M335 | Mobile-App

Teil 1 - Planung

Ronny müller und nadja mullis

2016

Inhaltsverzeichnis

[Kurzfassung 2](#_Toc469383967)

[Mockup 2](#_Toc469383968)

[Programmabläufe 4](#_Toc469383969)

[Nice to have 4](#_Toc469383970)

[Aufgabenaufteilung 4](#_Toc469383971)

[Schnittstellendefinitionen 4](#_Toc469383972)

[ERM 5](#_Toc469383973)

[JS-Framework 5](#_Toc469383974)

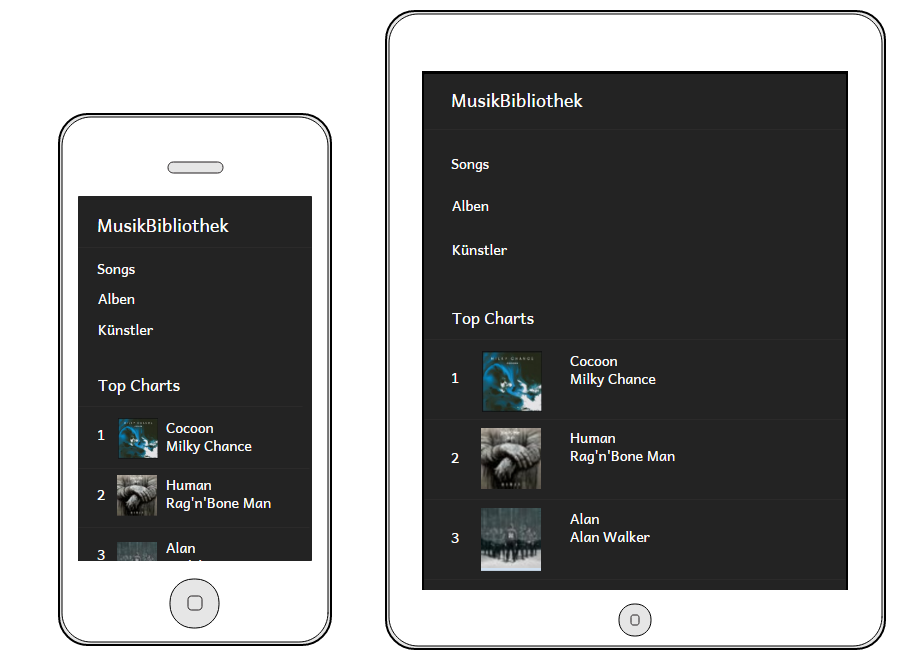
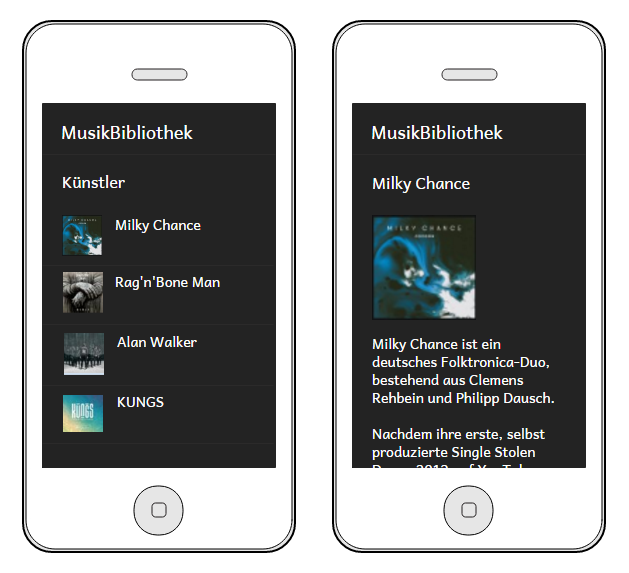
[PHP-Framework 5](#_Toc469383975)

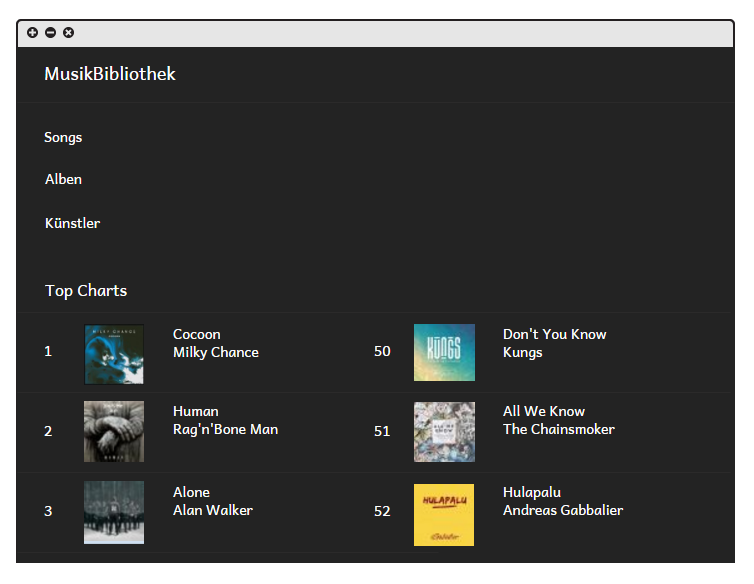
# Kurzfassung

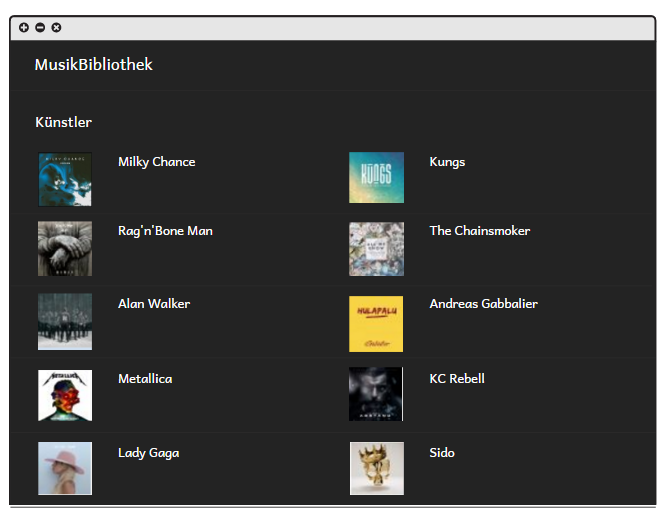
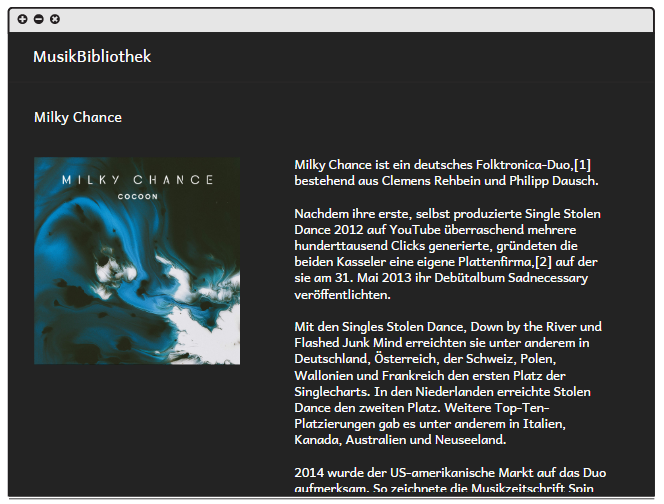
Die grundsätzliche Idee ist eine Mobile Applikation zu programmieren, um eine Musik Bibliothek zu verwalten. Es sollte so einfach wie möglich gestaltet werden Informationen über ein Album, Song oder Künstler zu erhalten. Die Applikation hilft Songs zu erkennen und diese einem Künstler/Album zuzuordnen oder um zu schauen was für Songs der Künstler/Album sonst noch hat. Die einzelnen Lieder kann man dann mit einem Klick auf Youtube öffnen. Zudem gibt es Top Charts dort werden die Aktuellsten Charts in einer Rangliste dargestellt. Zudem kann man jedes Lied Up- oder Downvoten und somit werden die Top 50 Songs bestummen.

# Mockup

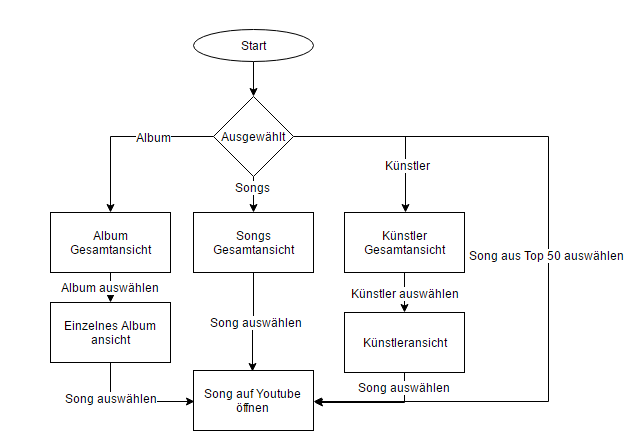
Es wurden in den Mockups als erstes die Hauptübersicht also die Startseite definiert. Sie wird auf Mobilen Geräten wie Tablet und Smartphone gleich aussehen. In der Desktopansicht werden die Top Charts auf zwei spalten aufgeteilt. Die Applikation verhält sich gleich in der Detailansicht. Die Verwaltungs-App ist vollumgänglich Responsive und wird auf allen End-Geräten passend dargestellt. Somit haben wir eine einheitliche Applikation erstellt die der Erwaratungskonformität und den weiteren sieben Grundsätze der Softwareergonomie entspricht.







Programmabläufe



Auf der Startseite werden zuoberst die «Filter Auswahl» dargestellt. Darunter werden die Top Charts Lieder dargestellt. Wenn man über den Filter etwas auswählt kommt man in eine Gesamtansicht wo alle Alben, Songs oder Künstler dargestellt werden. Wenn man einen Datensatz auswählt gelangt man in die Detailansicht der Künstler oder des Albums. Wenn man einen Song anklickt gelangt man direkt auf YouTube. Ebenfalls wenn man einen Song in der Top 50 anklickt kommt man auch direkt auf YouTube.

## Nice to have

Zusätzlich gibt es noch «Nice to have». Falls es Zeitlich noch drinnen liegt werden wir eine Art Up and Down Vote System machen. Somit kann man dann die Top Charts beeinflussen und selbst seine Meinung Einfliessen lassen.

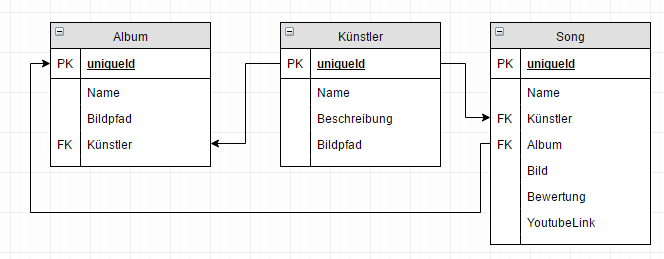
Aufgabenaufteilung

Da wir nur zu zweit sind ist die Aufgabenaufteilung grundsätzlich einfach. Der eine macht das Backend und der andere das Frontend. Wir haben uns geeinigt das Ronny Müller das Backend macht und Nadja Mullis das Frontend. Zusätzlich werden wir zusammen die nötige Dokumentation machen und die Testfälle. Am Schluss werden wir zusammen noch die Nice to have Ideen umsetzen, wenn es Zeitlich noch drinnen liegt. Wir werden das Projekt in Zusammenarbeit über GIT umsetzen. Die Ordnerstruktur wurde deshalb schon im Vorhinein definiert.

# Schnittstellendefinitionen

Wir verwenden keine Schnittstellen, weil wir nicht sicher sind ob es dann noch mit der Zeit reicht, das ganze Projekt umzusetzen.

# ERM



# JS-Framework

Wir werden Angular verwenden, weil wir keine Erfahrungen in diesem Bereich gesammelt haben bis jetzt. Und da wir Angular angeschaut haben hier im ÜK werden wir deshalb Angular nehmen.

# PHP-Framework

Grundsätzlich werden wir kein PHP-Framework verwenden. Wir werden das Projekt aber nach dem MVC-Muster umsetzen.

# Backend-Informationen

## Definition PHP-Files

**initDB.php: (muss erstmalig ausgeführt werden)**

|  |  |
| --- | --- |
| Beschreibung: | Öffnet   * /sql/createTables.php * /sql/fillTables.php   Welche die Tabellen in der DB erstellen und befüllen. |

**Uebersicht.php**

|  |  |
| --- | --- |
| Beschreibung: | Holt das JSON im /class/ Unterordner, welches per html GET übergeben wurde. Wird verwendet für alle Übersichten Seiten.  (class/album.php, class/kuenstler.php, class/song.php, class/top20.php) |

/class/\*.php

|  |  |
| --- | --- |
| Beschreibung: | Hier sind alle Files, welche die JSON’s erstellen. |