Projektdokumentation

Titel-Editor in Java

Johann Houszka, 0625523

Technische Universität Wien

[johann\_houszka@gmx.at](mailto:johann_houszka@gmx.at)

SS 2012

Matthias Eigner, 0625203

Technische Universität Wien

[matthias.eigner@student.tuwien.ac.at](mailto:matthias.eigner@student.tuwien.ac.at)

SS 2012

**Betreuung:**

Dr. Horst Eidenberger

Associate Professor

[eidenberger@tuwien.ac.at](mailto:eidenberger@tuwien.ac.at)

Inhaltsverzeichnis

1 Projektbeschreibung 3

[2 Requirements 4](#_Toc358316993)

[3 Arbeitsstruktur 5](#_Toc358316994)

[4 Ergebnisse 6](#_Toc358316995)

[4.1 Hauptfenster 6](#_Toc358316996)

[4.2 Effekte 7](#_Toc358316997)

[4.3 Titel 7](#_Toc358316998)

[4.4 Bilder 8](#_Toc358316999)

[4.5 Untertitel 8](#_Toc358317000)

[4.6 Ausgabe 9](#_Toc358317001)

[5 Probleme 9](#_Toc358317002)

[6 Stundenliste 10](#_Toc358317003)

1. **Projektbeschreibung**

Das Projekt wurde für die Lehrveranstaltung „From Design to Software 1“ abgewickelt.

Aufgabe war es, einen OpenSource Titel-Editor zu entwickeln, der folgende Vorgaben erfüllt:

* Laden und Anzeigen von Videos
* Laden und Anzeigen von vorhandenen Untertiteln
* Erstellen von neuen Untertiteln
* Erstellen von Titeln mit folgenden Parametern:
  + Schriftart
  + Farbe
  + Schriftgröße
  + Position
  + Anzeigedauer
* Anzeigen von Grafiken im Video mit folgenden Parametern:
  + Größe (Prozentuell)
  + Position
  + Anzeigedauer
* Auswahl und Anwendung von Videoeffekten auf das Video
* Speichern des bearbeiteten Videos auf 2 Arten
  + Speichern eines separaten Untertitelfiles
  + Rendering der Untertitel auf das Video
  + wenn möglich, verschiedene Videoformate (Container)

# Requirements

Es gibt 2 wichtige Requirements für diese Applikation:

* VLC-Player
* FFMPEG

Der VLC-Player wird deswegen benötigt, da die Applikation auf vlcj - das Java Framework für den VLC-Player baut.

FFMPEG muss deswegen installiert werden, da dies zum Konvertieren der Videos und Speichern der Subtitles benötigt wird.

Beide Tools sind OpenSource.

Zu dieser Applikation wird eine ini-Datei bereitgestellt, in der die Pfade zu den benötigten Libraries stehen.

Darin gibt es zwei Werte:

* **vlc:** gibt den Pfad zu den VLC-dlls (libvlc.dll, libvlccore.dll) an. Diese finden sich im Installationsordner des VLC-Players.
* **ffmpeg:** gibt den Pfad zur ffmpeg-Installation an. Bei der PC-Version muss dabei auf die "ffmpeg.exe" Datei verwiesen werden, bei der OSX-Version reicht ein Verweis auf "ffmpeg".

Nun folgt eine genauere Beschreibung der benötigten Libraries. Die Installation dieser Libraries ist vom Betriebssystem abhängig.

**OSX:**

Die Libraries FFMPEG und LAME müssen laut folgender Anleitung installiert werden (<http://jungels.net/articles/ffmpeg-howto.html>). Außerdem müssen XCode bzw. die XCode Command Line Tools installiert sein, da es sonst bei den Installationen von FFMPEG und LAME zu Problemen kommen kann.

Mögliche Fehlermeldungen während der Installation:

1. gcc is unable to create an executable file.

Lösungen (laut obiger Nummerierung):

1. Installieren der XCode Command Line Tools

Um Subtitles ins Video zu Rendern müssen folgende Libraries installiert werden:

* XCode downloaden
* MacPorts downloaden und installieren <http://www.macports.org/install.php>
* Pkg-config: Dieses Library wird benötigt, um libass erfolgreich zu installieren. Zur Installation muss folgender Anleitung gefolgt werden. <http://www.linuxfromscratch.org/blfs/view/svn/general/pkgconfig.html>
* fribidi: Dependency für Libass
* freetype2: Dependency für Libass
* fontconfig: Dependency für Libass
* Libass: Dieses Libary wird benötigt, um Untertitel direkt ins Video zu Rendern.

**ACHTUNG:** Damit unter Windows Subtitles in das Video gespielt werden können dürfen nur \*.ass Dateien verwendet werden.

Eine genaue Schritt-für-Schritt Anleitung für die Installation unter OSX bzw. Windows findet sich im Dokument "Installationsanleitung.docx".

**WINDOWS:**

Die Grundlagen für die Installation sind dieselben wie oben beschrieben, jedoch auf Windows ausgelegt. Für eine korrekte Installation ist folgender Installationsanleitung zu folgen (<http://www.wikihow.com/Install-FFmpeg-on-Windows>).

Falls dieser Weg nicht wie gewünscht funktioniert oder diese Installation nicht alle Funktionen enthält (z.B.: libass) wird hier ein weiterer Weg vorgestellt. Dieser wurde getestet und funktioniert.

Folgende Libraries bzw. Programme werden benötigt.

* MingW: Für die weitere Installation der Libraries wird die hier zur Verfügung gestellte MingW-Shell benötigt.
* pkg-config: Dependency für fontconfig
* fribidi: Dependency für libass
* freetype2: Dependency für fontconfig
* libxml2: Dependency für fontconfig
* fontconfig: Dependency für libass
* libass: wird benötigt, um Subtitles ins Video zu Rendern
* FFmpeg

**ACHTUNG:** Damit unter Windows Subtitles in das Video gespielt werden können dürfen nur \*.srt Dateien verwendet werden.

Eine genaue Schritt-für-Schritt Anleitung für die Installation unter OSX bzw. Windows findet sich im Dokument "Installationsanleitung.docx".

# Arbeitsstruktur

Die Arbeit lässt sich in verschiedene Schritte unterteilen.

Zuerst erfolgte die Literaturrecherche, bei welcher wir uns über die verschiedenen Frameworks, die für unser Projekt in Frage kommen, informiert haben.

Als nächster Schritt folgte das Interface Design. Dieses beinhaltete Brainstorming, Skizzieren des Interfaces und der einzelnen Komponenten, Generierung von Mockups (siehe Abbildung 1) und schlussendlich Layout-spezifische Designs, um die Größe der einzelnen Komponenten zu bestimmen.

Danach begann der Implementierungsprozess. Dieser wurde auf die 2 Entwickler aufgeteilt. Dabei wurden immer Implementierungsziele für eine Woche gesetzt, die danach umgesetzt wurden. Kam es zu Problemen, wurden diese bei Treffen oder via Skype besprochen.

Neben dem Implementieren wurden die einzelnen Komponenten getestet, um Probleme so früh wie möglich zu erkennen und zu vermeiden.

Nach dem Abschluss der Implementierung und des Testvorgangs, wurde der Projektbericht erstellt, in dem alle Schritte des Projekts festgehalten sind.

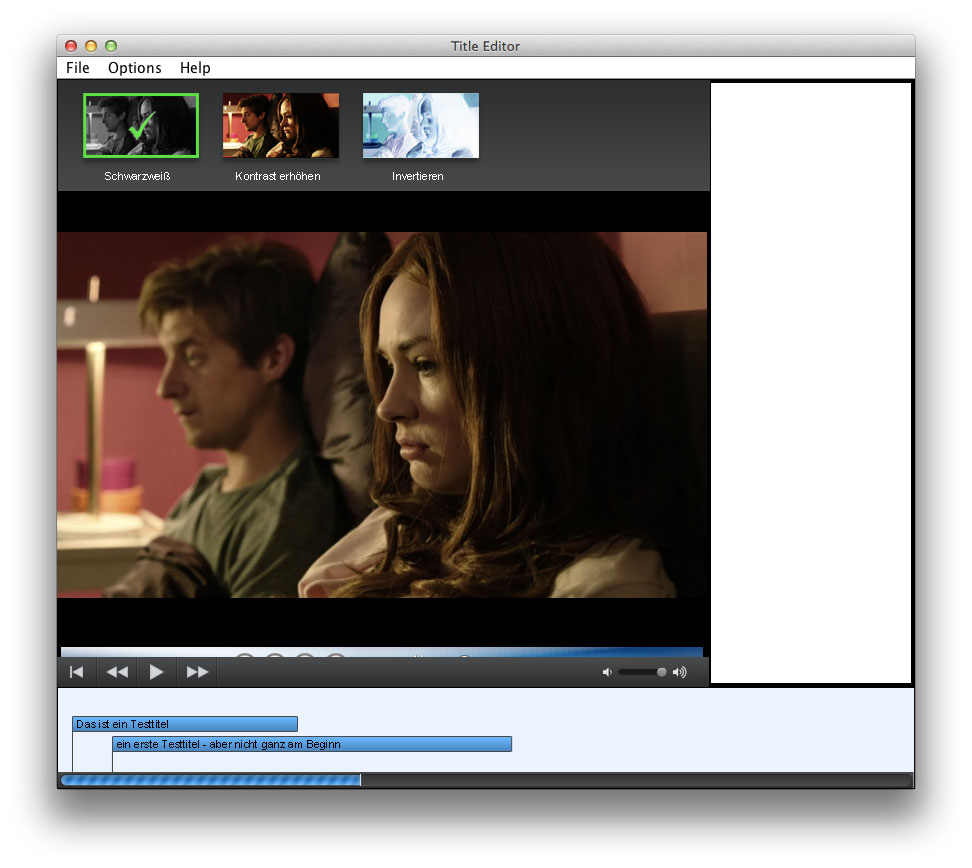


Abbildung 1: Fortgeschrittenes Mockup

# Ergebnisse

## 4.1 Hauptfenster

Das Hauptfenster besteht aus 4 Bereichen wie in Abbildung 1 ersichtlich.

Unten befindet sich eine Leiste die Position im Video anzeigt sowie Positionen von Untertiteln grafisch darstellt. In der Mitte befindet sich das Wiedergabefenster des Videos mit üblichen Kontrollelementen wie „Pause“/“Play“, „Vor- und Zurückspulen“ und „zum Anfang springen“. Rechts befinden sich die Einstellungsfenster in denen man Titel, Untertitel und Grafiken bearbeiten kann.

Über dem Wiedergabefenster befindet sich die Effektleiste. Hier sind alle Effekte über Vorschaubilder auswählbar. Der ausgewählte Effekt ist nach Auswahl sofort im angezeigten Video ersichtlich.

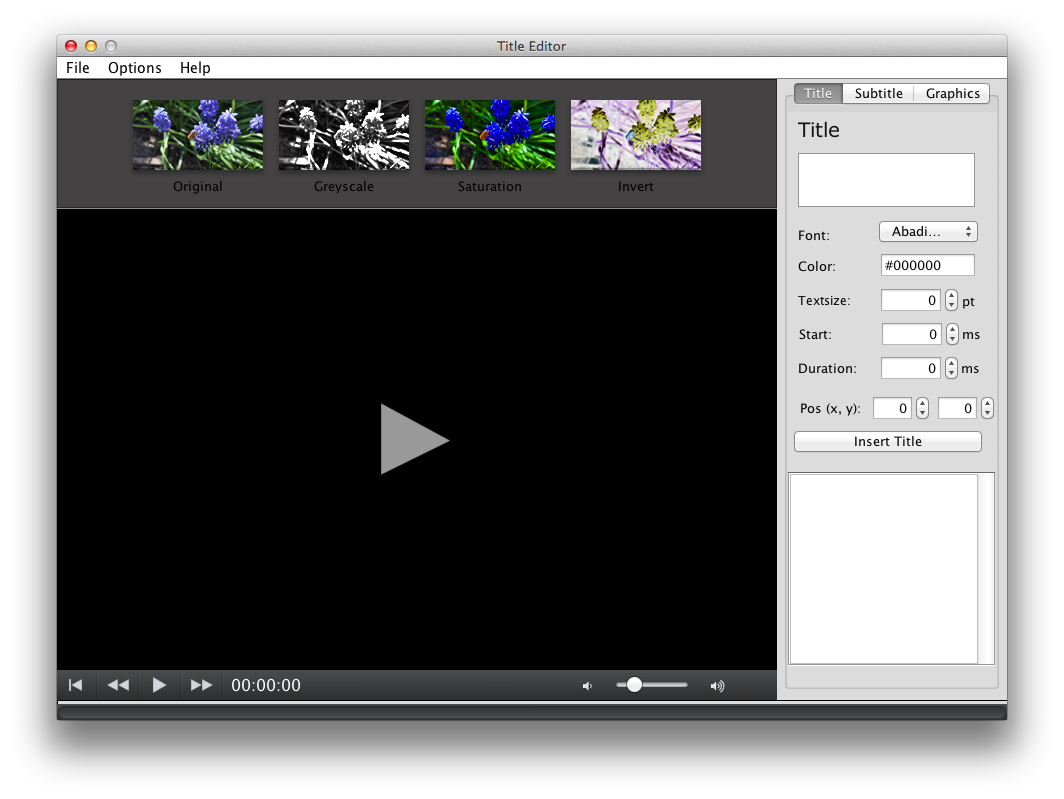


Abbildung 2: Hauptfenster

## 4.2 Effekte

Die Effekte wurden mit Hinblick auf Erweiterbarkeit und Offenheit entwickelt. Es wurde ein Interface mit den notwendigen Methoden erstellt, um einzelne Effekte zu implementieren. Danach kann man die einzelnen Frames in Pixelebene bearbeiten. Zusätzlich kann beim Starten des Effekts optional ein Einstellungspanel geöffnet werden. Dieses ist standardmäßig beim Effekt „Saturation“ aktiviert.

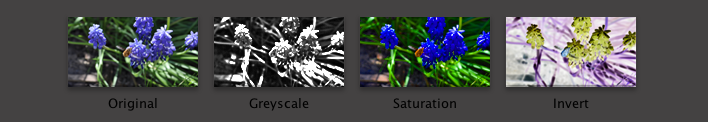


Abbildung 3: EffektPanel

## 4.3 Titel

Das Speichern von Titeln und Bildern folgt grundlegend der selben Art. So gibt es hier eine Titel-Klasse, in der alle notwendigen Methoden zum Speichern und Auslesen von Titeln implementiert sind. Diese sind großteils Getter/Setter-Methoden zum Speichern und Auslesen von spezifischen Daten wie dem Text, der Schriftart, der Schriftfarbe und anderen.

Abbildung 4 bietet einen gute Überblick über das Titel-UI.

## 4.4 Bilder

Das Speichern von Titeln und Bildern folgt grundlegend der selben Art. So gibt es hier eine Image-Klasse, in der alle notwendigen Methoden für das Speichern und Auslesen von Bildern implementiert sind. Diese sind großteils Getter/Setter-Methoden zum Speichern und Auslesen von spezifischen Daten wie dem Bildpfad, der Größe, der Position und der Anzeigedauer.

Abbildung 6 bietet einen gute Überblick über das Graphik-UI.

## 4.5 Untertitel

Das Speichern von Untertiteln erfolgt ebenfalls in einer speziellen Klasse. Für das Speichern werden die Daten in ein Textfile (Typ \*.srt) geschrieben und fürs Anzeigen aus einem Textfile (Typ \*.srt) ausgelesen.

Abbildung 5 bietet einen gute Überblick über das Untertitel-UI.

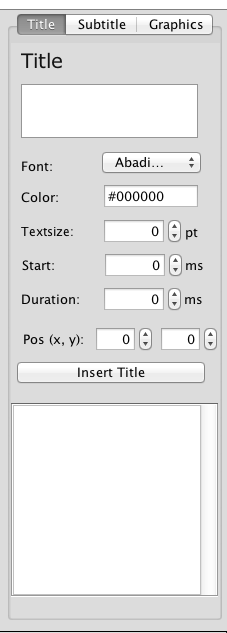
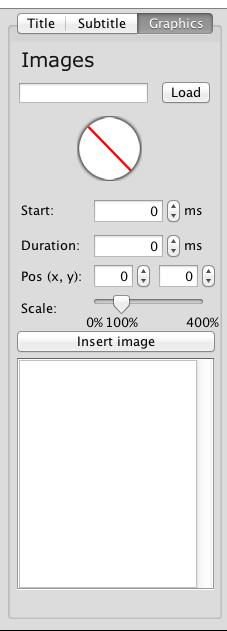
  

Abbildung 4: TitelPanel Abbildung 5: SubtitlePanel Abbildung 6: GraphicPanel

## 4.6 Ausgabe

Die Ausgabe des Videos erfolgt über den Menüpunkt „Start“ -> „Save“. Hier können einige Speicherparameter angegeben werden. Das Speichern wurde durch eine Kombination mehrere Frameworks realisiert. Xuggler ist für das Einfügen der Titel und Graphiken in das Video zuständig. FFMPEG kümmert sich um das Einfüge der Untertitel und das Konvertieren des ursprünglichen \*.mp4-Videos auf andere Formate (zurzeit nur \*.mkv). Einen genaueren Überblick über das UI des Speicherdialogs bietet Abbildung 7

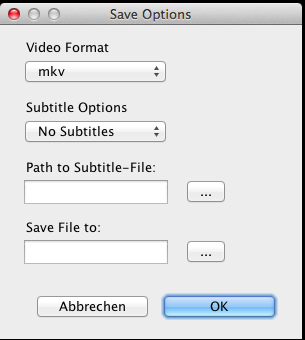


Abbildung 7: Speicherdialog

# Probleme

Während des Projekts gab es auch einige Probleme. So war die Suche nach dem geeigneten Framework sehr aufwendig.

Zuerst haben wir uns für ***Xuggler*** (<http://www.xuggle.com/xuggler/>)entschieden.

Aufbauend auf diesem Framework haben wir begonnen, das Interface zu designen und darauf aufbauend, den Player zur Wiedergabe von Videos zu implementieren. Dies war jedoch schwieriger als erwartet. Es kam immer wieder zu Problemen mit der Synchronisierung zwischen Audio und Video. Xuggler war somit für unsere Anforderungen nicht das richtige Framework, um zumindest das ganze Projekt darauf aufzubauen.

Nach weiteren Recherchen sind wir auf ***VLCJ*** gestoßen (<http://www.capricasoftware.co.uk/vlcj/>). Diese OpenSource-Lösung bindet den VLC-MediaPlayer in Java ein. VLCJ erlaubt im Gegensatz zu Xuggler eine einfachere Bedienung und Implementierung vieler VLC Funktionen.

Nach dem Umstieg auf das neue Framework haben wir erneut mit dem Implementierungsprozess begonnen und haben schon nach einigen Stunden gemerkt, dass wir mit diesem Framework viel besser zurechtkommen. Außerdem besitzt es alle Eigenschaften, die wir für unser Projekt benötigen.

In einer vorangehenden Analyse haben wir trotzdem einige Bereiche spezifiziert, bei denen möglicherweise mit Problemen zu rechnen ist. Diese sind das Editieren und Neu laden von bereits bestehenden Untertitelfiles und das Speichern des Videos mit darüber gerenderten Untertiteln.

Diese Analyse baut auf der Tatsache auf, dass aus der Dokumentation VLCJ-Dokumentation keine klaren Erkenntnisse über diese Punkte gewonnen werden konnten.

Das größte Problem dieses Projekts war das Speichern des Videos samt den Titeln, Untertiteln und Grafiken. VLCJ erlaubt zwar das simple Speichern von Videos – jedoch gibt es keine Möglichkeit, Titel oder Effekte ins Video zu speichern. Dies hat folgende Gründe:

* VLCJ zeigt Effekte und Titel nur „onDisplay“ an – die Effekte werden also nur über dem Original-Video angezeigt
* Für das Rendern der Titel und Grafiken in das Video benötigt man Images (Frames) um diese Frame für Frame zu einem Video zusammenzufügen – VLCJ erlaubt nur Zugriff auf das data-Array vom Typ int.

Nach eingehender Recherche wurde zum Speichern das zuvor verworfene Framework ***Xuggler*** (<http://www.xuggle.com/xuggler/>)verwendet.

Xuggler ist in der Applikation nun für folgende Schritte zuständig:

* Speichern des Videos
* Rendern der Titel, Grafiken und Effekte in das Video

Laut derzeitiger Implementierung sind folgende Container zulässig:

* mkv
* mp4

Das Einfügen der Subtitles in den Videocontainer stellte das nächste Problem dar. VLCJ kann zwar Subtitles erkennen und wiedergeben, speichern jedoch nicht. Auch Xuggler stellt keine Funktionalität in diese Richtung zur Verfügung.

Nach weiterführender Recherche übrnimmt das OpenSource-Projekt ***FFMPEG*** (<http://www.ffmpeg.org/>) diese Aufgabe.

Laut Anforderung sind folgende Speicherarten zulässig:

* Rendern der Subtitles in das Video
* Speichern der Subtitles in ein externes File

Ein nächstes Problem ist die mäßige Videoqualität bei durch Xuggler konvertierte Videos. Da dieses das Ausgangsvideo Frame-für-Frame wiederherstellt, kommt es zu einem Qualitätsverlust. Für diese Tatsache konnte von den Entwicklern keine Lösung gefunden werden.

# Stundenliste

Kürzel:

T (Tests), E (Entwicklung), B (Besprechung/Recherche), D (Dokumentation)

*Stundenliste – Johann Houszka*

|  |  |  |  |
| --- | --- | --- | --- |
| **Datum** | **Zeit(Std)** | **Beschreibung (Hauptaufgabe)** | **Kürzel** |
| 09.10.2012 | 5 | Ordnerstruktur erstellt, Libraries hinzugefügt, Recherche | E/ B |
| 10.10.2012 | 2 | Images(Buttons, Background) hinzugefügt | E |
| 11.10.2012 | 7 | GUI erstellt, GUI Listeners | E |
| 15.10.2012 | 7 | GUI Erweiterungen | E |
| 16.10.2012 | 3 | Background Tests | E |
| 17.10.2012 | 7,5 | GUI Design, Layout Fix, Separators, ControPanel erweitert | E |
| 18.10.2012 | 4,5 | ControlPanel Design, Zeitanzeige, ReloadPanel hinzugefügt | E |
| 19.10.2012 | 4 | EffectPanel erstellt, Effect-Klasse, Thumbnails, erste Effekt-Prototypes | E |
| 20.10.2012 | 2 | Effekt Hue hinzugefügt, Effekt selektieren möglich | E |
| 22.10.2012 | 2 | Weitere Effekte hinzugefügt | E |
| 23.10.2012 | 4 | Automatische Thumbnail-Generierung bei Laden des Videos, SettingsPanel hinzgefügt | E |
| 08.11.2012 | 4 | Funktion Vor-/Zurückspulen implementiert, SettingsPanel erweitert, Start Video-Speicherung | E |
| 12.11.2012 | 1 | Erste Umsetzung der Titelanzeige (via marquee) | E |
| 13.11.2012 | 2,5 | SubtitleFile laden, Lesen des SubtitleFiles | E |
| 14.11.2012 | 4 | Subtitle Änderungen, Zeitberechnung für Subtitles | E |
| 15.11.2012 | 3,5 | Dokumentation, Thumbnail Generierung neu | E/ D |
| 16.11.2012 | 1,5 | Thumbnail Generierung neu | E |
| 21.11.2012 | 3,5 | Dokumentation Code & Bericht | D |
| 29.11.2012 | 3,5 | SettingsPanel TabbedPane, SubtitleList | E |
| 30.11.2012 | 5,5 | GraphicsPanel hinzugefügt, ImageScaler Klasse erstellt, Speichern der Graphiken implementiert, Vorschau der Graphiken in Panel | E |
| 5.12.2012 | 1 | Libraries hinzugefügt, Dokumentation | E/ D |
| 6.12.2012 | 4 | ID für Graphiken, unnötigen Code entfernt, GraphicsPanel erweitert, Graphik in Video einfügen, Opacity Slider & Scale Slider GraphicsPanel | E |
| 7.12.2012 | 1,5 | Positionierung der Graphiken, Scale Fix | E |
| 10.12.2012 | 5 | Liste für Graphik Speicherung, Fehlerbehandlung Code | E/ T |
| 10.12.2012 | 4 | Major Changes: EmbeddedMediaPlayer change to DirectMediaPlayer (only audio, no video), EmbeddedMediaPlayer change to DirectMediaPlayer (audio & video, not resizable) | E |
| 11.12.2012 | 3,5 | Einigung auf eine fixe Größe des Panels, SpinnerModel GraphicsPanel, Effekt Black/White implementiert | B/ E |
| 12.12.2012 | 2 | SaveDialog erstellt | E |
| 14.12.2012 | 1 | SaveDialog erweitert (als PopUp) | E |
| 19.12.2012 | 1,5 | SaveDialog erweitert, Anpassungen des MainPanels | E |
| 16.01.2013 | 4,5 | Anpassungen UIListener, SaveDialog | E |
| 17.01.2013 | 1 | Task: mehr als ein Video laden (in Arbeit) | E |
| 18.01.2013 | 2,5 | Mehrere Videos hintereinander laden nun möglich, EffectsPanel Scrollbar Fix | E |
| 21.01.2013 | 2,5 | Effect Thumbnails nach gemeinsamer Besprechung nun hardcoded, Probleme bei Auswahl d. richtigen Frames | E/ B |
| 23.01.2013 | 2 | Unbenutzte Codestellen entfernt, Dokumentation | D |
| 25.01.2013 | 3,5 | SaveDialog erweitert, Menüpunkt „Create new Subtitles“, Bugfix | E/ T |
| 06.03.2013 | 2 | UIListener Update | E |
| 13.03.2013 | 2,5 | Subtitles Update, AboutPanel hinzugefügt | E |
| 29.03.2013 | 1 | AboutDialog hinzugefügt, PlayerSettingsPanel Update | E |
| 01.04.2013 | 3 | GraphicPanel Exception Handling bei Graphik Update, Graphiken löschen implementiert, JavaDoc erstellt | T/ E |
| 05.04.2013 | 2 | GraphicPanel neues Feature, JavaDoc, Code Formatierungen | E/ D |
| 10.04.2013 | 2,5 | Besprechung bzgl. Speichern von Videos, Xuggler Libraries Import, neue Klasse SaveVideo, | B/ E |
| 26.04.2013 |  | Testvideo hinzugefügt, Subtitles in Container speichern implementiert | E |
| 27.04.2013 | 6 | Besprechung Framework, ffmpeg Libraries hinzugefügt, ffmpeg auf System installiert | B/ E/ T |
| 01.05.2013 | 3 | SaveDialog erweitert, Projektbericht angepasst | E/ D |
| 03.05.2013 | 1 | SaveDialog Bugfixing | T/ E |
| 04.05.2013 | 3 | StatusMonitor hinzugefügt, Projektbericht erweitert | E/ D |
| 05.05.2013 | 2 | StatusMonitor Bugfixing, | T/ E |
| 09.05.2013 | 1,5 | Installationsanleitung der Libraries erstellt | D |
| 12.05.2013 | 1,5 | Code an optimale Library-Nutzung angepasst | E |
| 20.05.2013 | 5,5 | Applikationstest WINDOWS, Anpassungen der GUI an Windows-OS | T/E |
| 01.06.2013 | 3 | ProgressBar statt ProgressMonitor, GUI Updates | E |
| 02.06.2013 | 3 | ProgressBar Cancel-Button hinzugefügt, neue Klasse um Betriebssystem zu bestimmten - beim Speichern benötigt | E |
| 05.06.2013 | 5,5 | Windows - Subtitles rendern funktioniert, CheckBox beim Speicher-Dialog -> schnelleres Speichern | E |
| 06.06.2013 | 3 | Projektbericht finalisiert, Abgabepaket erstellt | D |
| 07.06.2013 | 1 | Zwischenabgabe1 "TitelEditor" | B |
| 07.06.2013 | 4 | Bugs behoben | E/T |
| 08.06.2013 | 3 | Änderungswünsche umgesetzt | E/T |
| 12.06.2013 | 3 | Änderungswünsche umgesetzt, Zwischenabgabe2 "TitelEditor" | E/T |
| 14.06.2013 | 3 | Änderungswünsche umgesetzt, Projektbericht finalisiert, Endabgabe "TitelEditor" | E/T/D |

*Stundenliste – Matthias Eigner*

|  |  |  |  |
| --- | --- | --- | --- |
| **Datum** | **Zeit** | **Beschreibung** | **Kürzel** |
| 09.10.2012 | 8 | Recherche, Startbesprechung, Entwicklungsumgebung | B, E |
| 10.10.2012 | 4 | Neue Folderstruktur, Program Grundgerüst | E |
| 11.10.2012 | 7 | Recherche, Frameworks, Grundgerüst | E |
| 12.10.2012 | 6 | Frameworks | E |
| 13.10.2012 | 4 | Frameworks komplett neu | E |
| 16.10.2012 | 2 | Timeline Grundgerüst und interface | E |
| 17.10.2012 | 5 | Neue Images, Timeline interface | E |
| 18.10.2012 | 4,5 | Effekte Grundgerüst, Timeline und Effekte entwicklung | E |
| 20.10.2012 | 4 | Effekte | E |
| 23.10.2012 | 3.5 | Setting Grundgerüst, Settings panel, Timeline | E |
| 15.11.2012 | 4 | Dokumentation | D |
| 21.11.2012 | 3,5 | Subtitle Grundgerüst | E |
| 30.11.2012 | 4 | Subtitle gui und funktionalität | E |
| 01.12.2012 | 2,5 | Subtitles funktionalität | E / T |
| 05.12.2012 | 2 | Neue Timeline, Program struktur update | E |
| 06.12.2012 | 3 | Imagespanel gui | E |
| 07.12.2012 | 3 | Imagespanel funktionalität, gui | E |
| 09.12.2012 | 3 | Entwicklung Imagespanel | E |
| 11.12.2012 | 3,5 | Imagepanel update – direktrendering, Besprechung | E / B |
| 13.12.2012 | 4 | Entwicklung verbesserungen, Testing | E / T |
| 16.12.2012 | 5 | Effecte, UI | E |
| 05.01.2013 | 2,5 | Effekte | E |
| 16.01.2013 | 1 | Effekte – direktrendering | E / T |
| 17.01.2013 | 5 | Testing | T |
| 21.01.2013 | 4,5 | Sättigungseffekt, Besprechung, Effekte, Architektur | E / T / B |
| 23.01.2013 | 3 | Effekte gui, funktionalität | E |
| 25.01.2013 | 3 | Imagepanel , Titels gui, funktionalität | E |
| 13.03.2013 | 3,5 | Title gui, funktionalität | E |
| 29.03.2013 | 2 | Title update | E |
| 05.04.2013 | 3,5 | Fixes | T / E |
| 10.04.2013 | 2,5 | Testing, small changes, besprechung saving | T / E / B |
| 17.04.2013 | 3 | Speichern von Images and Titles, Testing | E / T |
| 26.04.2013 | 3 | Testing, fixed timing bugs | T / E |
| 27.04.2013 | 6 | ffmpeg besprechung, installation | T / B |
| 29.04.2013 | 3,5 | Ffmpeg + upload | T |
| 01.05.2013 | 1 | Saving with Effects | E |
| 05.05.2013 | 4 | Testing | E |
| 20.05.2013 | 2,5 | Ini file with paths | E |
| 24.05.2013 | 4 | Testing, fixed bugs | T |
| 25.05.2013 | 4 | Testing | T |
| 27.05.2013 | 5 | Testing | T |
| 01.06.2013 | 4 | Testing, fixed bugs, added new gui-images, saving gui update | T |
| 02.06.2013 | 4 | Testing, Bugs beheben | T |
| 05.06.2013 | 3,5 | Testing, Bugs beheben | T |
| 06.06.2013 | 3 | Abschlussbesprechung, Dokumentation, Abgabepaket | B / D |
| 07.06.2013 | 1 | Zwischenabgabe1 "TitelEditor" | B |
| 07.06.2013 | 4 | Bugs behoben | E/T |
| 08.06.2013 | 3 | Änderungswünsche umgesetzt | E/T |
| 12.06.2013 | 3 | Änderungswünsche umgesetzt, Zwischenabgabe2 "TitelEditor" | E/T |
| 13.06.2013 | 4 | Änderungswünsche umgesetzt | E/T/D |