

PSE 2012

OQAT

Objective Quality Assessment Toolkit

Praxis der Softwareentwicklung

WS 2012

Implementierung und Testbericht



Auftraggeber

Karlsruher Institut für Technologie

Institut für Technische Informatik

CES - Chair for Embedded Systems

Prof.Dr.J.Henkel

Betreuer: S. Kobbe

Auftragnehmer

Name	E-Mail-Adresse
Eckhart Artur	artur.eckhart@gmail.com
Ermantraut Georg	georg.ermantraut@gmail.com
Leidig Sebastian	sebastian.leidig@gmail.com
Monev Alexander	bcclan@mail.bg
Sailer Johannes	johsailer@gmail.com

Karlsruhe, 5.9.2012

Inhaltsverzeichnis

Abbildungsverzeichnis

1 Einleitung

Das vorliegende Dokument bildet den Abschluss der Implementierungs- und Testphase des PSE-Projekts »OQAT « und soll verdeutlichen wie diese abgelaufen ist.

Die Verteilung der Aufgaben und des Aufwandes wurde mit Hilfe von Gantt-Diagrammen geplant. Während der Implementierung sind mehrere kleine Schwachstellen und Unklarheiten im Entwurf gefunden worden und die Architektur wurde entsprechend angepasst. Die Änderungen werden im Abschnitt »Änderungen am Entwurf« aufgeführt. Das fertig implementierten OQAT wurde intensiven Tests unterzogen. Dazu gehören automatisierte Unittests sowie manuelle Tests.

2 Planung und Aufwand

Nach Fertigstellung des Architektur-Entwurfs wurde zunächst der Aufwand für die verschiedenen Implementierungsschritte geschätzt und eine sinnvolle Reihenfolge für die Implementierung der verschiedenen Komponenten festgelegt. Die hier dargestellten Gantt-Diagramme geben einen groben Überblick des geplanten und tatsächlichen Ablaufs. Außerdem wird eine Gegenüberstellung des erwarteten und tatsächlichen Implementierungs-Aufwands in Stunden gegeben.

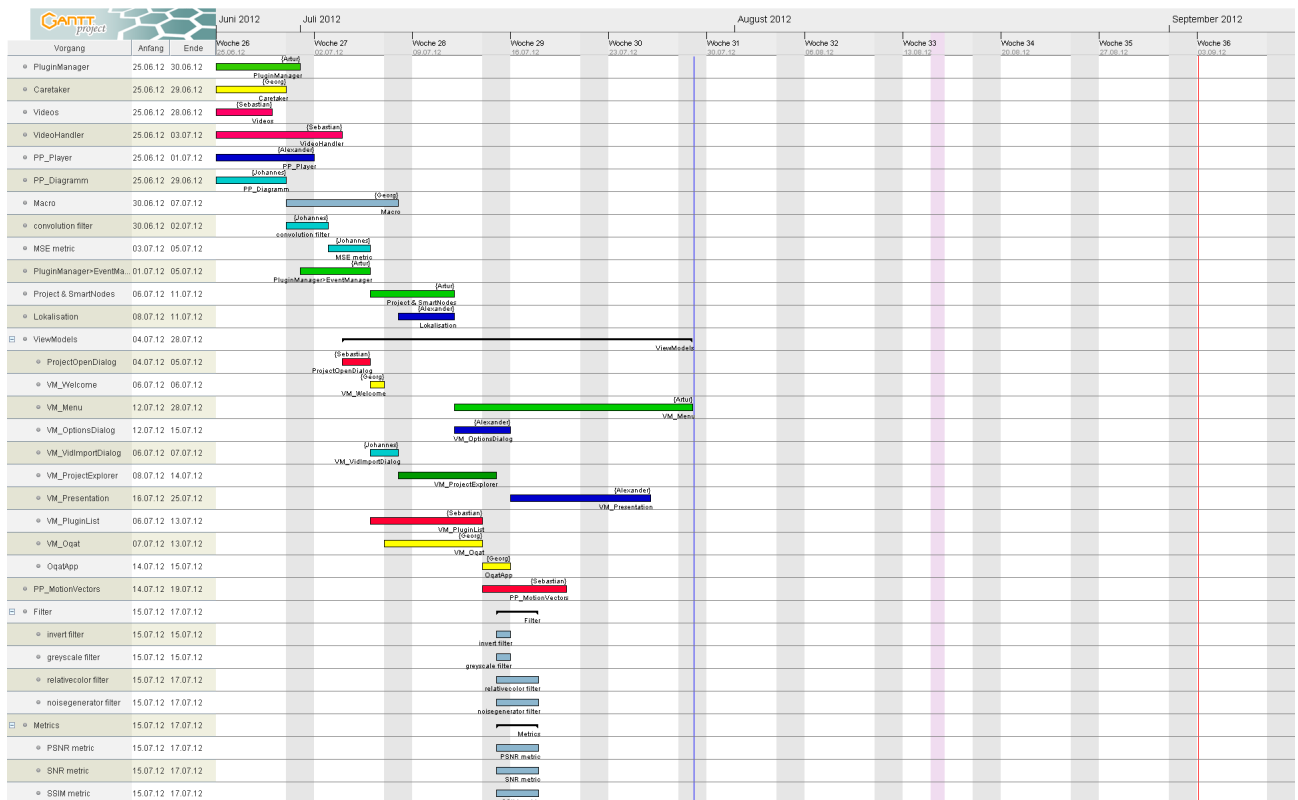


Abbildung 2.1: geplanter Implementierungsablauf

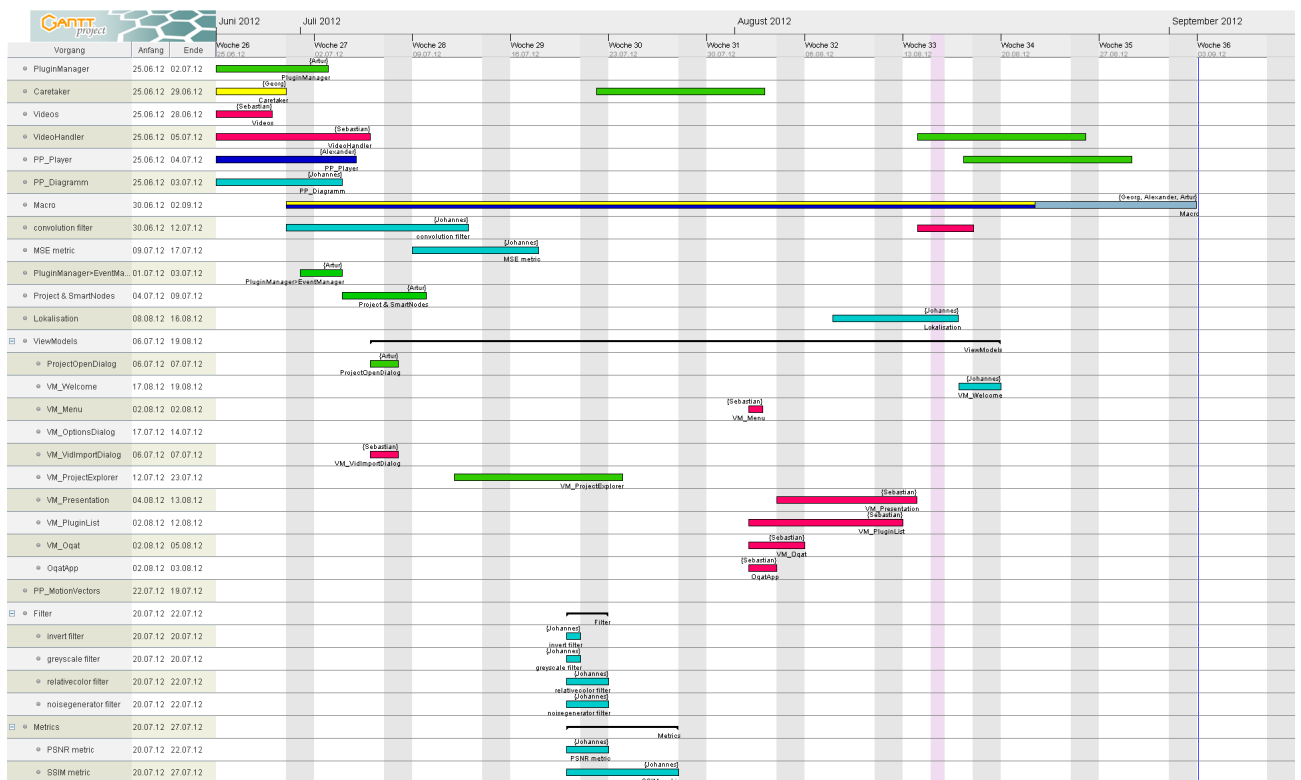


Abbildung 2.2: ungefährender tatsächlicher Implementierungsablauf

	<i>geplant</i>	<i>tatsächlich</i>
PluginManager	22	48
Caretaker	10	40
YuvVideoHandler	18	64
Video	4	3
Player	16	45
Diagram	10	24
Macro	24	195
Project & SmartNodes	12	24
Localisation	8	16
Menu	8	3
WelcomeView	3	12
ProjectOpenDialog	4	14
VidImportDialog	4	10
ProjectExplorer	14	26
PluginList	15	6
Presentation	20	17
Oqat	18	4
Convolution	5	17
Invert	2	3
Greyscale	2	1
RelativeColor	6	6
NoiseGenerator	6	4
PSNR	6	6
MSE	4	12
SSIM	6	22
MotionVectors	12	0
Gesamt	259	622

Abbildung 2.3: Vergleich des erwarteten und tatsächlich Aufwands in Stunden

3 Änderungen am Entwurf

- Viele Klassen implementieren nun ISerializable, damit Mementos abgespeichert werden können.
- Smartnode
Es wurden die Attribute name, id und fatherId hinzugefügt. Außerdem wurde der Konstruktor geändert, da es nun einfacher ist Kinder hinzuzufügen. Er nimmt nun ein Video, id und optional einen Smarttree als Argumente.
- Caretaker
Die Struktur wurde kompakter, für größere Dynamik und weniger organisatorischen Aufwand.
- Project
Hinzugefügt wurde: Attribut unusedId um ungenutzte Werte zu finden, saveProject Methode und Methoden zum Hinzufügen und Löschen von Smartnodes. Gelöscht wurde die Memento Funktionalität, sowie die Konstruktoren, welche eine Videoliste bzw. nur einen Pfad als Argument benötigten.
- Video
Die Mementofunktionalität wurde entfernt. Hinzugefügt wurde eine Liste der Plugins, die das Video bearbeitet haben, sowie die getExtraHandler Methode.
- VideoSource
Wurde entfernt, da nicht benötigt.
- OqatSerializationBinder
Wurde Hinzugefügt.
- class MementoListMemento
Wurde hinzugefügt um alle Mementos eines Objektes in einem einzigen File zu speichern.
- PluginManager:
Der Typ der PluginTable wurde auf Enumerable geändert, da es praktischer ist. getPluginmanager, wurde in den getter des Pluginmanagers verschoben, da es besser aussieht und weniger Code ist. Die Methode loadPluginTable wurde gelöscht, da ihre Funktion automatisch abläuft. Außerdem wurde die HandlerTable gelöscht.

- Macro
Wurde neu designed.
- Oqat
Event onNewProjectCreated wurde hinzugefügt und Initialisierungsmethoden wurden gelöscht, da nicht verwendet.
- Pluginlist
Es wurde CurrentViewType und das Pluginmanger Attribut entfernt. Der Konstruktor nimmt nun den PluginTyp als Argument.
- ProjectExplorer
Pluginmanger Attribut wurde gelöscht und importVideos Methode hinzugefügt, um durch den Video import Dialog zu führen.
- Welcome
Konstruktor nimmt kein Panel mehr an.
- VidImportOptionsDialog
Nimmt nun eine Liste von Videos an um mehrere Videos zu importieren.
- WindowsErrorConsole
implementiert zum Debugging.
- WindowOqatInfo
Implementiert um allgemeine Informationen über OQAT anzuzeigen.
- Sandbox
Klasse lädt temporär ein Plugin für Debugging-Zwecke.
- EntryEventArgs
Implementiert um String in EventArgs zu verpacken.
- PluginType
Einträge umbenannt zum jeweiligen Interface Namen, da es praktischer ist um an das jeweilige Gegenstück zu kommen.
- EventType
Wurde um neue Eventtypen erweitert.
- IVideoInfo
Implementiert nun IClonable und hat drei zusätzlichen Attribute width, height und framecount.
- MetaData
Plugins müssen dieses Interface nun nutzen um richtig geladen zu werden.

- IVideo
Wurde hinzugefügt.
- IPlugin
namePlugin und type Attribute wurden gelöscht und werden nun in IPluginMetadata bereitgestellt. propertiesView Attribut wurde hinzugefügt sowie die createExtraInstacne Methode, falls man ein weiteres Objekt des Typs braucht. Ungenutzte Methoden wurden entfernt.
- IPresentation
Wurde wegen Player und Handler neu designed.
- IMacro
Implementiert nun IPlugin, da sonst der Pluginmanager das Interface nicht erkennen kann.
- Player
Neu designed, auf Grund Performance Problemen.
- YuvVideoHandler
Neu designed, auf Grund Performance Problemen.
- Plugins
Minor changes with setVideo. SSIM wurde nicht realisiert, da der Wertebereich falsch war.

4 Automatisierte Tests

Fast alle Teile des Projekts wurden mit Hilfe von automatisierten Komponententests (unit tests) getestet. Im folgenden eine Übersicht über die Testfälle aller Komponenten, sowie Statistiken über die erreichte Codeabdeckung. Im Zusammenhang mit den beschriebenen Tests werden auch damit entdeckte und behobene Fehler dargestellt.

4.1 PluginManager

Testfälle

- pluginManager singleton getter
- Initialization: no PluginDir
- Initialization: invalid dll
- Initialization: valid dll, no plugin
- Initialization: plugin dll, outdated interface IPlugin
- Initialization: plugin dll, outdated interface IFilterOqat
- Initialization: add plugins
- Initialization: add Plugin dll after loading
- Initialization: add duplicate Plugin
- getPluginNames: ok
- getPluginNames: empty list
- getPlugin: IPlugin + name null
- getPlugin: IPlugin + name empty
- getPlugin: IPlugin + name not existing
- getPlugin: IPlugin + name existing
- getPlugin: detail interface + name existing

- getPlugin: name existing for different type
- addMemento: pluginname null
- addMemento: no plugin
- addMemento: overwrite existing memento
- addMemento: add several memento
- getMementoNames: pluginname null
- getMementoNames: pluginname empty
- getMementoNames: pluginname unknown
- getMementoNames: no Memento for plugin
- getMementoNames: mementos listed
- getMemento: mementoname null
- getMemento: mementoname empty
- getMemento: memento not existing
- getMemento: pluginname null
- getMemento: pluginname empty
- getMemento: pluginname unknown

Codeabdeckung 73% automatisiert

gefundene Probleme

- beschädigte Plugins werden nicht korrekt ausgefiltert
- nach Initialisierung des PluginManagers hinzugefügte Plugins werden nicht nachgeladen

4.2 Caretaker

Testfälle

- caretaker singleton getter
- getMemento: path null
- getMemento: path empty
- getMemento: targetfile not valid memento
- getMemento: targetfile valid memento but no regular fileextension (.mem)
- writeMemento / getMemento: valid write and read
- writeMemento: file already exists
- writeMemento: folder does not exist
- writeMemento: file already exists and is readonly

Codeabdeckung 100% automatisiert

gefundene Probleme

- Filestream wurde bei Ausnahmen nicht immer korrekt geschlossen

4.3 Diagram

Testfälle

- flushing a new Diagram
- Clone
- createExtraPluginInstance
- setVideo: null
- setVideo: null data
- setVideo: empty data
- setVideo: correct data
- setVideo: processedBy null
- setVideo: processedBy empty

- setMemento
- getMemento
- propertyView
- type
- presentationType
- namePlugin
- threadSafe

Codeabdeckung 100% automatisiert

gefundene Probleme keine

4.4 Macro

Testfälle

- createExtraPluginInstance
- setMemento: null
- setMemento: null state
- setMemento: state not valid
- setMemento: state valid
- getMemento: set before get
- namePlugin
- type
- addMacroEntry: child valid macro, parent null
- addMacroEntry: child valid filter, parent null
- addMacroEntry: child filter add
- addMacroEntry: child filter add with index
- addMacroEntry: child added to itself
- addMacroEntry EventHandler

- removeMacroEntry: non existent
- removeMacroEntry
- moveMacroEntry
- clearMacroEntryList

Codeabdeckung 29% automatisiert

gefundene Probleme

- Da der Macro erst später vollständig implementiert war, traten eine Reihe Probleme zu Tage, die gerade noch in der Implementierung steckten.

4.5 Smartnode

Automatische Unittests

- smarttree
- name
- id
- idfather
- Konstruktor: mit Smarttree
- Konstruktor: ohne Smarttree
- (equals)

4.6 Codeabdeckung

82 %

4.7 Project

- Konstruktor
- Smarttree
- description
- addNode: unbekannte father id
- addNode: vid = null
- addNode: id father = -1
- addNode: id father = 0
- rmNode: vid = null
- rmNode: force = false
- rmNode: force = true
- rmNode: Knoten existiert nicht

Codeabdeckung 96 %

4.8 ProjectOpenDialog

Automatische Unittests

- Konstruktor

Automatische UI Tests

- Abbrechen Button
- Projekt erstellen in vorgegebenen Ordner: mit und ohne vorgegebenem Namen
- Projekt erstellen über »Browse« funktion

4.9 ProjectExplorer

Automatische UI Tests

- Videos hinzufügen: als Kind von anderen und als oberstes Element
- Videos Löschen
- Projekt öffnen
- Video auswählen
- Video als Referenz auswählen
- Analyse betrachten

Manuell getestet

- Analyse exportieren
- Falsche .yuf Dateien einlesen
- falsche Dateien einlesen

4.10 Video

Testfälle

- Initialization: constructor
- Initialization: extra resources
- Initialization: frame metric values
- Handler: handler of video constructed with video info
- Handler: handler of video constructed without video info
- Handler: handler of video with invalid path

Codeabdeckung 100% automatisiert

gefundene Probleme keine

4.11 YuvVideoInfo

Testfälle

- Initialization: constructor with path
- Initialization: constructor without path
- Initialization: constructor with invalid path
- Clone: create second instance of a video info
- Frame Count: calculate frame count

Codeabdeckung Unbekannt

gefundene Probleme keine

4.12 WelcomeView

Testfälle

- Initialization: constructor
- Initialization: add projects to list
- getMemento
- setMemento
- GUI Test: create new project with valid path
- GUI Test: create new project with invalid path
- GUI Test: open existing project from file explorer
- GUI Test: select and open project with valid path from last projects
- GUI Test: select and open non-existent project from last projects

Codeabdeckung 80% automatisiert

gefundene Probleme keine

4.13 VideoImportOptionsDialog

Testfälle

- Initialization: constructor with valid path list
- Initialization: constructor with null
- Showing property view of video handler
- Importing video
- Importing invalid video
- Cancelling import
- Closing window

Codeabdeckung 49% automatisiert

gefundene Probleme keine

4.14 YuvVideoHandler

Testfälle

- Initialization: constructor of handler
- Initialization: set read context to valid path and video info
- Initialization: set read context to invalid path and video info
- Initialization: set read context to invalid path with valid video info
- Initialization: set read context to valid path with invalid video info
- Initialization: set import context to valid path
- Initialization: set import context to invalid path
- Initialization: set write context without setting read context
- Initialization: set write context after setting read context
- Initialization: set write context to invalid path and video info
- Initialization: set write context to invalid path with valid video info
- Initialization: set write context to valid path with invalid video info

- Flushing the reader
- Flushing the writer
- getMemento
- setMemento
- getFrame: reading all frames of a video
- getFrame: reading a frame out of video range
- writeFrames: writing a frame array within bounds of frame count
- writeFrames: writing a frame array out of bounds
- writeFrames: writing at position -1
- writeFrames: writing null frames

Codeabdeckung Unbekannt

gefundene Probleme getFrame might return null if a race condition between the loop containing the call to getFrame, and the buffer occurs

4.15 Pluginlist

Manuell getestet

- Plugins erkennen
- Mementos öffnen
- Mementos Speichern
- Mementos löschen
- dem Macro mementos übertragen
- dem Macro Plugins Übertragen
- PluginProperties View zeigen: vorhanden
- PluginProperties View zeigen: nicht vorhanden

4.16 Presentation

Manuell getestet (wurde anhand des alten Macros getestet)

- Player anzeigen
- zwei Player anzeigen
- Diagramm anzeigen
- Ansichtswechsel: Filter zu Metrik und umgekehrt
- Ansichtswechsel: von Metrik und Filter zu Analyse
- Drag-and-Drop auf beide Player
- flush

4.17 Player

Testfälle

- Initialization: constructor
- InitializeComponentTest
- createExtraPluginInstanceTest
- CopyMemoryTest
- OnPropertyChangedTest: random Jump Position Update
- OnPropertyChangedTest: next Frame Position Update
- OnPropertyChangedTest: pos Read Pro Name
- OnPropertyChangedTest: fps Ind Pro Name
- PauseClickTest
- PlayClickTest
- ConnectTest
- flushTest
- getMementoTest
- jumpToFrameTextBoxGotFocusTest
- nextFrameClickTest

- positionSliderDragStartedTest
- positionSliderDragDeltaTest
- setMementoTest
- setVideoTest
- setVideoTest1
- slowDownButtonClickTest
- speedUpButtonClickTest
- fpsIndicatorValueTest
- namePluginTest
- pausePlayTickerTest
- playTickerTimeoutTest
- positionReaderTest
- presentationTypeTest
- propertyViewTest
- typeTest
- getFrameTest
- jumpToFrameClickTest
- playTickerThreadTest

Codeabdeckung 80% automatisiert

gefundene Probleme keine

4.18 Invert

Testfälle

- Initialization: constructor
- getMementoTest: start Memento is set
- getMementoTest: is Memento
- localTest
- processTest
- processTest: does not return same Bitmap
- processTest: empty Bitmap
- setMementoTest: empty
- setMementoTest: not Memento
- namePluginTest
- propertyViewTest
- typeTest

Codeabdeckung 81% automatisiert

gefundene Probleme keine

4.19 Convolution

Testfälle

- Initialization: constructor
- getMementoTest: start Mememento is set
- processTest
- setMementoTest
- matrixTest
- namePluginTest
- propertyViewTest

- typeTest

Codeabdeckung 81% automatisiert

gefundene Probleme

- Matrix muss auf ein quadratisches Rechteck von der Größe [3,25] beschränkt sein

4.20 NoiseGenerator

Testfälle

- Initialization: constructor
- getMementoTest: start Mememento is set
- localTest
- processTest
- setMementoTest
- namePluginTest
- propertyViewTest
- typeTest

Codeabdeckung 69% automatisiert

gefundene Probleme keine

4.21 Greyscale

Testfälle

- Initialization: constructor
- getMementoTest: start Mememento is set
- localTest
- processTest

- processTest: empty Bitmap
- setMementoTest
- namePluginTest
- propertyViewTest
- typeTest

Codeabdeckung 69% automatisiert

gefundene Probleme keine

4.22 RelativeColor

Testfälle

- Initialization: constructor
- getMementoTest: start Mememento is set
- localTest
- processTest
- processTest: empty Bitmap
- setMementoTest
- namePluginTest
- propertyViewTest
- typeTest

Codeabdeckung 73% automatisiert

gefundene Probleme keine

4.23 MSE

Testfälle

- Initialization: constructor
- getMementoTest: start Mememento is set
- localTest
- analyseTest
- analyseTest: empty Bitmap
- setMementoTest
- setMementoTest: Object has to be in range [0, 3]
- setMementoTest: Object is not type int
- namePluginTest
- propertyViewTest
- typeTest

Codeabdeckung 51% automatisiert

gefundene Probleme keine

4.24 PSNR

Testfälle

- Initialization: constructor
- getMementoTest: start Mememento is set
- analyseTest
- setMementoTest
- namePluginTest
- propertyViewTest
- typeTest

Codeabdeckung 80% automatisiert

gefundene Probleme keine

5 System und Integrationstests

5.1 Qualitätsbestimmung

- /Q-10/ Die GUI soll falsche Benutzereingaben weitestgehend vermeiden.
Der Punkt wurde durch Einsatz von Komponenten, welche solche eingaben vermeiden(z.B.Slider) wenn möglich erfüllt.
- /Q-20/ Fehlerhafte Eingaben für Pfad, Frames per second, Filter und Analysemetriken werden vom Programm nicht angenommen und der Benutzer wird gefordert, sie zu korrigieren.
Wurde durch Messageboxen realisiert.
- /Q-30/ Ein Filter- oder Analysevorgang findet nur für gültige YUV-Dateien statt, d.h. wenn eine YUV-Datei nicht gelesen werden kann, stürzt das Programm nicht ab und gibt eine entsprechende Fehlermeldung aus.
Die Dateien werden vor dem hinzufügen auf Gültigkeit getestet.
- /Q-40/Oqat wird Benutzern eine Hilfe, in Form von Tooltips und einer kurzen Einleitung die wichtigsten Programmfunktionen anbieten.
Es wurde eine Dokumentation über die einzelnen Ansichten angefertigt. Auch einige Tooltips wurden eingefügt.

5.2 Testfälle

- /T-10/ Projekt erstellen, speichern und laden
funktioniert
- /T-20/ Projekt bearbeiten, speichern und laden
Videos lassen sich hinzufügen/löschen und verändern.
- /T-30/ Projekt auf anderem Rechner öffnen
Wenn die Pfadressourcen stimmen. Ansonsten gehen die Videos Verloren.
- /T-40/ Filter auswählen ohne Video ausgewählt zu haben - Fehlermeldung
Funktioniert

- /T-50/ Filter auswählen, Vorschau betrachten
Nicht Implementiert
- /T-60/ Filter anwenden, generierte Video-Datei überprüfen
Filter anwenden geht, Überprüfung macht keinen Sinn mehr da das Format und die Gültigkeit durch den Handler Gewährleistet werden.
- /T-70/ Filter-Einstellungen verändern
Funktioniert über Properties View bzw über Memento-Funktionalität.
- /T-80/ Mehrere Filter auf ein Video anwenden, Reihenfolge verändern
Funktioniert per Drag-and-Drop
- /T-90/ Analysemetrik auswählen ohne zwei Videos ausgewählt zu haben - Fehlermeldung
Funktioniert
- /T-100/ Analyse starten ohne Metrik auszuwählen - Fehlermeldung
Funktioniert
- /T-110/ Analyse durchführen, Ergebnisse anzeigen
Funktioniert
- /T-120/ Analyseergebnisse speichern und laden
Funktioniert
- /T-130/ Analyseergebnisse exportieren (CSV)
Funktioniert
- /T-140/ GUI Funktionalität
Funktioniert

6 Testszzenarien

- /TS-10/ Neues Projekt und neues Video zum Projekt hinzufügen.
Manuell getestet, funktioniert.
- /TS-20/ Bestehendes Projekt öffnen, mehrere Filter anwenden und deren Reihenfolge abspeichern.
Manuell getestet, funktioniert.
- /TS-30/ Analyse durchführen.
Manuell getestet, funktioniert aber Beschreibung für Analyse wurde nicht Implementiert.
- /TS-40/ Kompletter Analyse-Ablauf.
Manuell getestet, funktioniert. Nicht Implementierte Beschreibung siehe /TS-30/.

7 Probleme

- Macro
Aufgrund von Änderungen anderer Komponenten wurde der Macro oft umgeschrieben. Daraus folgte eine Unübersichtlichkeit im Code. Deswegen wurde beschlossen diesen von Grund auf neu zu schreiben.
- YuvVideoHandler
Aufgrund schwacher Performance wurde der Handler neugeschrieben.
- Caretaker
Die Serialisierung von Objekten wurde unterschätzt und musste auf andere Klassen erweitert werden.
- Filter und Metriken
Das Binding der GUI Elemente der meisten Filter und Metriken musste umgeschrieben werden, aufgrund von Threadingzugriffsfehlern.
- SSIM
Die Metrik hat bis zuletzt nicht immer Werte im Bereich von 0 bis 1 geliefert und wurde deswegen nicht eingefügt.

8 Zusammenfassung

Soweit wie möglich wurden zum Testen automatisierte Unittests geschrieben. Für GUI-Komponenten wurden zum Teil automatische UI-Tests genutzt, da die Vorteile dieser gegenüber manuellen Tests allerdings nur gering waren und z.T. Probleme mit der Kompatibilität von Dateipfaden auf verschiedenen Rechnern auftraten wurden auch manuell getestet.