**DOCUMENT DE**

**SYNTHESE**



**Sujet**

**S22 : Réalisation d’une interface graphique**

**pour un logiciel de traitement vidéo**

Tuteur :

OUNI Slim

Membre de l’équipe du projet :

DA SILVA CARMO Alexandre

PALMIERI Adrien

HUBLAU Alexandre

CHEVRIER Jean-Christophe

Groupe :

S3B

Ce document réunit :

La liste des fonctionnalités

Les maquettes

Les diagrammes de cas d’utilisation

Les diagrammes de classes

Déposé sur le dépôt bitbucket le 09/12/2018

**SOMMAIRE**

INTRODUCTION

SUJET DU PROJET

CHOIX DE CONCEPTION

QUELQUES PRECISIONS

LISTE DES FONCTIONNALITES

PRINCIPAUX GROUPES DE FONCTIONNALITES

GROUPE DE FONCTIONNALITES : « VIA LA FENETRE DE TRAITEMENT »

MENUS

PANELS BIBLIOTHEQUES

BIBLIOTHEQUES VERS CHRONOLOGIES

PANEL CHRONOLOGIES

PANEL BARRE DE LECTURE

PANEL VIDEO

GROUPE DE FONCTIONNALITES : « VIA LA FENETRE DE CONVERSION »

MENUS

PANEL BIBLIOTHEQUE

PANEL « RESUME »

PANELS PARAMETRES DU FICHIER

GROUPE DE FONCTIONNALITES : « OPTIONNEL »

MAQUETTES DES FENTRES DU LOGICIEL

PREMIER POTOTYPE ET REFLEXIONS

MAQUETTE DE LA FENETRE DE TRAITEMENT

MAQUETTE DE LA FENETRE DE CONVERSION

DIAGRAMMES DE CAS D’UTILISATION

DIAGRAMME DE CAS D’UTILISATION D’OUVERTURE DU LOGICIEL

DIAGRAMME DE CAS D’UTILISATION TRAVAILLER DANS LE MODE TRAITEMENT

DIAGRAMME DE CAS D’UTILISATION TRAVAILLER DANS LES BIBLIOTHEQUES

DIAGRAMME DE CAS D’UTILISATION TRAVAILLER DANS LES CHRONOLOGIES

DIAGRAMME DE CAS D’UTILISATION TRAVAILLER DANS LE MODE CONVERSION

DIAGRAMME DE CAS D’UTILISATION TRAVAILLER DANS LA BIBLIOTHEQUE

DIAGRAMME DE CAS D’UTILISATION TRAVAILLER SUR UN FICHIER

PROGRAMMATION : DIAGRAMMES DE CLASSES ET FONCTIONNEMENT

PROGRAMMES FEDERATEURS

DIAGRAMME DE CLASSES

DESCRIPTION DU FONCTIONNEMENT

Main

OpeningWindow

ProcessingPanel

ConversionPanel

PACKAGE files

DIAGRAMME DE CLASSES

DESCRIPTION DU FONCTIONNEMENT

SelectedFile

SettingsFile

ProcessedFile

PACKAGE ffmpeg\_tools

DIAGRAMME DE CLASSES

DESCRIPTION DU FONCTIONNEMENT

FfmpegRuntime

UserRequests

SystemRequests

PACKAGE tools

DIAGRAMME DE CLASSES

DESCRIPTION DU FONCTIONNEMENT

TemporaryFilesManager

Id Séquence

TemporaryFilesFilter

DirectoryAbsentException

PACKAGE gui\_processing

DIAGRAMME DE CLASSES

DESCRIPTION DU FONCTIONNEMENT

ConversionPanel

Conversion Model

VUES ET CONROLEURS

LibraryView et LibraryController

VideoView et VideoController

PlaybackBarView et PlaybackBarController

TimelineView et TimelineController

GeneralKeyBoardController

PACKAGE gui\_conversion

DIAGRAMME DE CLASSES

DESCRIPTION DU FONCTIONNEMENT

ConversionPanel

Conversion Model

VUES ET CONROLEURS

LibraryView, et LibraryController

GeneralKeyBoardController

SummaryView

TabsView, SoundSettingsPanel, et VideoSoundSettingsPanel

CONCLUSION

INTRODUCTION

*SUJET DU PROJET*

On rappelle que le sujet de notre projet consiste en la réalisation d’une interface graphique. Cette interface graphique a pour mission d’adapter FFmpeg un logiciel de montage vidéo, utilisable qu’en ligne de commande. Le but principal est de rendre accessible FFmpeg aux utilisateurs non informaticiens, peu coutumier de l’invite de commande sous Windows, du terminal sous Linux, ou même de tout type de Shell (= interpréteur en ligne de commandes).

*CHOIX DE CONCEPTION*

Nous avons choisi d’utiliser comme technologie le langage informatique JAVA. FFmpeg étant codé en C, nous ne pourrons pas communiquer directement avec ses librairies. Ce pourquoi nous aurons des classes java réalisant un interfaçage entre notre logiciel et FFmpeg (Cf Package ffmpeg\_tools).

Notre logiciel sera découpé en plusieurs packages. Et dans chaque package, des patrons de conception seront utilisés. Ces patrons de conception seront adaptés aux fonctionnalités des classes en cause. Principal but recherché : établir une architecture logicielle cohérente et logique, avoir un code factorisé, éviter toute redondance dans les programmes, temps d’exécution optimal, économie de la mémoire.

Un choix a été réalisé dans notre équipe de projet, nous avons choisis de réaliser tout le code du logiciel en ANGLAIS. Enfin… On parle essentiellement ici des noms des classes, des méthodes et des attributs, qui seront donc écrits en anglais.

*QUELQUES PRECISIONS*

Des maquettes de l’interface graphique seront présentés plus tard dans le document, on tient à préciser que ce sont les composants graphiques qu’il faut retenir. En effet les thèmes, les couleurs choisis ici ne sont pas définitif. Les thèmes et autres petits détails seront peut-être modifiés plus tard, au gré de nos envies et de nos préférences.

Ensuite, des diagrammes de classes seront présentés par la suite. Nous nous devons de préciser quelques détails. Il est possible lors de la programmation concrète du logiciel que nous rajoutions des méthodes internes (méthodes déclarées en private) pour factoriser le code, ou même que nous rajoutions des classes, ou des exceptions personnalisées (Classes héritant d’Exception ou même de Throwable). Soyons honnête nous avons pu oublier certains détails, ou même certains détails améliorant le codage du logiciel, c’est humain… Il y aura donc peut-être au cours des étapes suivantes de l’étude du logiciel, des modifications minimes des diagrammes de classes.

Les maquettes, les diagrammes de cas d’utilisation, et els diagrammes de classes sont – vous allez le voir - plutôt conséquents. Nous vous conseillons donc d’en plus de les regarder ce document de consulter les répertoires suivants du dépôt :

s3b\_s22\_chevrier\_dasilvacarmo\_hublau\_palmieri\doc\etape\_2\maquettes\

s3b\_s22\_chevrier\_dasilvacarmo\_hublau\_palmieri\doc\etape\_2\ diagrammes\_de\_\_cas\_d\_utilisation\

s3b\_s22\_chevrier\_dasilvacarmo\_hublau\_palmieri\doc\etape\_2\ diagrammes\_de\_classes\

s3b\_s22\_chevrier\_dasilvacarmo\_hublau\_palmieri\doc\etape\_2\autres\

Ainsi vous pourrez voir les images en grand format.