Como gravar videos em Java ME

Aqui vai um exemplo de código para gravação de vídeo em Java ME:

dcrocha 19:48, 22 October 2007 (UTC)

```
package example;

import java.io.OutputStream;

import javax.microedition.io.Connector;
import javax.microedition.io.file.FileConnection;
import javax.microedition.lcdui.Alert;
import javax.microedition.lcdui.AlertType;
import javax.microedition.lcdui.Canvas;
import javax.microedition.lcdui.Command;
import javax.microedition.lcdui.CommandListener;
import javax.microedition.lcdui.Display;
import javax.microedition.lcdui.Displayable;
import javax.microedition.lcdui.Graphics;
import javax.microedition.media.Manager;
import javax.microedition.media.Player;
import javax.microedition.media.PlayerListener;
import javax.microedition.media.control.RecordControl;
import javax.microedition.media.control.VideoControl;

public class VideoCanvas extends Canvas implements PlayerListener, CommandListener {

 private VideoRecordMIDlet midlet = null;
 private Command start = new Command("Start",Command.OK,0);
 private Command stop = new Command("Stop",Command.OK,1);
 private Command exit = new Command("Exit",Command.BACK,0);
 private Command play = new Command("Play",Command.OK,2);
 private String status = "Not initialized";
 private Player player = null;
 private VideoControl control = null;
 private RecordControl record = null;
 private boolean recording = false;
 private FileConnection conn = null;
 private OutputStream stream = null;
 private String PATH = System.getProperty("fileconn.dir.videos") + "vi.3gp";

 /**
  * Constructor
  *
  * @param midlet
  */

 public VideoCanvas(VideoRecordMIDlet midlet) {
                System.out.println(PATH);
  this.midlet = midlet;
  addCommand(play);
  addCommand(start);
  addCommand(stop);
  addCommand(exit);
  setCommandListener(this);
 }

 public void commandAction(Command c, Displayable arg1) {
            if(c == start) {
                recording = true;
                startRecording();
            }
            else if(c == stop) {
                stop();
                recording = false;
            }
            else if(c == play) {
                recording = false;
                repaint();
                startPlaying();
            }
            else if(c == exit) {
                midlet.notifyDestroyed();
            }
 }

 /**
  * Paint
  */

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

        public void startRecording() {
            try {
                player = Manager.createPlayer("capture://video");
  player.addPlayerListener(this);
  player.realize();

                //setup recording
  record = (RecordControl)player.getControl("RecordControl");
  record.setRecordSizeLimit(300000);
  conn = (FileConnection)Connector.open(PATH,Connector.READ_WRITE);
  if(!conn.exists()) conn.create();
  stream = conn.openOutputStream();
  record.setRecordStream(stream);

  // Grab the video control and set it to the current display.
  control = (VideoControl)player.getControl("VideoControl");
  if (control != null) {
                    control.initDisplayMode(VideoControl.USE_DIRECT_VIDEO,this);
                    control.setDisplaySize(getWidth(),getHeight());
      control.setVisible(true);
                }
```


(C) Copyright Nokia 2013. All rights reserved.

```
                player.start();
                record.startRecord();
            }
            catch(Exception e) {
                Alert erro = new Alert("Erro",e.toString(),null,AlertType.ERROR);
                Display.getDisplay(midlet).setCurrent(erro);
                e.printStackTrace();
            }
}

/**
 * Start
 *
 */

public void startPlaying() {
            try {
                player = Manager.createPlayer(PATH);
 player.addPlayerListener(this);
 player.realize();

                // Grab the video control and set it to the current display.
 control = (VideoControl)player.getControl("VideoControl");
 if (control != null) {
                    control.initDisplayMode(VideoControl.USE_DIRECT_VIDEO,this);
                    control.setDisplaySize(176,144);
                    control.setDisplayLocation((getWidth() - control.getDisplayWidth()) / 2,(getHeight() - control.getDisplayHeight(
                    control.setVisible(true);
                }

                player.start();
            }
            catch(Exception e) {
                Alert erro = new Alert("Erro",e.getMessage(),null,AlertType.ERROR);
                Display.getDisplay(midlet).setCurrent(erro);
                e.printStackTrace();
            }
}

public void stop() {
            if(player != null) {
                try {
                    if(recording) {
                        record.stopRecord();
                        record.commit();
                        stream.close();
                    }
                    player.stop();
                    player.deallocate();
                    player.close();
                    player = null;
                    if(recording) {
                        Alert alert = new Alert("Mensagem","Gravado em " + PATH,null,AlertType.INFO);
                        Display.getDisplay(midlet).setCurrent(alert,this);
                        stream = null;
                    }

                    if(conn != null)
                        conn.close();
                }
                catch(Exception e) {

                }
            }
}

public void playerUpdate(Player p, String s, Object o) {

            if(p.getState() == Player.STARTED) {
                status = control.getDisplayHeight() + "=" + control.getDisplayWidth();
                setTitle(status);
            }

}

}
```


(C) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2013. All rights reserved.