

MAKALAH SEMINAR TUGAS AKHIR
LAYANAN WEB BERGERAK PEMUTAR VIDEO ONLINE
PADA SYMBIAN OS MENGGUNAKAN JAVA
M. Dani Budiman*, Aghus Sofwan**, Maman Somantri**

Abstrak – Teknologi bergerak (*mobile technology*) semakin berkembang ditandai dengan meningkatnya pasar perangkat bergerak terutama ponsel pintar (*smartphone*) dan PDA (*Personal Digital Assistant*) yang tidak hanya digunakan untuk komunikasi suara saja namun juga untuk komunikasi data. Penelitian tugas akhir ini bertujuan untuk membuat sebuah aplikasi pemutar video online.

Aplikasi yang dikembangkan pada penelitian ini merupakan sebuah aplikasi Layanan Web model klien-server yang berjalan pada Symbian OS. Penelitian ini meliputi perancangan basisdata, pengembangan aplikasi manajemen basisdata pada sisi server, pembangunan sebuah aplikasi MIDlet untuk klien yang berjalan pada perangkat bergerak, serta pengujian dari aplikasi tersebut secara keseluruhan.

Pengujian aplikasi pemutar video dilakukan dengan metode kotak hitam dan metode kotak putih dengan menggunakan perangkat yang sesungguhnya (pada penelitian digunakan telepon selular Nokia 6600). Hasil pengujian aplikasi tersebut menunjukkan hasil yang baik dan stabil. Hal ini menunjukkan bahwa Java sesuai untuk pengembangan aplikasi pada perangkat bergerak, khususnya pada model Layanan Web bergerak.

Kata-kunci: Symbian OS, J2ME, Mobile Media API, Record Management System, Layanan Web bergerak.

I PENDAHULUAN

1.1 Latar Belakang

Informasi saat ini telah menjadi komoditas penting pada saat ini. Penyampaian informasi pun dapat dilakukan melalui banyak media. Melalui media cetak konvensional seperti koran dan majalah, maupun media elektronik seperti radio yang berbentuk suara dan televisi yang mampu menampilkan gambar bergerak dan suara.

Untuk mempercepat dan mempermudah akses terhadap informasi, diperlukan suatu aplikasi yang mampu menyediakan fasilitas akses yang dapat digunakan dimanapun dan kapanpun. Karena itu penulis membuat aplikasi pemutar 3gp (*3gp player*) yang mampu menampilkan video di layer ponsel yang berbasis J2ME.

Teknologi JAVA dipilih, selain karena teknologi J2ME (Java 2 Micro Edition) saat ini telah banyak dipakai dalam berbagai jenis dan merek ponsel juga karena dengan teknologi JAVA dimungkinkan untuk membangun satu sistem penuh, mulai dari aplikasi *mobile* di sisi pengguna sampai aplikasi di sisi *server*,

menggunakan JSP (Java Server Pages), yang berfungsi sebagai administrasi sistem.

1.2 Tujuan

Tujuan penyusunan tugas akhir ini adalah untuk Merancang dan membangun suatu aplikasi *video player* menggunakan bahasa pemrograman java yang akan diimplementasikan pada sistem operasi Symbian Series 60.

1.3 Batasan Masalah

Pembatasan masalah dalam tugas akhir ini adalah sebagai berikut.

1. Sistem Operasi yang dituju untuk pengembangan maupun pemasangan aplikasi adalah Symbian OS versi 7.0 Series 60.
2. MIDP yang didukung adalah MIDP 2.0.
3. Web Container yang digunakan adalah Tomcat.
4. Bahasa scripting yang digunakan adalah JSP.
5. Basisdata yang digunakan adalah MySQL.
6. Video yang dipergunakan adalah format 3gpp atau file yang berekstensi .3gp.
7. Tidak menjelaskan mekanisme pendistribusian voucher.

II DASAR TEORI

2.1 Symbian OS

Symbian OS pada intinya terdapat lapisan dasar (*microkernel* dan *device driver*), lapisan *middleware* (sistem server, keamanan, *framework* aplikasi) dan lapisan komunikasi (telepon, pesan, dan *personal area networking*). Tiga bagian lapisan utama yaitu sebagai berikut.

1. Lapisan Dasar

Lapisan Dasar (*Base Layer*) merupakan lapisan paling dasar dalam struktur sistem operasi Symbian OS. Bagian ini terdiri atas *Mikrokernel*, *Device Driver* dan *Pustaka Pengguna (user library)*. *Mikrokernel* berhubungan langsung dengan processor dan bertanggung jawab pada manajemen daya dan memori. *Device Driver* adalah antarmuka antara perangkat keras dan perangkat lunak yang dibutuhkan untuk mengakses berbagai keperluan misalnya yang berhubungan dengan gambar dan suara serta menangani jalur komunikasi. *Pustaka Pengguna* menyediakan bermacam fungsionalitas seperti penanganan kesalahan (*error handling*), manajemen memori yang digunakan oleh program, serta menangani bermacam proses dan interupsi (*interrupt*).

2. Lapisan Perangkat Tengah

Middleware terdiri dari sistem server, sistem keamanan dan *framework* aplikasi. Sistem server menyediakan fungsi untuk manajemen *window*. Sistem keamanan

* Mahasiswa Teknik Elektro UNDIP

** Dosen Teknik Elektro UNDIP

menyediakan pustaka keamanan dasar seperti enkripsi data. Framework aplikasi adalah sekumpulan pustaka yang digunakan untuk menangani tampilan antar muka user, proses *starting* suatu aplikasi dan grafis.

Application engines Phone book, calendar, photo album, notepad, to-do, pinboard	Installed applications Games, self-developed applications, utility software	Messaging SMS, MMS, e-mail, fax	MIDP Java KVM
Application framework GUI framework (Avkon, standard Eikon, Uikon), application launching, and graphics		Personal area networking Bluetooth, infrared	
Multimedia Images, sounds, graphics		Communications infrastructure TCP/IP stack, HTTP, WAP stack	
Security Cryptography, software	Telephony HSCSD, GPRS	Base User library, kernel, device drivers	

Gambar 2.1 Struktur Symbian OS secara umum.

3. Lapisan Komunikasi

Lapisan Komunikasi (*Communication Layer*) terdiri dari Platform Telepon (*telephony*) dan sistem pesan (*messaging system*). Lapisan ini menyediakan arsitektur dan protokol untuk melakukan komunikasi baik berupa suara maupun data, misalkan untuk penanganan panggilan telepon, pengiriman pesan teks, email dan konektivitas ke Internet. Struktur sistem operasi Symbian OS secara umum bisa dilihat pada Gambar 2.1

2.2 Series 60 Platform

Series 60 platform merupakan desain referensi lengkap dari sebuah ponsel pintar, yang dikembangkan oleh Nokia dan dilisensi oleh beberapa perusahaan perangkat bergerak untuk keperluan desain perusahaan itu sendiri. Terdapat beberapa aplikasi *default* seperti *personal information management*, multimedia, protokol komunikasi, penerimaan dan pengiriman pesan, kemampuan untuk *browsing* Internet. Sistem operasi Symbian OS Series 60 membutuhkan spesifikasi perangkat keras minimal sebagai berikut.

1. Layar (*Display*)

Layar yang dipakai pada Symbian OS Series 6.0 adalah berorientasi vertikal dengan resolusi layar sebesar 176x208 piksel dengan kemampuan menampilkan warna minimal sebanyak 4096 warna (12bit).

2. Metode Input (*Input Method*)

Symbian OS Series 60 didesain untuk dapat digunakan hanya dengan menggunakan satu tangan saja. Metode input yang digunakan adalah menggunakan papan ketik (*keypad*) 12 tombol sesuai dengan standar dari ITU-T (*International Telecommunication Union – Telecommunication Standardization Sector*).

3. Prosesor

Prosesor yang digunakan pada sistem operasi Symbian Series 60 adalah ARM Processor 32 bit.

4. Memori

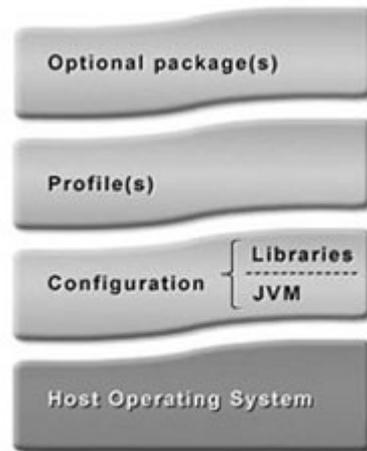
Kapasitas memori minimal yang dibutuhkan Symbian Series 60 adalah 16 MB untuk ROM (*Read Only*

Memory) serta 8MB untuk RAM (*Random Access Memory*).

2.3 Java 2 Micro Edition

J2ME merupakan kumpulan pustaka atau library yang didesain untuk perangkat yang memiliki keterbatasan memory dan prosesor dengan kecepatan rendah, seperti perangkat ponsel pintar, personal data assistants (PDA). Pada akhirnya para pengembang java memiliki pilihan untuk menghasilkan aplikasi yang dapat berjalan pada semua perangkat bergerak yang mendukung J2ME.

Untuk mendukung arsitektur yang mudah disesuaikan, J2ME menerapkan konsep Konfigurasi (*Configurations*), Profil (*Profiles*), dan Paket Tambahan (*Optional Package*). Gambar 2.2 menunjukkan susunan hirarki antarkonsep arsitektur pada J2ME dengan sistem operasi di mana J2ME diimplementasikan.



Gambar 2.2 Susunan Arsitektur J2ME.

Konsep *Configuration* dan *Profile* pada J2ME dapat dijelaskan sebagai berikut.

1. Konfigurasi J2ME

Configuration ini menerangkan spesifikasi minimum bagi sebuah perangkat dan fitur apa saja yang dapat didukung oleh J2ME pada perangkat tersebut. Spesifikasi perangkat yang diatur oleh *Configuration* antara lain ditentukan oleh kecepatan prosesor, jumlah memori dan jenis konektivitas yang tersedia.

2. Profil J2ME

Sebuah *Profile* melengkapi *Configuration* dengan menambahkan kelas-kelas (*classes*) yang menyediakan fitur khusus untuk perangkat tertentu atau keperluan tertentu. Kedua *Configuration* pada J2ME (CDC dan CLDC) mempunyai satu atau lebih *Profile*. Beberapa *Profile* J2ME yang utama antara lain MIDP (*Mobile Information Device Profile*), PDA *Profile* (PDAP), *Foundation Profile*, *Personal Basis* dan *Personal Profile*, RMI *Profile* dan *Game Profile*.

3. J2ME Optional Package

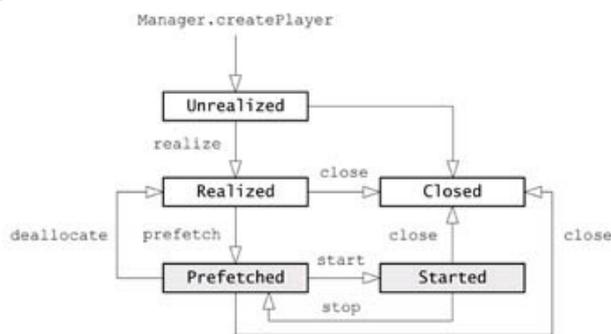
Pada dasarnya J2ME *Optional Package* merupakan pustaka yang digunakan untuk perluasan dari profile

yang sudah ada. *Optional Package* menyediakan pustaka yang bertujuan untuk digunakan ulang oleh desain tim profile dan vendor dari lingkungan J2ME. *Optional Package* dalam J2ME didefinisikan dalam *Java Specification Request (JSR)*, diantara JSR yang terkenal antara lain JSR 120 *Wireless Messaging API*, JSR 135 *Mobile Media API*, JSR 172 *Web Service Specification*, JSR 177 *Security and Trust Service*, JSR 179 *Location API*.

a) Mobile Media Api

Mobile Media API, JSR 135 memperluas fungsionalitas dari platform J2ME dengan menyediakan kemampuan suara (*audio*), gambar bergerak (*video*). Dijadikannya Mobile Media Api sebagai paket tambahan karena secara realitas terdapat perbedaan pada kebutuhan dan kemampuan masing-masing perangkat. Jadi adalah hal yang mustahil bagi semua perangkat untuk mendukung keseluruhan spesifikasi MMAPi.

Player memiliki lima status yaitu *Unrealized*, *Realized*, *Prefetched*, *Started*, and *Closed*. Pada awalnya objek *player* akan berada pada status *unrealized*. Pada saat peralihan dari keadaan *unrealized* ke *realized* objek *player* akan melakukan komunikasi untuk mendapatkan konten sumber, komunikasi dilakukan terhadap sistem server atau file system. Kondisi *realized* bisa dicapai dengan memanggil method *realize*. Pemanggilan method *prefetch* menyebabkan objek *player* berusaha mendapatkan sumber daya keluaran (*output*), seperti perangkat suara (*speaker*) pada perangkat bergerak. Proses tersebut kemungkinan harus menunggu bilamana terdapat aplikasi lain yang sedang menggunakan sumber daya tersebut. Bila objek *player* berhasil mendapatkan sumber daya keluaran, maka *player* telah mencapai status *prefetched*. Pada keadaan ini *player* sudah dapat memulai memainkan data. Pemanggilan method *start* berarti *player* sudah melakukan proses memainkan data dan pada tahap ini *player* memasuki tahap *started*.



Gambar 2.3 Siklus hidup dari objek player.

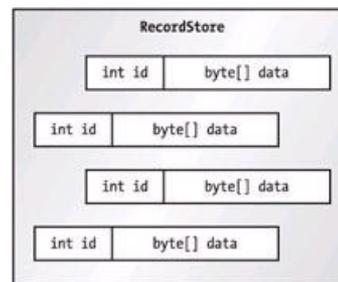
4. MIDP (Mobile Information Device Profile)

MIDP merupakan salah satu platform dari J2ME yang berdasarkan pada CLDC dan KVM yang dijalankan pada perangkat dengan sumber daya terbatas seperti telepon selular dan PDA. Aplikasi dari MIDP disebut MIDlet.

a) Penyimpanan Permanen Pada MIDlet.

MIDP memberikan suatu mekanisme penyimpanan data yang bersifat permanen. Mekanisme penyimpanan ini disebut sebagai RMS (*Record Management System*). Sebuah record store berisi kumpulan *record* yang datanya tidak akan berubah walaupun diakses oleh beberapa MIDlet. J2ME platform bertanggung jawab untuk menjaga semaksimal mungkin integritas/keutuhan data terhadap kegiatan normal dari platform, *reboot* dan penggantian baterai (catu daya).

Isi data dari setiap record yang tersimpan dalam *record store* yaitu dalam bentuk larik (*array*) dari byte. Masing-masing dari record dapat memiliki panjang record yang berbeda-beda. Sebagai identitas yang bersifat unik maka masing-masing record memiliki penanda berupa *recordId*. *RecordId* memiliki tipe data integer. Selain sebagai penanda, *recordId* juga berfungsi sebagai argumen pada method *getRecord(int recordId)* untuk mengeluarkan data yang tersimpan. Karena *recordId* bertipe data integer maka nilai maksimum yang dapat dihasilkan adalah 2,147,483,647. Karakteristik yang lain adalah setiap *record store* hanya akan menciptakan sebuah nilai *recordId* satu kali, karena itu tidak akan ada nilai *recordId* yang sama dalam satu *record store*. Struktur dari record store ditunjukkan pada Gambar 2.4.



Gambar 2.4 Struktur penyimpanan Record Store.

2.4 Layanan Web Bergerak

Layanan Web adalah suatu antar muka yang menyediakan berbagai aktivitas aplikasi yang dapat diakses melalui jaringan. Layanan Web merupakan suatu antar muka yang menyediakan berbagai kumpulan operasi yang dapat diakses pada jaringan.

Perbedaan utama antara Layanan Web dengan web adalah pada sisi arsitekturnya. Sebuah web bisa mempunyai berbagai macam fungsi seperti penyedia informasi ataupun hiburan sedangkan *Web Service* lebih berorientasi pada layanan. Sebuah web dirancang untuk interaksi antara program dengan pengguna, sedangkan *Web Service* dirancang untuk interaksi antara program dengan program.

Dengan adanya *Web Service*, sebuah aplikasi dapat dengan mudah diintegrasikan dengan aplikasi lainnya dan memiliki interoperabilitas yang tinggi. *Web Service* menggunakan berbagai standar yang telah ada seperti HTTP (*Hyper Text Transfer Protocol*), XML (*eXtensible Markup Language*), dan SOAP (*Simple Object Access Protocol*).

Java sebagai salah satu bahasa pemrograman yang sejak awal didesain untuk bekerja dalam jaringan telah mendukung konsep Layanan Web ini. Dukungan Java terhadap *Web Service* juga dibawa ke dalam J2ME antara lain dengan dimunculkannya *Java Specification Request (JSR) 172, Web Service Specification*.

III PERANCANGAN PERANGKAT LUNAK

Aplikasi yang dibangun pada tugas akhir ini adalah sebuah aplikasi Layanan Web bergerak pada Symbian OS yang bernama MobileVideoShop. Fungsi utamanya adalah memberikan kemudahan kepada pengguna untuk memutar sekaligus membeli video yang ditawarkan oleh penyedia/*server*. Kemudahan penggunaan dapat terlihat dari daftar video(*playlist*) yang disediakan, sehingga pengguna tinggal memilih video yang diinginkan.

3.1 Perancangan Sisi Klien

Perancangan pada sisi klien ini menggunakan analisis berorientasi objek.

1. Use Case.

Uraian berikut ini adalah use case dari sistem sisi server MobileVideoShop dari sisi klien. Kata yang bercetak tebal menggambarkan calon objek sedangkan kata yang bercetak miring menggambarkan operasi.

Use Case:

Aplikasi klien MobileVideoShop *dijalankan* oleh **pengguna**. Pertama kali pengguna *mendapatkan* tampilan **splashscreen**. sistem kemudian *menampilkan* **halaman login**. Pada **halaman login** terdapat dua **menu**, yang pertama **pengguna** dapat *login* ke **aplikasi sisi server** dan yang kedua **pengguna** dapat *membrowse* isi dari **tempat penyimpanan video**. Apabila terjadi *kesalahan pengisian* maka sistem akan *menampilkan* **pesan kesalahan**. Setelah *login*, pengguna akan mendapati **halaman utama** yang berisi **form playlist** yang masih kosong dan beberapa **menu** yang *muncul* setelah **pengguna** *menekan* **softkey Option**. Pengguna *memilih* **menu Unduh Playlist**. Sistem kemudian *menghubungi* **aplikasi sisi server**. Selama *proses query*, sistem *menampilkan* **animasi download(gauge)**. Setelah *proses query* selesai, **aplikasi server** *mengirim balik* **hasil query** ke sistem. **playlist** kini sudah terisi **daftar video** yang dapat diunduh beserta **informasi harga video**. Pada pilihan Option *bertambah* satu menu yaitu **menu putar**. **Pengguna** kemudian *memilih* **daftar video**. Selanjutnya pengguna *memilih* **menu putar** Sistem *mengecek* apakah pengguna mempunyai cukup **saldo** untuk *mengunduh* video. Sistem *menampilkan* **animasi gauge**. **aplikasi sisi server** *mengirimkan* **string url video**, string url video *diproses* oleh **fileUrl**. Setelah *selesai diunduh*, sistem kemudian *memproses*, lalu video *diputar* oleh **video player** *ditampilkan* pada **video canvas**. Apabila pengguna *menyimpan* data video, pengguna *memilih* **menu simpan**. Pada **form video** pengguna dapat *memasukan* **judul video**, kemudian sistem *menyimpan* video pada **tempat penyimpanan**.

Pengguna dapat *mengecek* sisa **saldo** dari **menu Cek Saldo**. Untuk *menambah* jumlah **deposit**, **pengguna** *memilih* **menu Isi Deposit**, kemudian memasukan kode voucher. Dengan *memilih* **menu Browse Video**, pengguna dapat *memutar* video, *rename* nama video, *menghapus* video.

2. Identifikasi Objek.

Setelah proses dilanjutkan dengan indentifikasi calon objek kemudian diikuti oleh hasil seleksi calon objek. Berikut ini merupakan daftar spesifikasi atribut dari masing-masing objek atau kelas.

Splashscreen = ukuran + gambar + durasi
Halaman login = judul halaman + tipe halaman + item login
Penyimpanan video = tipe data + nama data + isi data + string besar penyimpanan
Form utama = judul halaman + item playlist
playlist = string judul video + string harga
Animasi gauge = judul gauge + tipe gauge + label gauge
Fileurl = string url video
Video player = ukuran + data video + string video
Video canvas = ukuran + tipe warna
Form video = judul halaman + item video
Deposit = string deposit + item voucher

3. Pemodelan Kelas-Tanggungjawab-Kolaborator

Setelah pengerjaan use case dan indentifikasi objek, tahapan selanjutnya melakukan pemodelan kelas-tanggungjawab-kolaborator(CRC). Metode ini disusun berdasarkan indentifikasi objek atau kelas. Objek merupakan instansiasi dari kelas, untuk mempermudah pada saat pembuatan program, tiap kelas diberi nama yang lebih sederhana.

4. Model Tingkah Laku Objek

Pemodelan objek yang dilakukan antara lain melalui CRC dan model hubungan antar objek berfungsi untuk mempresentasikan elemen statis dari model analisis berorientasi objek. Langkah selanjutnya adalah memodelkan elemen dinamis dari model analisis, hal ini dilakukan dengan memodelkan tingkah laku objek, antara lain melalui penelusuran kejadian (event trace) dan membangun Diagram Transisi Keadaan (State Transition Diagram).

3.2 Perancangan Sisi Server

Perancangan pada sisi server ini menggunakan analisis berorientasi objek.

1. Use Case.

Uraian berikut ini adalah use case dari sistem sisi server MobileVideoShop dari sisi server. Kata yang bercetak tebal menggambarkan calon objek sedangkan kata yang bercetak miring menggambarkan operasi.

Administrasi Pengguna

Administrator memiliki wewenang untuk *mengedit* **data pribadi pengguna** seperti **nama**, **nama login** dan **password**. **Administrator** juga dapat *menghapus* **data user** dari sistem, berikut **deposit** yang dimiliki. **Administrator** berwenang untuk *mengupload* **video** ke **server**, *mengedit* atribut **video** tersebut bahkan

menghapusnya. Seorang administrator dapat *menciptakan kode voucher* yang akan dipergunakan oleh *pengguna*, dan dapat juga menghapusnya.

Kegiatan Pengguna

Seorang pengguna terlebih dahulu *mendaftar* dengan *memasukan data pribadi* pada sistem. Setelah itu *pengguna* dapat *login* ke sistem, jika terjadi *kesalahan pengisian* maka sistem akan *menampilkan pesan kesalahan*. *Pengguna* dapat *mengedit data pribadi*, dengan memilih menu *ubah akun*, seorang *pengguna* juga dapat *melihat* nilai *deposit* yang dimilikinya dengan *memilih* menu *cek deposit*. Untuk menambah nilai *deposit* *pengguna* harus memasukan *kode voucher*, apabila *kode voucher* yang dimasukan salah maka sistem akan menampilkan *pesan kesalahan*. Untuk mengetahui *daftar video* *pengguna* dapat memilihnya pada menu *daftar video*.

2. Identifikasi Objek.

Setelah proses dilanjutkan dengan indentifikasi calon objek kemudian diikuti oleh hasil seleksi calon objek. Berikut ini merupakan daftar spesifikasi atribut dari masing-masing objek atau kelas.

- Administrator = id + nama + nama login + password
- Pengguna = id + nama + nama login + password
- Deposit = id + nilai deposit
- Video = id + judul, nama file + harga
- Kode voucher = id + kode voucher

Sama seperti perancangan pada sisi klien, langkah selanjutnya adalah melakukan tahapan Pemodelan Kelas-Tanggungjawab-Kolaborator dan Model Tingkah Laku Objek

IV ANALISA DAN PEMBAHASAN

4.1 Aplikasi Sisi Klien

Aplikasi pada sisi klien adalah sebuah MIDlet Java yang berjalan pada Symbian OS. Aplikasi ini diuji pada perangkat bergerak sesungguhnya (pada penelitian ini digunakan telepon selular Nokia 6600).

1. Instalasi Aplikasi

Aplikasi yang dihasilkan dari J2ME Wireless Toolkit adalah berupa dua buah berkas berekstensi .jad dan .jar. Berkas berekstensi .jar adalah aplikasi utamanya sedangkan berkas berekstensi .jad adalah berkas penyerta yang mengandung informasi tentang aplikasi. Kedua file ini ditransfer pada perangkat tujuan kemudian dijalankan. AMS (*Application Management Software*) dari perangkat tujuan akan mengenalinya dan proses instalasi akan dimulai. Setelah proses instalasi selesai, aplikasi dapat dijalankan dengan mengakses *icon* pada menu utama perangkat. Tampilan utama aplikasi sisi klien diperlihatkan pada Gambar 4.1.



Gambar 4.3 Menu utama aplikasi klien.

2. Pengujian Aplikasi

Aplikasi dinyatakan berhasil melewati pengujian apabila dalam percobaan-percobaan berikut tidak aplikasi tidak menjadi *error* atau melaksanakan perintah dengan salah. Pengujian aplikasi baik pada sisi klien maupun server menggunakan metode kotak putih (*white box*) dan metode kotak hitam (*black box*).

Dalam pengujian kotak putih aplikasi dihadapkan pada kondisi yang memungkinkan aplikasi untuk mengalami error saat dijalankan (*run-time error*), antara lain dengan melakukan proses isi ulang deposit tanpa memasukan kode voucher. Hasil pengujian aplikasi klien menggunakan metode kotak putih ditunjukkan pada Tabel 4.1.

TABEL 4.1 HASIL PENGUJIAN PROSES ISI DEPOSIT DENGAN METODE KOTAK PUTIH.

Kondisi 1	Kondisi 2	Output
TRUE	diabaikan	Pesan: Maaf kode voucher harus diisi
diabaikan	TRUE	Pesan: Kode voucher yang Anda masukan salah!

Keterangan :

Kondisi 1 : *Field* kode voucher tidak diisi dengan lengkap.

Kondisi 2 : Proses isi deposit menggunakan voucher yang tidak valid.

Pengujian yang kedua adalah menggunakan metode kotak hitam. Pada metode pengujian perbandingan ini, aplikasi diberikan berbagai kondisi masukan, kemudian keluaran yang dihasilkan sistem dibandingkan dengan keluaran yang diharapkan. Hasil pengujian aplikasi klien menggunakan metode kotak hitam ditunjukkan pada Tabel 4.2.

Hasil pengujian menggunakan dua metode diatas, aplikasi tidak menjadi *error* atau menampilkan keluaran yang salah. Semua kondisi penggunaan telah diujikan dan berhasil, dengan demikian aplikasi telah berhasil melewati pengujian.

TABEL 4.2 HASIL PENGUJIAN PROSES ISI DEPOSIT DENGAN METODE KOTAK HITAM.

Input Kd voucher	Output	
	Output diharapkan	Output sistem
-	Pesan: kode voucher harus diisi	Pesan: Maaf kode voucher harus diisi
√	Proses isi deposit dilaksanakan	Proses isi deposit Dilaksanakan

Keterangan :

- √ : Data diisi.
- : Data kosong.

4.2 Tinjauan Kritis Sistem

Pembuatan program aplikasi merupakan proses yang paling sulit dalam rangkaian pembuatan Tugas Akhir. Banyak permasalahan yang muncul pada saat pengerjaan aplikasi. Pada bagian ini akan dibahas perihal permasalahan yang muncul pada saat pengerjaan aplikasi MobileVideoShop serta solusi terhadap masalah tersebut. Permasalahan yang muncul meliputi tipe pemutaran data video secara online, ukuran berkas *record management system*, dan pembatasan karakter pada *text field*.

1. Tipe Pemutaran Data Video

Ada dua macam tipe pemutaran data video yang dapat dilakukan. Tipe yang pertama adalah mekanisme *streaming* dan yang kedua dengan mengunduh data secara utuh/komplit. Transfer data secara *streaming* merupakan sebuah mekanisme pengiriman data dengan dipotong dengan ukuran tertentu kemudian potongan data tersebut langsung diputar oleh player tanpa harus menunggu data secara keseluruhan. Tipe kedua merupakan tipe mengunduh data secara utuh dari server, baru kemudian data video diputar oleh player.

Setelah dilakukan penelitian ternyata tipe *streaming* tidak dapat diterapkan pada aplikasi ini karena potongan data yang tidak lagi dikenali oleh player sebagai bagian dari data video format 3gpp.

2. Ukuran Record Management System

Record Management System merupakan suatu mekanisme penyimpanan data permanen pada ponsel. Pada sistem ini proses yang dapat dilakukan sama dengan proses yang terjadi pada penanganan berkas (*file*). Oleh karena itu sistem ini memiliki empat operasi utama, yaitu memasukan, menghapus, mengubah dan mengeluarkan data. Semua operasi diatas pada prinsipnya dikelola oleh berkas yang bernama rms.db. Berkas rms.db memiliki ukuran sebesar data recordstore yang tersimpan didalamnya.

Pada proses untuk memasukan dan mengeluarkan data tidak ditemui suatu permasalahan. Lain halnya dengan proses mengubah dan menghapus data. Pada proses ini terjadi permasalahan yang melibatkan ukuran rms.db. Misalkan pada berkas rms.db terdapat recordstore berukuran 300KB, yang dipecah menjadi 3 buah record masing-masing berukuran 100KB. Operasi penghapusan (*delete*) dengan memanggil method *deleteRecord* mencoba menghapus record pertama sebesar 100KB. Setelah dilakukan operasi

penghapusan, seharusnya ukuran berkas rms.db sekarang menjadi 200KB, tetapi pada kenyataannya besar berkas rms.db tetap pada ukuran semula 300KB.

Permasalahan ini dapat diatasi dengan membuat recordstore baru, menghapus record pada recordstore lama, memindahkan data recordstore lama ke recordstore baru, kemudian menghapus recordstore lama.

3. Pembatasan Panjang Karakter Pada Form Text Field

Form *text field* salah satunya digunakan untuk menginput string kode voucher. Panjang dari string kode voucher adalah 10 karakter. Permasalahan muncul pada saat pengetikan karakter terakhir dari kode voucher, karakter yang dapat ditampilkan hanya karakter pertama dari pilihan karakter lainnya, selanjutnya diikuti oleh peringatan "Maximum number of characters reached". Hal ini tentu saja akan menyulitkan karena tidak selalu karakter terakhir pada kode voucher adalah karakter pertama pada tombol *alphanumeric*.

Solusi yang diberikan adalah dengan jalan menambah panjang pembatas karakter dari 10 menjadi 11 karakter.

4.3 Aplikasi Sisi Server

Aplikasi MobileVideoShop pada sisi server terbagi menjadi dua bagian, yaitu *frontend* dan *backend*. Aplikasi *frontend* menangani masukan dari pengguna yang menggunakan aplikasi klien sedangkan aplikasi *backend* merupakan aplikasi untuk administrasi basisdata yang digunakan. *Backend server* menangani tugas-tugas administrasi basisdata seperti menampilkan data, menambah data, menghapus data, serta mengedit data. Tampilan aplikasi pada sisi server ditunjukkan pada Gambar 4.2.



Gambar 4.4 Aplikasi sisi server

1. Pengujian Aplikasi

Aplikasi dinyatakan berhasil melewati pengujian apabila dalam percobaan-percobaan berikut tidak aplikasi tidak menjadi *error* atau melaksanakan perintah dengan salah. Pengujian aplikasi baik pada sisi klien maupun server menggunakan metode kotak putih (*white box*) dan metode kotak hitam (*black box*).

Dalam pengujian kotak putih aplikasi dihadapkan pada kondisi yang memungkinkan aplikasi untuk mengalami error saat dijalankan (*run-time error*).

Prosedur pengujian aplikasi dilakukan seperti pada pengujian aplikasi sisi klien. Aplikasi dihadapkan pada berbagai kondisi yang memungkinkan untuk menimbulkan kesalahan, seperti misalnya pengisian *form* yang salah maupun tidak lengkap. Aplikasi harus mampu mengenali kondisi-kondisi tersebut dan menyiapkan fasilitas untuk mengatasinya, sehingga tidak menjadi *error* atau menjalankan fungsi yang salah.

Dari semua hasil pengujian dengan metode kotak putih maupun metode kotak hitam, aplikasi tidak menjadi *error* atau menampilkan keluaran yang salah. Semua kondisi penggunaan telah diujikan dan berhasil, dengan demikian aplikasi telah berhasil melewati pengujian.

V PENUTUP

5.1 Kesimpulan

1. Berdasarkan hasil pengujian dengan metode kotak putih (*white box*) maupun dengan metode kotak hitam (*black box*), aplikasi bergerak yang dibangun pada sisi klien telah cukup baik dan stabil, hal ini menunjukkan bahwa J2ME sebagai bahasa pemrograman cukup baik untuk pengembangan aplikasi pada perangkat bergerak.
2. Berdasarkan hasil pengujian dengan metode kotak putih (*white box*) maupun dengan metode kotak hitam (*black box*), aplikasi yang dibangun pada sisi server cukup baik dan stabil, hal ini menunjukkan bahwa JSP sebagai bahasa pemrograman berorientasi pada server mampu bekerja sama dengan bahasa pemrograman yang berbeda, seperti J2ME.
3. Tipe pemutaran video secara *streaming* pada prosesnya akan menghasilkan *error*, hal ini disebabkan karena potongan data yang diterima tidak dapat dikenali oleh aplikasi pemutar.
4. Ukuran berkas record management system yang tidak dapat menyusut dapat diatasi dengan pembuatan *recordstore* baru, kemudian data pada *recordstore* lama dipindahkan ke *recordstore* yang baru.
5. Pada penggunaan form *text field* terdapat masalah pada pengetikan karakter yang terakhir, hal ini dapat diatasi dengan cara panjang maksimum karakter ditambah dengan 1.

5.2 Saran

1. Aplikasi pemutar video perlu dikembangkan lebih lanjut untuk format data video yang lainnya seperti *.avi, *.mpg, dll.
2. Pada saat terjadi komunikasi antara sisi klien dan server sangat mungkin terjadi penyadapan informasi. Untuk mencegah hal tersebut faktor keamanan perlu ditingkatkan lebih lanjut, misalnya dengan menggunakan teknik enkripsi data.

DAFTAR PUSTAKA

- [1]. Digia Information Atchitect, *Programming for the Series 60 Platform and Symbian OS*, John Willey and Sons Publishing, Chicester UK
- [2]. Jode, M. et al. *Programming Java™ 2 Micro Edition on Symbian OS*, John Willey and Sons Publishing, Chicester UK, 2003.
- [3]. Edward L., and R. Baker, *Developing Series 60 Applications: A Guide for Symbian OS C++ Developers*, Addison Wesley, 2004.
- [4]. Keogh, J., *J2ME: The Complete Reference*, Mc Graw Hill Publishing, California, 2003.
- [5]. Rikyanto, Isak, *Belajar Sendiri Java Server Pages*, Elex Media Komputindo, Jakarta, 2002
- [6]. Turner, James, *MySQL and JSP Web Applications*, Sams Publishing, Indianapolis, 2002
- [7]. Wicaksono, Ady, *Pemrograman Aplikasi Wireless dengan Java*, Elex Media Komputindo, 2002.
- [8]. Wicaksono, Ady, *Pemrograman Internet dan XML Pada Ponsel*, Elex Media Komputindo, 2003.
- [9]. -----, *Exploring the J2ME Mobile Media APIs*. Mei 2004.
<http://www.devx.com/wireless/Article/20911/0/page/index.html>
- [10].-----, J2ME official site. April 2005
<http://java.sun.com/product/j2me>.
- [11].-----, J2ME documentation site. April 2005
<http://java.sun.com/j2me/reference/docs/index.html>



**[12].Moch. Dani
Budiman**

(L2F000619) lahir di Jakarta, 6 Agustus 1982. Menempuh pendidikan dasar di SDN Cipayung 04 Jakarta lulus tahun 1994, kemudian melanjutkan ke SLTPN 9 Jakarta lulus tahun 1997, dilanjutkan lagi di SMUN 39 Jakarta lulus 2000, dan sampai saat ini masih menyelesaikan studi S1 di Jurusan Teknik Elektro Fakultas Teknik Universitas Diponegoro Semarang Konsentrasi Informatika.

Menyetujui dan Mengesahkan,

Pembimbing I,

Aghus Sofwan, S.T., M.T.
NIP. 132 163 757
Tanggal

Pembimbing II,

Maman Somantri, S.T., M.T.
NIP. 132 231 133
Tanggal