

PEMANFAATAN TEKNOLOGI E-MAIL DAN BLUETOOTH DALAM APLIKASI BLUE PUSH MAIL

Titin Fatimah¹⁾, Nurullah²⁾

Fakultas Teknologi Informasi, Universitas Budi Luhur
 Jl.Raya Ciledug, Petukangan Utara, Kebayoran Lama, Jakarta Selatan 12260, Indonesia
 E-mail : ¹⁾ titin.fatimah@budiluhur.ac.id, ²⁾ uyoh_jkt@yahoo.com

Abstract -- *The Internet has transformed the business information systems. Transformation that makes the system connected only in the internal network, becoming a system which can be accessed by the public. Transformation moving more faster, in line with popularity of mobile devices, such as mobile phones and PDAs. Besides the rapid development of hardware, the development of mobile applications are also increasingly supportive. Mobile applications experience a significant acceleration, when Sun Microsystems released the Java mobile edition. Now Java mobile edition became one of the standard features in mobile devices. These revolutionary developments, triggering proliferation of mobile business solutions. This solution enables mobile elements communicate with back-end systems in every place and time. Java has provided a framework that bridges between the back-end systems with mobile clients. So the choice of Java is one solution that should frame the development of use. Mobile-based applications are expected to provide more convenient facilities in exchanging information. By utilizing e-mail technology and Bluetooth, Blue Push Mail application created to help distribute information better, faster and precise.*

Keywords : *mobile device application, information exchange, E-mail, Bluetooth*

1. PENDAHULUAN

Dewasa ini dunia teknologi informasi mengalami perkembangan yang sangat pesat, bahkan telah mencakup hampir semua aspek kehidupan manusia. Kebutuhan manusia yang beragam mengakibatkan teknologi informasi maju berkembang. Salah satu kebutuhan pokok manusia adalah mereka ingin agar semua pekerjaannya dapat

diselesaikan dengan cepat sehingga waktu dan tempat tidak menjadi suatu halangan. Kebutuhan inilah yang membuat teknologi *mobile* semakin berkembang.

Dan saat ini hampir semua perangkat *mobile* sudah dilengkapi dengan teknologi Bluetooth, yang memungkinkan kita membangun jaringan kecil antar perangkat *mobile* tersebut. Dengan dukungan teknologi Bluetooth, pengguna perangkat *mobile* memiliki kemudahan untuk melakukan transfer data ataupun informasi antar perangkat.

Dengan telah meluasnya penggunaan perangkat *mobile* dengan dukungan Bluetooth, maka dapat dikembangkan sebuah aplikasi yang dapat memanfaatkan dukungan teknologi yang telah ada, sehingga dapat dihindari penggunaan biaya yang besar. Teknologi Bluetooth sudah menjadi pelengkap yang sangat umum pada setiap *mobile device* dewasa ini.

2. LANDASAN TEORI

a. Java

Java merupakan bahasa pemrograman yang awalnya dikembangkan oleh James Gosling di Sun Microsystem. Java pertama kali di-release pada tahun 1995 sebagai komponen utama dari Java *platform*.

Sun Microsystems telah mendefinisikan tiga *platform* Java yang masing-masing diarahkan untuk tujuan tertentu dan untuk lingkungan komputasi yang berbeda-beda:

- 1) *Standard Edition* (J2SE), didesain untuk jalan pada komputer desktop dan komputer *workstations*.
- 2) *Enterprise Edition* (J2EE), dengan *built-in* mendukung untuk Servlets, JSP, dan XML, edisi ini ditunjukkan untuk aplikasi berbasis *server*.

- 3) *Micro Edition (J2ME)*, didesain untuk piranti dengan memori terbatas, layar *display* terbatas dan kemampuan pemrosesan yang juga terbatas.

b. J2ME (Java 2 Micro Edition)

Kelahiran *platform* J2ME timbul karena dibutuhkan adanya sebuah *platform* komputasi yang mengakomodasi piranti *consumer electronics* dan *embedded*. Piranti ini dikelompokkan menjadi dua kategori yaitu:

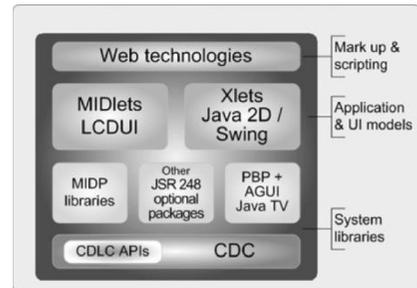
- 1) Personal, piranti *mobile* yang dapat digunakan untuk komunikasi melalui jaringan tertentu misalnya ponsel, *Personal Digital Assistants* (PDA), Palm, Pocket PC dan *organizer*.
- 2) Piranti informasi yang digunakan bersama dengan jaringan tetap (*fixed*), koneksi jaringan yang tidak putus-putus misalnya TV Internet, *high-end communicators*, dan sistem navigasi mobil.

Banyaknya jenis dan tipe peranti *mobile* membuat sulit pencapaian standar kinerja dan portabilitas. Meskipun J2ME menerapkan konsep *run everywhere*, pengembang J2ME menspesifikasikan beberapa arsitektur yang terbagi atas: konfigurasi, profil, dan paket opsi (*optional package*). Tujuan dan spesifikasi J2ME itu demi mencapai kinerja dengan memanfaatkan kelebihan piranti sekaligus mencapai portabilitas.

Konfigurasi adalah *virtual machine* yang menyediakan beberapa pustaka kelas. Konfigurasi menyediakan fungsi dasar dengan karakteristik yang sama. Contohnya: fungsi koneksi jaringan dan manajemen *memory*.

Sementara itu, *profile* menyediakan lingkungan pustaka-pustaka API untuk membangun aplikasi *mobile*. Paket opsi dibuat untuk menyediakan fungsi-fungsi pada piranti *mobile* yang lebih spesifik. Contohnya: piranti yang memiliki

akses Bluetooth memerlukan API Bluetooth.



Gambar 1 : Arsitektur J2ME

Dari Gambar 2.1, bisa dilihat bahwa J2ME memiliki dua konfigurasi *virtual machine*. Dua konfigurasi itu adalah:

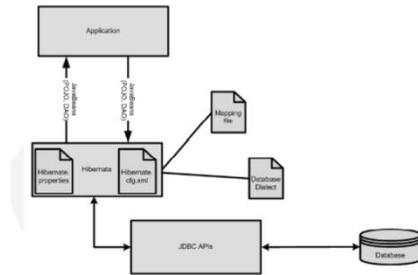
- 1) *Connected Limited Device Configuration* (CLDC) didesain untuk piranti *mobile* terkecil dengan 128 512 Kb memori, prosesor 16 32 bit. Profil dasar yang berjalan di atas CLDC adalah MIDP.
- 2) *Connected Device Configuration* (CDC) adalah konfigurasi *high-end* yang membutuhkan memori minimum 2 Mb dan prosesor 32 bit. Profil dasar yang berjalan di atas CDC adalah *foundation profile* (FP).

c. Hibernate

Hibernate adalah sebuah *tool mapping* objek/relasional *open source* untuk Java yang dapat membuat kita mengembangkan *persistent class* dan *persistent logic* tanpa perlu memikirkan bagaimana menangani data. *Hibernate* tidak hanya menangani *mapping* dari Java *class* ke tabel *database* (dan dari tipe data Java ke tipe data SQL), tetapi juga menyediakan data *query* dan fasilitas *retrieval* yang dapat secara signifikan mengurangi *development time* yang dihabiskan untuk *manual data handling* di SQL dan JDBC.

Hibernate mempermudah untuk membangun aplikasi *database* berkinerja tinggi dengan Java. Hanya dengan menuliskan POJO (*Plain Old Java Object*) sederhana, membuat file XML *mapping* yang mendeskripsikan *relationship* antara atribut *database* dan kelas dan memanggil

beberapa *Hibernate* API untuk me-load atau menyimpan *persistent object*.



Gambar 2:
Arsitektur Hibernate

d. *BlueCove*

BlueCove merupakan *library* Java untuk *Bluetooth* (JSR-82 *implementation*). *BlueCove* dapat bekerja dengan *driver* Mac OS x, WI DCOMM, *BlueSoleil*, *Bluetooth stack* pada Windows XP SP2 atau Windows Vista dan *Bluetooth stack* pada Windows *Mobile*.

BlueCove menyediakan Java interface JSR-82 untuk beberapa *profile* *Bluetooth*, diantaranya:

1) *Service Discovery Application Profile* (SDAP)

2) *Serial Cable Emulation Protocol* (RFCOMM)

Protokol RFCOMM merupakan padanan dari koneksi serial *port* RS-232 antar dua *device* dalam jaringan *wireless*.

3) *Logical Link Control and Adaptation Protocol* (L2CAP)

L2CAP adalah *multiplexing layer* yang memungkinkan *protocol level* tinggi atau aplikasi untuk menggunakan komunikasi *Bluetooth*.

4) *OBEX Generic Object Exchange Profile* (GOEP)

OBEX adalah *protocol* komunikasi yang digunakan untuk bertukar *binary data* antar *device*. OBEX dikembangkan oleh *Infrared Data Association*, oleh sebab itu OBEX juga dikenal sebagai IrOBEX atau *Infrared OBEX*.

e. *JavaMail API (Application Programming Interface)*

JavaMail API merupakan satu set API abstrak yang memodelkan sebuah sistem mail. *JavaMail* API merupakan paket opsional untuk J2SE. *JavaMail* menyediakan seluruh fungsi standar bagi para *developer* untuk membangun aplikasi klien untuk email, antara lain:

- 1) Membuat surat elektrik yang terdiri dari sekumpulan atribut *header* dan blok data dari beberapa tipe data yang dikenal dalam *Content-Type header*. *JavaMail* memanfaatkan Java *interface Part* dan Java kelas *Message* untuk mengartikan sebuah pesan surat elektrik.
- 2) Membuat objek sesi yang mengautentikasi identitas pengguna dan mengontrol akses terhadap tempat penyimpanan dan pengiriman pesan di *server*.
- 3) Mengirimkan pesan ke alamat email tujuan.
- 4) Menerima pesan dari tempat penyimpanan pesan di *server* (*message store*).

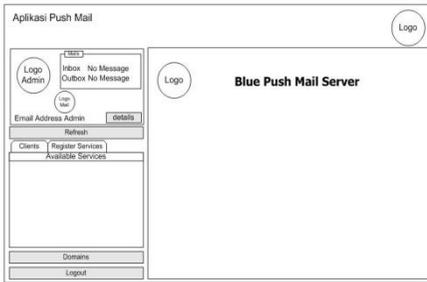
3. RANCANGAN APLIKASI

Aplikasi *Blue Push Mail* memiliki dua buah program, satu program untuk sisi *server* dan satu program untuk sisi *client*. Aplikasi *server* merupakan *threaded* program yang harus selalu aktif, agar proses *push mail* dapat berjalan dengan baik. Antara program *server* dengan program *client* akan terhubung dalam *wireless network* dengan menggunakan gelombang radio *Bluetooth*.

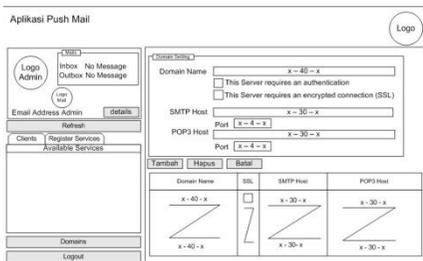
Agar jaringan *wireless* antara *server* dengan *client* dapat terbentuk, masing-masing *service Bluetooth* pada perangkat *mobile client* dan komputer *server* harus diaktifkan. Juga harus dipastikan bahwa setiap *service Bluetooth* dapat ditemukan (*visible*) oleh *service Bluetooth* yang lain.

Untuk memulai proses otomatisasi *push mail*, setiap alamat *Bluetooth* perangkat *mobile* harus didaftarkan terlebih dahulu pada *database server*. Program *server* dilengkapi dengan *service* aktif yang dapat mendeteksi lokasi alamat *Bluetooth* perangkat *mobile client*. Setiap *service* aplikasi *client* yang

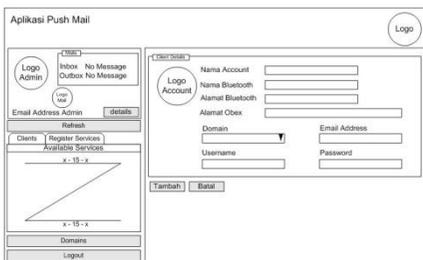
ditemukan akan dimasukkan ke dalam daftar *service* aktif. Gambar rancangan layar dapat dilihat pada gambar 3 sampai dengan gambar 9.



Gambar 3 : Rancangan Layar Menu Utama



Gambar 4 : Rancangan Layar Halaman Input Data Domain



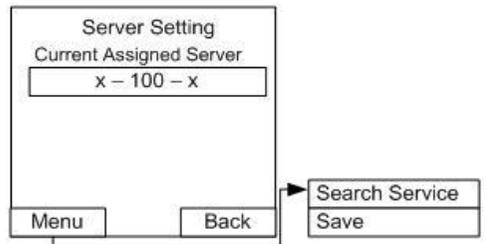
Gambar 5 : Rancangan Layar Halaman Register Client



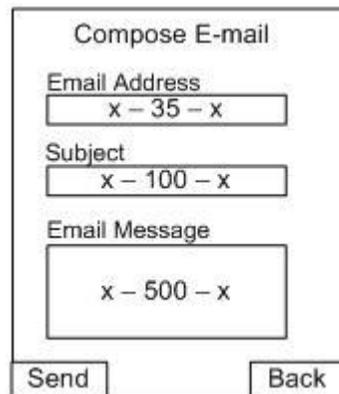
Gambar 6 : Rancangan Layar Halaman Kirim Email



Gambar 7 : Rancangan Layar Halaman Utama Client



Gambar 8 : Rancangan Layar Server Setting Client



Gambar 9 : Rancangan Layar Kirim Email Client

4. IMPLEMENTASI

Pada aplikasi Blue Push Mail ada dua tahap implementasi program yaitu program aplikasi Blue Push Mail *server* dan program kirim email yang berjalan pada *device handphone*.

Pengetesan utama pada *client* meliputi:

- Apakah program *client* dapat mendeteksi kehadiran Bluetooth *service* komputer *server*.
- Apakah aplikasi *client* dapat mengirimkan data *email* ke Bluetooth komputer *server*.

Pengetesan utama pada aplikasi Blue Push Mail di sisi *server* meliputi:

- Apakah *thread* POP3 pada *server* dapat membaca dan mengunduh data *email* pada *store mail server*. *Email* yang diunduh akan disimpan dalam *table* MailRecieve. Adapun *attachment* yang dilampirkan akan disimpan dalam *folder attachment* dan id referensinya akan disimpan dalam *table* MailAttachment.
- Apakah *thread* SMTP pada *server* dapat mengirimkan antrian *email* yang harus didistribusikan ke masing-masing alamat *email* tujuan. *Email* yang telah terkirim ke *store* tujuan akan diberikan tag terkirim.
- Apakah *thread* Bluetooth *service server* dapat mendistribusikan *email* dan *attachment* yang belum terkirim dalam *table* MailRecieve dan MailAttachment ke masing-masing *mobile device* terdaftar.

Pembuatan Aplikasi Blue Push Mail ini menggunakan bahasa pemrograman Java. Agar aplikasi Blue Push Mail dapat mengunduh *email*, PC tempat Aplikasi Blue Push Mail *server* dipasang harus memiliki koneksi internet aktif. Aplikasi Blue Push Mail *server* hanya akan mengirimkan (mem-*push*) *email* alamat Bluetooth *handphone* yang terdaftar dalam *database server* dan berada dalam jangkauan Bluetooth *server*. Untuk mendaftarkan alamat Bluetooth *handphone*, *service* Bluetooth

handphone harus diaktifkan. Aplikasi Blue Push Mail *server* akan mendeteksi keberadaan *service* Bluetooth *handphone client*.

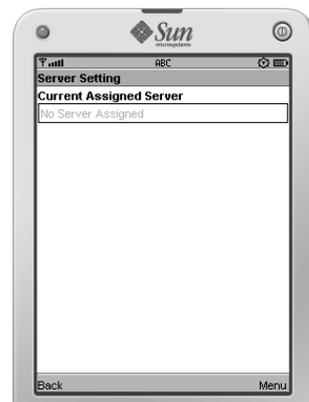
Tidak ada langkah khusus yang harus dilakukan agar aplikasi Blue Push Mail *server* dapat mengunduh dan mem-*push email* terdaftar. *Threaded* program pada aplikasi Blue Push Mail *server* akan secara berkala memeriksa status *email* pada *mail server*.

5. TAMPILAN LAYAR

Aplikasi Blue Push Mail *client* hanyalah sebuah aplikasi untuk mengirimkan *email* ke aplikasi Blue Push Mail *server* yang selanjutnya akan diteruskan (di-*push*) oleh aplikasi *server* ke alamat *email* yang dituju.



Gambar 10: Tampilan Aplikasi Blue Push Mail Client



Gambar 11: Tampilan Server Setting



Gambar 12: Tampilan Kirim Email

6. KESIMPULAN

- a. Dengan adanya aplikasi Blue Push Mail yang memanfaatkan koneksi Bluetooth antara server dengan client memberikan keuntungan dari segi pengurangan biaya untuk koneksi internet, karena koneksi internet hanya dibutuhkan oleh aplikasi *server*.
- b. Dengan adanya aplikasi Blue Push Mail, proses distribusi informasi menjadi lebih praktis.
- c. Karena menggunakan bahasa pemrograman Java, aplikasi Blue Push

Mail berjalan di berbagai macam *platform*.

- d. Dengan menggunakan Hibernate framework sebagai penghubung antara Java dengan database server, menjadikan aplikasi Blue Push Mail menjadi RDBMS independent software.

DAFTAR PUSTAKA

- [1] Eckel, Bruce, 2006, *Thinking in Java Fourth Edition*, United States : Pearson Education, Inc.
- [2] Hartanto, Antonius Aditya, 2003, *Java 2 Micro Edition Tingkat Lanjut*, Jakarta : Elex Media Komputindo.
- [3] Kline, Paul J., C Bala Kumar, dan Timothy J. Thompson, 2004, *Bluetooth Application Programming with The Java APIs*, United States of America : Morgan Kaufmann Publishers.
- [4] Knudsen, Jonathank dan, Sing Li 2005, *Beginning J2ME: From Novice to Professional, Third Edition*, United States : Apress.
- [5] Wicaksono, Andi, 2002, *Pemrograman Aplikasi Wireless dengan Java*, Jakarta : Elex Media Komputindo