|  |
| --- |
| **Page 1** |

**Лекция 1. ВВЕДЕНИЕ В РАЗРАБОТКУ МОБИЛЬНОГО ПО**

**1.1 ВВЕДЕНИЕ**

Android - операционная система для мобильных устройств: смартфонов,

планшетных компьютеров, КПК. В настоящее время именно Android является самой широко

используемой операционной системой для мобильных устройств. подтверждение

этого факта можно найти в таблице, составленной по данным аналитической компании

Gartner.

Таблица 1.1. Мировые продажи смартфонов конечным пользователям, распределение по ОС

**Операционная**

**система**

**продано**

**(тыс.шт.) III кв.**

**2013**

**доля рынка**

**(%) III кв.**

**2013**

**продано**

**(тыс.шт.) III кв.**

**2012**

**доля рынка**

**(%) III кв.**

**2012**

**Android**

205022,7

81,9

124552,3

72,6

**iOS**

30330,0

12,1

24620,3

14,3

**Microsoft**

8912,3

3,6

3993,6

2,3

**BlackBerry**

4400,7

1,8

8946,8

5,2

**Bada**

633,3

0,3

4454,7

2,6

**Symbian**

457,5

0,2

4401,3

2,6

**другие**

475,2

0,2

683,7

0,4

**Общее в:**

**250231,7**

**100,0**

**171652,7**

**100,0**

Источник: Gartner (ноябрь 2013)

Внимательное изучение таблицы позволяет увидеть большую популярность

смартфонов под управлением ОС Android в мире, доля таких устройств не первый

год превышает половину от общего числа купленных смартфонов. Кроме всего

прочего, эта популярность продолжает расти. Очевидно, что армия пользователей

смартфонов под управлением Android будет искать дополнительные программы для своих

устройств, в связи с этим умение разрабатывать эти самые программы может принести много

пользы своему владельцу. Например, можно разрабатывать для себя полезные, интересные,

интересные (нужное подчеркнуть) приложения, а можно, разведав обстановку и

|  |
| --- |
| **Page 2** |

оглянувшись, сделать разработку мобильных приложений своей профессиональной деятельностью,

основной или дополнительной.

**Т рохи истории**

Рассмотрим, как все начиналось. В 2003 году в Пало Альто, штат Калифорния

Энди Рубин с единомышленниками (Рич Майнер, Ник Сирс и Крис Уайт) основали компанию

Android Inc. Сначала в компании занимались проектированием мобильных гаджетов,

на основе геолокационных данных автоматически подстраивались под нужды

пользователей.

В августе 2005 года Android Inc. стала дочерней компанией Google. Энди

Рубин, Рич Майнер и Крис Уайт остались в Android Inc. и начали работать над

операционной системой, базирующейся на ядре Linux. В Google задумали реализовать

самую мощную платформу, пригодную к использованию на тысячах различных моделей

телефонов. В связи с этим был создан Open Handset Alliance (OHA) -

консорциум, состоящий из более чем 80 компаний, направляет свои усилия на

разработку открытых стандартов для мобильных устройств. В состав OHA входят

такие гиганты, как Google (организатор и идейный вдохновитель), HTC, Sony, Dell, Intel,

Motorola, Qualcomm, Texas Instruments, Samsung Electronics, LG Electronics, T-

Mobile, Sprint Corporation, NVIDIA и многие другие.

Первая версия Android была представлена 23 сентября 2008 года, версии было

дано название Apple Pie (можно заметить созвучие с прямым конкурентом). дальше так

повелось, что название каждой очередной версии представляет какой-либо десерт, при

этом первые буквы наименований в порядке версий соответствуют буквами

латинского алфавита по порядку.

**1.2 УСТРОЙСТВО ПЛАТФОРМЫ ANDROID**

Платформа Android объединяет операционную систему, построенную на основе ядра

ОС Linux, промежуточное программное обеспечение и встроенные мобильные приложения.

Разработка и развитие мобильной платформы Android выполняется в рамках проекта

AOSP (Android Open Source Project) под управлением OHA (Open Handset

Alliance), управляет всем процессом поисковый гигант Google.

|  |
| --- |
| **Page 3** |

Android поддерживает фоновое выполнение задач; 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, система Android представляет собой полный

программный стек, в котором можно выделить следующие уровни:

• **Базовый уровень (Linux Kernel)** - уровень абстракции между аппаратным уровнем и

программным стеком;

• **Набор библиотек и среда исполнения (Libraries & Android Runtime)**

обеспечивает важнейший базовый функционал для приложений, содержит виртуальную

машину Dalvik и базовые библиотеки Java необходимые для запуска Android приложений;

• **Уровень каркаса приложений (Application Framework)** обеспечивает разработчикам

доступ к API, предоставляемые компонентами системы уровня библиотек;

• **Уровень приложений (Applications)** - набор установленных базовых приложений.

**Рис.** **1.1.** архитектура Android

Рассмотрим компоненты платформы более подробно.

В основе компонентной иерархии лежит ядро ОС Linux 2.6 (несколько урезано),

служит промежуточным уровнем между аппаратным и программным обеспечением, обеспечивает функционирования системы, оказывает системные службы ядра: управление памятью,энергосистемой и процессами, обеспечение безопасности, работа с сетью и драйверами.

Уровнем выше располагается набор библиотек и среда исполнения.

Библиотеки реализуют следующие функции:

• предоставляют реализованы алгоритмы для высших уровней;

• обеспечивает поддержку файловых форматов;

• осуществляет кодирование и декодирование информации (например, мультимедийные

кодеки)

• выполняет отрисовку графики и т.д.

Библиотеки реализованы на С / С ++ и скомпилированные под конкретное аппаратное

обеспечение устройства, вместе с которым они и поставляются производителем в

предустановленном виде.

Рассмотрим некоторые библиотеки:

**Surface Manager**

- Композитный менеджер окон. Поступающие команды отрисовки собираются в закадровый буфер, где они накапливаются,

составляя некую композицию, а затем выводятся на экран. это

позволяет системе создавать интересные бесшовные эффекты, прозрачность

окон и плавные переходы.

**Media**

**Framework**

- Библиотеки, реализованные на базе PacketVideo OpenCORE.

Используются для записи и воспроизведения аудио и видео

контента, а также для вывода статических изображений.

Поддерживаются форматы: MPEG4, H.264, MP3, AAC, AMR, JPG и

PNG.

**SQLite**

- легковесный и продуктивный реляционная СУБД, используется в

Android в качестве основного движка для работы с базами данных.

**3D библиотеки**

- Используются для высокооптимизированные отрисовки 3D-

графики, при возможности используют аппаратное ускорение.

Библиотеки реализованы на основе API OpenGL | ES. OpenGL | ES

(OpenGL for Embedded Systems) - подмножество графического

программного интерфейса OpenGL, адаптированное для работы на

встроенных системах.

**FreeType**

- Библиотека для работы с битовыми картами, для растеризации

|  |
| --- |
| **Page 5** |

шрифтов и осуществления операций над ними.

**LibWebCore**

- Библиотеки браузерного движка WebKit, используемого

также в известных браузерах Google Chrome и Apple Safari.

**SGL (Skia**

**Graphics**

**Engine)**

- Открытый движок для работы с 2D-графикой. графическая

библиотека является продуктом Google и часто используется в других

программах.

**SSL**

- Библиотеки для поддержки одноименного криптографической

протокола.

**Libc**

- Стандартная библиотека языка С, а именно ее BSD реализация,

настроена для работы на устройствах на базе Linux.

Среда выполнения включает в себя библиотеки ядра, обеспечивающие

большую часть низкоуровневой функциональности, доступной библиотекам ядра языка

Java, и виртуальную машину Dalvik, что позволяет запускать приложения. каждое приложение

запускается в своем экземпляре виртуальной машины, тем самым обеспечивается

изоляция работающих приложений от ОС и друг от друга. Для исполнения на

виртуальной машине Dalvik Java - классы компилируются в выполняемые файлы с

расширением .dex с помощью инструмента dx, входящей в состав Android

SDK. DEX (Dalvik EXecutable) - формат исполняемых файлов для виртуальной

машины Dalvik, оптимизированный для использования минимального объема памяти. при

использовании IDE Eclipse и плагина ADT (Android Development Tools) компиляция

классов Java в формат .dex происходит автоматически.

Архитектура Android Runtime такова, что работа программ осуществляется строго в

рамках окружения виртуальной машины позволяет защитить ядро ОС от возможной

вреда со стороны других ее составляющих. Поэтому код с ошибками или вредное ПО не

смогут испортить Android и устройство на его базе, когда сработают.

На еще более высоком уровне располагается каркас приложений (Application

Framework), архитектура которого позволяет любому приложению использовать уже

реализованы возможности других программ, к которым разрешен доступ. в состав

каркаса входят следующие компоненты:

|  |
| --- |
| **Page 6** |

• богатый и расширяемый набор представлений **(Views),** который может быть использован для создания визуальных компонентов приложений, например, списков, текстовых полей, таблиц, кнопок или даже встроенного web-браузера; • контент-провайдеры **(Content Providers),** управляющие данными, одни приложения

открывают для других, чтобы те могли их использовать для своей работы;

• менеджер ресурсов **(Resource Manager),** что обеспечивает доступ к ресурсам

без функциональности (не несет кода), например, в срочных данным, графике,

файлов и другим;

• менеджер уведомлений **(Notification Manager),** что позволяет приложениям

отображать свои сообщения для пользователя в строке состояния;

• менеджер действий **(Activity Manager),** управляющий жизненными циклами приложений,

что сохраняет историю работы с действиями, предоставляет систему навигации по действиям;

• менеджер местоположения **(Location Manager),** что позволяет приложениям

периодически получать обновленные данные о текущем географическом положении

устройства.

Application Framework предоставляет в распоряжение приложений в ОС Android

вспомогательный функционал, благодаря чему реализуется принцип многократного

использование компонентов приложений и ОС. Естественно, в рамках политики безопасности. И, наконец, самый высокий, близкий к пользователю уровень приложений. Именно на этом уровне

пользователь взаимодействует с устройством, управляемым ОС Android. здесь

представлен набор базовых приложений, который предустановлен на ОС Android.

Например, браузер, почтовый клиент, программа для отправки SMS, карты,

календарь, менеджер контактов и др. Список интегрированных программ может

меняться в зависимости от модели и версии Android. К этому уровню

также относятся все пользовательские приложения.

Разработчик обычно взаимодействует с двумя верхними уровнями архитектуры Android

для создания новых приложений. Библиотеки, система исполнения и ядро Linux скрытые

по каркасом приложений.

Повторное использование компонентов других приложений приводит к идее

задач в Android. Приложение может использовать компоненты другого Android

приложения для решения задачи, например, если разрабатываемое приложение

предполагает использование фотографий, оно может вызвать приложение, управляющее

фотографиями и зарегистрировано в системе Android, выбрать с его помощью

фотографию и работать с ней.

Для пополнения коллекции приложений мобильного устройства пользователь

может воспользоваться приложением Google Play, которое позволяет покупать и устанавливать

приложения с сервиса Google Play. Разработчики, в свою очередь, могут выкладывать свои

приложения в этот сервис, Google Play отслеживает появление обновлений программы, сообщает

пользователям этого приложения об обновлении и предлагает установить его. также

Google Play предоставляет разработчикам доступ к услугам и библиотекам, например, доступ

к использованию и отражению Google Maps.

Для установки программы на устройствах с ОС Android создается файл с

расширением \* .apk (Android package), который содержит исполняемые файлы, а также

вспомогательные компоненты, например, файлы с данными и файлы ресурсов. после

установки на устройство каждое приложение "живет" в своем собственном изолированном

экземпляре виртуальной машины Dalvik.