetup (SPb): Appium в мобильном тестировании [Электронный ресурс]. URL: <https://www.youtube.com/watch?v=OKA1OUIAM> (Дата обращения 11.02.2020). Загл. с экр. Яз. рус.
7. Android Debug Bridge (adb). [Электронный ресурс]. URL: <https://developer.android.com/studio/command-line/adb> (Дата обращения 14.02.2020). Загл. с экр. Яз. англ.
8. Introducing the HeadSpin Connected Intelligence Platform™ [Электронный ресурс]. URL: <https://www.headspin.io/platform/> (Дата обращения 05.03.2020). Загл. с экр. Яз. англ.