

# 1 Aufgabe: GUI-Frameworks (4)

Untersuche die folgenden Frameworks zur Entwicklung einer grafischen Benutzeroberfläche in Python:

1. pyGTK mit Glade
2. Tkinter
3. PyGUI
4. libavg
5. PyGObject
6. Pyforms
7. PyQt5 mit QtCreator
8. wxWidgets

Betrachte dabei unter anderem:

- Stärken und Schwächen
- Plattformunabhängigkeit
- Videodarstellung
- Qualität der Dokumentation
- Schwierigkeitsgrad der Benutzung
- Lernbarkeit
- Maintenance
- Lizenz (academic/commercial use)
- Anpassbarkeit (eigene Widgets)
- Python3 Unterstützung

Erstelle die in der Abbildung 1 dargestellte GUI möglichst detailgetreu mit jedem Framework in beiden Designs und vergleiche die Handhabung.

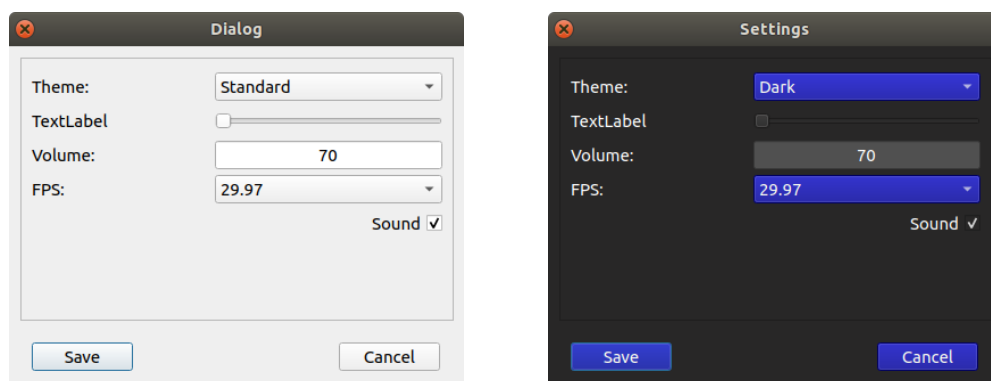


Abbildung 1: Test-GUI in hellem und dunkeltem Design

## Ansprechpartner:

Tim Jeske (*Softwarearchitekt*), tim.jeske@stud.htwk-leipzig.de

## 2 Aufgabe: Videoediting (1)

Untersuche das Videoediting Python Module MoviePy auf:

- Eignung für das Projekt
- Rendering und Export von Videos
- Qualität der Dokumentation
- Benutzung für Videos und andere Mediendateien (Bild, PDF, Ton)
- Schwierigkeitsgrad der Benutzung
- Maintenance
- Lizenz (academic/commercial use)
- Python3 Unterstützung

Suche nach alternativen Videomodulen und vergleiche diese mit MoviePy (evtl. OpenShot Library).

### **Ansprechpartner:**

Tim Jeske (*Softwarearchitekt*), tim.jeske@stud.htwk-leipzig.de

### 3 Aufgabe: OpenCV (1)

Untersuche die Bibliothek OpenCV und erlautere Benutzung und Möglichkeiten. Untersuche die Eignung dieser Bibliothek für unser Projekt und finde gegebenenfalls Alternativen. Finde heraus, ob OpenCV kompiliert werden muss und wie man es in einem Projekt einbindet, sodass es auf unterschiedlichen Betriebssystemen funktioniert (Linux, Windows, Mac).

Untersuche NumPy und beschreibe den Nutzen dieser Bibliothek. Erläutere die Verwendung und die Vor- und Nachteile der Bibliothek.

**Ansprechpartner:**

Tim Jeske (*Softwarearchitekt*), tim.jeske@stud.htwk-leipzig.de

## 4 Aufgabe: Face-Tracking (2)

Untersuche verschiedene Face Tracking Methoden (z.B. Haar, LBP) auf ihre Eignung für das Projekt. Gehe dabei möglichst auf folgende Punkte ein:

- Sidedetection
- maximale Entfernung zur Kamera
- Voraussetzungen
- minimale Auflösung
- Ausgabedaten
- Ressourcenverbrauch
- Echtzeitfähigkeit
- Sind Trainingsdaten erforderlich?
- Bestehende, nutzbare Implementierungen
- Qualität (Sensibilität und Entstehung von Ghosts)

### **Ansprechpartner:**

Tim Jeske (*Softwarearchitekt*), tim.jeske@stud.htwk-leipzig.de

## 5 Aufgabe: Face-Tracking-Bibliotheken (1)

Untersuche die API Face Recognition und die API Openface auf:

- Eignung für das Projekt
- Verwendete Methode zur Gesichtserkennung
- Voraussetzungen
- Qualität der Gesichtserkennung
- Qualität der Dokumentation
- Schwierigkeitsgrad der Benutzung
- Maintenance
- Ausgabedaten
- Lizenz (academic/commercial use)
- Plattformunabhängigkeit
- Python3 Unterstützung

### **Ansprechpartner:**

Tim Jeske (*Softwarearchitekt*), tim.jeske@stud.htwk-leipzig.de

## 6 Aufgabe: PDF-Konverter (1)

Untersuche, wie PDF-Dateien mit Python in Bilder und andere Formate konvertiert werden können. Betrachte und Vergleiche dabei diese Bibliotheken:

1. Ghostscript
2. ImageMagick

Untersuche dabei auch:

- Benutzung
- Plattformunabhängigkeit
- Python3 Unterstützung
- Größe der Bibliothek
- Maintenance

Finde gegebenenfalls Alternativen und vergleiche diese mit den oben genannten Bibliotheken.

## 7 Aufgabe: Benutzereinstellungen (1)

Untersuche wie Einstellungen eines Benutzers gespeichert werden können. Betrachte dabei die Dateiformate ini, json und yaml und die Bibliotheken:

1. ConfigParser
2. PyYaml
3. json

Eine ausführliche Erläuterung des Themas findest du hier:

<https://pyvideo.org/pycon-us-2009/pycon-2009--data-storage-in-python---an-overview0.html>

**Ansprechpartner:**

Tim Jeske (*Softwarearchitekt*), [tim.jeske@stud.htwk-leipzig.de](mailto:tim.jeske@stud.htwk-leipzig.de)

## 8 Aufgabe: Cross-Plattform-Deployment von Python (1)

Untersuche wie ein Python Projekt auf Windows und auf Linux deployed werden kann.  
Untersuche dabei die Module:

1. Pyinstaller
2. fbs
3. cx\_freeze

Suche nach Alternativen und vergleiche diese gegebenenfalls mit den oben genannten Modulen. Erfüllen diese Frameworks die Anforderungen:

- Windows und Linux Executable
- mit OpenCV
- mit GUI, wie PyQt
- Config-Dateien nach Freeze benutzbar (vergleiche Aufgabe 7) Zum Test deploye deine Programmieraufgabe oder die Vorlage der Programmieraufgabe für Windows und für Linux. Gibt es bestimmte Voraussetzungen, die erfüllt sein müssen, damit ein Projekt gefreezed werden kann?

### **Ansprechpartner:**

Tim Jeske (*Softwarearchitekt*), tim.jeske@stud.htwk-leipzig.de