

Pengembangan Aplikasi Daftar Perencanaan Kegiatan Berbagi Pakai pada Sistem Operasi Android

Kadek Anggrian Mahendra Putra, Umi Laili Yuhana, dan Rully Soelaiman
Teknik Informatika, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember (ITS)
Jl. Arief Rahman Hakim, Surabaya 60111 Indonesia
e-mail: yuhana@cs.its.ac.id

Abstrak—Teknologi pada perangkat bergerak sudah banyak melahirkan berbagai aplikasi perangkat bergerak untuk memajemen aktivitas manusia. Penelitian ini bertujuan untuk membangun sebuah aplikasi manajemen perencanaan kegiatan yang dapat mendistribusikan perencanaan kegiatan dengan banyak pengguna. Aplikasi ini dilengkapi dengan fitur perintah suara, penentuan prioritas berdasarkan nilai bobot dari kegiatan tersebut dengan menggunakan algoritma TF-ICF (*Term Frequency – Inverse Corpus Frequency*), serta pengurutan perencanaan kegiatan berbasis *location-aware*. Aplikasi ini dibangun dalam 2 bagian, yaitu sisi klien pada sistem operasi Android, dan sisi *server* sebagai pusat data dan kontrol. Media komunikasi yang digunakan adalah internet.

Pengguna dapat memasukkan data kegiatan baru menggunakan *keyboard* atau perintah suara dari aplikasi klien. Data tersebut lalu dikirimkan ke *web service* (sisi *server*). Aplikasi klien akan menampilkan data secara berurut berdasarkan lokasi terdekat (menggunakan GPS), waktu terdekat, serta nilai bobot terkecil.

Pada penggunaannya, aplikasi klien akan mengeluarkan suara pemberitahuan dan memunculkan notifikasi pada layar perangkat bergerak untuk memberitahu pengguna ketika pengguna berpindah lokasi dan di lokasi tersebut terdapat beberapa data rencana kegiatan miliknya, dan juga ketika sebuah perencanaan telah mendekati *deadline*-nya. Pengguna juga dapat mengundang pengguna lainnya untuk menjadi peserta atau pengamat dalam kegiatannya. Suatu perencanaan kegiatan dapat ditandai sebagai telah selesai dikerjakan atau masih ditunda, memiliki anak kegiatan lainnya, serta dapat disisipkan dengan arsip yang berkaitan.

Kata Kunci—Android, Daftar Kegiatan, Metode Pembobotan, Pengenalan Suara, Sadar Lokasi

I. PENDAHULUAN

SEORANG aktivis memiliki berbagai macam kegiatan yang tidak boleh sampai terlewatkan. Beragam jenis perencanaan kegiatan, waktu, dan lokasi yang telah ditentukan harus segera diselesaikan sehingga menuntut seseorang untuk memiliki kemampuan memajemen waktu untuk memajemen perencanaan kegiatan yang dimilikinya, terutama aktivitas yang dikerjakan secara berkelompok. Bagi seseorang yang buruk dalam memajemen waktu, tentu faktor ini akan menjadi hambatan bagi aktivitasnya, bahkan dapat menurunkan produktivitas kelompok kerjanya. Lain halnya pada seseorang yang pintar dalam memajemen waktu namun berada di tempat yang padat kendaraan, tentu

waktu perjalanan akan menjadi pertimbangan ekstra dalam menentukan aktivitas mana yang harus diprioritaskan.

Di era teknologi yang semakin maju, beragam aplikasi pintar diciptakan untuk memudahkan dan mempercepat pengerjaan aktivitas manusia, bahkan dapat menggantikan peran manusia untuk mengerjakan tugasnya secara mandiri dan dinamis. Aplikasi-aplikasi pintar tersebut dapat menjadi asisten pribadi sehari-hari yang membantu dirinya untuk mengelola dan mengerjakan sebagian pekerjaannya.

Saat ini sudah banyak aplikasi daftar perencanaan kegiatan untuk perangkat bergerak, khususnya pada sistem operasi Android. Satu di antaranya adalah Any.DO, sebuah aplikasi daftar perencanaan kegiatan yang memiliki fitur unggulan seperti terhubung dengan Google Tasks, memiliki antarmuka yang menarik dan sederhana, membagi daftar kegiatan dengan pengguna lainnya, pilihan *widget*, menambahkan pengingat, tambahan tema, fitur *pop-up* pesan baru dan panggilan tak terjawab. Namun aplikasi tersebut dan aplikasi sejenisnya hanya dapat mencatat kegiatan dari Google Tasks API, membagikan kegiatan, dan menggunakan fitur *voice search* pada Android. Pada penelitian ini akan mencoba menggabungkan fitur lainnya seperti penggunaan Google Calendar API, Google Maps API, *voice command*, *location-aware*, *push notification* dari Google Cloud Messaging, dan Google Drive.

II. KAJIAN PUSTAKA

A. To-do List

Makalah tentang *Task Manager* [1] menjelaskan konsep dasar sebuah aplikasi daftar perencanaan kegiatan yang dapat dibagi, di mana pada dasarnya memiliki bagian-bagian kerangka seperti komponen, atribut, dan operasi-operasinya yang telah umum digunakan. Beberapa bagian kerangka akan diterapkan pada penelitian dengan uraian sebagai berikut.

1. Komponen

- Penugasan: komponen utama pada daftar perencanaan kegiatan yang dibagi menjadi beberapa anak penugasan lainnya dan berhubungan dengan penugasan/dokumen lainnya.
- Pengguna: terdiri dari administrator dan pengguna terdaftar. Pengguna terdaftar dapat mengundang pengguna terdaftar lainnya untuk bergabung dalam penugasan yang sama sebagai peserta atau pengamat. Peserta yang tergabung dalam sebuah penugasan memiliki hak akses membaca dan mengubah atribut

penugasan seperti tanggal mulai penugasan dan tanggal selesai penugasan. Seorang pengamat hanya memiliki hak akses membaca saja berupa data-data pesan atau arsip dari sebuah penugasan yang dimilikinya.

- Dokumen: sumber informasi tambahan yang bersifat opsional untuk menunjang aktivitas pengguna, dapat dibagikan dengan pengguna terdaftar lainnya.

2. Atribut

- Judul Penugasan: bersifat *mandatory*, wajib memiliki sebuah judul yang mendeskripsikan secara singkat dari penugasan dan tujuannya.
- Status Penugasan: menjelaskan informasi keadaan penugasan yang terakhir, misalnya status *finished*, *not finished*, *pending*, dan *not pending*.
- Pelaku Penugasan: orang yang bertanggung jawab menyelesaikan tugas tersebut, memberikan informasi bahwa siapa saja yang bertanggung jawab terhadap penugasan tersebut.
- *Deadline* Penugasan: batas akhir dari waktu pengerjaan sebuah penugasan. Sistem hanya dapat mengingatkan *deadline* dari sebuah penugasan dan tidak akan memberikan sebuah aksi ketika penugasan telah mencapai *deadline*-nya
- Tanggal mulai Penugasan: memberitahu peserta kapan penugasan akan dimulai.
- Relasi Penugasan: relasi dengan penugasan lainnya dan/atau dokumen lainnya.
- Lampiran Penugasan: informasi tambahan yang dapat menunjang aktivitas penugasan.

B. Algoritma Term Frequency – Inverse Corpus Frequency

Term Frequency dan *Inverse Corpus Frequency* (TF-ICF) adalah algoritma pembobotan kata yang digunakan untuk menghitung nilai bobot setiap kata berdasarkan frekuensi kata tersebut dalam kumpulannya [3]. Algoritma ini merupakan gabungan dari dua algoritma, yaitu *Term Frequency* dan *Inverse Corpus Frequency*. Algoritma ini diterapkan dalam aplikasi untuk menentukan prioritas kegiatan dalam suatu daftar kegiatan.

Ide penggunaan metode pembobotan kata sudah pernah diterapkan sebelumnya pada sebuah aplikasi pengingat kegiatan “*TaskMinder*” [2]. Perhitungan nilai bobot pada algoritma TF-ICF menghitung nilai bobot setiap kata pada sebuah nama perencanaan kegiatan. Perhitungannya dengan menggunakan (1).

$$W_{ij} = \log(1 + f_{ij}) \times \log\left(\frac{N+1}{n_j+1}\right) \quad (1)$$

Keterangan:

- W_{ij} = nilai bobot kata j pada perencanaan kegiatan i
- f_{ij} = jumlah munculnya kata j pada perencanaan kegiatan i
- N = jumlah total perencanaan kegiatan
- n_j = jumlah perencanaan kegiatan di mana kata j muncul paling tidak sekali.

Kemudian nilai bobot rata-rata (W_i) dihitung setelah nilai bobot setiap kata didapatkan. Nilai tersebut nantinya yang akan dipakai sebagai nilai bobot dari suatu perencanaan kegiatan. Perhitungan nilai bobot rata-rata dihitung dengan (2).

$$W_i = \frac{(w_{ij1} + w_{ij2} + \dots + w_{ijN})}{N} \quad (2)$$

Keterangan:

- W_i = bobot perencanaan kegiatan i
- w_{ij1} = bobot kata pertama (kata j_1) pada perencanaan kegiatan i dihitung dengan TF-ICF
- N = total kata yang terdapat di perencanaan kegiatan i

Pengurutan daftar perencanaan berdasarkan nilai bobot merupakan solusi dari suatu kondisi di mana pada saat proses pengurutan ditemukan 2 atau lebih kegiatan dengan waktu akhir pengerjaan (*deadline*) yang sama, serta jarak lokasi terdekat yang sama pula. Dengan begitu, pengguna dapat memprioritaskan kegiatannya berdasarkan frekuensi kegiatan tersebut telah dilakukan.

C. Teknologi Location-Aware

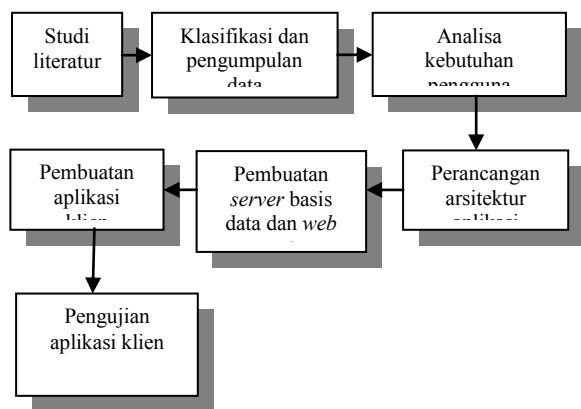
Teknologi *location-aware* adalah sebuah bentuk umum dari teknologi yang dapat menentukan posisi secara geografi. Contoh umumnya adalah *Global Positioning System* (GPS), sebuah sistem teknologi navigasi yang telah tertanam pada ponsel-ponsel pintar, terutama ponsel berbasis Android. GPS menggunakan satelit untuk menentukan posisinya pada permukaan bumi, lalu ditampilkan dalam bentuk sebuah peta [4]. Komponen GPS telah wajib hadir di setiap perangkat bergerak berbasis Android, membuat pengguna sudah tidak asing lagi dengan informasi berbasis lokasi.

Informasi lokasi ini digunakan dalam penelitian untuk membantu membuat konteks dalam penyajian data menjadi bentuk lain, sehingga tidak semua data yang ada akan ditampilkan serta membantu pengguna untuk memahami informasi berdasarkan lokasi.

D. RESTful API

RESTful API adalah sebuah *web service* yang diimplementasikan dengan menggunakan HTTP dan prinsip REST (*Representational State Transfer*) [5], dan sumber dayanya tersimpan di penyimpanan data. *Web service* ini digunakan sebagai media pertukaran data antara sisi klien dengan sisi *server* yang menyimpan basis data. Klien dapat mengirimkan permintaan dan *server* akan memproses permintaan tersebut (seperti permintaan membuat, menerima, merubah, dan menghapus sumber daya). Setelah *server* selesai melakukan pemrosesan permintaan, *server* akan mengirimkan respon menuju klien sebagai hasil dari selesainya sebuah aksi. Format data yang dihasilkan dapat berupa xml atau json. Kedua format ini sangat umum digunakan dalam bidang pertukaran data dan didukung oleh banyak bahasa pemrograman. Format ini sangat ringan dan terbukti lebih hemat penggunaan memori saat melakukan pertukaran data. Aplikasi klien hanya perlu membaca format ini sehingga dapat diolah kembali untuk dipergunakan.

Dalam penelitian ini, pembangunan RESTful API mengembalikan data dengan format json. Format ini dipilih karena terbukti lebih cepat dan membutuhkan sedikit sumber daya dibandingkan dengan penggunaan xml berdasarkan berbagai penelitian [6]. Format json juga telah didukung oleh sistem operasi Android dengan menggunakan *library* yang sudah cukup banyak tersedia.



Gambar. 2. Tahap-tahap Pengerjaan Aplikasi.

E. Android OS

Android adalah sistem operasi yang bersifat *open source*. Sistem operasi ini memiliki lisensi Apache yang sangat terbuka dan bebas. Android awalnya dikembangkan oleh Android Inc., kemudian diakuisisi oleh Google pada tahun 2005 sehingga menjadi salah satu bagian dari produk Google [7]. Sifat *open source* ini membuat daya tarik tersendiri bagi para pengembang aplikasi perangkat bergerak untuk mencoba menciptakan aplikasi hasil karyanya. Berbagai macam dokumentasi resmi dan tutorial pengembangan dari para pengembangnya tersedia dan sangat mudah dijangkau. Hal ini membuat sistem operasi Android menjadi sistem operasi yang sangat populer bagi berbagai produsen perangkat bergerak untuk meraih konsumen [8].

III. METODOLOGI

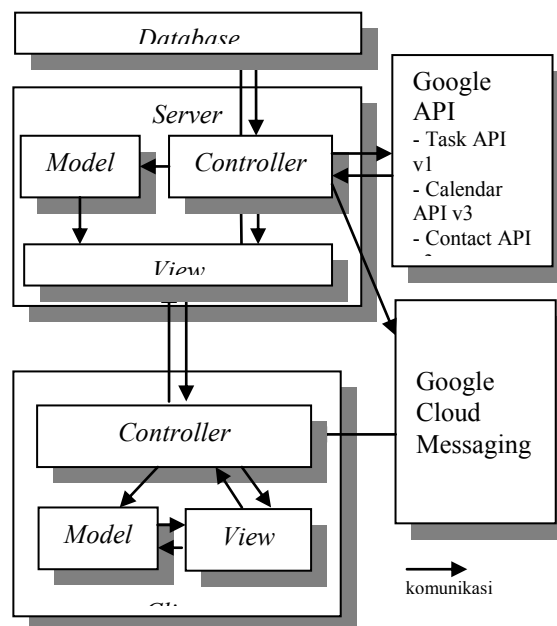
Pada Gambar. 1 dijelaskan tahap-tahap pengerjaan sistem dibagi menjadi 7 tahap. Masing-masing tahap dijelaskan sebagai berikut.

A. Studi Literatur

Literatur yang dipelajari adalah konsep dasar dan kerangka umum dari aplikasi perencanaan kegiatan (makalah *Task Manager* [1] dan makalah pendukung penelitian [2], penerapan teknologi Google API pada penelitian, serta tutorial pendukung lainnya. Literatur yang digunakan sebagian besar berasal dari materi perkuliahan, artikel *tutorial*, *e-book*, forum *online*.

B. Klasifikasi dan Pengumpulan Data

Data yang digunakan sebagai masukan adalah data masukan dari pengguna (pengguna memasukkan data sendiri saat menggunakan aplikasi), data jadwal kegiatan dari akun Google Calendar pengguna (seluruh daftar jadwal kegiatan pengguna beserta *deadline* yang telah tersimpan pada akun pengguna), data perencanaan kegiatan dari akun Google Tasks pengguna (seluruh daftar perencanaan kegiatan yang telah tersimpan pada akun pengguna), dan data kontak dari akun Google Contacts pengguna (seluruh daftar kontak yang telah tersimpan pada akun pengguna).



Gambar. 1. Perancangan Arsitektur Sistem Aplikasi.

C. Analisis Kebutuhan Pengguna

Pada tahap ini dilakukan penggalan kebutuhan pengguna seperti kebutuhan apa saja yang diperlukan untuk dapat membantu penggunanya sehingga permasalahan utama dapat terselesaikan. Kebutuhan pengguna ini digambarkan ke dalam sebuah diagram kasus penggunaan yang menggambarkan hubungan antara pengguna dan fungsionalitas sistem.

D. Perancangan Arsitektur Sistem

Pada tahap ini dilakukan analisa rancangan arsitektur sistem yang digambarkan seperti pada Gambar. 2. Aplikasi utama merupakan sebuah aplikasi Android yang dipasang di perangkat bergerak pengguna. Aplikasi ini dapat dicopot dari perangkat bergerak pengguna. Untuk mempermudah penyimpanan data, maka dibangun sisi *server* untuk menangani masalah penyimpanan data. Jalur komunikasi antara klien dan *server* menggunakan internet.

Pada sisi *server* dibangun sebuah *web service* yang berfungsi sebagai gerbang komunikasi antara klien dan *server*. Penyimpanan basis data di sisi *server* memberi keuntungan penggunaannya untuk dapat langsung berpindah dari satu perangkat bergerak ke perangkat bergerak lainnya tanpa perlu melakukan perpindahan data.

Penelitian ini menggunakan layanan Google Cloud Messaging untuk menangani segala pemberitahuan perubahan data di *server* ke setiap perangkat bergerak yang terkait. Ini mempermudah pengguna dalam mendapatkan informasi terbaru secara cepat di waktu yang sama.

E. Pembuatan Server Basis Data dan Web Services

Pada tahap ini dilakukan pembangunan sisi *server* dengan menggunakan perkakas pembangunan Netbeans IDE 7.0.1 dengan bahasa pemrograman Java. Sisi *server* yang dibangun adalah *web services* sebagai jalur komunikasi antara aplikasi klien dan *server*, *controller* sebagai jalur komunikasi *server* (menuju GCM, Google API dan *server* basis data) dan *server* basis data sebagai penyimpanan data. Contoh data sebagai

masukkan pada sisi *server* yaitu data dari layanan Google API (Calendar API, Task API, dan Contact API dalam format json) dan data permintaan dari klien (dalam format json). Data keluaran dari *server* adalah data dalam format json.

F. Pembuatan Aplikasi Klien

Pada tahap ini dilakukan pembangunan aplikasi klien menggunakan perangkat pembangunan Android Developer Tools (Eclipse v21.0.0-531062) dengan bahasa pemrograman Java Android. Antarmuka-antarmuka yang dibangun yaitu antarmuka untuk menampilkan daftar perencanaan kegiatan (dalam bentuk daftar dan peta), antarmuka untuk memasukkan data perencanaan baru, antarmuka untuk merubah data perencanaan kegiatan yang telah ada, dan antarmuka pengaturan aplikasi. Fitur-fitur yang akan diterapkan yaitu penggunaan teknologi sadar lokasi (menggunakan GPS), perintah suara (menggunakan *speech recognition*), dan notifikasi secara *real-time* (menggunakan Google Cloud Messaging). Data masukan pada aplikasi klien ini berupa data teks dari pengguna (nama kegiatan, tanggal, daftar peserta yang bertanggung jawab), data lokasi (titik koordinat), data arsip (berformat docx, xlsx, pptx, pdf, dan format lainnya), dan data suara pengguna (perintah suara). Data keluarannya berupa daftar perencanaan kegiatan, data lokasi berupa peta, dan suara.

G. Pengujian Aplikasi Klien

Pada tahap ini akan dilakukan pengujian terhadap sistem yang telah dibuat, mengamati kinerja sistem, serta mengidentifikasi kendala yang mungkin timbul. Pengujian dilakukan dengan metode *black box* (untuk menguji ketepatan pemberian nilai bobot), metode *white box* (untuk menguji kebenaran kasus penggunaan), dan survei pengguna untuk menguji pendistribusian informasi ke pengguna lainnya.

IV. EVALUASI DAN PENGUJIAN

Pengujian yang dilakukan meliputi pengujian kebutuhan fungsionalitas, pengujian kebutuhan non-fungsionalitas, pengujian validasi dan juga pengujian subyektifitas. Pengujian dilakukan pada lingkungan pengujian yang telah ditentukan. Setelah seluruh pengujian dilaksanakan, maka dilakukan evaluasi terhadap hasil akhir dari pengujian. Hasil evaluasi dari pengujian adalah sebagai berikut.

1. Fungsional aplikasi untuk memberi nilai bobot pada tiap perencanaan kegiatan telah berfungsi dengan baik dan sesuai dengan yang diharapkan.
2. Fungsional aplikasi untuk menyajikan prioritas kegiatan secara tepat sebagian besar sesuai. Nilai kesesuaian yang didapatkan dari 5 orang penguji adalah 75% sesuai.
3. Fungsional aplikasi untuk berhubungan dengan layanan Google bekerja dengan cukup handal. Nilai kehandalan yang didapat dari 5 orang penguji adalah 75% handal.
4. Fungsional aplikasi sebagai aplikasi berbagi informasi perencanaan kegiatan bekerja dengan cukup tepat. Nilai ketepatan yang didapatkan dari 5 orang penguji adalah 90% tepat.
5. Fungsional aplikasi untuk mengurutkan daftar perencanaan berdasarkan lokasinya sebagian besar sesuai dengan kondisi nyata.

6. Fungsional aplikasi untuk mengoperasikan perintah suara bekerja dengan cukup tepat. Hasil pengujian menyimpulkan bahwa aplikasi dapat merespon perintah suara pengguna dengan ketepatan 75%.

V. KESIMPULAN/RINGKASAN

Dari hasil pengamatan selama perancangan, implementasi, dan proses pengujian yang dilakukan, kesimpulan yang dapat diambil berdasarkan rumusan masalah yang ada adalah sebagai berikut ini.

1. Daftar perencanaan kegiatan dapat diberikan nilai bobot dengan menggunakan algoritma TF-ICF. Algoritma ini cocok digunakan karena algoritma ini menghitung nilai pembobotan secara independen tanpa perlu mengetahui terlebih dahulu informasi-informasi lainnya. Tingkat kesesuaian dari hasil pengujian oleh 5 orang penguji adalah 75%.
2. Berdasarkan hasil pengujian penggunaan algoritma TF-ICF pada daftar perencanaan kegiatan, pemberian nilai prioritasnya disimpulkan tepat dan berjalan sesuai yang diharapkan.
3. Google API dan sistem dapat dihubungkan menggunakan *library* yang telah didistribusikan oleh Google. *Library* ini digunakan pada sisi *server*, sehingga segala jenis pertukaran data dilakukan oleh *server*. Fitur ini dapat dijalankan dengan hasil pengujian kehandalan sebesar 75%.
4. Perintah suara pada aplikasi klien diimplementasikan menggunakan fitur Recognizer Intent pada aplikasi. Berdasarkan pengujian aplikasi, perintah suara dapat bekerja dengan ketepatan 75%.

DAFTAR PUSTAKA

- [1] Thomas Kreifelts, Elke Hinrichs, and Gerd Woetzel. (1993, September) CiteSeerX. [Online]. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.38.2100&rep=rep1&type=pdf&ei=kbA1UdOxGcL7kgWM-IHgCQ&usq=AFQjCNEBxLEW_hfw-sXgECAUBpQnMa4mJg&sig2=03xV0bxxnii5FNiBrsq_3A&bvm=bv.43148975.d.dGI
- [2] Brian M. Laundry, Rahul Nair, Zach Pousman, and Tungare Manas. (2006) Rahul Nair. [Online]. http://www.rahulnair.net/files/taskminder_paper.pdf
- [3] Joel W. Reed et al. (2006, Desember) Oak Ridge National Laboratory. [Online]. <http://aser.ornl.gov/publications/ICMLA06.pdf>
- [4] Location Aware Tech - Education Technology Services. [Online]. http://ets.tlt.psu.edu/wp-content/uploads/psuets_locationawaretech.pdf
- [5] Roger L. Costello. XFront - tutorials and articles on XML and web technologies. [Online]. <http://www.xfront.com/REST-Web-Services.html>
- [6] Nurzhan Nurseitov, Michael Paulson, Randall Reynolds, and Clemente Izurieta. (2009) Montana State University. [Online]. <http://www.cs.montana.edu/izurieta/pubs/caine2009.pdf>
- [7] Mohamad Sani. (2010, Oktober) Tutorial Pengantar Pengembangan Aplikasi Android. [Online]. <http://referensi.dosen.narotama.ac.id/files/2011/12/Pengantar-Membuat-Aplikasi-Android.pdf>
- [8] Okto Hendrik Sinaga, Shelti E. S. Teresa, and M. Jamil. (2010, Oktober) Sheltiteresaes - just another wordpress.com site. [Online]. <http://sheltiteresaes.files.wordpress.com/2011/10/android.docx>