

## Webanwendung zur Videoanalyse mit dem Django-Framework

### Hintergrund

Video ist eine wichtige Dokumentationsform in Requirements Engineering, z.B. Vision-Videos für die Anforderungserhebung. Vision-Videos zeigen Funktionalitäten einer Software, damit gemeinsames Verständnis zu den Anforderungen zwischen Entwicklern und Kunden geschaffen werden kann. Für diesen Fall müssen Vision-Videos nicht perfekt, aber „gut genug“ sein, um diese Rolle zu spielen.

Allerdings werden Vision-Videos oft von Entwicklern erstellt, die keine Profis in der Videoproduktion sind. Wir entwickeln daher einen Online-Dienst, um Vision-Videos automatisch zu bewerten. Entwicklern bekommen dann schnelle Rückmeldungen im Browser und können ihre Vision-Videos noch verbessern.

### Aufgabe

Im Rahmen dieser Arbeit sollen Sie eine Webanwendung mit dem Django-Framework programmieren. Backend-Server der Webanwendung soll [Debian 10](#) sein.

Die Webanwendung soll

1. folgende Funktionalitäten haben:
  - a. Authentifizierung der Benutzer;
  - b. Hochladen des Videos;
  - c. Abspielen des Videos;
  - d. Abspeicherung der personalisierten Daten (Werte, Bewertungen, Verbesserungsvorschläge usw.);
2. Video aufgrund folgender Merkmale analysieren:
  - a. für das ganze Video: Auflösung, Shutter Speed, Frame Rate, Bit Rate, Bit Depth, Sample Rate, Videolänge;
  - b. für alle Einstellungen (shots) des Videos: Shotlänge, Hintergrundfarbe, Kontrast;
3. jedes Merkmal einzeln bewerten (z.B. schlecht/neutral/gut);
4. Bewertung mit Visualisierungen und Verbesserungsvorschläge den Nutzern anzeigen.

Zu der Arbeit ist ein Bericht von etwa 50 Seiten zu erstellen, in dem alle genannten Punkte beschrieben werden.

### Organisatorisches

**Betreuer:** M. Sc. [Jianwei Shi](#)    **Prüfer:** Prof. Dr. Schneider    **Beginn:** ab 12.10.2020

#### Erforderliche Erkenntnisse:

Bildverarbeitung, Python-, Linux Bash-, und JavaScript-Programmierung

#### Vorgeschlagene Pakete für Programmierung:

[OpenCV](#) für Bildverarbeitung, [PySceneDetect](#), [Shotdetect](#), [ffprobe](#) für Shot-Erkennung

#### Hilfsmittel und Materialien:

[Django Dokumentation](#), [Virtual Box](#)