

Министерство образования и науки Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г. ЧЕРНЫШЕВСКОГО»

Кафедра компьютерной физики
и метаматериалов

Основы разработки приложений для платформы Android

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ ДИПЛОМНОЙ РАБОТЫ

студента 4 курса 431 группы

направления 03.03.02 «Физика» физического факультета

Пажитнова Сергея Александровича

Научный руководитель

к. ф.-м. н., доцент

_____ А. С. Ремизов

Зав. кафедрой

д. ф.-м. н., профессор

_____ В. М. Аникин

Саратов 2016 год

Актуальность работы. Вопрос развития сферы мобильных устройств особенно актуален в наши дни. По сути, смартфоны и планшетные компьютеры стали компактной альтернативой персональному компьютеру, с которой в любом месте и в любое время пользователь способен взаимодействовать. Сейчас большинство моделей смартфонов и планшетных компьютеров выпускаются на базе платформы Android. Также на платформе Android работают не только смартфоны и планшетные компьютеры, но и часы, тренажёры, телевизионные приставки, фото-рамки, автомобили и многое другое.

Целью данной квалификационной работы является изучение архитектуры и функций платформы Android, знакомство со средой разработки и создание Android-приложений.

В задачи данной квалификационной работы входит:

1. Изучение устройства платформы Android;
2. Рассмотрение архитектуры Android-приложений, особенностей их функционирования;
3. Изучение среды разработки Android-приложений, средств отладки и эмуляции устройств;
4. Создание демонстрационных приложений;

Структура и объем работы. Выпускная квалификационная работа изложена на 52 страницах, состоит из введения, 3 глав (Платформа Android; Жизненный цикл Android-приложений; Примеры приложений) и заключения. Библиографический список включает 12 наименований. Текст иллюстрирован 18-ю рисунками.

Основное содержание работы

Платформа Android. ОС Android – операционная система, программа, позволяющая создавать Java-приложения, управляющие устройством, через разработанные Google библиотеки будь то смартфон, планшет или часы.

Данная платформа основана на ядре операционной системы Linux и собственной реализации виртуальной машины Java от Google.

Android поддерживает фоновое выполнение задач – работы приложения продолжается, в то время как само приложение не отображается на экране или во время работы с другими приложениями; предоставляет богатую библиотеку элементов пользовательского интерфейса; поддерживает 2D и 3D графику, используя OpenGL стандарт; поддерживает доступ к файловой системе и встроенной базе данных SQLite.

С точки зрения архитектуры, операционная система Android представляет собой полный программный стек, в котором можно выделить 4 основных уровня:

1) Базовый уровень(Linux Kernel) – уровень абстракции между аппаратным уровнем и программным стеком;

2) Набор библиотек и среда выполнения(Libraries & Android Runtime) – обеспечивает важнейший базовый функционал для приложений. Содержит виртуальную машину Dalvick / ART и базовые библиотеки Java, необходимые для запуска Android приложений;

3) Уровень каркаса приложений(Application Framework) – обеспечивает разработчикам доступ к API, предоставляемым компонентами системы уровня библиотек;

4) Уровень приложений(Applications) – набор предустановленных базовых приложений;

Средства разработки. Последние несколько лет Android поощрял разработчиков использовать Eclipse ADT (Android Developer Tools) Bundle в качестве среды для разработки приложений. Ситуация изменилась, когда была анонсирована и стала доступна для загрузки более продвинутая среда разработки Android Studio. В отличие от ADT Bundle, базируемой на Eclipse IDE и системе Apache Ant, Android Studio работает на основе IntelliJ IDEA и системе Gradle.

К обязательным инструментам разработки относится Android SDK - набор средств программирования, который содержит инструменты, необходимые для создания, компиляции и сборки мобильного приложения. Наиболее важные инструменты, входящие в состав Android SDK:

1) SDK Manager - инструмент, позволяющий загрузить компоненты Android SDK;

2) Debug Monitor - самостоятельный инструмент, предоставляющий графический интерфейс к нескольким инструментам, предназначенным для анализа и отладки Android приложений;

3) Android Emulator (emulator) - виртуальное мобильное устройство, которое создается и работает на компьютере разработчика, используется для разработки и тестирования мобильных приложений без привлечения реальных устройств;

4) AVD Manager - предоставляет графический интерфейс для создания виртуальных Android устройств (AVDs), предусмотренных Android Emulator, и управления ими;

5) Android Debug Bridge (adb) - гибкий инструмент, позволяющий управлять состоянием эмулятора или реального Android устройства, подключенного к компьютеру.

Виды Android-приложений. Для разработки приложений необходимо понимать, с какой целью создаётся приложение. Для этого следует изучить виды приложений для операционной системы Android. Приложения делятся на следующие виды:

1) Приложения переднего плана – пользователь имеет возможность взаимодействовать с приложением только во время его видимости на экране устройства. Если приложение не отображается на экране, то его деятельность приостанавливается или завершается полностью;

2) Фоновые приложения – взаимодействие таких приложений с пользователем минимально. Свои основные функции приложение выполняет в фоновом режиме без участия пользователя. Примером такого приложения

служит SMS-автоответчик. Приложение автоматически отвечает на SMS, поступившие на устройство, без вмешательства пользователя, в соответствии с настройками, который установил пользователь;

3) Смешанные приложения – большую часть времени такие приложения работают в фоновом режиме, но при необходимости пользователь может взаимодействовать с приложением напрямую. Примером таких приложений может служить skype. Данное приложение реагирует на действия пользователя в видимом режиме, например, на набор или отправку текста, и может выводить на экран уведомление о поступивших сообщениях, принимать звонки, находясь в фоновом режиме;

4) Виджеты – это приложения, расположенные на рабочем столе устройства. Это могут быть часы, календарь, заряд батареи или прогноз погоды. Как правило, такие приложения предустановлены на устройстве.

Архитектура приложения. Архитектура Android приложений основана на идее многократного использования компонентов, которые являются основными строительными блоками. Каждый компонент является отдельной сущностью и помогает определить общее поведение приложения.

Система Android выстроена таким образом, что любое приложение может запускать необходимый компонент другого приложения. Когда система запускает компонент, она запускает процесс приложения, которому принадлежит компонент, если он еще не запущен, и создает экземпляры классов, необходимых компоненту. Поэтому в отличие от большинства других систем, в системе Android приложения не имеют единой точки входа (нет метода main(), например).

Можно выделить четыре различных типа компонентов, каждый тип служит для достижения определенной цели и имеет свой особый жизненный цикл, который определяет способы создания и разрушения соответствующего компонента. Рассмотрим основные компоненты Android-приложений.

1) **Активности (Activities).** Активность - это видимая часть приложения (экран, окно, форма), отвечает за отображение графического интерфейса пользователя. При этом приложение может иметь несколько активностей, например, в приложении, предназначенном для работы с электронной почтой, одна активность может использоваться для отображения списка новых писем, другая активность - для написания, и еще одна - для чтения писем. Несмотря на то, что для пользователя приложение представляется единым целым, все активности приложения не зависят друг от друга. В связи с этим любая из этих активностей может быть запущена из другого приложения, имеющего доступ к активностям данного приложения.

2) **Сервисы (Services).** Сервис - компонент, который работает в фоновом режиме, выполняет длительные по времени операции или работу для удаленных процессов. Сервис не предоставляет пользовательского интерфейса. Например, сервис может загружать данные из сети, не блокируя взаимодействие пользователя с активностью. Сервис может быть запущен другим компонентом и после этого работать самостоятельно, а может остаться связанным с этим компонентом и взаимодействовать с ним.

3) **Контент-провайдеры (Content providers).** Контент-провайдер управляет распределенным множеством данных приложения. Данные могут храниться в файловой системе, в базе данных SQLite, в сети, в любом другом доступном для приложения месте. Контент-провайдер позволяет другим приложениям при наличии у них соответствующих прав делать запросы или даже менять данные.

4) **Приемники широковещательных сообщений (Broadcast Receivers).** Приемник - компонент, который реагирует на широковещательные извещения. Большинство таких извещений порождаются системой, например, извещение о том, что экран отключился или низкий заряд батареи. Приложения также могут инициировать широковещание, например, разослать другим приложениям сообщение о том, что некоторые данные загружены и доступны для использования. Хотя

приемники не отображают пользовательского интерфейса, они могут создавать уведомление на панели состояний, чтобы предупредить пользователя о появлении сообщения, а также могут запускать соответствующий какому-либо событию сервис.

Все рассмотренные компоненты являются наследниками классов, определенных в Android SDK.

Примеры приложений

В рамках данной квалификационной работы были созданы три приложения. Первое приложение определяет набор датчиков, доступных на устройстве пользователя. Второе приложение демонстрирует работу одного из датчиков. И наконец, третье приложение демонстрирует некоторые возможности работы с картой.

Датчики устройств на базе Android. Несмотря на скромные размеры, современные смартфоны — очень сложные устройства с мощными многоядерными процессорами, камерами с автофокусом и экранами большого разрешения. Кроме того, любой смартфон оснащён различными датчиками, которые делают использование устройства более удобным, либо расширяют их возможности.

Перечислим наиболее популярные датчики смартфонов на платформе Android: акселерометр, гироскоп, магнитометр, датчик приближения, датчик света. На этом можно завершить список наиболее популярных датчиков, которые используются в смартфонах. Но на самом деле существует огромное количество различных датчиков таких, как: барометр, дозиметр, термометр, гигрометр, датчик Холла, пульсометр, шагомер, сканер отпечатков пальцев.

Как было сказано выше, первое приложение определяет набор датчиков определённого устройства и выводит список доступных датчиков на экран устройства. Данное приложение было протестировано на реальном устройстве.



Рисунок 1 – Результат работы приложения, определяющего набор доступных датчиков устройства.

На рисунке 1 представлен результат работы приложения. Было определено два датчика — датчик ускорения и датчик приближения CM36671.

Демонстрация работы датчиков ускорения и положения. Акселерометр это первый датчик, который стали широко применять в смартфонах. Также акселерометр часто называют G-сенсором. Как можно понять из его названия, этот датчик служит для измерения ускорения устройства по трём осям. Очевидно, что ускорение есть только тогда, когда устройство перемещается или поворачивается в пространстве, поэтому положение неподвижного смартфона акселерометр определить не может. Чтобы исправить этот недостаток акселерометра, совместно с ним в подавляющем большинстве современных устройств, применяется гироскоп. В отличие от акселерометра, гироскоп может определять положение в пространстве даже неподвижного устройства. Погрешность откалиброванного гироскопа в современных смартфонах, как правило, не превышает 1-2 градусов. Гироскоп и акселерометр широко используются во многих мобильных играх для управления, а также в других приложениях — с самыми разными задачами.

Вторым приложением, созданным в рамках данной квалификационной работы, стала программа, выводящая на экран показания датчика ускорения и положения в пространстве. Результаты работы данного приложения приведены на рисунке 2.

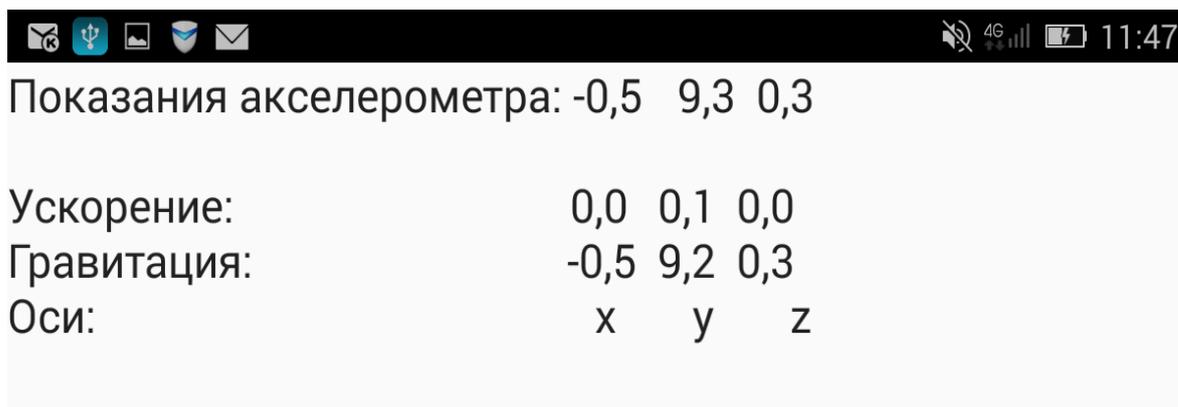


Рисунок 2 – Результат работы приложения, демонстрирующего работу датчиков ускорения и положения в пространстве.

Интеграция Google Maps в Android-приложение. Для работы с картами Google мы должны установить соответствующий пакет служб Google Play Services SDK. Также для доступа к серверам Google карт нашему приложению необходим ключ Google Maps API. После завершения подготовки среды разработки к работе с картами Google можем начинать создавать приложение. В рамках данной квалификационной работы, было создано приложение, использующее Google Maps. Пользователь данного приложения может просматривать карту мира в двух режимах:

1. Политическая карта мира.
2. Карта, основанная на снимках спутников.

Также доступна функция установки маркера в любом месте карты по нажатию на экран устройства, и установка маркера в центре карты нажатием кнопки. Результаты работы приложения продемонстрированы на рис.3.

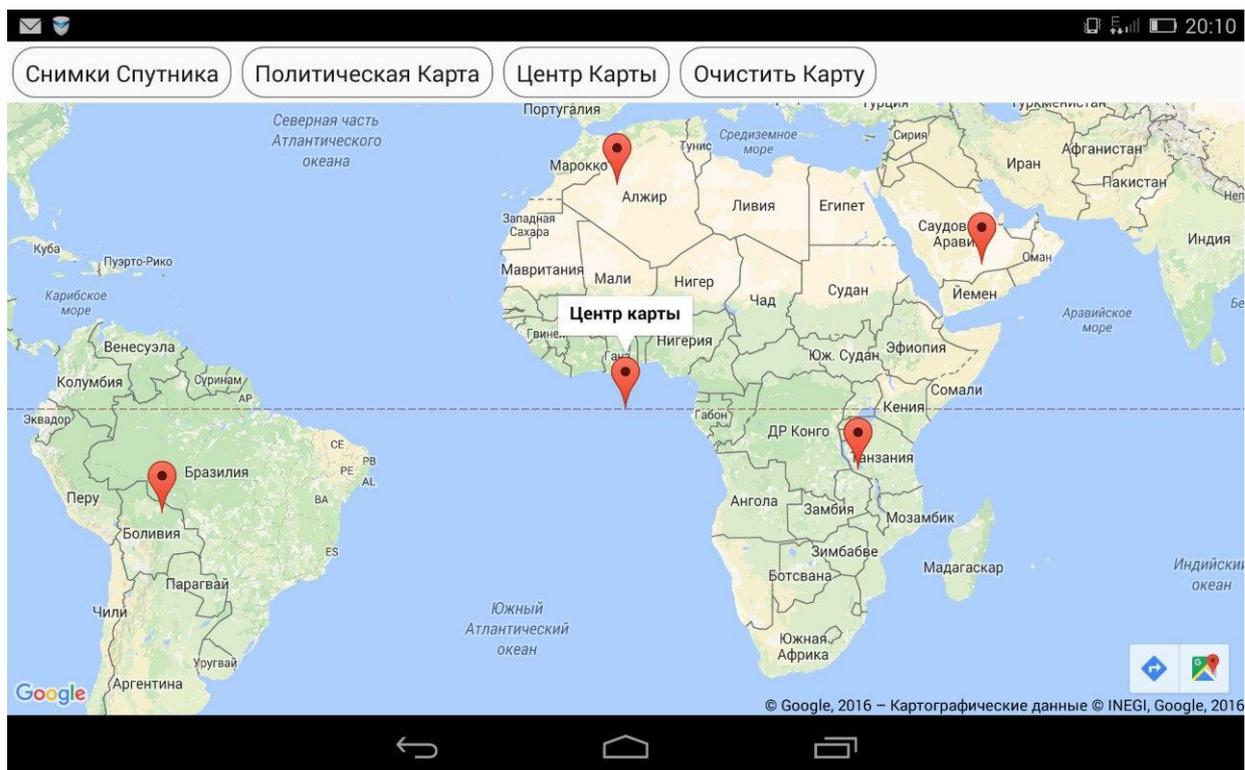


Рис. 3 Режим политической карты мира.

ВЫВОДЫ

В данной работе рассмотрено устройство платформы Android, архитектура и жизненный цикл приложений в этой системе. В практической части работы были созданы три демонстрационных приложения для устройств с операционной системой Android:

- приложение для работы с картой мира,
- приложение, определяющее наличие различных датчиков в устройстве
- приложение, демонстрирующее работу датчиков перемещения и положения в пространстве

Приложения созданы с помощью среды разработки Android Studio и протестированы на реальном устройстве.