

# Aplikasi J2ME Untuk Video Player Online

**Robertus Lilik Haryanto**

*lilik.haryanto@gmail.com*

*http://lharyanto.ifastnet.com*

## **Lisensi Dokumen:**

*Copyright © 2003-2006 IlmuKomputer.Com*

*Seluruh dokumen di IlmuKomputer.Com dapat digunakan, dimodifikasi dan disebarkan secara bebas untuk tujuan bukan komersial (nonprofit), dengan syarat tidak menghapus atau merubah atribut penulis dan pernyataan copyright yang disertakan dalam setiap dokumen. Tidak diperbolehkan melakukan penulisan ulang, kecuali mendapatkan ijin terlebih dahulu dari IlmuKomputer.Com.*

Aplikasi pemutar video pada perangkat *mobile* memang sudah tidak asing lagi, antara lain aplikasi pemutar video dengan format file 3GPP, MPEG, dll. Aplikasi video player online ini tidak jauh berbeda dengan aplikasi pemutar video lainnya, hanya saja aplikasi ini akan membaca file video pada sebuah server menggunakan koneksi GPRS. File tersebut akan dijalankan oleh video player ini seperti halnya aplikasi video player lainnya.



Dengan menggunakan aplikasi ini, pengguna tidak perlu *men-download* file video dari server secara manual terlebih dahulu, karena secara otomatis aplikasi ini akan mengambil file tersebut dari server dan akan menyimpannya pada memori untuk dijalankan.

## **Pendahuluan**

Aplikasi *video player online* ini digunakan untuk menampilkan video berekstensi **.mpg**, dimana file video yang akan dijalankan berada pada salah satu server. Pada artikel ini, aplikasi yang akan dikembangkan hanya menggunakan server lokal sebagai media penyimpanan file video tersebut. File tersebut akan dijalankan oleh aplikasi ini secara melalui koneksi GPRS.

## **Isi**

Untuk mengembangkan aplikasi ini, penulis membuat dengan 2 kelas, yaitu kelas **DoraMPEGPlayer** yang merupakan turunan dari kelas MIDlet dan merupakan implementasi dari kelas CommandListener, digunakan sebagai antarmuka antara pengguna dengan aplikasi, dan kelas **DoraMPEGCanvas** yang merupakan turunan dari kelas Canvas dan merupakan implementasi dari kelas PlayerListener, digunakan untuk menjalankan file berekstensi .mpg yang berada di server.

Anda dapat mengembangkan aplikasi menggunakan editor apa saja, dimana dapat memudahkan Anda dalam proses pengembangan. Dalam hal ini, penulis menggunakan NetBeans 5.5, karena IDE ini memiliki plug-in untuk *mobile application development*, sehingga sangat membantu

dalam proses pengembangan aplikasi.

Berikut listing program untuk kedua kelas tersebut:

**1. Kelas DoraMPEGPlayer (DoraMPEGPlayer.java)**

```
/*
 * Nama File      : DoraMPEGPlayer.java
 * Developed by   : Lilik Haryanto
 */

import javax.microedition.midlet.*;
import javax.microedition.lcdui.*;

public class DoraMPEGPlayer extends MIDlet implements CommandListener
{

    private Command cmdExit;
    private Command cmdPlay;

    private Display display;
    private TextField textField;
    private Gauge gauge;
    public Form form;

    private static final int GAUGE_LEVELS = 4;
    private static final int GAUGE_MAX = 12;
    private static final String DEFAULT_URL =
        "http://localhost:8087/lilik/video/test.mpg";

    public DoraMPEGPlayer() {
        display = Display.getDisplay(this);
        form = new Form("DoraMPEGPlayer");
        textField = new TextField("MPEG Video URL", DEFAULT_URL, 100,
            TextField.ANY);
        gauge = new Gauge("Loading MPEG Video", false, GAUGE_MAX, 0);
        cmdExit = new Command("Exit", Command.EXIT, 2);
        cmdPlay = new Command("Play", Command.SCREEN, 1);
        form.addCommand(cmdPlay);
        form.addCommand(cmdExit);
        form.setCommandListener(this);
        form.append(textField);
    }

    public void startApp() {
        display.setCurrent(form);
    }

    public void pauseApp() {
    }

    public void destroyApp(boolean unconditional) {
    }

    public void commandAction(Command c, Displayable d) {
        if (c == cmdExit) {
            destroyApp(false);
            notifyDestroyed();
        } else if (c == cmdPlay) {

```

```
        gauge.setValue(0);
        form.append(gauge);
        DoraMPEGCanvas mpegCanvas = new DoraMPEGCanvas(this);
        mpegCanvas.initializeVideo(textField.getString());
    }
}

public void updateGauge(){
    int current = gauge.getValue();
    current = (current + GAUGE_MAX/GAUGE_LEVELS);
    gauge.setValue(current);
}
}
```

## 2. Kelas DoraMPEGCanvas (DoraMPEGCanvas.java)

```
/*
 * Nama File      : DoraMPEGCanvas.java
 * Developed by   : Lilik Haryanto
 */

import javax.microedition.midlet.*;
import javax.microedition.lcdui.*;
import javax.microedition.media.*;
import javax.microedition.media.control.*;
import java.io.*;

public class DoraMPEGCanvas extends Canvas implements CommandListener,
PlayerListener, Runnable {

    private DoraMPEGPlayer parent;
    private Display display;

    private Player player;
    private VideoControl videoControl;

    private String url;
    private Thread initializer;

    private Command btnClose;
    private Command btnReplay;

    public DoraMPEGCanvas(DoraMPEGPlayer parent) {
        super();
        this.parent = parent;
        display = Display.getDisplay(parent);
        btnClose = new Command("Close", Command.SCREEN, 1);
        addCommand(btnClose);
        setCommandListener(this);
    }

    public void initializeVideo(String url){
        this.url = url;
        initializer = new Thread(this);
        initializer.start();
    }

    public void run() {
        try {
```

```
        player = Manager.createPlayer(url);
        parent.updateGauge();
        player.addPlayerListener(this);
        player.realize();
        parent.updateGauge();
        player.prefetch();
        parent.updateGauge();
    } catch (IOException ioe) {
        Alert alert = new Alert("IOException thrown",
ioe.getMessage(),
        null, AlertType.ERROR);
        display.setCurrent(alert);
    } catch (MediaException me) {
        Alert alert = new Alert("MediaException thrown",
me.getMessage(),
        null, AlertType.ERROR);
        display.setCurrent(alert);
    }
    playVideo();
}

public void playVideo() {
    try {
        /* ambil video control dan set menjadi display sedang aktif
*/
        videoControl
        (VideoControl)player.getControl("VideoControl");
        if (videoControl != null) {
            videoControl.initDisplayMode(videoControl.USE_DIRECT_VIDEO,
this);
        }

        parent.updateGauge();
        int cHeight = this.getHeight();
        int cWidth = this.getWidth();
        videoControl.setDisplaySize(cWidth, cHeight);
        display.setCurrent(this);
        videoControl.setVisible(true);
        player.start();
    } catch (MediaException me) {
        Alert alert = new Alert("MediaException thrown",
me.getMessage(),
        null, AlertType.ERROR);
        display.setCurrent(alert);
    }
}

public void paint(Graphics g) {
    g.setColor(0, 0, 0);
    g.fillRect(0, 0, getWidth(), getHeight());
}

public void playerUpdate(Player p, String event, Object eventData)
{
    /* tambahkan pilihan "Replay" ketika video telah selesai */
    if (event == PlayerListener.END_OF_MEDIA) {
        if (btnReplay == null) {
            btnReplay = new Command("Replay", Command.SCREEN, 1);
        }
    }
}
```

```
        addCommand(btnReplay);
    }
}

public void commandAction(Command c, Displayable d) {
    if (c == btnReplay) {
        try{
            player.start();
        } catch (MediaException me) {
            Alert alert = new Alert("MediaException thrown",
                me.getMessage(), null, AlertType.ERROR);
            display.setCurrent(alert);
        }
    } else if (c == btnClose) {
        player.close();
        parent.form.delete(1);
        display.setCurrent(parent.form);
        url = null;
        parent = null;
    }
}
}
```

Dalam pengetesan, penulis menggunakan emulator SonyEricsson, karena emulator ini memiliki aplikasi untuk *serial networking* antara perangkat mobile dengan server (baca: Device Explorer). Untuk proses koneksi, bisa menggunakan kabel data (COM, USB), IrDA, ataupun Bluetooth.

Berikut adalah tampilan untuk aplikasi video player online:



Gambar 1. URL dari file .mpg pada Web Server



Gambar 2. Aplikasi menjalankan file .mpg

## Penutup

Manfaat dari aplikasi ini antara lain dapat menjadi dasar dari pengembangan aplikasi-aplikasi pemutar video online lainnya. Selain itu juga dapat dijadikan suatu media untuk e-Learning.

## Referensi

1. Sony Ericsson J2ME MMAPI 1.1 API Documentation (JSR-135).

## **Biografi Penulis**

**Robertus Lilik Haryanto.** Lahir di Klaten, 2 Oktober 1983. Menyelesaikan pendidikan program S1 pada program studi Teknik Informatika Universitas Sanata Dharma (USD), Yogyakarta pada bulan Agustus 2005. Pernah bekerja di sebuah perusahaan penyedia layanan *web hosting*, PT. Inter Lintas Media Yogyakarta sebagai *web programmer*. Selain itu, pernah mengajar pelatihan Java dan Delphi di Gama Learning Center (GLC), Yogyakarta. Saat ini bekerja sebagai developer di PT. Jati Piranti Solusindo (Ecom), Jakarta, sejak tahun 2005.



Menggeluti dunia pemrograman sejak tahun 1999 dan bahasa pemrograman yang paling digemari adalah Java (J2SE, J2ME, dan J2EE), C++, Microsoft C#.NET dan PHP. Selain melakukan beberapa penelitian, juga sedang mendalami beberapa bahasa pemrograman di atas.