

1 Funktionale Anforderungen

Eigentlicher Gegenstand ist die Software für die Regieführung, Videoanzeige- und Verarbeitung live bei Kicker-Spielen. Außerdem gibt es als Nebenprodukt eine Clientsoftware für den Raspberry Pi.

1.1 Videoanzeige

- eingehendes Bildmaterial wird standardmäßig in der GUI als weiteres Element angezeigt
- Der Nutzer kann zur Vollbildanzeige wechseln und die Videoanzeige in einem Eigenen Fenster öffnen (und somit das Bild z.B. über den Beamer anzeigen und gleichzeitig im Hintergrund Regie führen)
- beim Eingang mehrerer Signale verschiedener Kameraclients kann die Quelle über zwei Buttons bestimmt werden (next Table, previous Table)
- entweder Anzeige eines oder mehrerer in einer Art Split-Screen (wird über einen speziellen Punkt in der Auswahl entschieden)
- über zwei Buttons kann die Anzeige gestartet und angehalten werden (Play, Stop)

1.2 Videoverarbeitung

- der Nutzer hat die Möglichkeit Mitschnitte vom Videostream anzufertigen
- über zwei Buttons kann die **Aufnahmefunktion** gesteuert werden (Start Rec, Stop Rec)
- über eine Auswahl können ein, oder Mehrere Datenträger bestimmt werden, auf jedes gewählte Gerät wird eine Kopie geschrieben
- in einem weiteren Auswahlbereich kann das Streaming an einen Online-Livestreaming-Dienst aktiviert werden.
- alternativ zur Standardqualität soll der Nutzer die Möglichkeit haben, die Auflösung und evtl. FPS einzustellen

1.3 Oberfläche

- zu jedem Tisch (also zu jeder Videoquelle) wird dem Nutzer Nummer und Hersteller des Tisches angezeigt
- ist der Online-Livestream aktiv, so werden dem Nutzer Statistiken angezeigt (Zuschauer, Anzahl d. Aufrufe usw.)

1.4 Sonstiges

- Es soll die Möglichkeit geben, über ein am Rechner angeschlossenes Mikrophon den Spielverlauf zu kommentieren
- Ton soll über Boxen wiedergegeben werden (evtl. auch online streamen)

2 Nicht-Funktionale Anforderungen

Aktuell wird angestrebt, jegliche zu erzeugende Software in Java zusammen mit JavaFX zu schreiben. Des Weiteren wird ein Repository auf GitHub angelegt. Für die Entwicklung werden Scene Builder und Netbeans eingesetzt.

2.1 Komponenten

- Bild und Ton sollen mit der Media-Klasse aus javaFx behandelt werden
- ein Kameraclient besteht aus einem Raspberry Pi und einer Kamera
- denkbare Kameras sind die Aktuelle Pi Cam und zum Testen eine einfache USB Cam von Lenco
- auf dem Pi soll eine Software laufen, mit welcher das Kamerabild abgegriffen und über LAN weiter an einen Rechner mit der Hauptanwendung, welche in den Funktionalen Anforderungen thematisiert wurde, gestreamt wird
- Kameraclients dürfen nicht auf einen speziellen Kamerateypen angewiesen sein, da zukünftig auch GoPros und andere USB-Kameras eingesetzt werden könnten
- über die Videoübertragung wird noch zu sprechen sein (evtl. Socket Server)

2.2 Oberfläche

- für Anzeige von Tischnummer und Hersteller wäre es vermutlich die beste Lösung die Zuordnung vom Nutzer machen zu lassen
- die GUI wird im Scene Builder aufgebaut

2.3 Video

- Einstellung der Qualität ergibt offensichtlich nur Sinn für Online-Livestream und Speichern auf dem Stick (zwei separate Dropdowns?)

3 Meilensteine

1. Herstellung der Kommunikation zwischen Pi und Rechner
2. rudimentäre Software für Videoübertragung von Pi zu Hauptrechner
3. Anpassung der GUI des Hauptprogramms
4. Funktionalität für Mitschnitte in unterschiedlicher Qualität
5. Verschiedene Anzeigemodi für Video
6. Audioverarbeitung
7. Vereinigung von Video und Audio

- 8. Anbindung Audio an Mitschnitte
- 9. Online-Livestream
- 10. Designüberarbeitung