Hinweise:

* Schriftart Calibri, 12pt, 1.15facher Zeilenabstand, Blocksatz
* Ränder „Layout→Normal“
* Umfang: mind. 10 und max. 12 Seiten einschl. Titelblatt
* Die dunkelgrau gedruckten Hinweise (Stil „subtle emphasis“) sind zu entfernen.

Praktikum zum Modul  
Software Engineering

Sommersemester 2019

Simon Heid, 3239637

Harry Heilmann, 3271740

<<Vorname Nachname>>, <<Matrikelnummer>>

# Kurzbeschreibung des Projekts

Unser Projekt ist ein Programm, dass es ermöglicht, schnell und einfach mehrere Audiodateien hinsichtlich ihrer Lautstärke zu normalisieren. Die Lautstärke der Aufzeichnung eines Liedes variiert je nach ihrer Quelle. Diese Lautstärkeschwankungen bei der Wiedergabe mehrerer Lieder sind unangenehm und werden von den meisten Wiedergabeprogrammen nur schlecht bis gar nicht kompensiert. Unsere Idee ist es daher, den Lautstärkenausgleich bereits vor dem Abspielen durchzuführen. Dazu haben wir ein Programm entwickelt, das ihm zugeführte Dateien bearbeitet, um deren durchschnittliche Lautstärken anzugleichen. Die Ziellautstärke errechnet es dabei aus dem Mittelwert der Lautstärken einer Auswahl an Referenzdateien oder wahlweise aus dem aller zu bearbeitenden Dateien.

# Anforderungsanalyse

## User Stories

Textuelle Beschreibung(en) wie der User die App bzw. das Produkt erlebt.

## Anforderungen

Mind. 5 nummerierte Anforderungen formuliert mit Satzschablone. Etwaige Auflistungen (z.B. Betriebssysteme, Plattformen, Browser, …) am Ende dieses Abschnitts einfügen. Sofern zutreffend oder möglich, erstellen Sie eine Priorisierung.

1. Das Programm muss fähig sein, eine gegebene Liste von WAV-Audiodateien so zu bearbeiten, dass diese Audiodateien mit demselben Lautstärkelevel, welches aus dem Durchschnitt der Dateien berechnet wird, gespeichert werden.
2. Das Programm muss über eine grafische Anwendungsoberfläche bedienbar sein.
3. Das Programm sollte die Möglichkeit bieten, eine globale Skalierung des Lautstärkenlevels im Bereich von 0,1 bis 10 vorzunehmen.
4. Das Programm sollte die Möglichkeit bieten, eine oder mehrere Audiodateien als Referenz(en) für das Lautstärkelevel der anderen Dateien anzugeben.
5. Das Programm sollte die Möglichkeit bieten, auch weitere Audiodateiformate (vorzugsweise FLAC und MP3) als Eingabe zu verwenden.

# Modellierung

## Use Cases (Diagramme und tabellarische Form)

Mind. 1 Use Case Diagramm, mind. 3 tabellarische Beschreibungen von Use Cases. Referenzieren Sie hier entsprechende Anforderungen.

## Klassendiagramme

Mind. 1 Klassendiagramm; externe Module nach Möglichkeit mit Schnittstellennotation („Lollipop“ bzw. Buchse/Stecker).

## Objektdiagramme

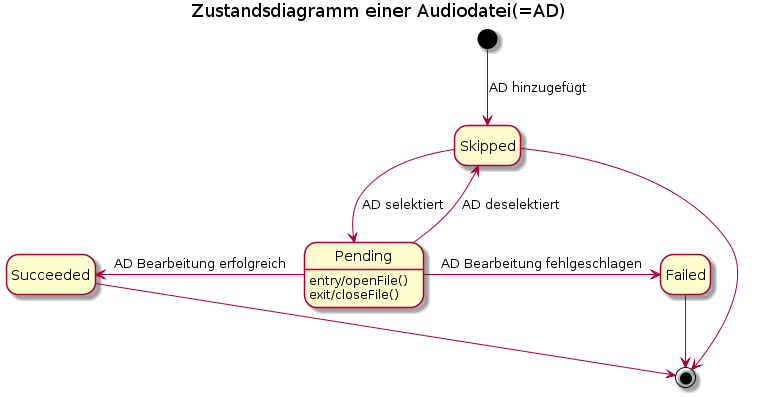
Mind. 1 Objektdiagramm für einen geeigneten Use Case; verzeichnen Sie auch externe Schnittstellen bzw. Module.

## Sequenzdiagramme

Mind. 1 Sequenzdiagramm für einen geeigneten Use Case.

## Zustandsdiagramme

Mind. 1 Zustandsdiagramm für eine Komponente, einen Use Case oder die gesamte Applikation.



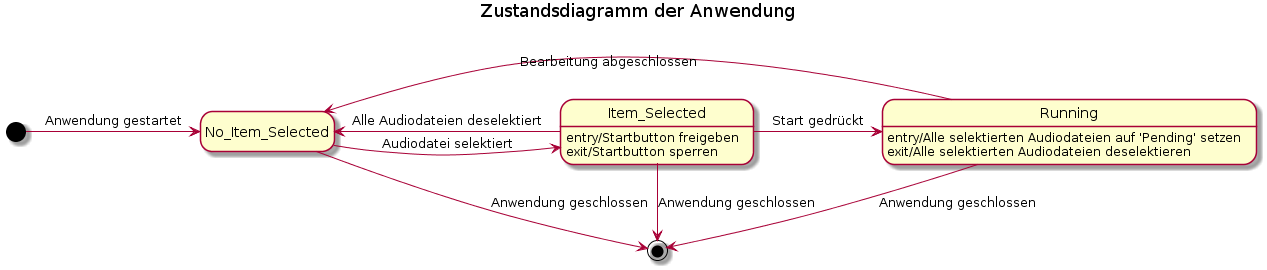
Abbildung 1: Zustandsdiagramm einer Audiodatei auf der Oberfläche

Abbildung 2: Zustandsdiagramm der Anwendungsoberfläche

# Implementierung

Beschreiben Sie die Rahmenbedingungen und begründen Sie kurz die Wahl der verwendeten Toolkits. Listen Sie die erfüllten sowie nicht erfüllten Anforderungen, und erläutern Sie, welchen individuellen Beitrag jedes Teammitglied geleistet hat.

## Erfüllte Anforderungen

## Nicht erfüllte Anforderungen

## User Interface

Zeigen Sie anhand von Fotos oder Screenshots, wie Ihr Prototyp die Eingangs beschriebenen User Stories nachbildet. Referenzieren Sie entsprechend die (nicht) erfüllten Anforderungen (max. 2 Seiten).

# Reflexion

Vergleichen Sie die frühe Planung in den ersten Wochen mit den dem Ergebnis am Ende.

* Entspricht das Ergebnis Ihrer ursprünglichen Vision?
* Was lief wie geplant, welche (überraschenden?) Änderungen haben sich ergeben?
* An welcher Stelle sind Sie nicht weitergekommen, wo haben Sie am meisten dazugelernt?
* Hat sich die geplante Arbeitsaufteilung bewährt?
* Was würden Sie für das nächste Projekt anders machen?