



INTERFACE₁ GRAPHIQUE POUR FFMPEG - ITERATION 2

HUBLAU Alexandre
PALMIERI Adrien
DA SILVA CARMO Alexandre
CHEVRIER Jean-Christophe

INTRODUCTION

INTRODUCTION

Où en sommes nous dans le projet ?

- la fenêtre de conversion :
 - l'interfaçage pour les opérations de conversion est quasiment terminé ;
 - la partie graphique a été améliorée et le sera encore par la suite.
- la fenêtre de traitement :
 - l'interfaçage pour les traitement a été commencé ;
 - la partie graphique a été commencé et a bien évoluée ;
 - **le lien entre l'interfaçage et la fenêtre a été à peine commencé;**

Nous affrontons des difficultés :

- FFmpeg est un logiciel très volumineux, complexe, et compliqué ;
- L'implémentation entre les formats et les codecs est compliquée et peu documentée
- les bibliothèques java.awt et javax.swing sont deprecated (= obsolètes) ;

BONNES PRATIQUES MISES EN PLACE

BONNES PRATIQUES MISES EN PLACE

- ajout de **classes pour la gestion des Threads** (fenêtres et ffmpeg) ;
- interfaçage repensé :
 - classes pour la **gestion des flux** (filtrage et consommation) ;
 - classes **iterator - iterable** pour la **manipulation des flux** ;
 - classes **répertoriant en constantes les flags** et autres des commandes ffmpeg ;
 - classes pour la **construction de requêtes** intuitives ;
- simplification de l'interface

CONCLUSIONS DU DERNIER RENDEZ-VOUS

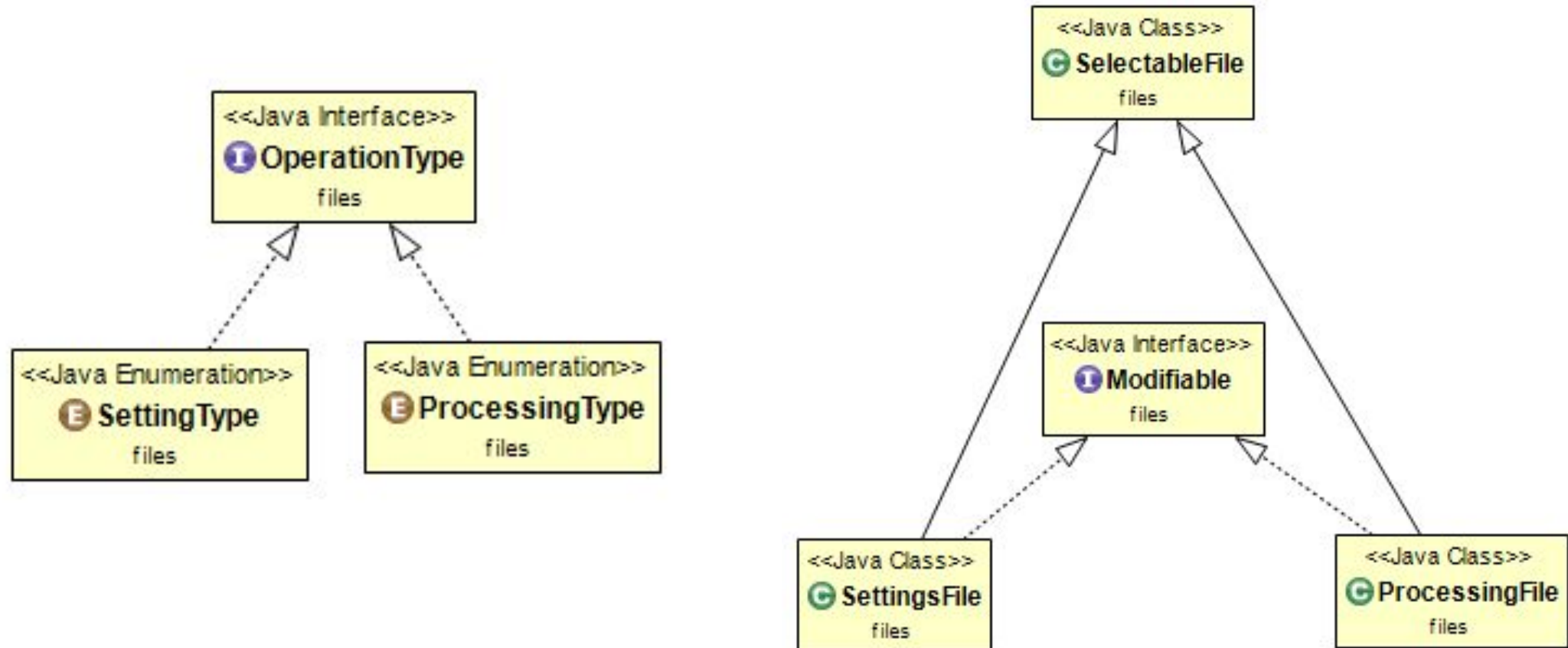
CONCLUSIONS DU DERNIER RENDEZ-VOUS

M. Ouni a suggéré de **repenser la fenêtre de traitement** :

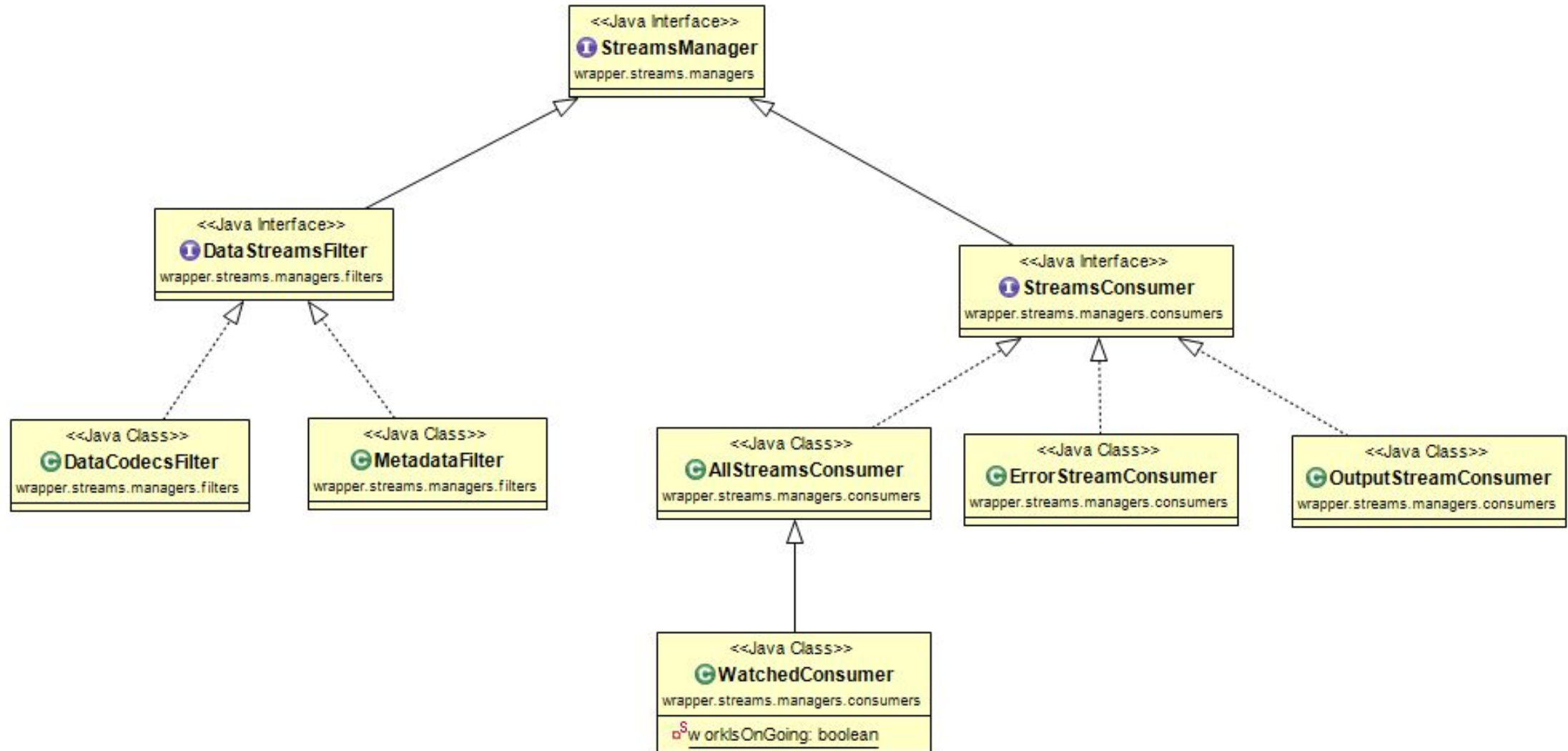
- réaliser des fonctionnalités **plus simples et moins classiques** ;
- **réaliser qu'en second plan** les fonctionnalités qui avaient été prévues de base ;
- **IMPORTANT** : avoir des fonctionnalités intéressantes à présenter en fin de projet, et **ne pas se laisser écraser par des objectifs d'itérations trop complexes et trop éloignées du sujet.**

PATRONS DE CONCEPTION

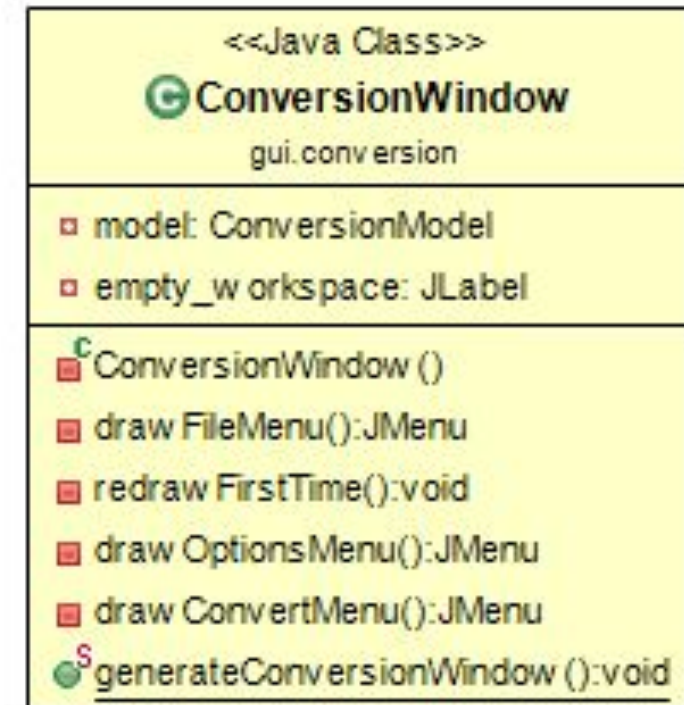
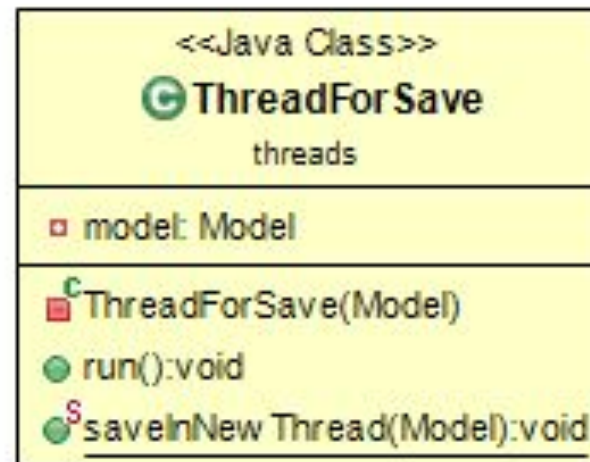
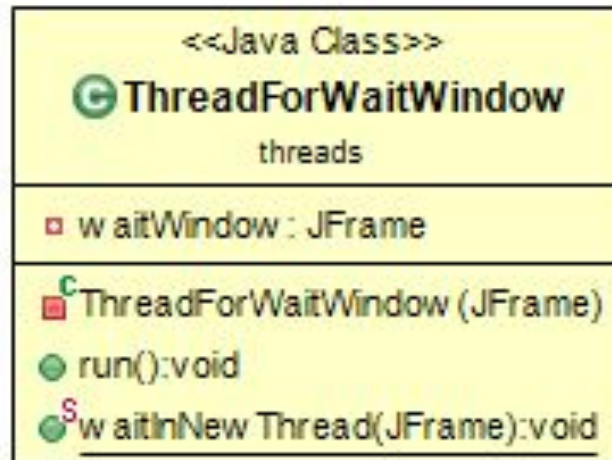
PATRON DE CONCEPTION STRATEGIE



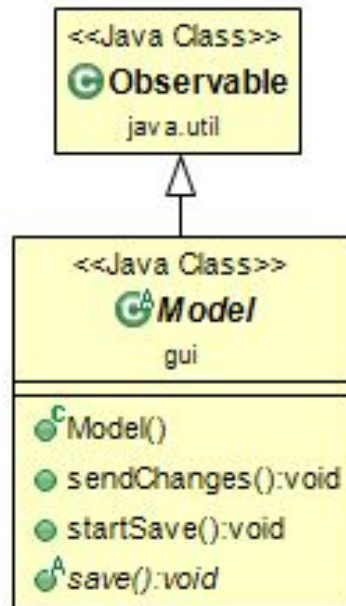
PATRON DE CONCEPTION STRATEGIE



PATRON DE CONCEPTION FACTORY



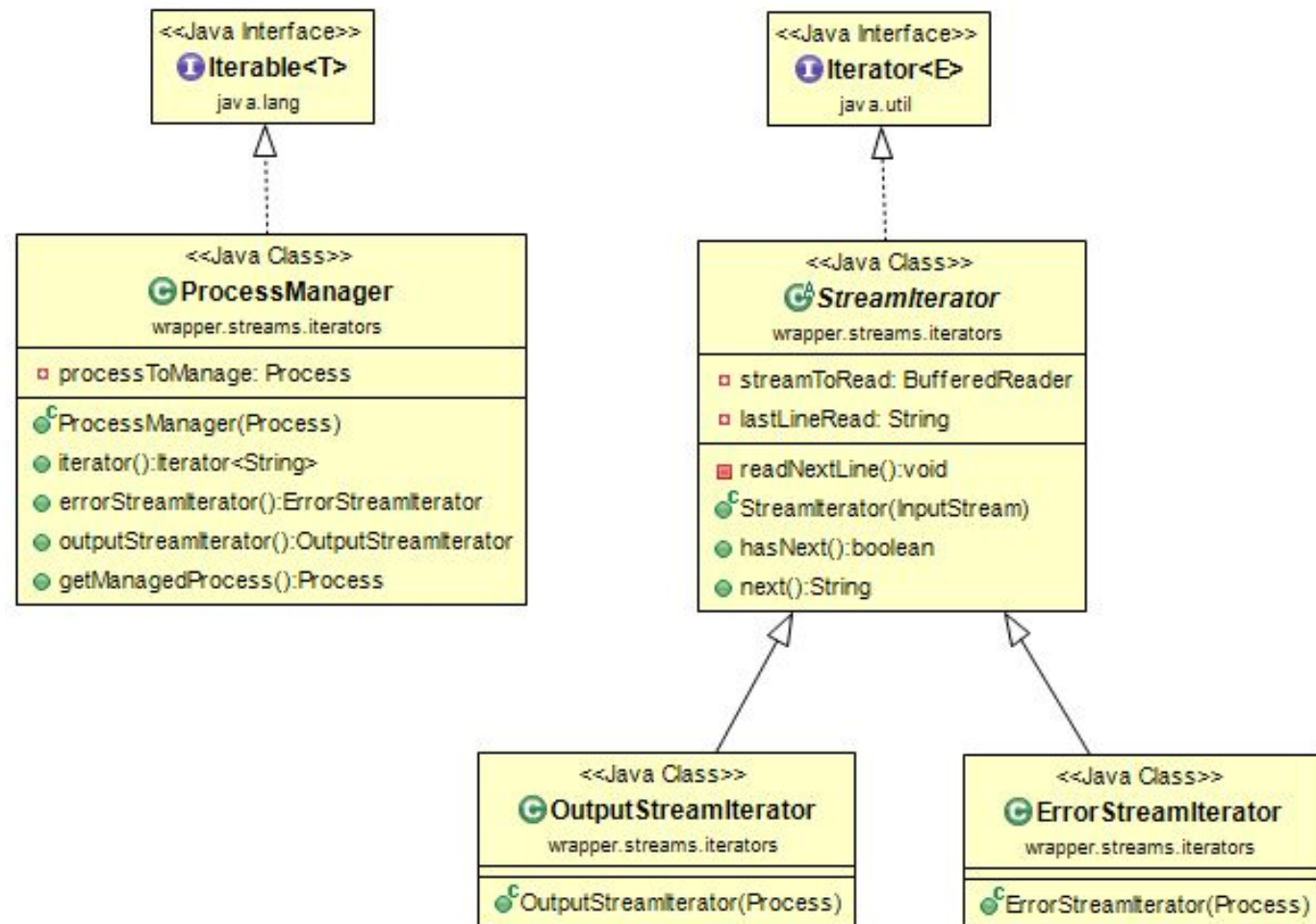
PATRON DE CONCEPTION MVC



etc...

PATRON DE CONCEPTION

ITERATOR-ITERABLE



“PSEUDO” PATRON DE CONCEPTION

<<Java Class>> Request wrapper.runtime.details
request: List<String> input: String output: String
Request() Request(String) Request(String,String) from(String):Request to(String):Request askSomethingElse(String[]):void codecs():Request videoCodec(String):Request videoBitrate(String):Request resolution(String):Request framerate(String):Request audioCodec(String):Request audioBitrate(String):Request samplingRate(String):Request numberAudioChannels(String):Request crop(String,String,String,String):Request rotate():Request extractImage(String):Request resizeImage(String,String):Request result():ProcessManager make():Request

```

Request()
Request(String)
Request(String,String)
from(String):Request
to(String):Request
askSomethingElse(String[]):void
codecs():Request
videoCodec(String):Request
videoBitrate(String):Request
resolution(String):Request
framerate(String):Request
audioCodec(String):Request
audioBitrate(String):Request
samplingRate(String):Request
numberAudioChannels(String):Request
crop(String,String,String,String):Request
rotate():Request
extractImage(String):Request
resizeImage(String,String):Request
result():ProcessManager
make():Request

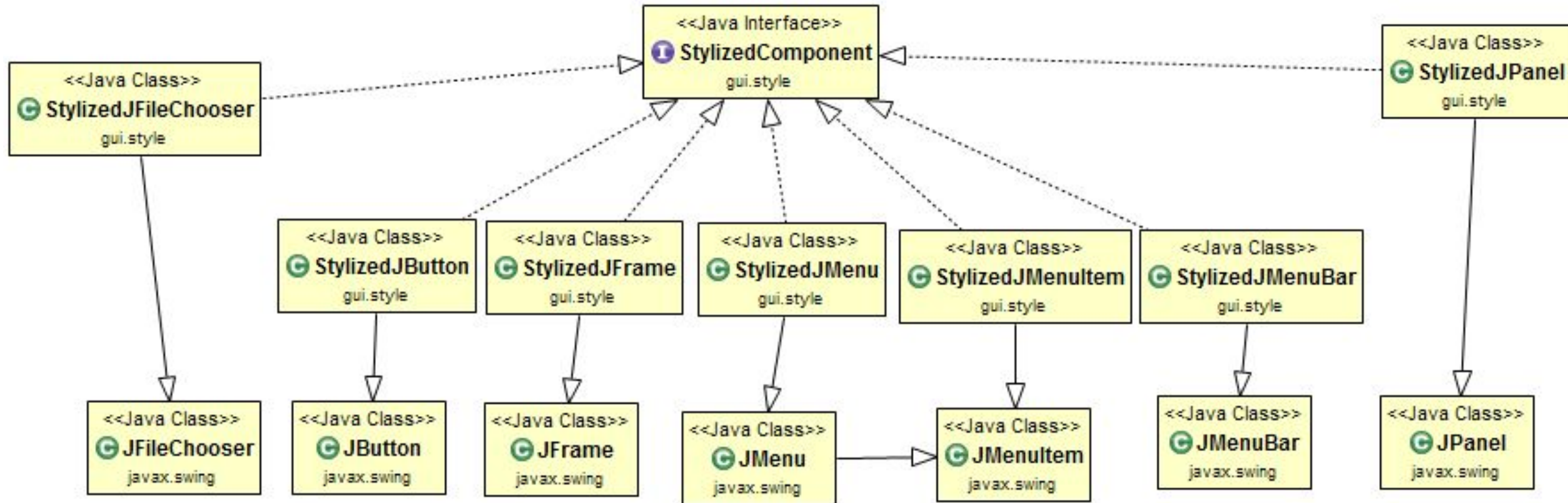
```

```

new Request()
  .from("exampleInput.avi")
  .videoCodec("mpeg4")
  .videoBitrate("500")
  .framerate("25")
  .audioCodec("mp3")
  .audioBitrate("400")
  .to("exampleOutput.mp4")
  .make();

```

“PSEUDO” PATRON DE CONCEPTION



GESTION DES RESSOURCES.

GESTIONNAIRE DES RESSOURCES

- méthodes de **mapage** des fichiers (check...) pour la **vérification** et la **régénération** des ressources durant l'**utilisation** du logiciel.

```
public static boolean checkConversionImports() {  
    return checkImports() && checkDirectory(CONVERSION_OLD_IMPORTS);  
}
```

```
public static void secureConversionImports() throws UnfindableResourceException {  
    if(!checkConversionImports())  
        throw new UnfindableResourceException(UnfindableResourceException.UAITSR_CONVERSION_IMPORTS);  
}
```

- méthode de **nettoyage** des ressources à la **fermeture** du logiciel.

```
public static boolean clearResources() {  
    boolean succeed = clearTemporaryFiles();  
    boolean succeed2 = clearErr();  
    boolean succeed3 = clearOut();  
    return succeed && succeed2 && succeed3;  
}
```

GESTIONNAIRE DES RESSOURCES

- utilisation de **fichiers horodatés** pour la garantie de **noms uniques**, et pour **éviter l'épuisement de l'espace de noms**.

```
public static String day() {  
    String now = Instant.now().toString();  
    return now.substring(0, now.indexOf("T"));  
}  
/**  
 * @return String      DATE COURANTE (YYYY-MM-DD) + MOMENT COURANT (EN MILLISECONDES).  
 */  
public static String now() {  
    return day()+"_"+(System.currentTimeMillis()+60*60*10*10*10);  
}
```

```
* @throws UnfindableResourceException  Exception sur les ressources introuvables.  
*/  
public static String _temporary() throws UnfindableResourceException {  
    ResourceManager.secureTemporaryFiles();  
    return TEMPORARY_FILES_PREFIX+ResourceConstants.now()+JPG;  
}
```

INTERFACAGE.

INTERFACAGE

Nous avons **répertorié les flags et autres arguments** acceptés en ligne de commande par FFmpeg.

```

7  * en heriter ( d'ou la presence du final devant class).
8  *
9  * Auteurs du projet :
10 * @author HUBLAU Alexandre, PAMIERI Adrien, DA SILVA CARMO Alexandre, et CHEV
11 */
12 public interface FlagConstants {
13     public final static String FLAG_SUPPORTED_CODECS = "-codecs";
14
15     public final static String FLAG_QUALITY = "-crf";
16
17     public final static String FLAG_VIDEO_CODEC = "-vcodec";
18     public final static String FLAG_VIDEO_BITRATE = "-b:v";
19     public final static String FLAG_FRAMERATE = "-r";
20     public final static String FLAG_RESOLUTION = "-s";
21
22     public final static String FLAG_AUDIO_CODEC = "-acodec";
23     public final static String FLAG_AUDIO_BITRATE = "-b:a";
24     public final static String FLAG_SAMPLING_RATE = "-ar";
25     public final static String FLAG_NUMBER_AUDIO_CHANNELS = "-ac";
26

```

```

9  *
10 * Auteurs du projet :
11 * @author HUBLAU Alexandre, PAMIERI Adrien, DA SILVA CARMO Alexandre, et CHEVRIER Jean-christophe
12 */
13 public interface ResolutionConstants {
14     public final static String[] CIF_RESOLUTIONS = {"128x96", "176x144", "352x288", "704x576", "14
15
16     public final static String[] VGA_RESOLUTIONS = {"160x120", "320x240", "640x480", "800x600", "8
17
18     public final static String[] GA_RESOLUTIONS = {"1024x768", "1600x1200", "1280x1024", "1366x768
19         "2560x1600", "2560x2048", "3200x2048", "3840x24
20
21     public final static String[] CGA_AND_EGA_RESOLUTIONS = {"320x200", "640x350"};
22
23     public final static String[] HD_RESOLUTIONS = {"852x480", "1280x720", "1920x1080"};
24
25     public final static String[] ALL_RESOLUTIONS = getAllResolutions();
26
27     public static String[] getAllResolutions() {
28         List<String> allResolutions = new ArrayList<String>();
29         allResolutions.addAll(Arrays.asList(CIF_RESOLUTIONS));
30         allResolutions.addAll(Arrays.asList(VGA_RESOLUTIONS));
31         allResolutions.addAll(Arrays.asList(GA_RESOLUTIONS));
32         allResolutions.addAll(Arrays.asList(CGA_AND_EGA_RESOLUTIONS));
33         allResolutions.addAll(Arrays.asList(HD_RESOLUTIONS));
34         return allResolutions.toArray(new String[allResolutions.size()]);
35     }
36

```


INTERFACAGE

Réalisation de **requêtes encadrées** par des **outils** adaptant ffmpeg en java.
Utilisation des flags pour constituer les requêtes.

```

    /**
     * @return la requete this.
     */
    public Request framerate(String framerate) {
        if(framerate==null) throw new NullPointerException("Framerate null !");
        if(Integer.parseInt(framerate)<=0) throw new IllegalArgumentException("Framerate negatif ou nul !");
        askSomethingElse(new String[]{FLAG_FRAMERATE, framerate});
        return this;
    }

```

```

    /**
     * @return la requete this.
     */
    public Request frame(String time) {
        if(time==null) throw new NullPointerException("Time null !");
        cut(time, ONE_SECOND);
        framerate("1");
        return this;
    }

```

```

/**
 * [ COUPER. ]
 *
 * [ ----- ] => input video
 *
 * [ ----- ] => output video
 *      time           time+period
 *
 *
 * @param time    le temps de depart de coupage de la video.
 * @param period  la periode a conserver de la video a partir du temps de depart.
 *
 * @return la requete this.
 */
public Request cut(String time, String period) {
    if(time==null) throw new NullPointerException("Time null !");
    if(period==null) throw new NullPointerException("Period null !");
    askSomethingElse(new String[]{FLAG_PERIOD[0], time, FLAG_PERIOD[1], period});
    return this;
}

```

GESTION DES CODECS.

Gestion des codecs

- Problématique : pour chaque codec il faut donner la bonne extension à FFMpeg
- Solution :
 - utilisateur choisit en premier le format en sortie
 - logiciel affiche uniquement codecs compatible avec le format
 - mise à jour en temps réel de l'interface selon le codec choisi
 - ajout automatique de l'extension par le logiciel

Gestion des codecs

- Première map association Format -> Codec vidéo
- Seconde map association Codec vidéo -> Codec audio

```
public final static String[] ALL_EXTENSIONS = {".3g2", ".3gp", ".asf", ".avi", ".flv", ".m4v", ".mov", ".mkv", ".mp2", ".mp4",  
                                              ".mpeg", ".mpg", ".ogg", ".webm", ".wmv"};  
public final static String[] ALL_SUPPORTED_VIDEO_CODECS = SystemRequests.askVideoCodecs();  
public final static String[] ALL_SUPPORTED_AUDIO_CODECS = SystemRequests.askAudioCodecs();  
// List who associates file extensions (key) with an hashmap containing compatible audio/video codecs  
public final static Map<String, Map<String, List<String>>> CORRESPONDING_EXTENSION;
```

Problématique : générer une hashmap finale statique

Gestion des codecs

- Générer une hashmap finale statique :
 - utilisation du bloc static
 - utilisation de la methode unmodifiableMap(Map)

```
static {
    Map<String, Map<String, List<String>>> initialize_extensions = new HashMap<String, Map<String, List<String>>>();
    Map<String, List<String>> compatible_codecs = new HashMap<String, List<String>>();
    List<String> audioCodecList = new ArrayList<String>();
    for(String extension : ALL_EXTENSIONS) {
        switch(extension){
            case ".3g2":
                //audio
                audioCodecList.add("aac");
                audioCodecList.add("amr_nb");
                audioCodecList.add("amr_wb");
                audioCodecList.add("evrc");
                audioCodecList.add("smv");
                //mpeg4
                compatible_codecs.put("mpeg4", audioCodecList);
                //h263
                compatible_codecs.put("h263", audioCodecList);
                //h264
                compatible_codecs.put("h264", audioCodecList);
                break;
            case ".3gp":
                //audio
                audioCodecList.add("aac");
                audioCodecList.add("amr_nb");
                audioCodecList.add("amr_wb");
                //mpeg4
                compatible_codecs.put("mpeg4", audioCodecList);
                //h263
                compatible_codecs.put("h263", audioCodecList);
                //h264
                compatible_codecs.put("h264", audioCodecList);
                break;
            (...).
        }
        initialize_extensions.put(extension, compatible_codecs);
        compatible_codecs = new HashMap<String, List<String>>();
        audioCodecList = new ArrayList<String>();
        audioCodecList.clear();
    }
    CORRESPONDING_EXTENSION = Collections.unmodifiableMap(initialize_extensions);
}
```

Gestion des codecs

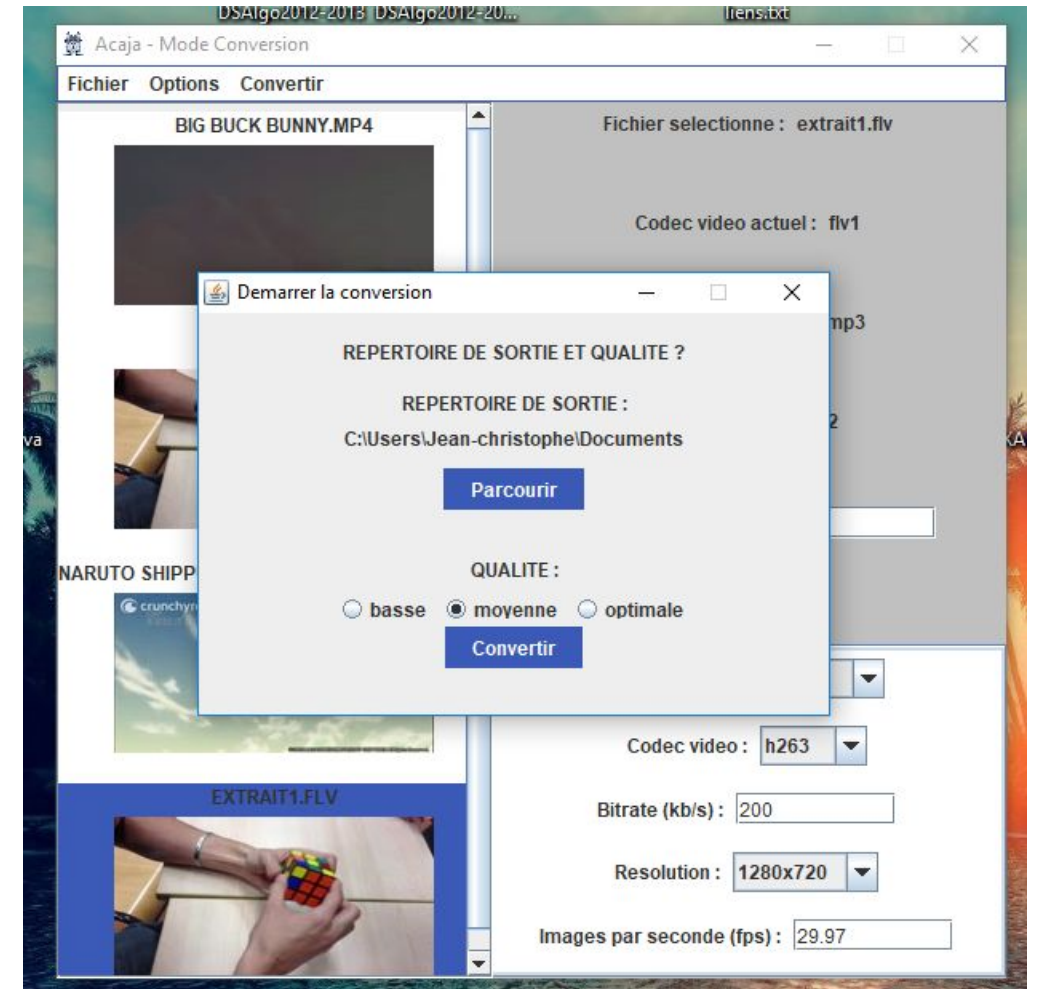
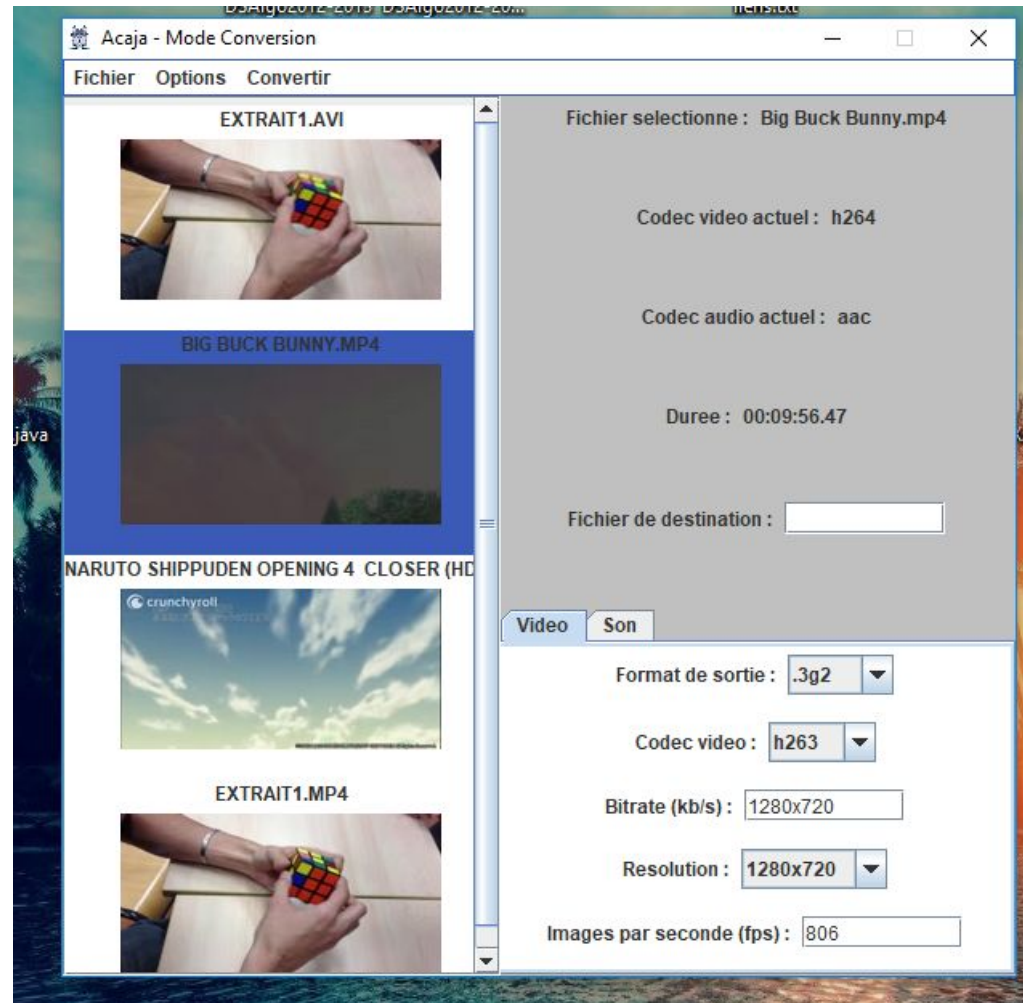
- Associations des formats / codecs audio/vidéo peu documentée
- Tests en cours, mise en place d'un document de test

<u>FORMAT ENTREE</u>	<u>CODEC VIDEO</u>	<u>CODEC AUDIO</u>	<u>RÉSOLU TION</u>	<u>BITRATE VIDEO</u>	<u>BITRATE AUDIO</u>	<u>FPS</u>	<u>ECHANTI LLONAG E</u>	<u>NB CANAUX</u>	<u>FORMAT SORTIE</u>	<u>STATUT</u>
MP4	H264	MP3	1280 x 720	1205	384	25	48000	2	FLV	<u>SUCCES</u>
MP4	H263	AAC	1280 x 720	1205	384	25	48000	2	3G2	<u>ECHEC : RESOLUTION</u>
MP4	H264	AAC	1280 x 720	1205	384	25	48000	2	3G2	<u>SUCCES</u>

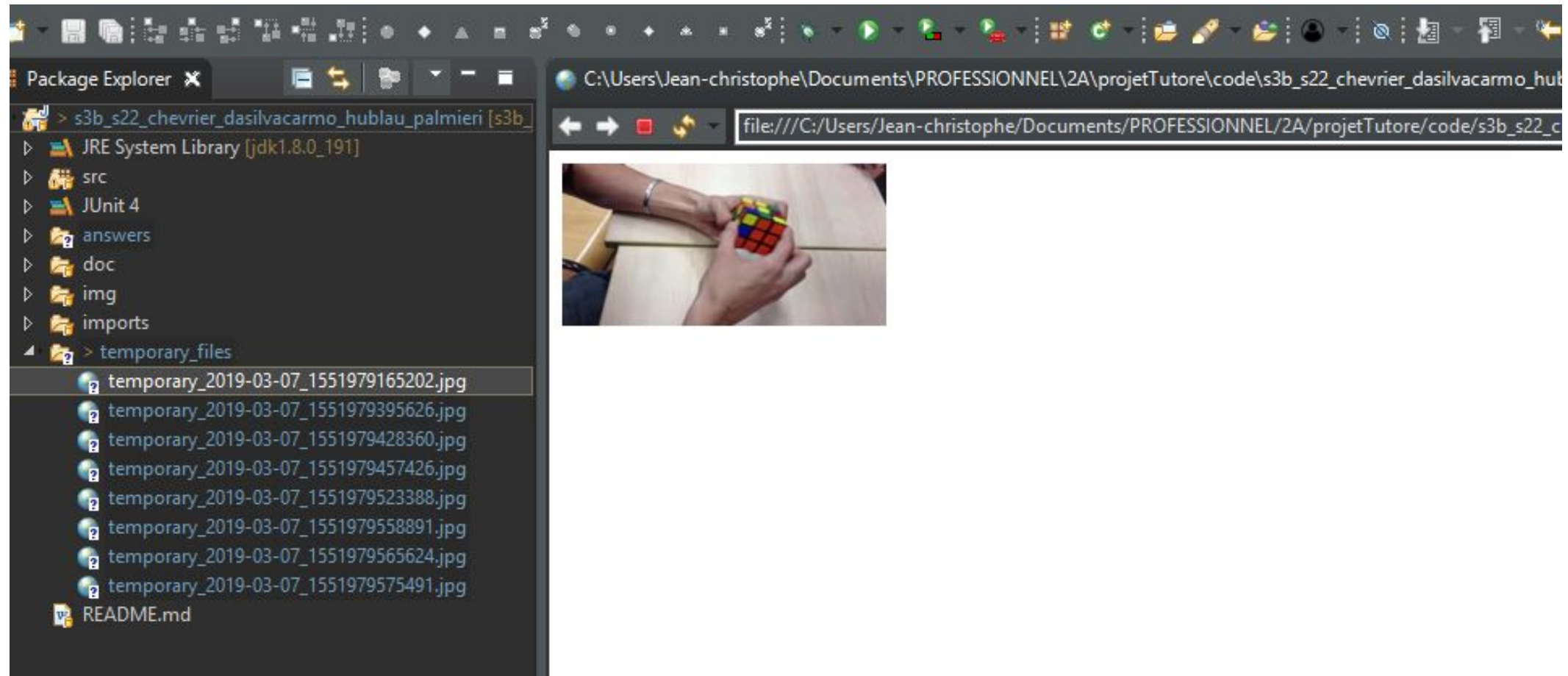
supplémentaire

LES FENETRES DU LOGICIEL.

FENETRE DE CONVERSION



FENETRE DE CONVERSION



FENETRE HISTORIQUE DES REPONSES

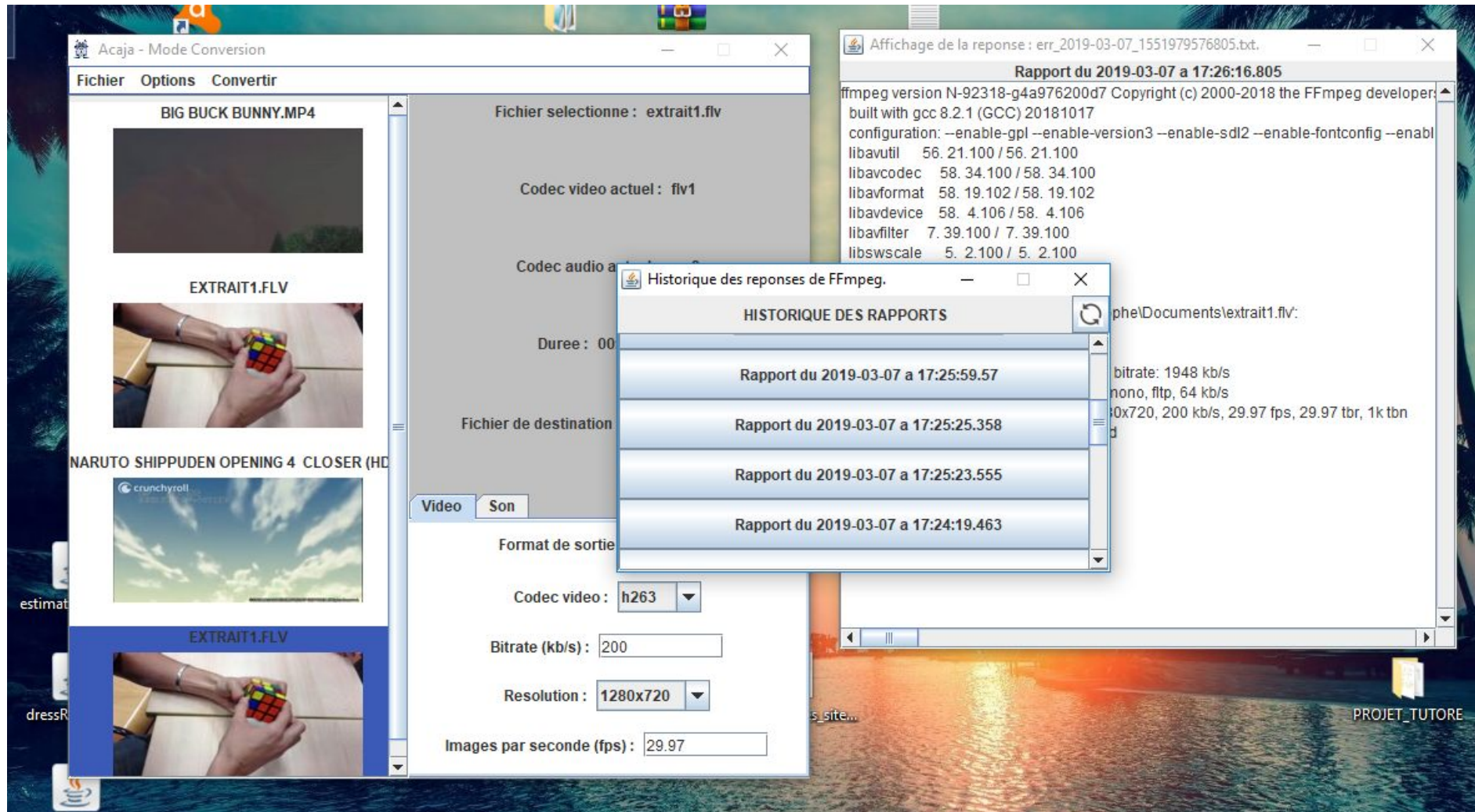
```

package Explorer X
  wrapper.streams.managers
  wrapper.streams.managers.consumers
  wrapper.streams.managers.filters
  uml.png
  uml.ucls
  JUnit 4
  answers
  stderr
    err_2019-03-07_1551979165421.txt
    err_2019-03-07_1551979167290.txt
    err_2019-03-07_1551979167665.txt
    err_2019-03-07_1551979167988.txt
    err_2019-03-07_1551979395796.txt
    err_2019-03-07_1551979397885.txt
    err_2019-03-07_1551979428527.txt
    err_2019-03-07_1551979430282.txt
    err_2019-03-07_1551979457586.txt
    err_2019-03-07_1551979459463.txt
    err_2019-03-07_1551979523555.txt
    err_2019-03-07_1551979525358.txt
    err_2019-03-07_1551979559057.txt
    err_2019-03-07_1551979560217.txt
    err_2019-03-07_1551979565821.txt
    err_2019-03-07_1551979567636.txt
    err_2019-03-07_1551979575660.txt
    err_2019-03-07_1551979576805.txt
  stdout
    out_2019-03-07_1551979167540.txt
    out_2019-03-07_1551979167863.txt

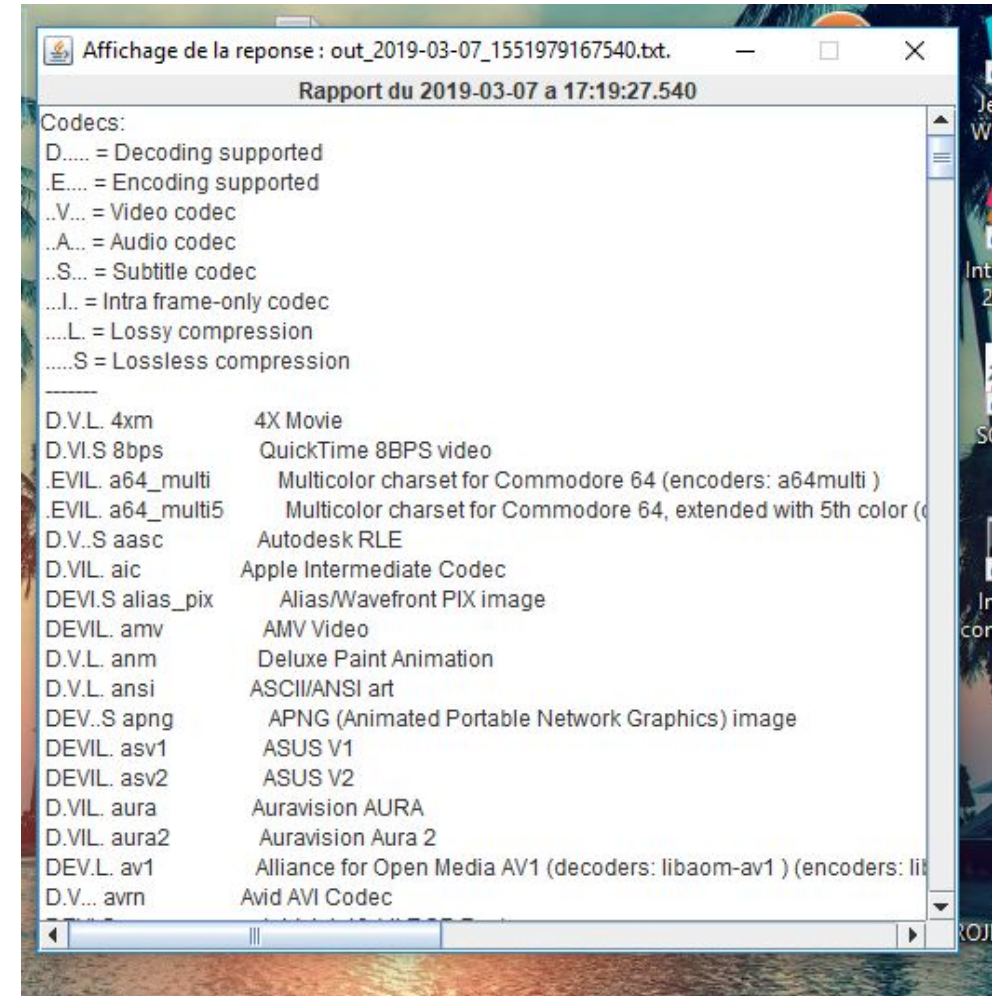
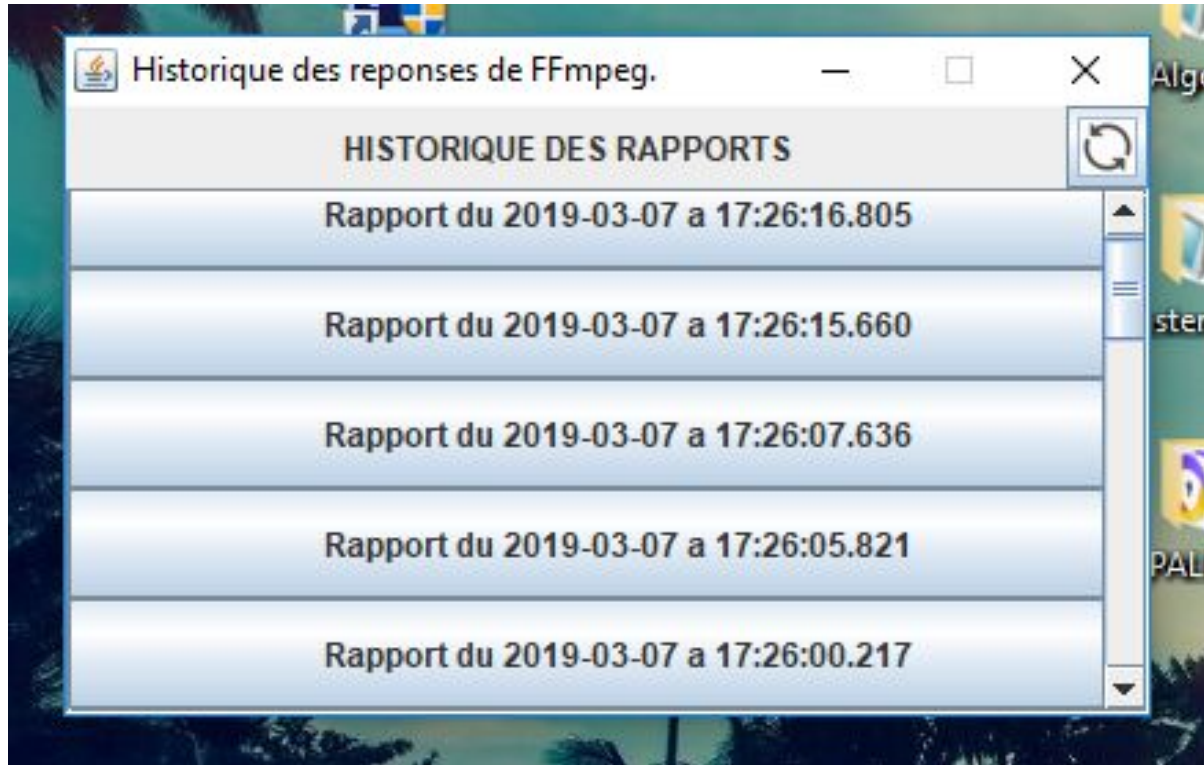
err_2019-03-07_1551979165421.txt X
1 ffmpeg version N-92318-g4a976200d7 Copyright (c) 2000-2018 the FFmpeg developers
2 built with gcc 8.2.1 (GCC) 20181017
3 configuration: --enable-gpl --enable-version3 --enable-sdl2 --enable-fontconfig --enable-gnutls --enable-iconv --enable-l
4 libavutil      56. 21.100 / 56. 21.100
5 libavcodec     58. 34.100 / 58. 34.100
6 libavformat    58. 19.102 / 58. 19.102
7 libavdevice    58.  4.106 / 58.  4.106
8 libavfilter     7. 39.100 /  7. 39.100
9 libswscale     5.  2.100 /  5.  2.100
10 libswresample  3.  2.100 /  3.  2.100
11 libpostproc   55.  2.100 / 55.  2.100
12 [avi @ 000001a1e6b5a540] non-interleaved AVI
13 Input #0, avi, from 'C:\Users\Jean-christophe\Documents\extrait1.avi':
14 Metadata:
15   encoder      : Lavf58.19.102
16 Duration: 00:00:30.10, start: 0.000000, bitrate: 1311 kb/s
17   Stream #0:0: Video: mpeg4 (Simple Profile) (FMP4 / 0x34504D46), yuv420p, 1280x720 [SAR 1:1 DAR 16:9], 1233 kb/s, 29.97 f
18   Stream #0:1: Audio: mp3 (U[0][0][0] / 0x0055), 44100 Hz, mono, fltp, 64 kb/s
19 Stream mapping:
20   Stream #0:0 -> #0:0 (mpeg4 (native) -> mjpeg (native))
21 Press [q] to stop, [?] for help
22 [swscaler @ 000001a1e7f70600] deprecated pixel format used, make sure you did set range correctly
23 Output #0, image2, to 'C:\Users\Jean-christophe\Documents\PROFESSIONNEL\2A\projetTutore\code\s3b_s22_chevrier_dasilvacarmo_h
24 Metadata:
25   encoder      : Lavf58.19.102
26   Stream #0:0: Video: mjpeg, yuvj420p(pc), 200x100 [SAR 8:9 DAR 16:9], q=2-31, 200 kb/s, 1 fps, 1 tbn, 1 tbc
27 Metadata:
28   encoder      : Lavc58.34.100 mjpeg
29 Side data:
30   cpb: bitrate max/min/avg: 0/0/200000 buffer size: 0 vbv_delay: -1
31 frame= 0 fps=0.0 q=0.0 size=N/A time=00:00:00.00 bitrate=N/A speed= 0x
32 frame= 1 fps=1.0 q=3.5 lsize=N/A time=00:00:01.00 bitrate=N/A dup=1 drop=1 speed=0.97x
33 video:4kB audio:0kB subtitle:0kB other streams:0kB global headers:0kB muxing overhead: unknown
34

```

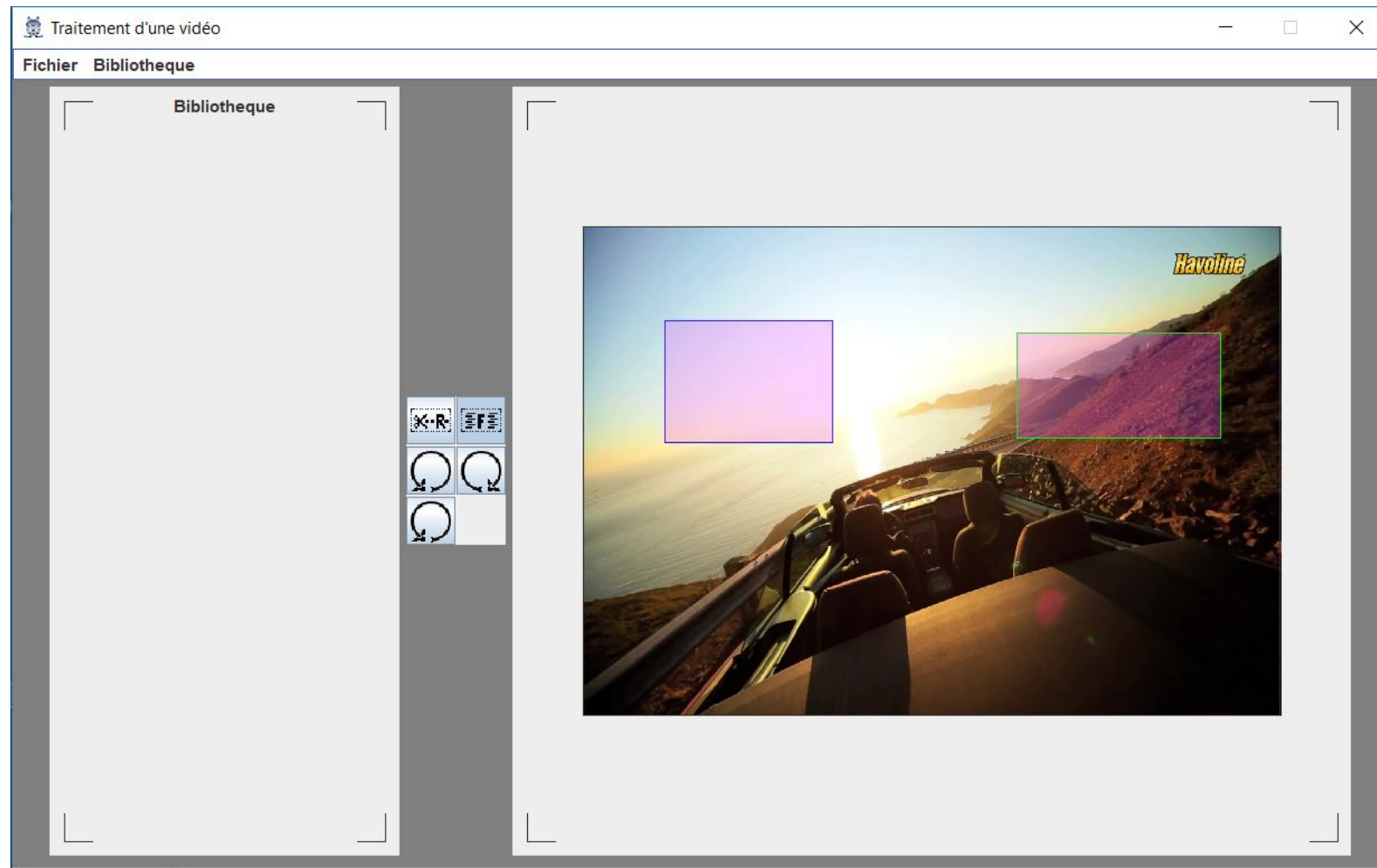

FENETRE HISTORIQUE DES REPONSES



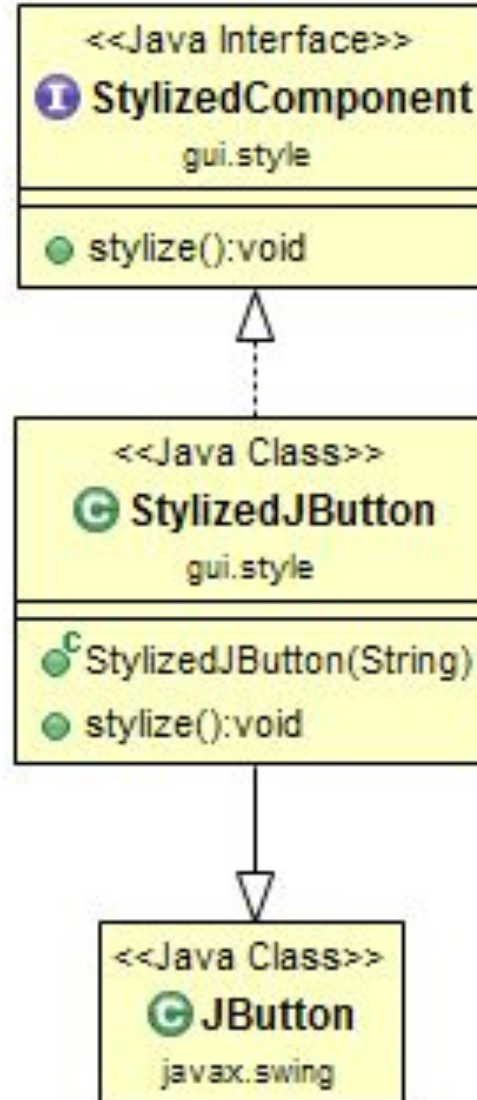
FENETRE HISTORIQUE DES REPONSES



FENÊTRE DE TRAITEMENT



DESIGN



DESIGN DE LA FENÊTRE

```
public class StylizedJButton extends javax.swing.JButton implements StylizedComponent {  
  
    public StylizedJButton(String text) {  
        super(text);  
        this.stylize();  
    }  
  
    @Override  
    public void stylize() {  
        this.setBackground(StyleTheme.BACKGROUND_COLOR_SECONDARY);  
        this.setOpaque(true);  
        this.setBorderPainted(false);  
        this.setForeground(Color.WHITE);  
        this.setFocusPainted(false);  
    }  
}
```

Itération 3

- Ajouter des fonctionnalités à la partie traitement comme par exemple des sous-titres.
- Finir quelques détails sur l'IHM de la partie conversion.
- Régler les problèmes de compatibilité des codecs dans la partie conversion.
 - résolution
 - format / codec

“DEMO”

CONCLUSION