

LAPORAN PENELITIAN

No. 121/Pen/Informatika/I/2012

**APLIKASI *CONTACT BACKUP MANAGEMENT*
PADA TELEPON SELULAR**

Oleh:
Justinus Andjarwirawan
Andreas Handojo
Angela Feliciano Soenjaya

JURUSAN TEKNIK INFORMATIKA



**FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS KRISTEN PETRA
SURABAYA
2012**

HALAMAN PENGESAHAN

1. a. Judul : Aplikasi Contact Backup Management
Pada Telepon Selular
- b. Nomor Penelitian : 121/Pen/Informatika/I/2012
- c. Bidang Ilmu : Teknologi Perangkat Lunak
2. Ketua Peneliti
- a. Nama Lengkap dan Gelar : Justinus Andjarwirawan, M.Eng.
- b. Jenis Kelamin : Pria
- c. Pangkat/Golongan/NIP : -/3C/98031
- d. Jabatan Akademik : Asisten Ahli
- e. Fakultas/Jurusan : Teknologi Industri/Teknik Informatika
- f. Universitas : Universitas Kristen Petra
3. Tim Peneliti
- a. Nama Lengkap dan Gelar : Andreas Handojo, M.MT.
- b. Jenis Kelamin : Pria
- c. Pangkat/Golongan/NIP : Pembina/4B/00016
- d. Jabatan Akademik : Lektor
- e. Fakultas/Jurusan : Teknologi Industri/Teknik Informatika
- f. Universitas : Universitas Kristen Petra
4. Tim Peneliti
- a. Nama Lengkap dan Gelar : Angela Feliciana Soenjaya, S.Kom.
- b. Jenis Kelamin : Wanita
- c. Pangkat/Golongan/NIP : -
- d. Jabatan Akademik : -
- e. Fakultas/Jurusan : Teknologi Industri/Teknik Informatika
- f. Universitas : Universitas Kristen Petra
5. Lokasi Penelitian : Universitas Kristen Petra
6. Kerjasama dengan Instansi Lain
- Nama Instansi : -
- Alamat : -
7. Jangka Waktu Penelitian : Januari 2012 s/d Juni 2012
8. Biaya
- a. Sumber dari UK Petra : Rp. 2.150.000
- b. Sumber lain : -
- Total : -

Surabaya, 8 Juli 2011

Mengetahui,

Ketua Jurusan Teknik Informatika

Ketua Peneliti

Yulia, M.Kom.
NIP: 99036

Justinus Andjarwirawan, M.Eng.
NIP: 98031

Menyetujui,
Dekan Fakultas Teknologi Industri

Djoni Haryadi Setiabudi, M.Eng.
NIP: 85009

ABSTRAK

Kebutuhan untuk berinteraksi dengan banyak orang dapat menjadi masalah ketika daftar *contact* pada telepon selular hilang karena suatu hal tertentu. Dengan ingatan manusia yang terbatas, cukup susah dan membutuhkan waktu untuk mengembalikan *contact* tersebut seperti sedia kala. Oleh karena itu, *back-up and synchronizing contact* dibutuhkan untuk mengatasi masalah tersebut.

Proses *back-up and synchronizing contact* dilakukan dengan bantuan server melalui koneksi Internet. Di mana pengguna telepon selular harus terkoneksi ke sebuah *account* pada *website*. Aplikasi juga harus telah ter-*install* pada telepon selular. Pertama kali aplikasi berbasis Java ME ini me-*retrieve contact list* dari database sehingga pengguna dapat menandai data mana saja yang ingin di-*backup*. Kemudian, proses back-up dilakukan. Pada proses *synchronizing*, aplikasi men-*download* data *contact* dari *server* terlebih dulu. Setelah proses *download* selesai, *synchronizing* data *contact* dilakukan.

Aplikasi ini dapat berjalan sempurna pada telepon selular Nokia dengan system operasi S40 6th *Editon* dan S60, serta Sony Ericsson dengan Java *Platform* 8 dan terbaru. Penggunaan aplikasi ini dapat berjalan dengan baik dengan ditunjang koneksi internet yang memadai dengan spesifikasi telepon selular yang sesuai.

Kata Kunci:

Personal Information Management, Backup, Synchronizing.

ABSTRACT

The development of time and technology has changed social lifestyle nowadays. Cellular phone is not luxury anymore and has become the communication life style of most society. The problem appears when the cellular phone's contact list was lost because something as the people has the necessity to interact with each other. It quite difficult and need more time to restore as before. Back-up and synchronizing contact could overcome for something like that.

The back-up and synchronizing contact process are performed by an application together with a server and internet connection. A cellular phone user must connect to an account in the specific website and install application on the cellular phone. Then, the users may use the features provided by application which builds on Java ME. First, the application which is builds on Java Me, retrieves all the contact list from the database and enable the user to mark the data before the back-up process performed. After the application downloading the contact data from the server, synchronizing process performed.

The result of the experiments shows that this application works on Nokia with operating system S40 6th Edition and S60, also on Sony Ericsson with Java Platform 8 and latest well. This application works well supported by the availability and stability of internet connection with compatible cellular phone specification.

Keywords:

Personal Information Management, Backup, Synchronizing.

KATA PENGANTAR

Puji syukur ke hadirat Tuhan Yesus Kristus atas segala kasih dan karunia-Nya, sehingga pada akhirnya penulis dapat menyelesaikan Laporan Penelitian yang ditujukan pada Fakultas Teknologi Industri Universitas Kristen Petra Surabaya. Penulis menyadari bahwa tanpa bantuan dari berbagai pihak, baik dari masa penelitian sampai pada penyelesaian laporan ini sangatlah sulit bagi penulis untuk menyelesaikannya. Penulis mengucapkan banyak terima kasih kepada semua pihak yang terlibat dalam penelitian ini.

Surabaya, 8 Juli 2012

Penulis

DAFTAR ISI

HALAMAN PENGESAHAN	ii
ABSTRAK	iv
KATA PENGANTAR	vi
DAFTAR ISI	vii
DAFTAR TABEL	ix
DAFTAR GAMBAR	x
BAB 1. PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Perumusan Masalah	2
1.3. Tujuan	2
1.4. Ruang Lingkup	2
1.5. Metodologi Penelitian	4
BAB 2. TINJAUAN PUSTAKA	6
2.1. <i>Personal Information Management (PIM) in Java ME</i>	6
2.2. Spesifikasi Telepon Selular	7
2.3. <i>File Synchronization</i>	8
2.4. vCard (.vcf, v.card)	8
BAB 3. METODE PENELITIAN	10
3.1. Variabel Penelitian	10
3.2. Teknik Pengambilan Sampel	10
3.3. Teknik Pengumpulan Data	10
3.4. Teknik Analisa Data	10
BAB 4. HASIL DAN PEMBAHASAN	11
4.1. Penggunaan Sistem	11
4.2. <i>Download dan Installing Aplikasi</i>	11
4.3. <i>Back-Up My Contact</i>	12
4.4. <i>Download and Synchronizing</i>	14
4.5. Pengujian Sistem	15
4.5.1. Pengujian Proses Retrieve Data Contact dari Contact Database pada Telepon Selular	15

4.5.2. Pengujian Proses Back-up Data Contact Berdasarkan Hasil Seleksi User pada Nokia C6-00	16
4.5.3. Pengujian Proses Download dan Sinkronisasi Data Contact dari Server ke Telepon Selular pada Nokia C6-00	17
BAB 5. KESIMPULAN DAN SARAN	19
5.1. Kesimpulan	19
5.2. Saran	19
DAFTAR PUSTAKA	21
LAMPIRAN	23

DAFTAR TABEL

Tabel 2.1. Spesifikasi Telepon Selular Nokia	7
Table 2.2. Spesifikasi Telepon Selular Nokia	8

DAFTAR GAMBAR

Gambar 2.1. Major Classes dan Interfaces dari JSR 75 PIM API	6
Gambar 4.1. Tampilan Menu Utama Aplikasi	12
Gambar 4.2. Tampilan <i>Contact List</i> dan Menu <i>Back-up</i> dan <i>Alert Info</i>	14
Gambar 4.5. Tampilan <i>Synchronizing Form</i>	14
Gambar 4.6. Hasil Pengujian Proses <i>Retrieve Data Contact</i> pada (a) Nokia C6-00, (b) Nokia 5730	15
Gambar 4.7. Hasil Pengujian Proses <i>Retrieve Data Contact</i> pada (c) Sonny Ericsson K660i dan (d) Sonny Ericsson J210i Hazel ...	16
Gambar 4.8. Data <i>Contact</i> Hasil Seleksi <i>User</i> yang Akan Di- <i>back-up</i> pada Nokia C6-00	17

BAB 1

PENDAHULUAN

1.1. Latar Belakang Masalah

Telepon selular telah menjadi sesuatu yang bagi sebagian besar orang wajib untuk dimiliki. Hal ini berkaitan dengan tren gaya hidup dan kebutuhan akan mobilitas yang semakin tinggi dari hari ke hari. Sejak pertama kali dikembangkan hingga saat ini, fitur yang sampai sekarang masih dibutuhkan ada dalam telepon selular adalah untuk menelepon dan mengirim pesan singkat yang lebih dikenal dengan SMS (*Short Messaging Service*).

Kedua fitur telepon selular di atas sering digunakan oleh pengguna untuk tetap dapat *keep in touch* dengan kerabat, relasi atau siapapun juga. Pengguna perlu mengetikkan nomor yang dituju untuk menelepon atau mengirim SMS. Faktanya, bukan hanya satu atau dua orang saja yang perlu diingat-ingat nomor teleponnya. Keterbatasan manusia dalam mengingat sesuatu membuat pihak pengembang telepon selular untuk menambahkan fitur *phonebook* atau *contact*. Dengan fitur *contact* ini pengguna telepon selular dapat menyimpan daftar nama dan nomor telepon layaknya buku telepon sesungguhnya. Hal ini memudahkan para pengguna telepon selular untuk menghubungi orang-orang yang ingin dihubungi. Semakin hari, kapasitas penyimpanan buku telepon ini menjadi semakin besar.

Salah satu hal yang paling menjadi masalah bagi sebagian besar orang ketika terjadi sesuatu dengan telepon selular yang dimiliki adalah kehilangan daftar nomor telepon yang telah tersimpan dalam *contact* mereka. Hal ini juga menjadi hal yang cukup merepotkan ketika mengganti telepon selular dengan yang baru, di mana harus memindahkan daftar nomor telepon tersebut. Memakan banyak waktu dan terkadang sulit untuk melakukan pengecekan nomor mana saja yang sudah dipindahkan dan mana yang belum. Akibatnya, ada yang terlewat atau tersimpan dua kali.

Diperlukan sebuah cara untuk mem-*back-up* data nomor telepon tersebut yang dapat diakses sewaktu-waktu ketika dibutuhkan. Pengguna telepon selular dapat meng-*upload* data pada *contact* ke suatu *server* melalui sebuah *website*.

Koneksi ke Internet harus dilakukan melalui telepon selular tersebut itu sendiri. Sebuah aplikasi pada telepon selular tersebut membaca data *contact* dan mengambilnya untuk di-*upload*. *File* yang berhasil ter-*upload* disimpan dalam *database* dengan format tertentu. Pengguna dapat men-*download* data tersebut kembali. Data yang berhasil di-*download* dapat langsung digunakan tanpa harus repot-repot meng-*input*-kan karena adanya aplikasi pada telepon selular tadi. Aplikasi seperti ini diharapkan dapat menjawab permasalahan yang sering dihadapi berkaitan dengan penggunaan fitur buku telepon pada telepon selular.

1.2. Perumusan Masalah

Masalah-masalah yang ada dalam implementasi:

- Bagaimana mendapatkan *input* berupa data *contact* beserta formatnya dari suatu telepon selular.
- Bagaimana mengekspor data *contact* ke suatu server.
- Bagaimana menyimpan data yang di-*upload* lewat sebuah *website* dengan dengan efektif dan efisien pada suatu *database*.
- Bagaimana suatu telepon selular dapat membaca *file* tersebut dan secara otomatis mengonversi data yang didapat sesuai dengan format *contact* pada telepon selular tersebut serta menginputkan data tersebut pada buku telepon.

1.3. Tujuan

Tujuan dari penelitian ini adalah membuat suatu aplikasi yang dapat mem-*back-up* data *contact* dari suatu telepon selular (*source*) ke telepon selular lainnya serta otomatis meng-*insert*-kan data tersebut pada telepon selular tujuan (*destination*).

1.4. Ruang Lingkup

Ruang lingkup penelitian ini dibatasi pada:

- Data yang diolah hanya data *contact* dari suatu telepon selular.
- Aplikasi yang dibuat memiliki fitur sebagai berikut:
 1. *Back-up* data *contact*.

Melalui fitur ini *user* dapat melakukan *back-up* data *contact* ke suatu *server*.

2. *Insert data contact.*

Melalui fitur ini *user* dapat meng-*insert*-kan data *contact* yang telah di-*download* dari suatu *server*.

3. Sinkronisasi data *contact*.

Data *contact* yang akan di-*upload* maupun yang telah di-*download* akan mengalami sinkronisasi. Data yang di-*upload* disinkronisasi sesuai format penyimpanan di *server*. Data yang di-*download* disinkronisasi sesuai format telepon selular yang digunakan.

4. *Marking data contact.*

Fitur *marking* ini memungkinkan *user* untuk melakukan *marking* (penandaan atau seleksi data) sebelum melakukan *back-up* maupun *insert*.

5. *View data contact.*

6. *Update data contact.*

User dapat melakukan *update* pada data *contact* miliknya.

7. *Delete data contact.*

Data *contact* yang dirasa sudah tidak diperlukan dapat dihapus menggunakan fitur *delete*.

8. *Collision checking.*

Fitur ini akan melakukan pengecekan kemungkinan data kembar (dalam kasus ini adalah atribut nama) untuk mengantisipasi kemungkinan *redundance*.

9. *Combine/merge data contact yang memiliki atribut nama yang sama.*

10. *Tranformation field.*

Fitur *transformation field* memungkinkan *user* untuk memindahkan atau mentransformasi data dari suatu *field* ke *field* yang lain.

- a. Telepon selular yang digunakan harus mendukung teknologi *Personal Information Management (PIM)* - Java ME (JSR 75).

- b. Telepon selular yang digunakan harus mendukung fitur dan layanan yang memungkinkan untuk dapat melakukan koneksi dengan internet.
- c. Pengujian terhadap aplikasi dilakukan dengan menggunakan telepon selular sesuai ketentuan pada nomor 2 dan nomor 3 dengan telepon selular produksi Nokia C6-00, Nokia 5730 XpressMusic dan Sony Ericsson J210i Hazel, Sony Ericsson K660i.
- d. Desain dan pembuatan *website* menggunakan HTML, PHP dan CSS.
- e. Desain dan pembuatan *database* menggunakan MySQL.
- f. Desain dan pembuatan aplikasi menggunakan Java ME.

1.5. Metodologi Penelitian

Langkah-langkah pekerjaan:

1. Studi literatur tentang
 - a. PIM Java ME
 - b. Spesifikasi Telepon Selular
 - c. File Synchronization
 - d. vCard
 - e. CSV
2. Perencanaan dan pembuatan aplikasi
 - a. *Retrieve data contact*
 - b. Konversi data ke format yang ditentukan untuk penyimpanan sebelum data *di-back-up*
 - c. Pembuatan *website* dan *database* serta format penyimpanan data
 - d. Konversi data ke format *contact* pada telepon selular tujuan setelah *download* dan meng-*insert*-kan pada *phonebook*
3. Pengujian dan analisa aplikasi
 - a. Pengujian aplikasi yang telah dibuat
 - Pengujian hasil *retrieve* data, penyimpanan data konversi data, dan *insert* data
 - Pengujian pada beberapa jenis telepon selular produksi Nokia dan Sony Ericsson
 1. Analisa keberhasilan dan ke-*valid*-an proses

4. Pengambilan kesimpulan
Penyusunan laporan

BAB 2

TINJAUAN PUSTAKA

2.1. *Personal Information Management (PIM) in Java ME*

Telepon selular pada umumnya memiliki fitur buku telepon untuk menyimpan data nomor telepon. Informasi ini termasuk informasi personal yang tidak dapat diakses secara langsung oleh aplikasi MIDP. Informasi personal ini dapat diakses dan di-*manage* menggunakan PIM API (*Application Programming Interface*) for Java ME. Informasi personal lainnya yang dapat diakses adalah *calendar* yang biasa digunakan untuk menyimpan *event* penting dan juga *to-do-list*.

Gambar 2.1 menunjukkan hirarki dari *major classes* dan *interfaces* di dalam PIM API.

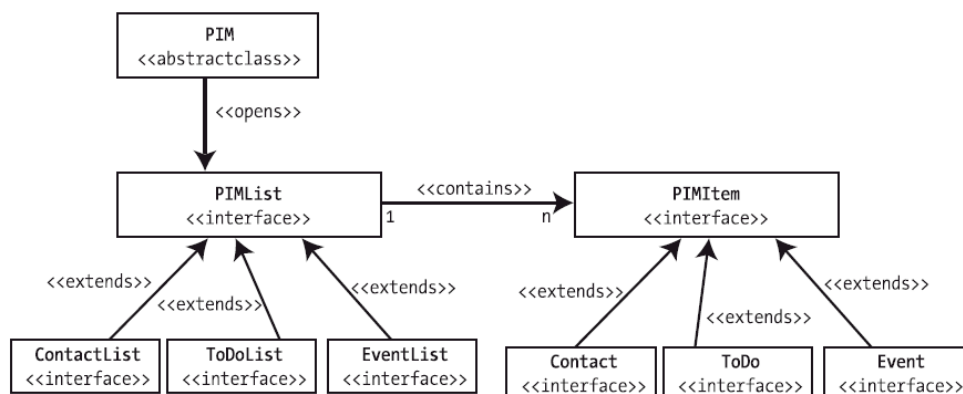


Figure 9-2. The major classes and interfaces of the JSR 75 PIM API

Gambar 2.1. Major Classes dan Interfaces dari JSR 75 PIM API

Sumber: Li & Knudsen (2005, p. 130)

PIM adalah sebuah obyek *singleton* yang digunakan untuk membuka *lists* dan *records*. PIM merepresentasikan PIM *database* dan *list* berisi *items* yang merepresentasikan *records* dalam *database*. *List* dan *item* diakses melalui PIMList interface dan PIMItem interface.

Sebelum menggunakannya, sebaiknya dilakukan pengecekan terlebih dahulu apakah PIM Optional Package ini tersedia, caranya:


```
String currentVersion = System.getProperty("microedition.pim.version");
```

Apabila *optional package* tidak tersedia fungsi akan mengembalikan nilai *null*. Sampai saat ini versi yang ada adalah "1.0".

Langkah pertama yang harus dilakukan untuk dapat menggunakan PIM API adalah memanggil sebuah *static method* untuk *me-retrieve singleton* PIM *instance*. PIM *instance* memberikan hak untuk mengakses PIM *lists/database* dan PIM *items/fields*. *Database* dapat dimanipulasi kemudian setelah *methods* *openPIMList()* dijalankan. Apabila *database* berhasil dibuka, *items* di dalamnya dapat diakses sesuai kebutuhan.

JSR 75 menyatakan *pimListType* dapat berupa PIM.CONTACT_LIST, PIM.EVENT_LIST, dan PIM.TODO-LIST. Berikutnya, untuk tipe akses mode dapat menggunakan salah satu dari PIM.READ_ONLY, PIM.READ_WRITE, atau PIM.WRITE_ONLY. Membuka *database* berdasarkan nama dan tipenya dapat dilakukan menggunakan method *openPIMList()* yang kedua.

Penghapusan data (dalam kasus ini digunakan *contact* sebagai contohnya) dilakukan satu per satu. *Reference* dari *item* yang akan dihapus di-*passing*-kan ke dalam *method* untuk menghapus. Penghapusan data sekaligus tidak dimungkinkan untuk dilakukan. Oleh karena itu harus dilakukan perulangan dengan *Enumeration* untuk menghapus semua data.

2.2. Spesifikasi Telepon Selular

Beberapa telepon selular digunakan dalam pengujian aplikasi ini. Spesifikasi telepon selular Nokia yang digunakan dapat dilihat pada Tabel 2.1.

Tabel 2.1. Spesifikasi Telepon Selular Nokia

Type	OS	Java Platform	PIM - JSR 75 Availability
Nokia C6-00	S60 5th Edition v.9.4	(unspecified)	Ok
Nokia E72	S60 3rd Edition FP 2 (S60 3.2.3) v9.3	(unspecified)	Ok
Nokia E63	S60 3.1 Edition v.9.2	(unspecified)	Ok
Nokia 5730 XpressMusic	S60 3rd Edition FP 2 v9.3	(unspecified)	Ok

Tabel 2.2. Spesifikasi Telepon Selular Sony Ericsson

Type	OS	Java Platform	PIM - JSR 75 Availability
Sony Ericsson J210i Hazel	(unspecified)	8.5	Ok
Sony Ericsson J108 Cedar	(unspecified)	8.5	ok
Sony Ericsson K660i	(unspecified)	8	ok
Sony Ericsson K810i	(unspecified)	7	ok
Sony Ericsson W700	(unspecified)	(unspecified)	(unspecified)
Sony Ericsson W8	Android	(unspecified)	(unspecified)

Spesifikasi untuk telepon selular BlackBerry mengenai Java Platform dan PIM – JSR75 tidak dapat diperoleh, bahkan pada situs resmi dari BlackBerry. BlackBerry memiliki Java SDK sendiri, yaitu Java SDK v6.0. dan juga Java *Development Environment* (JDE), yaitu Eclipse.

2.3. File Synchronization

File synchronization adalah sebuah proses untuk memastikan bahwa suatu *file* saling ter-update pada dua atau lebih lokasi dengan aturan tertentu. Secara garis besar ada dua macam *file synchronization*, yaitu *one-way file synchronization* dan *two-way file synchronization*. *One-way file synchronization* biasanya disebut *mirroring*. Pada tipe sinkronisasi ini updated file di-copy-kan dari lokasi asal ke satu atau lebih lokasi tujuan, tetapi tidak ada file yang di-copy-kan kembali pada lokasi asal. Sebaliknya, pada *two-way file synchronization* updated file di-copy-kan pada kedua lokasi asal dan tujuan. Hal ini dimaksudkan agar kedua lokasi tersebut memiliki data yang sama persis (*File Synchronization*, para. 1-2).

2.4. vCard (.vcf, v.card)

vCard ada format *file* standar untuk menyimpan informasi *business card* elektronik. Atribut yang terdapat pada *vCard* pada umumnya meliputi *name*,

address, phone number, email address, birthday, juga men-support custom fields seperti images. vCard sering kali digunakan untuk import dan export contact pada phonebook.

BAB 3

METODE PENELITIAN

3.1. Variabel Penelitian

Variabel penelitian diperoleh dengan langkah-langkah sebagai berikut:

- Melakukan pengamatan dengan keadaan yang sebenarnya.
- Melakukan studi literatur tentang *PIM*, *vCard* dan *Java ME*.
- Melakukan desain diagram alir untuk proses-proses dalam aplikasi.
- Melakukan pengambilan kesimpulan dan saran berdasarkan analisa keseluruhan hasil pengujian sistem.
- Melakukan penyusunan laporan.

3.2. Teknik Pengambilan Sampel

Pengambilan sampel dilakukan dengan menginstal aplikasi pada beberapa vendor telepon selular. Sebuah *server* disiapkan untuk menampung backup *contacts*. Hasil pengujian dibandingkan dengan data sebenarnya saat sebelum ditransfer baik yang ada di server maupun telepon selular.

3.3. Teknik Pengumpulan Data

Pengumpulan data dilakukan dengan cara melakukan pengujian ekspor, impor dan sinkronisasi *contacts*. Data adalah yang dapat diterima oleh server dan kembali lagi ke telepon selular yang sama maupun yang lain. Data di server dan telepon selular yang berbeda dapat dijadikan sebagai pembanding dalam pengujian.

3.4. Teknik Analisa Data

Data hasil rata-rata dari setiap pengujian dijadikan dasar untuk melakukan analisa tentang kinerja sistem dalam kondisi yang berbeda-beda. Hasil analisa disimpulkan pada bab lima.

BAB 4

HASIL DAN PEMBAHASAN

Pengujian dilakukan untuk melihat apakah proses *retrieve* data *contact*, proses *back-up* data *contact*, dan proses sinkronisasi berhasil dilakukan. Aplikasi diujicobakan pada empat telepon selular yang berbeda, yaitu Nokia C6-00, Nokia 5730, Sony Ericsson K660i, dan Sony Ericsson J210i Hazel.

4.1. Penggunaan Sistem

Setiap tahapan proses utama yang dijalankan oleh aplikasi ini memiliki tampilan yang berbeda. Tampilan dibuat dengan *form* dan *list*. Pada setiap *form* terdapat *command* untuk menjalankan menu yang terdapat pada *form* tersebut. Aplikasi ini juga menampilkan *alert* saat proses yang sedang dijalankan telah selesai atau menampilkan *exception*.

Tampilan *website* terbagi atas *page-page* sesuai dengan fasilitas yang ada. Data disajikan dalam bentuk tabel sehingga mudah dibaca. Tampilan untuk melakukan *insert* data baru maupun *edit* data lama disajikan dalam bentuk *form* isian.

4.2. Download dan Installing Aplikasi

Hal pertama yang harus dilakukan adalah men-*download* aplikasi ini dari *website* Sync IT!. Setelah berhasil men-*download*, *install downloaded file* pada telepon selular. Apabila proses instalasi berhasil dilakukan, hal yang perlu dilakukan selanjutnya adalah *account sign-up* (dijelaskan pada bagian selanjutnya).

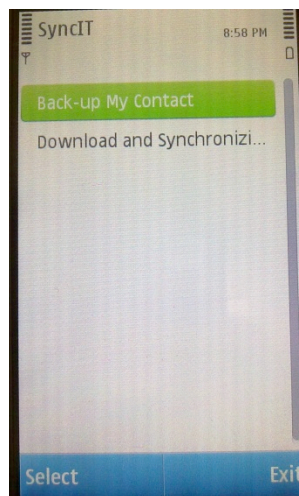
Pada halaman utama *website* ini terdapat beberapa fasilitas. Perhatikan kalimat “*Download the application here*” pada keterangan di bagian kanan halaman. Klik pada kata “*here*” dan *installer* aplikasi dapat langsung di-*download*.

Setelah *installer* aplikasi berhasil di-*download*, *install* aplikasi pada telepon selular. Apabila, aplikasi telah berhasil di-*install* pada telepon selular akan muncul icon aplikasi di lokasi di mana aplikasi di-*install*.

Sebelum proses *back-up* dan *sinkronisasi* dapat dilakukan, *user* terlebih dahulu harus melakukan *account sign-up* pada *website*. Fasilitas ini dapat diakses pada halaman utama *website*. *User* cukup mengisi *username* dan *password* pada *field* yang tersedia. Kemudian klik tombol “OK” setelah selesai pengisian. *Link* “*Main Page*” akan mengembalikan *user* pada tampilan halaman utama.

4.3. *Back-up My Contact*

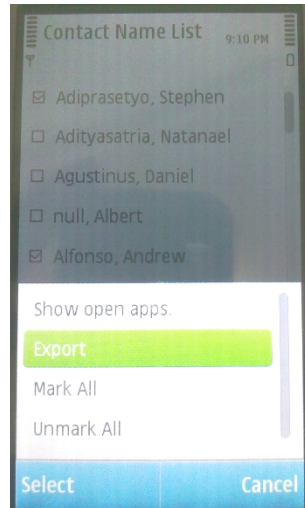
Setelah selesai meng-*install* aplikasi pada telepon selular dan *sign-up account* pada *website*, *user* dapat melakukan proses *back-up* data *contact* maupun sinkronisasi (dijelaskan pada bagian selanjutnya). Saat menjalankan aplikasi pada telepon selular, maka tampilan menu utama akan ditampilkan. Kemudian, pilih menu “*Back-up My Contact*” (tampilan dapat dilihat pada Gambar 4.1).



Gambar 4.1. Tampilan Menu Utama Aplikasi

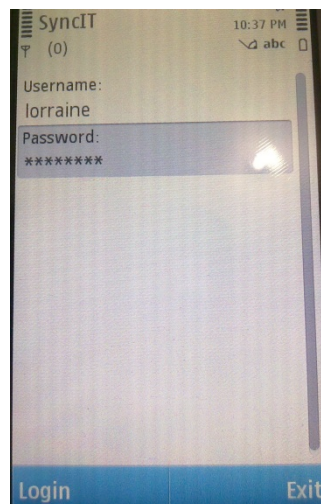
Selanjutnya, aplikasi akan menampilkan *contact list* yang ada pada telepon selular. *User* dapat memilih *contact* mana saja yang hendak di-*export* dengan menandai *checkbox* yang terletak di sebelah kiri *contact name*. *User* juga dapat melakukan *mark all/unmark all* dengan mengakses menu “*Option*”. Setelah

selesai melakukan penandaan proses *back-up* dilakukan dengan memilih menu “*Export*”. Tampilan untuk hal ini dapat dilihat pada Gambar 4.2.



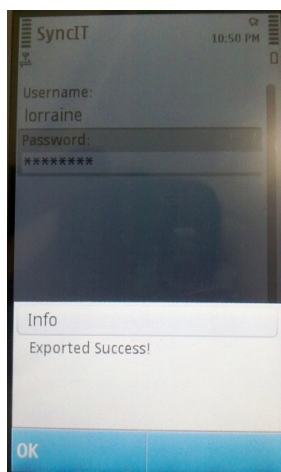
Gambar 4.2. Tampilan *Contact List* dan Menu *Back-up* dan *Alert Info*

Sebelum dapat melakukan proses *export*, user harus melakukan *login* terlebih dahulu seperti yang ditampilkan pada Gambar 4.3.



Gambar 4.3. Tampilan *Form Login*

Alert yang menyatakan bahwa proses ini telah selesai dilakukan akan muncul setelah semua *contact* yang ditandai berhasil di-*export*. Tampilan untuk hal ini dapat dilihat pada Gambar 4.4.

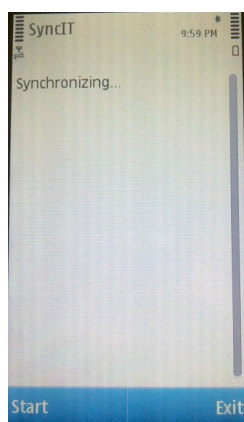


Gambar 4.4. Tampilan *Alert Info* jika Proses *Export* telah Selesai

4.4. *Download and Synchronizing*

Hal yang dapat dilakukan segera setelah proses instalasi pada telepon selular dan *sign-up account* pada *website* adalah *download and synchronizing*. Sama seperti *back-up my contact*, pilih menu *download and synchronizing* pada tampilan menu utama. Setelah itu *user* juga harus melakukan proses *login* terlebih dahulu.

Kemudian tampilan akan berubah menjadi *form synchronizing* setelah *user* berhasil melakukan *login*. Pilih menu “*Start*” untuk memulai proses. Setelah proses selesai aplikasi akan menampilkan *alert* yang menandakan bahwa proses *download and synchronizing* telah selesai. Tampilan untuk hal ini dapat dilihat pada Gambar 4.5.



Gambar 4.5. Tampilan *Synchronizing Form*

4.5. Pengujian Sistem

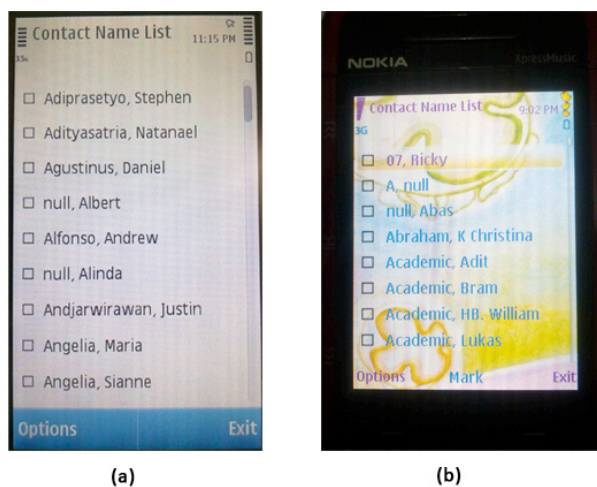
Pengujian dilakukan pada setiap proses-proses utama yang dijalankan oleh aplikasi. Adapun jenis-jenis pengujian yang akan dilakukan antara lain:

- Pengujian proses *retrieve* data *contact* dari *contact database* pada telepon selular dengan menggunakan PIM Interface.
- Pengujian proses *back-up* data *contact* berdasarkan hasil seleksi *user*.
- Pengujian proses *download* dan sinkronisasi data *contact* dari *server* ke telepon selular.

4.5.1. Pengujian Proses *Retrieve Data Contact* dari *Contact Database* pada Telepon Selular

Proses *retrieve* data *contact* memerlukan *permission* untuk dapat mengakses *contact database* sehingga dapat ditampilkan maupun digunakan untuk hal yang lain. Pengujian ini dilakukan dengan jenis telepon selular dan data *contact* yang berbeda. Adapun jenis telepon selular yang digunakan yang digunakan antara lain:

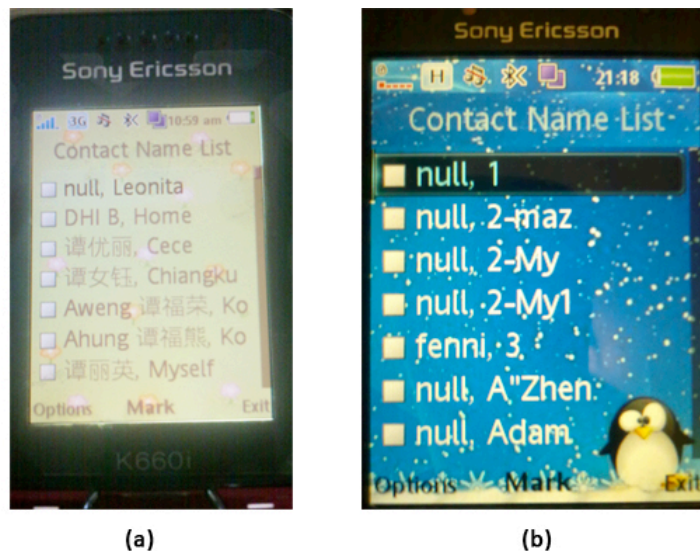
- Nokia C6-00.
- Nokia 5730.
- Sony Ericsson K660i.
- Sony Ericsson J210i Hazel.



Gambar 4.6. Hasil Pengujian Proses *Retrieve Data Contact* pada

(a) Nokia C6-00, (b) Nokia 5730

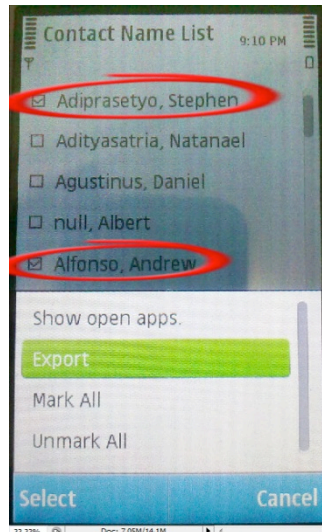
Pada Gambar 4.6 di atas, dapat dilihat bahwa proses *retrieve* data *contact* dari *contact database* pada telepon selular Nokia C6-00 dan Nokia 5730 berhasil dilakukan. Sedangkan hasil proses *retrieve* data *contact* pada *contact database* telepon selular Sonny Ericsson K660i dan Sonny Ericsson J210i Hazel dapat dilihat pada Gambar 4.7.



Gambar 4.7. Hasil Pengujian Proses *Retrieve* Data *Contact* pada (c) Sonny Ericsson K660i dan (d) Sonny Ericsson J210i Hazel

4.5.2. Pengujian Proses *Back-up* Data *Contact* Berdasarkan Hasil Seleksi *User* pada Nokia C6-00

Proses *back-up* terdiri dari beberapa proses. Pertama, aplikasi akan meng-*eksport* seluruh data *contact* berdasarkan hasil seleksi *user* ke sebuah *server*. Proses ini harus mendapatkan *permission* untuk dapat mengirim data melalui koneksi internet. Data yang dikirim dari telepon selular ini diterima oleh sebuah halaman *.php untuk kemudian diproses lebih lanjut. Proses ini meliputi pemilihan data dan konversi struktur sehingga dapat disimpan pada *database* di *server*. Kemudian, data yang sudah diolah disimpan ke dalam *database* di *server*.



Gambar 4.8. Data *Contact* Hasil Seleksi *User* yang Akan Di-*back-up* pada Nokia C6-00

Gambar 4.8. di atas menunjukkan proses *export* pada telepon selular Nokia C6-00. Proses *back-up* di atas berhasil dilakukan. Telah terlihat juga bahwa data *contact*, dalam kasus ini “Adiprasetyo Stephen” dan “Alfonso, Andrew” tersimpan dalam *database server*.

4.5.3. Pengujian Proses *Download* dan Sinkronisasi *Data Contact* dari *Server* ke *Telepon Selular* pada Nokia C6-00

Pengujian ini dilakukan untuk melihat apakah proses *download* dan sinkronisasi berhasil dilakukan. Bila ditemukan data yang sama persis, maka data yang dari *server* tidak akan di-*insert* ke dalam telepon selular. Proses *merge* akan dilakukan apabila ditemukan data *contact* dengan nama yang sama. Kemudian, apabila telepon selular tidak men-*support* suatu *field* tertentu, maka *field* akan di-*transform* ke *field* lain yang *support*.

Terlihat ada data baru pada *server* yang belum ada sebelumnya pada telepon selular Nokia C6-00. Data tersebut dalam kasus ini adalah data *contact* dengan nama “Saputra, Rean”. Setelah proses *download* dan sinkronisasi, maka data *contact* tersebut ditambahkan pada telepon selular Nokia C6-00.

Berdasar hasil pengujian, aplikasi dapat berjalan dengan sempurna pada telepon selular Nokia dengan sistem operasi S40 6th *Edition* dan S60, antara lain Nokia C6-00, Nokia 5730 XpressMusic, dan Nokia E63. Aplikasi tidak dapat dijalankan pada Nokia E72, meskipun bersistem S60 dan memiliki spesifikasi yang sama dengan Nokia 5730 XpressMusic. Hal ini mungkin disebabkan aplikasi gagal mengakses *array contact list* pada *PIM database*. Kegagalan akses ini mungkin disebabkan oleh *incompatible software specification* pada Nokia E72 yang digunakan dalam pengujian, mengingat pemilik tidak pernah melakukan *update* pada telepon selularnya.

Aplikasi juga berjalan dengan sempurna pada telepon selular Sony Ericsson dengan Java *Platform* 8 dan terbaru. Aplikasi *incompatible* pada Sony Ericsson tanpa atau dengan Java Platform versi sebelumnya sehingga tidak dapat dijalankan pada tipe tersebut. Pada telepon selular dengan sistem operasi Android aplikasi tidak dapat berjalan dan harus di-*built* ulang menggunakan (IDE) tersendiri.

Pada telepon selular BlackBerry, dalam pengujian ini digunakan BlackBerry Bold 9900 dan BlackBerry Curve 8520 (Gemini), Aplikasi tidak dapat berjalan dengan sempurna. Hal ini dikarenakan sistem *display interface* yang tidak sesuai. Untuk mengatasi masalah ini aplikasi harus di-*built* ulang menggunakan *Java Development Environment (JDE)* khusus untuk BlackBerry.

BAB 5

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Berdasarkan hasil pengujian dapat disimpulkan beberapa hal sebagai berikut:

- Aplikasi saat ini hanya bisa me-*retrieve* adalah *contact list* yang tersimpan pada *memory* telepon selular.
- Keberhasilan proses *back-up* data bergantung pada keberhasilan pengiriman data serta keberadaan dan kestabilan koneksi internet.
- Keberhasilan proses *download* bergantung pada keberadaan dan kestabilan koneksi internet.
- Aplikasi dapat berjalan seratus persen pada telepon selular produksi Nokia dengan sistem operasi S40 6th Edition dan S60 (kecuali pada Nokia E72). Hal ini mungkin disebabkan aplikasi gagal mengakses *array contact list* pada PIM *database*.
- Aplikasi dapat berjalan seratus persen pada telepon selular produksi Sony Ericsson dengan Java Platform 8 dan terbaru.
- Aplikasi tidak dapat berjalan dengan sempurna pada telepon selular BlackBerry dan harus di-*built* ulang menggunakan JDE khusus untuk BlackBerry.

5.2. Saran

Adapun beberapa hal yang dapat dijadikan sebagai saran dalam proses pengembangan selanjutnya, antara lain:

- Aplikasi yang sekarang belum men-support pemakaian karakter asing sebagai value dari sebuah field. Untuk dapat mengakomodasi pemakaian karakter asing diperlukan sistem *encoding* tertentu.
- *Contact item* dengan *field* bertipe data tertentu memerlukan sistem *encoding* tertentu sehingga dapat di-*back-up* pada database dan juga di-*input*-kan ke telepon selular dari *database*.

- Adanya peluang untuk mengembangkan sistem menjadi lebih efisien dan efektif sehingga proses dapat berjalan lebih cepat. Hal ini tentunya perlu didukung teknologi yang memadai (*hardware*, koneksi internet yang lebih cepat, dll).
- Pada telepon selular dengan system operasi Android, aplikasi harus di-built ulang menggunakan IDE yang lebih spesifik dan *compatible*.

DAFTAR PUSTAKA

- BlackBerry (2011). *BlackBerry - java application development*. Retrieved September 10, 2011, from <http://us.blackberry.com/developers/javaappdev/>
- BlackBerry (2011). *BlackBerry - learning resources*. Retrieved September 10, 2011, from <http://us.blackberry.com/developers/javaappdev/learningresources/>
- Developer Sony Ericsson (2011). *Cedar™ - developer world - phones - devworld - sony ericsson*. Retrieved September 10, 2011, from <http://developer.sonyericsson.com/wportal/devworld/phones/phone-overview/cedar?cc=gb&lc=en>
- Developer Sony Ericsson (2011). *Hazel™ - developer world - phones - devworld - sony ericsson*. Retrieved September 10, 2011, from <http://developer.sonyericsson.com/wportal/devworld/phones/phone-overview/hazel?cc=gb&lc=en>
- Developer Sony Ericsson (2011). *K660i - developer world - phones - devworld - sony ericsson*. Retrieved September 10, 2011, from <http://developer.sonyericsson.com/wportal/devworld/phones/phone-overview/k660i?cc=gb&lc=en>
- Developer Sony Ericsson (2011). *K810i - developer world - phones - devworld - sony ericsson*. Retrieved September 10, 2011, from <http://developer.sonyericsson.com/wportal/devworld/phones/phone-overview/k810i?cc=gb&lc=en>
- Firtman, Maximiliano Brian. (2009). *Programming the mobile web*. Sebastopol: O'Reilly Media, Inc.
- Fling, Brian. (2009). *Mobile Design and Development*. Sebastopol: O'Reilly Media, Inc.
- Forum Nokia Wiki. (2011). *How to use http POST request in Java ME - forum nokia wiki*. Retrieved February 15, 2011, from http://wiki.forum.nokia.com/index.php/How_to_use_Http_POST_request_in_Java_ME

- Forum Nokia Wiki. (2011). *How to write data to a file in Java ME - nokia developer wiki*. Retrieved February 15, 2011, from http://www.developer.nokia.com/Community/Wiki/How_to_write_data_to_a_file_in_Java_ME
- Li & Knudsen. (2005). *Beginning J2ME Platform: From Novice to Professional, 3rd Ed.* Berkeley: Apress.
- Nokia Europe (2011). *Nokia europe - nokia C6-00 - QWERTY phone specifications*. Retrieved September 10, 2011, from <http://europe.nokia.com/find-products/devices/nokia-c6-00/specifications>
- Nokia Europe (2011). *Nokia europe - nokia 5730 XpressMusic - specifications*. Retrieved September 10, 2011, from <http://europe.nokia.com/find-products/devices/nokia-5730xpressmusic/specifications>
- Nokia Europe (2011). *Nokia europe - nokia E72 - technical specifications, business phone with free gps*. Retrieved September 10, 2011, from <http://europe.nokia.com/find-products/devices/nokia-e72/specifications>
- Nokia Europe (2011). *Nokia europe - technical specifications - nokia E63 support*. Retrieved September 10, 2011, from <http://europe.nokia.com/support/product-support/e63/specifications>
- Ortiz, C. Enrique. (2010). *Managing personal information – an introduction to the PIM API for Java ME*. Retrieved June 10, 2010, from <http://developers.sun.com/mobility/apis/pim/pim1/>
- Wikipedia, the free encyclopedia. (2011). *File synchronization*. Retrieved September 10, 2011, from http://en.wikipedia.org/wiki/File_synchronization

LAMPIRAN

Biaya

1. Biaya Seminar BISSTECH 2011	Rp.	500.000
2. Transport (SUB-DPS-SUB)	Rp.	850.000
3. Transport lokal	Rp.	200.000
4. Akomodasi 2 malam	Rp.	600.000
TOTAL	Rp.	2.150.000