Richtlinie zum „Software-Sprint“

master_me (vormals soundsgood)

Schlussbericht

Zuwendungsempfänger:

Klaus Scheuermann

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS22S24 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Vorbemerkung: In Absprache mit dem Prototype Fund, wurde das Projekt umbenannt in 'master_me'. Der Name ist passender und findet mehr Anklang.

Kurze Darstellung der Aufgabenstellung und Motivation

Mit dem ersten Lockdown in der Covid19 Pandemie 2020 stolperten viele unabhängige Konferenzen und Veranstaltungen aus dem zivilgesellschaftlichem Feld ins Digitale. Klanglich ließ das oft zu wünschen übrig. Sie konnten mit kommerziell produzierten Inhalten nicht mithalten und man konnte sich schlecht auf die Inhalte konzentrieren.

Aufgabenstellung war die Entwicklung eines Open Source Plugins zur Verbesserung der Audioqualität bei Live-Streaming-Events. Es sollte über gängige Schnittstellen in Open-Source Streaming Software einzubinden sein.

Die Problemlösung bestand darin, einen DSP (digital signal processing) Prozess zu schaffen, der Lautstärke, Frequenzgang, Dynamik und Fehler kontrolliert. Diesen galt es in ein Plugin zu giessen, welches über gängige Schnittstellen (VST,LV2,JACK) in bestehende Open-Source Streaming Software integriert werden kann. Im Vordergrund stand die Audioqualität und die einfache Usability der Software.

Beitrag des Projektes zu den Zielen der Förderinitative „Software-Sprint“

Zur Zielgruppe zählen alle, die live streamen und die Audioqualität des Streams verbessern möchten. Beispielsweise Live-Streams von Konferenzen, Konzerten und sonstigen Events. Zudem gibt es offenbar ein reges Interesse an master_me aus der Podcasting-Szene. Auch eine Anwendung bei Internet-Radiosendern ist denkbar.

Durch master_me werden Open-Source Streaming Strukturen konkurrenzfähig zu kommerzieller Software.

Selbst Menschen, die wenig Expertise im Bereich Audio haben, können durch einfaches Einbinden des Plugins eine erhebliche Verbesserung des Klangs erreichen, was auf der Empfängerseite zu besserer Hörbarkeit und letztlich zu höherer Konzentration führt.

Ausführliche Darstellung der Ergebnisse

In der ersten Stufe wurden Anforderungen gesammelt, bestehende Module getestet, neue Module definiert und erste Tests gemacht.

Im zweiten Schritt wurde mit potenziellen Anwender*innen Tests vorgenommen. Darunter war ein Test mit einem Mitglied des VOC (Video Operation Center) des Chaos Computer Clubs im Hackspace X-Hain um die Nutzbarkeit in der Streaming Architektur des VOC zu evaluieren. Auf der Konferenz re:publica 2022 wurde master_me drei Tage lang an Bühne 6 im Dauerbetrieb getestet und Parameter optimiert. Das Streaming Team der re:publica (wecap) war begeistert und wird master_me voraussichtlich bei zukünftigen Live-Stream-Events einsetzen.

Nach Fertigstellung des Plugins wurde in Stufe 3 ein Video für den Demo-Day des Prototype Fund mit Inhalten von der re:publica 2022 erstellt. Der Originalton von Bühne 6 wird mit dem, in Echtzeit bearbeiteten Sound von master_me, im A-B Vergleich gegenübergestellt wird. (https://cloud.4ohm.de/s/WowqnDkqRMHW9tf)

Mit dem Erreichen von Stufe 4, wurde master_me zum Demo-Day des Prototype Fund am 31.8. in verschiedenen Plugin-Formatenauf Github veröffentlicht. Dort findet sich zudem eine ausführliche Dokumentation und Anleitung. https://github.com/trummerschlunk/master_me

master_me kann nun als VST-Plugin in Hostsoftware wie OBS und Ardour integriert werden.

master_me existiert jetzt als fertige Stable-Release. Es war mir wichtig, nicht nur einen Prototypen als proof-of-concept zu schaffen, sondern eine belastbare Software, die für alle Anwender nutzbar ist.

Erkenntnisse:

Die DSP Programmierung klappte wie geplant.
Die Umsetzung in ein plattformübergreifend lauffähiges Plugin, in allen gängigen Formaten, erforderte erheblich mehr Aufwand als erwartet, da einige bestehende Architekturen der Programmiersprache Faust nicht so fehlerfrei liefen, wie gedacht.

Mitte August lief master_me schon sehr gut, sah aber noch nicht all zu ansprechend aus. Um mehr Akzeptanz und möglichen Einsatz in der Öffentlichkeit zu erreichen, wurde ein ansprechendes GUI geschaffen, welches sich schon jetzt großer Beliebtheit erfreut.

Die Open Knowledge Foundation stand mir mit dem Team des Prototyp Fund tatkräftig zur Seite. Ein Coaching wurde absolviert, welches zentrale Fragen zu GUI und Usability herausarbeitete.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Mit master_me klingen Inhalte besser, die mit Open-Source Software gestreamt werden. Sie werden konkurrenzfähig zu kommerziell produzierten Inhalten und sind für den Empfänger angenehmer zu hören und damit leichter zugänglich. Ein Fachwissen zum Thema Audio ist nicht nötig, wodurch keine Ressourcen unnötig auf die Optimierung des Klangs aufgewendet werden müssen.

Die Veröffentlichung als Open-Source Projekt ermöglicht die kollektiven Weiterentwicklung, welche schon jetzt Fahrt aufgenommen hat. Mehrere Entwickler tragen als Kollaborateuren zum Projekt bei. Jede*r kann aus dem Source-Code lernen.

Eine Idee, die immer wieder für Begeisterung sorgt, ist die mögliche Implementation von master_me in Hörgeräte. Dies sollte man auf jeden Fall weiter verfolgen.

Ich habe durch die Entwicklung von master_me sehr viel gelernt und bin dankbar, dass ich das Projekt mit der Hilfe des BMBF, der OKF und des Prototype Funds durchführen konnte.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Ursprünglich gab es die Idee, master_me auch in Video-Konferenz Software wie Jitsi oder BigBlueButton einzubinden. Dies ist jedoch schwierig, da es keine offenen, standardisierten Schnittstellen gibt. Nun, da das Projekt veröffentlicht ist und ausprobiert werden kann, werde ich den beiden Entwickler-Communities das Projekt vorstellen und bei der Anpassung des Codes helfen, sofern Interesse besteht.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

https://github.com/trummerschlunk/master_me

Ein Video des bekannten Open-Source-Audio Youtubers 'unfa' wird sich demnächst mit master_me beschäftigen.

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Die Arbeits- und Kostenplanung wurde eingehalten.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Wie oben schon kurz erwähnt, gab es fehlerhafte Architectures in der Programmiersprache Faust, auf die ich mich ursprünglich verlassen hatte. Das machte viel zusätzliche Arbeit, die aber letztlich erfolgreich war. Einige dieser Architetures wurden verbessert, sodass andere Programmierer*innen davon profitieren.