Inhaltsverzeichnis

[1. Einführung 1](#_Toc380743710)

[2. Vorgeschlagenes System 2](#_Toc380743711)

[2.1. Übersicht: 2](#_Toc380743712)

[2.2. Funktionale Anforderungen 2](#_Toc380743713)

[2.3. Nichtfunktionale Anforderungen 3](#_Toc380743714)

[2.4. Systemmodelle 4](#_Toc380743715)

[2.4.1. Szenarien 4](#_Toc380743716)

[2.4.2. Anwendungsfallmodell: Use case-Diagramm + Use case Beschreibungen 4](#_Toc380743717)

[*2.4.3. Statisches Modell:* 4](#_Toc380743718)

[2.4.4. Dynamisches Modell: Sequenzdiagramme + Zustandsdiagramm(e) 4](#_Toc380743719)

[3. Glossar 4](#_Toc380743720)

[Anhang A: GUI-Skizzen 4](#_Toc380743721)

# 1. Einführung

Das hier vorgestellte System ist eine Social-Media-Plattform auf der Benutzer Musik von Künstlern hören und in persönlichen Wiedergabelisten speichern können. Jeder Benutzer hat ein eigenes Profil und kann sich die Profile aller anderen Benutzer anschauen. Der Benutzer kann durch eine Follow-Funktion über Beiträge und neue Wiedergabelisten oder Alben von Benutzern und Künstlern informiert werden.

Ein Benutzer kann zum Künstler werden und dann selbst Musik publizieren, verbreiten und für sich werben. Diese wird dann in seinem Profil angezeigt und ist für jeden sicht- und hörbar.

Neben den Benutzer existieren auch sogenannte Labels, die die Profile von den ihnen zugehörigen Künstlern in deren Namen verwalten können. Über diese Funktion hinaus haben Labels ebenfalls ein eigenes Profil wo sie eigene Playlist zusammenstellen und für ihre Künstler werben können.

Jedes Label wird von mindestens einem Manger verwaltet, der dadurch über mehr Funktionalität verfügt als der normale Benutzer. Er kann, wie der Künstler auch, Musik hochladen, in Alben gruppieren und in Genre einteilen.

Alle erzeugten oder hochgeladenen Daten werden vom System auf einem lokalen Server konsistent gespeichert.

Im Hintergrund des Systems existiert eine Kontrollinstanz in Form von realen Administratoren. Ihre Aufgabe ist es Beschwerden zu bearbeiten und die Authentizität von Künstlern, Labeln und Label-Managern mit Hilfe eines Drittsystems zu verifizieren.

# 2. Vorgeschlagenes System

## 2.1. Übersicht:

Kurze textuelle Beschreibung

## 2.2. Funktionale Anforderungen

##### Benutzerfreundlichkeit

* Kenntnisstand des Benutzers:  
  Er ist in der Lage Anwendungen zu bedienen.
* Normen für die Benutzerschnittstelle:  
  Wir haben in der Vorlesung Normen jeglicher Art nicht behandelt.
* Welche Dokumentation soll dem Benutzer übergeben werden:  
  Eine Video Einführung in das Programm.

##### Zuverlässigkeit, Verfügbarkeit, Robustheit

* Wie zuverlässig, verfügbar, robust soll das System sein:  
  Das System soll nach Möglichkeit Abstürze vermeiden.
* Ist ein Systemneustart im Falle eines Fehler akzeptabel:  
  Nein.
* Wie viele Daten darf das System verlieren:  
  Keine bereits gesicherten Daten.
* Wie soll das System mit Ausnahmen umgehen:  
  Ausnahmen sollen abgefangen und dem Benutzer als Dialog angezeigt werden.
* Gibt es Betriebssystem Anforderungen:  
  Das System soll auf einem Computer funktionsfähig sein.
* Gibt es Datenschutzanforderungen:  
  Der Zugriff darf nur durch eingeloggte (insbesondere registrierte) Benutzer erfolgen dürfen. Des Weiteren sollen die persönlichen Passwörter verschlüsselt gespeichert werden.

##### Leistungsanforderungen

* Wie schnell muss das System reagieren:  
  Es sollte so schnell wie möglich reagieren.
* Gibt es zeitkritische Benutzeraufgaben:  
  Nö.
* Wie viele Anwender soll das System gleichzeitig unterstützen:  
  Eine Person.
* Welche Latenz wird noch akzeptiert:  
  Eine Latenz von 42.
* Wie groß sind Daten Speicher vergleichbarer Systeme:  
  Petabyte.

##### Unterstützungen

* Welche Erweiterungen sind geplant:  
  Keine.
* Wer wartet das System:  
  Die Administratoren (bis auf weiteres die Gruppenmitglieder).
* Gibt es Pläne zur Portierung auf andere Systeme:  
  Nein.

##### Implementierung

* Gibt es Beschränkungen auf gewisse Hardware- und oder Softwareplattform  
  Java 7
* Gibt es Beschränkungen durch die Wartungsarbeitsgruppe  
  Loginsperre während der Wartung
* Gibt es Beschränkungen durch die Testarbeitsgruppe  
  Nein, es wird ausschließlich in Testumgebung getestet

##### Schnittstelle

* Soll das System mit anderen bereits vorhandenen Systemen interagieren  
  Nein
* Wie werden Daten in das System exportiert/ aus das System exportiert  
  Mit Hilfe einer Datenbank und dem Dateisystems.
* Welche Normen sollen vom System unterstützt werden:  
  Wir haben in der Vorlesung Normen jeglicher Art nicht behandelt.

##### Betrieb

* Wer kümmert sich um das laufende System:  
  Die Administratoren (bis auf weiteres die Gruppenmitglieder).

##### Installation

* Wer installiert das System:  
  Der Benutzer.
* Wie viele Installationen sind geplant:  
  Eine.
* Gibt es zeitlich Beschränkungen für die Installation:  
  Nein.

##### Rechtliches

* Wie soll das System lizensiert werden:  
  Es wird keine Lizensierung erforderlich sein.
* Wer trägt die Verantwortung bei einem Systemfehler:  
  Keiner.
* Werden durch die Benutzung bestimmter Algorithmen oder Komponenten Lizenzen fällig:  
  Nein.

## 2.3. Nichtfunktionale Anforderungen

Textuelle Beschreibung der relevanten nichtfunktionalen Anforderungen

## 2.4. Systemmodelle

### 2.4.1. Szenarien

#### Anwendungsfall Listener:

##### Anwendung starten

* User lädt die Anwendung herunter, installiert und führt sie aus.

##### Registrierung

* User registriert sich über das Registrierungsformular und gibt dabei seine persönlichen Informationen (Password, Benutzer Name, etc...) an.

##### Einloggen/Abmelden

* User logt sich mit seinen Login Daten ein.

##### Suche Medium nach Kriterium XY

* User gibt in die Suchleiste ein Kriterium (Genre, Band, Label) ein und erhält eine Liste mit Ergebnissen. Aus diesen Ergebnissen kann er nacheinander eins auswählen und anschließend betrachten.

##### Suche von Benutzern

* User gibt einen anderen Benutzernamen oder eine andere Email Adresse in das Suchfeld ein und erhält eine Liste mit übereinstimmenden Benutzerprofilen.

##### Playlist erstellen

* Der User wählt die Funktion: neue Playlist erstellen. Diese wird unmittelbar darauf benannt. anschließend kann der User beliebige Titel der wiedergabeliste hinzufügen. Gespeichert wird automatisch.

##### Playlist bearbeiten

* der User fügt einen beliebige Titel der wiedergabeliste hinzu. Gespeichert wird automatisch.

##### Playlist löschen

* der User lässt sich die Wiedergabeliste anzeigen und wählt einen Titel oder mehrere Titel aus, die/den er gerne löschen möchte und bestätigt die Sicherheitsabfrage. gespeichert wird automatisch.

##### Persönliche Daten bearbeiten/...

* der User wählt die Maske „profilbearbeiten“ und verändert den Inhalt der Textfelder, die sein Profil beschreiben/darstellen.
* nach dem erfolgreichen speichern durch Bestätigung wird das veränderte Profil gespeichert und ist nun online für alle anderen User verfügbar.

##### Abspielen von Titeln/Playlists/

* Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. in Dauerschleife/einmal/

##### Interagieren mit anderen Benutzer/Informationen teilen

* User sucht nach Olaf und sieht sich sein Profil an. da er ihm als sehr sympathisch erscheint möchte er ihn als Freund hinzufügen und klickt auf den entsprechenden Button. Nun ist er Freund von Olaf.

##### Vorschläge für Titel/Playlists/Benutzer/ unterbreiten

* User schickt Olaf eine private Nachricht und verweist auf den Titel/Playlist/Benutzer in Form eines links.

##### Bewertungen abgeben an Titel/Künstler/Label

* Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

### 2.4.2. Anwendungsfallmodell: Use case-Diagramm + Use case Beschreibungen

### 2.4.3. Statisches Modell:

Klassendiagramm für Entitätsklassen + Klassenbeschreibungen für Entitäts-, Grenz- und Kontrollklassen

### 2.4.4. Dynamisches Modell: Sequenzdiagramme + Zustandsdiagramm(e)

# 3. Glossar

Lexikonartige Auflistung und Kurzerklärung wichtiger Begriffe

# Anhang A: GUI-Skizzen

GUI-Skizzen der zentralen Grenzklassen