

Advancement in Mobile Communication using Android

J.Mani Bharathi
Kalasalingam University
295'A', Rathinamnagar,
Sivakasi, India.

S.Hemalatha
Kalasalingam University
9-12-4, Nethaji 2nd St
Viswanathapuram, India.

V.Aishwarya
Kalasalingam
University
24 Rajam Aveune,
Kalaingar Nagar,
Poonga,
Ellis Nagar
Madurai, India.

C.Meenapriya
Kalasalingam University
170, Ramnagar
Senbagathoppu Road
Rajapayalam, India.

L.Hepzibha Shekinah
Grace
Kalasalingam university
Pandian Nagar Kosakulam
Madurai, India.

ABSTRACT

This paper deals about the introduction of android in mobile devices. The recent trends in android give the future development in software development kit (SDK) which is ubiquitous in all mobile phones. With this use of android in our mobile phone the compatibility accessibility and the interface of the mobile phone version ones has been considerably increased. The latest of android allows the speech recognition facility in the mobile devices. With this further development of android allows us to install video or picture recognition with moving sensors in our mobile phones.

Keywords

SDK -Software Development Kit, API-Application Package Index,WII

1. INTRODUCTION

There were nearly 700 million cell phones sold each year compared with fewer than 200 million PCs — and the gap was widening. Increasingly, phones were the way people wanted to connect with each other and with everything else. Phones were going to replace PCs as the main gateway to the Internet, and they were going to do it soon. The cell phones ran on different software, had less memory, and operated under the constraints of pay-per-byte wireless networks; the mobile Web was a stripped-down, mimeographed version of the real thing. To avoid this, Android had the solution. Android is a free, open source mobile platform that any coder could write for and any handset maker could install. It would be a global, open operating system for the wireless future. Android is software stack for mobile devices that includes operating system, middleware and key applications. Android is an Open Platform for Mobile development. It is based on Linux operating system and developed by Google and Open Handset Alliance.

2. SCOPE OF THE WORK

- Ability to record and watch videos with the camcorder mode
- Uploading videos to YouTube and pictures to Picasa directly from the phone
- A new soft keyboard with an "Autocomplete" feature
- Ability to automatically connect to a Bluetooth headset within a certain distance

- New widgets and folders that can populate the desktop
- Animations between screens and moving sensors to play Wii games
- Expanded ability of Copy and paste to include web pages

Android is based on the Linux Kernel and has some interesting particularities. "Android does not differentiate between the phone's core applications and third-party applications. They can all be built to have equal access to a phone's capabilities providing users with a broad spectrum of applications and services. Android breaks down the barriers to building new and innovative applications. For example, a developer can combine information from the web with data on an individual's mobile phone -- such as the user's contacts, calendar, or geographic location -- to provide a more relevant user experience. With Android, a developer could build an application that enables users to view the location of their friends and be alerted when they are in the vicinity giving them a chance to connect. Android provides access to a wide range of useful libraries and tools that can be used to build rich applications. For example, Android enables developers to obtain the location of the device, and allow devices to communicate with one another enabling rich peer-to-peer social applications."

3. ANDROID RUNTIME

3.1 Dalvik virtual machine:

Every Android application runs in its own process, with its own instance of the Dalvik virtual machine. Dalvik has been written so that a device can run multiple VMs efficiently. The Dalvik VM executes files in the Dalvik Executable (.dex) format which is optimized for minimal memory footprint. The VM is register-based, and runs classes compiled by a Java language compiler that have been transformed (as ByteCodes, which is good for run fast) into the .dex format by the included "dx" tool.

3.2 Core Libraries:

It contains all of the collection classes, utilities, io etc.

3.2.1 Linux Kernel:

Android relies on Linux version 2.6 for core system services such as security, memory management,

process management, network stack, and driver model. It also act as an abstraction layer between the hardware and rest of the software stack.

4. DEVELOPMENT ENVIRONMENT

4.1 Supported Operating Systems

- Windows XP (32-bit) or Vista (32- or 64-bit)
- Mac OS X 10.4.8 or later
- Linux

4.2 Supported Development Environments

4.2.2 Eclipse IDE

4.2.2.1 Eclipse 3.3 (Europa), 3.4 (Ganymede)

Recommended Eclipse IDE packages: Eclipse IDE for Java EE Developers, Eclipse IDE for Java Developers, Eclipse for RCP/Plug-in Developers Eclipse JDT plugin Eclipse Classic IDE package is not supported. JDK 5 or JDK 6 (JRE alone is not sufficient) Android Development Tools plugin.

5. SOFTWARE DEVELOPMENT KIT

It is a typically a set of development tools that allows a engineer to create applications for a certain software package, software framework, hardware platform, computer system, video game console, operating system or similar platform.

5.1 Android 1.1

5.1.1 External Libraries

The system image includes these external libraries, which you can access from your application by adding a <uses-library>.

5.1.2 Built-in Applications

The system image includes these built-in applications: Alarm Clock, API Demos, Browser, Calculator, Camera, Contacts, Dev tools, Dialer, Email, Maps (and Street View), Messaging, Music, pictures and settings.

5.1.3. Resolved Issues

Alarm Clock alert now plays audio/vibe directly, rather than through AlarmManager. AlarmClock alert starts playing audio/vibe audio/vibe in itsIntent Receiver, rather than on activity start. These changes should prevent alarms from being blocked by modal dialogs. Fixes to device sleep. Single tap no longer opens the in-call dial pad; users now need to touch a and drag it. Fixes a bug causing approximately 1 in 25 outbound messages to free up the IMAP connection (to a Gmail based server) when transferred to the Sent folder. Removes automatic account setup entries that were broken or not testable. Adds minor fixes to a few of the remaining entries. Makes improvements to warning dialogs used for a few special cases. Changes default mail checking interval to every 15 minutes (instead of defaulting to "never"). Fixes password-quoting bugs in IMAP, so that users can include special characters in passwords (e.g. spaces). Fixes various errors in auto and manual account setup Improves reporting for various connection errors, making it easier for to the user desire failed account setups. Fixes new-mail notifications for POP3 accounts. Ensures proper auto-checking of accounts marked as "never check". Now displays date and time using user preference (e.g. 24 hr vs. AM/PM). Now shows cc: in message view. Improves recovery from POP3 connection

failures. POP3 parser rules loosened, so the application can work with complaint email servers.

5.1.4 Features

- Maps: Adds details and reviews when a user does a search on Maps and clicks on a business to view its details.
- Dialer: In-call screen timeout default is now longer when using the speakerphone.
- Dialer: Adds a "Show dial pad" / "Hide dial pad" item to the in-call menu, to make it easier to discover the DTMF dial pad.
- Adds support for saving attachments from MMS
- Adds support for marquee in layouts.

5.1.5 Overview

- Adds annotations for test systems, no actual (nontest) API changes.
- Adds a method to allow a process to easily determine its UID.
- Adds support for marquee in layouts.
- Adds new methods for determining padding in views. Adds new permissions that allow an application to broadcast an SMS or WAP Push message.
- API cleanup: removes protected constructor from SDK-bound system images.

6. ANDROID 1.5

6.1 Built-in Applications:

The system image includes these built-in applications: Alarm Clock , Browser , Calculator, Camcorder, Camera, Contacts, Custom Locale (developer app), Dev Tools (developer app), Dialer, Email, Gallery, IME for Japanese text input, Messaging, Music, Settings, Spare Parts (developer application).

6.1.1 Overview

6.1.2 UI framework:

Frame work for easier background/UI thread interaction.

New Sliding Drawer widget.

New Horizontal Scroll view widget.

App Widget framework.

APIs for populating Live Folders with custom content .

6.1.3 Media frame work:

Raw audio recording and playback APIs.

Interactive MIDI playback engine.

Video recording APIs for developers (3GP format).

Video and photo sharing Intents.

Media search Intent.

Input Method framework.

Input Method Service framework.

Text-prediction engine.

Ability to provide downloadable IMEs to users.

6.1.4 Application-defined hardware requirements:

Applications can now use a new element in their manifest files, <uses-configuration> to indicate to the Android system what hardware features they require in order to function properly. For example, an application might use the element to specify that it requires a physical keyboard or a particular navigation device, such as a trackball. Prior to installing the application, the Android system checks the attributes defined for the <uses-configuration> element and allows the installation to continue only if the required hardware is present.

6.1.5 Speech recognition framework:

Support for using speech recognition libraries via Intent.

6.1.6 Miscellaneous API additions:

- Location Manager - Applications can get location change updates via Intent.
- Web View - Touch start/end/move/cancel DOM event support.
- Redesigned Sensor Manager APIs GLSurfaceView - convenience framework for creating OpenGL applications.
- Broadcast Intent for app update install succeeded for smoother application upgrade experience.

6.1.6 Performance Improvement:

Faster Camera start-up and image capture. Much faster acquisition of GPS location (powered by SUPL AGPS). Smoother page scrolling in Browser and Speedier Gmail conversation list scrolling.

7. NEW FEATURES

7.1 On-screen soft keyboard:

It works in both portrait and landscape orientation. Its main support is for user installation of 3rd party keyboards. It also for user dictionary for custom words.

7.1.1 Home screen:

The home screens are having the widgets like, bundled home screen widgets include: analog clock, calendar, music player, picture frame, and search Live folders.

7.1.2 Camera & Gallery

Video recording.

Video playback (MPEG-4 & 3GP formats).

7.1.3 Bluetooth:

Stereo Bluetooth support (A2DP and AVCRP profiles).

Auto-pairing.

Improved hands free experience.

7.1.4 Browser:

Updated with latest Web kit browser & Squirrelfish JavaScript engines and Copy 'n paste in browser. It also having the facility with search within a page and user selectable text-encoding.

7.1.5 UI changes include:

Unified Go and Search box.

Tabbed bookmarks/history/most-visited screen.

7.1.6 Contacts:

Shows user picture for Favourites.

Specific date/time stamp for events in call log.

One-touch access to a contact card from call log event.

7.1.7 System Requirements:

- New Linux kernel (version 2.6.27).
- SD card file system auto-checking and repair SIM Application Toolkit 1.0.

7.1.8 Google applications (not available in the Android 1.5 System Image that is included in the Android SDK :

View Google Talk friends' status in Contacts, SMS, MMS, Gmail, and Email applications. Batch actions such as archive, delete, and label on Gmail messages. Upload videos and photos in You tube and Picasa.

8. ANALYSIS OF ANDROID 1.6 SDK:

The android 1.6 system is going to deliver the sdk is one of the development from the android 1.5, which gives compatible to android powered handsets. The system has fully comfort and includes the map external library.

The android 1.6 system delivers an updated version of the frame work API and the android 1.5. As with previous sensors, the android 1.6 has sensor bar with softwares of wii games and also sensor software. The updated API is assigned an integer identifier that is stored in the system itself. It is used to determine whether the application is compatible with the system, prior to installing the applications.

8.1 BUILT IN APPLICATIONS:

In the developer application the inbuilt of sensors with software provide the best playment of wii games and other 3D games. Some spare parts are software of wii games which is installed through API and sensors to capture from the wii remote. Other applications are as same as the android 1.5 sdk. The spare parts are the includement of map external library and other external libraries. The software of wii and sensors are transferred through Bluetooth so it brings autopairing effect.

8.1.1 WII REMOTE:

The main feature of wii remote is the gesture recognition and pointing devices. The gesture recognition detects motion and rotation in three dimension through the use of accelerometer technology. By separating the controller from the gaming console, the accelerometer data can be used as an input for the gesture recognition. The accelerating sensors represented the gestures by characteristic patterns of incoming signals data ie. Vectors ,representing the current accelerations of controller in all three dimensions. By filtering and quantizer technique the unwanted vector data can be identified and cleared. This can be done by k-mean algorithm. In this recognition all different size objects, and other images in 3D can be recognized.

8.1.2 VECTOR QUANTIZATION:

Like other acceleration-sensors the one integrated into the Wiimote delivers too much vector data to be put into a single HMM. In order to cluster and abstract this data the common k mean algorithm is applied with k being the number of clusters or codes in the so-called codebook. A codebook size delivering satisfying results and its empirically identified $k = 8$ for gestures in a two-dimensional plane. However, adopt the idea of arranging the 8 cluster centres on a circle by extending it to the 3D case. Instead of distributing the centres uniformly on a two-dimensional circle its possible to put them on a three dimensional sphere, intersecting two circles orthogonal to each other. Consequently this leads to $k = 8 + 6 = 14$ centres. The radius of each circle/sphere dynamically adapts itself to the incoming signal data.

9. IMPLEMENTATION

The implementation of the gesture recognition is in Java using the standardization of Java APIs for Bluetooth Wireless technology (JABWT) defined by the JSR-82 specification. Using Java ensures platform independency, for developing and testing purposes the usage of the GNU/Linux platform with the Avetana Bluetooth implementation is needed. The recognition process is realized as a reusable and extensible gesture recognition library based on an event-driven design pattern. The library provides an interface for basic functions, e.g. acceleration readout with the Wii Listener interface, as well as recognition functions using a Gesture-Listener interface. Through its modularity it is easy to adapt the prototype to other acceleration-based controllers.

10. SENSING

The wii remote has the ability to sense acceleration along with three axes through the use of an ADXL 330 accelerometer. The wii remote also has the feature of pix art optical sensors, allowing it to determine where the wii remote is pointing. Unlike a light gun that senses light from a mobile screen, the wii remote senses light from the sensor which is inbuilt in the form of chip in the android phones allows consistent usage regardless of the mobile sizes. The infrared LED's in the sensor gives an accurate pointer. The sensor allows the wii remote to be used as an accurate pointing device 30 meters away from the mobile. The wii remote's image sensor is used to locate the sensor's points of light in the wii remote's field of view. The inbuilt wii software in the mobile devices calculates the distance between the wii remote and sensor using triangulation. Games can be programmed to sense whether the image sensor is covered, which is demonstrated in a microgame of smooth moves, where if the player does not uncover the sensor, the champagne bottle that the remote represents will not open.

11. MAP EXTERNAL LIBRARY:

The map external library is made to add powerful mapping capabilities to this sdk and the good application package index add on includes a map external library. The classes of the map library offer built in downloading, rendering, caching of map files as well as the variety of display options and controls. The key class in the maps library is map view and a subclass of view group is in the android standard library. When the map view has focus, it can capture key presses and touch gestures to pan and zoom the map automatically, including handling network requests for additional map files. It also provides all of the UI elements necessary for users to control the map.

12. PERFORMANCE IMPROVEMENT

The performance improvement of the sdk 1.6 is very high and it didn't get lose memory due to the new software

13. REFERENCES

- [1] Van Riper, A. Bowdoin (2002). Science in popular culture: a reference guide. Westport: Greenwood Press. p. 10. ISBN 0-313-31822-0.
- [2] Dinello, Daniel (2005). Technophobia!: Science Fiction Visions of Posthuman Technology. University of Texas Press. p. 109.
- [3] Shilov, Anton (07 July 2009). "ARM Doubts Android is Good Enough for Netbooks, Smartbooks". X-bit labs. <http://www.xbitlabs.com/news/mobile/display/20090707162820.html>
- [4] "Android 1.5 Version Notes". Android Developers, April 2009. <http://developer.android.com/sdk/android-1.5.html#features>. Retrieved 2009-09-03
- [5] Android Overview. Open Handset Alliance. http://www.openhandsetalliance.com/android_overview.htm Retrieved 2008-09-23
- [6] Wii technical notes- Nitendo of Europe.
- [7] Apache Felix on Androids written by Marcel Offermans and Christian van Spaandonk
- [8] Casamassina, Matt (2006-09-12). "IGNcube's Nintendo "Revolution" FAQ" Retrieved 2006-09-07.
- [9] GameSpy Staff (2003-07-30). "Beginner's Guide: GameCube". GameSpy. Retrieved 2006-09-07.
- [10] Nintendo of America. "Repair Form for U.S. Residents". Nintendo of America. Retrieved 2008-03-09.
- [11] Electronic Arts (2008-01-31). "Supplemental Segment Information" (PDF). *Thomson Financial*.p. 4. Retrieved 2008-02-09.
- [12] "Wii — Frequently Asked Questions". Nintendo. Retrieved 2008-03-31.