

Audit 1

**Ermöglichung einer demokratisch basierten Musikwahl bei
Partys und Veranstaltungen**

Bogdan Krenz, Carlos Bystron



Der Inhalt

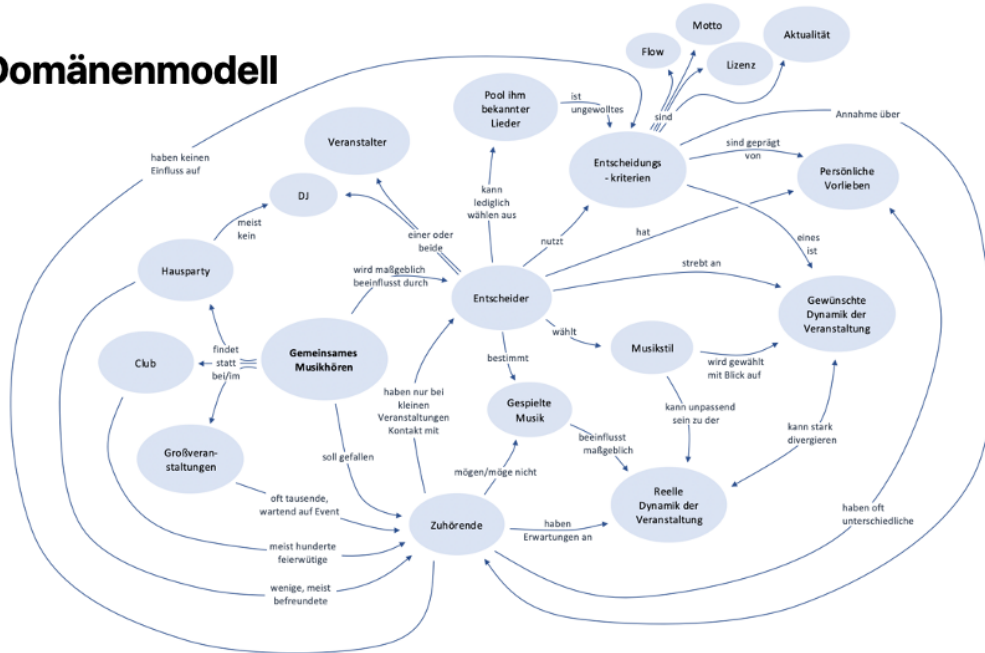
- Domänenmodell •
- Fishbone Diagramm •
- Zielhierarchie •
- Architekturmodell •
- API Analyse •
- Projektplan •



Persönliche Motivation:

Wir vermissen Partys, ganz klar. Was wir nicht vermissen ist die teilweise sehr schlechte Musik auf Partys. Egal ob Hausparty, im Club oder bei Großveranstaltungen: Viel zu oft läuft Musik, die zu vielen Anwesenden nicht gefällt. Das wollen wir ändern.

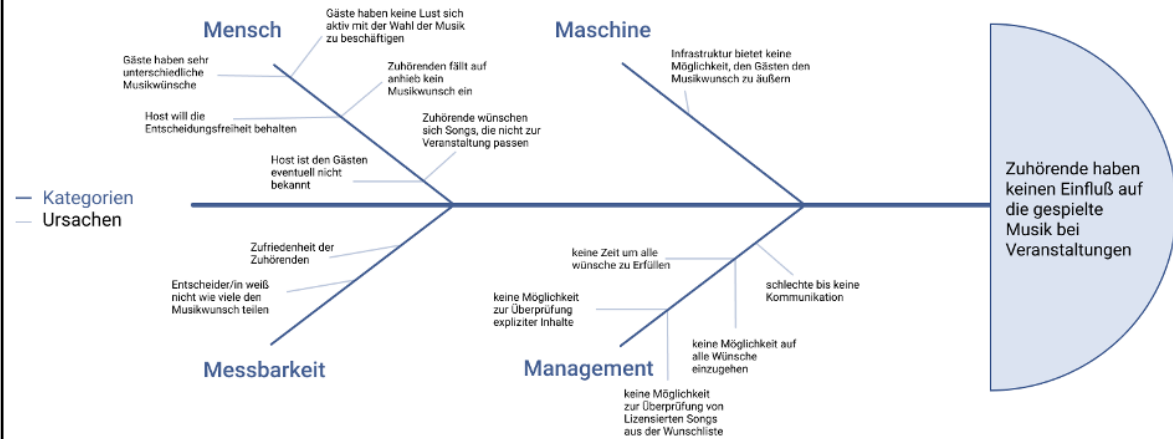
Das Domänenmodell



Das Domänenmodell entstand iterativ um das Konzept des gemeinsamen Musikhörens. Bereits in der ersten Iteration wurde herausgearbeitet, dass bei den meisten Veranstaltungen eine oder wenige Entscheider/innen für die Wahl der Musik verantwortlich sind. Die Zuhörenden haben meist keine Möglichkeit die gespielte Musik zu beeinflussen.

Der entscheidende Zielkonflikt kann dabei zwischen der gewünschten Veranstaltungsdynamik des/der Entscheiders/in und der reell empfundenen Dynamik der Zuhörenden entstehen. Diese können teilweise stark divergieren, bei Großveranstaltungen insbesondere ohne dass der Entscheider dies bemerkt. Der/Die Entscheider/in trifft seine/ihre Musikwahl mit, meist unterbewusst, gewählten Entscheidungskriterien. Diese können beispielsweise mit Lizenzen und der Aktualität der Songs zusammenhängen, werden jedoch auch von persönlichen Vorlieben beeinflusst. In jedem Fall kann der Entscheider lediglich Lieder wählen, die ihm bekannt sind was die Auswahl potentiell gespielter Lieder enorm einschränkt.

Das Fishbone Diagramm



Um den Problemraum näher zu beleuchten, wurde ein Fishbone Diagramm erstellt. Im Diagramm wird zwischen Problem, Kategorien und Ursachen unterschieden. Das Problem liegt im "Kopf" des Diagramms, die großen Knochen repräsentieren die Kategorie und kleine Knochen die Ursache. Bei der näheren Betrachtung des Problems sind folgende Kategorien in den Vordergrund gekommen: Management, Mensch, Messbarkeit und Maschine.

Es fällt auf dem ersten Blick auf, dass es sich grundsätzlich sowohl um Management- als auch Menschenbedingte Ursachen für die Problemstellung handelt. Zumal haben Gäste einen speziellen Musikwunsch den sie gerne äußern wollen, doch es wird in den meisten Fällen ihnen nicht einmal die Möglichkeit geboten diesen zu äußern. Das ist jedoch nicht einseitig, denn die Veranstalter, DJ's oder in unserem Fall die Host's, haben meist nicht die Möglichkeit mit den Menschen auf der Veranstaltung zu interagieren und die Lieder auf mögliche Kompatibilität oder explizite Inhalte zu überprüfen.

Wissenschaftliche Evidenz

„[...] listeners reported more intense emotions (1) to self-chosen music than to randomly selected music [...]"

Liljeström, Juslin & Västfjäll (2013)

„[...] emphasised the important role of engaging with music in the company of others with regard to SWB [subjective wellbeing]"

Weinberg & Joseph (2017)

"Participants exposed to their self-selected music were most efficient, perceived lowest distraction, highest enjoyment [...]"

Cassidy & Macdonald (2009)

Eine Auswahl der Rechercheergebnisse, die uns in unserer Wahrnehmung der Evidenz des Problemraums bestärkt haben:

"Consistent with predictions based on previous field studies, listeners reported more intense emotions (1) to self-chosen music than to randomly selected music[...]"

Liljeström, S., Juslin, P. N., & Västfjäll, D. (2013). Experimental evidence of the roles of music choice, social context, and listener personality in emotional reactions to music. *Psychology of Music*, 41(5), 579–

599. <https://doi.org/10.1177/0305735612440615> (abgerufen am 15.11.2020)

"The findings also emphasised the important role of engaging with music in the company of others with regard to SWB [subjective wellbeing], highlighting an interpersonal feature of music."

Weinberg, M. K., & Joseph, D. (2017). If you're happy and you know it: Music engagement and subjective wellbeing. *Psychology of Music*, 45(2), 257–

267. <https://doi.org/10.1177/0305735616659552> (abgerufen am 15.11.2020)

"Participants exposed to their self-selected music were most efficient, perceived lowest distraction, highest enjoyment, liking and appropriateness, and experienced a

reduction in tension-anxiety. In contrast, performance and experience were poorest when exposed to High-Arousal experimenter-selected music. Participants were most inaccurate, perceived highest distraction, lowest liking, enjoyment and appropriateness, and experienced an increase in tension-anxiety.”

Cassidy, G., & Macdonald, R. (2009). The effects of music choice on task performance: A study of the impact of self-selected and experimenter-selected music on driving game performance and experience. *Musicae Scientiae*, 13(2), 357–386. <https://doi.org/10.1177/102986490901300207> (abgerufen am 15.11.2020)

Die Zielhierarchie

Projektziel

Bei Veranstaltungen soll die Entscheidung über die gespielte Musik demokratisiert werden. Dem/Der Entscheidenden über die Musik (Host) muss dazu eine Möglichkeit geboten werden zu erfahren, welche Musik den meisten Veranstaltungsgästen (Guests) gefällt.

Strategische Ziele

- Veranstaltungsgäste müssen Musik teilen können, die ihnen gefällt ohne viel Zeit für die Suche aufzubringen.
- Hosts müssen Musikvorschläge erhalten, die möglichst vielen Gästen gefallen
- Es muss ein lauffähiger Prototyp erstellt und getestet werden
- Veranstaltungsgäste sollten während der Veranstaltung die Qualität der gespielten Musik bewerten können.
- Hosts sollten während der Veranstaltung um live Feedback bitten und dieses einsehen können

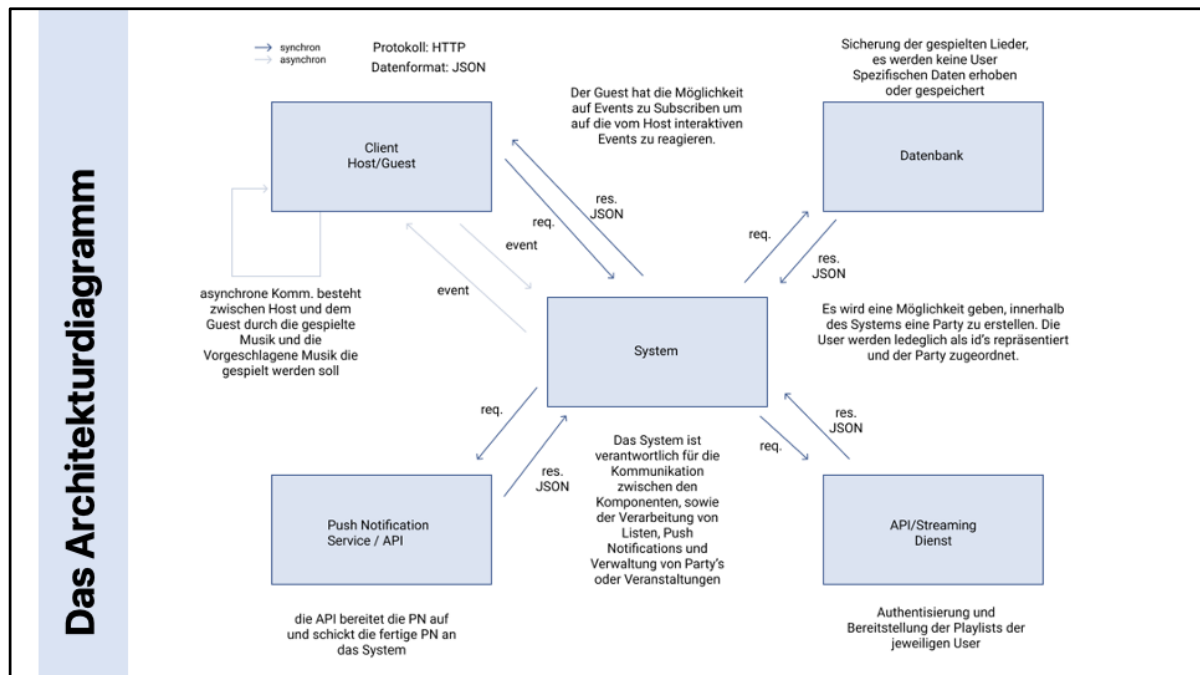
Die Einteilung der Zielhierarchie wurde wie im Medien Informatik Glossar definiert vorgenommen. Eine stetige Erweiterung insbesondere der operativen Ziele ist angedacht.

Taktische Ziele

- Guests müssen sich mit ihrem Konto eines Streaming Anbieters anmelden können
- Hosts müssen Liste vorgeschlagener Songs, die möglichst vielen Gästen gefallen, erhalten können
- Hosts müssen die Liste nach verschiedenen Kategorien filtern können
- Guests sollten eine Benachrichtigung bekommen, wenn der Host um Feedback zu der gespielten Musik bittet
- Guests sollten nach Abstimmen
- Der Hosts sollte Abstimmungen starten sowie deren Ergebnisse und resultierende Handlungsvorschläge einsehen können
- Das System kann mit verschiedenen Streaming Anbietern genutzt werden

Operative Ziele

- Es müssen zwei User Interfaces – eines für die Hosts, eines für die Guests – design und implementiert werden
- Es muss die API eines Streamingdienstes ausgewählt und an das System angebunden werden
- Es muss eine umfängliche Anwendungslogik implementiert werden, welche aufgrund der Musikvorlieben der anwesenden Gäste passende Musik für die Veranstaltung findet
- Es muss eine geeignete Architektur für das verteilte System gewählt werden
- Es muss ein backend Service umgesetzt werden, der die Daten der API sowie Zugriffe auf die Datenbank verwaltet
- Es muss eine Datenbank eingebunden werden welche die Musikvorlieben der Gäste sowie die Musikvorschläge für den Host speichert
- Es muss ein PoC umgesetzt werden, um eine erste Funktionsfähigkeit des Systems einschätzen zu können
- Das System sowie sämtliche Komponenten sollten ausreichend getestet werden



Das Architekturdiagramm beinhaltet die Systemkomponenten. Im Frontend haben wir die Clients mit dem Host und den Guests. Hier haben die User die Möglichkeit sich mit dem von ihnen gewünschten Musik Streaming Plattform anzumelden. Das System ist verantwortlich für die Kommunikation zwischen den Komponenten, sowie der Verarbeitung von Listen, Push Notifications und Verwaltung von Party's oder Veranstaltungen. Durch die API's der Streaming Dienste wird eine Authentisierung des Users ermöglicht, und es werden die Playlists der User an das System geschickt. Die Listen werden bearbeitet und in einer Datenbank gesichert. Die User werden hier lediglich als id's gespeichert und es werden keine userspezifischen Daten erhoben oder gesichert. In allen Teilen des Systems herrscht eine synchrone Kommunikation zwischen den Komponenten. Überall wo eine Anfrage an eine Komponente geschickt wird, kommen Daten im JSON Format in der Antwort zurück.

Es gibt zwei Ausnahmen, es herrscht eine asynchrone Kommunikation zwischen dem Host und dem Client, der Host kommuniziert mit den Gästen über die gespielte Musik. Die Guests kommunizieren mit dem Host über die Listen und Vorschläge. Der Host kann Events starten, die auch asynchron sind, da der Guest auf diese erst subscriben muss. Auf diesen Events kann der User Feedback zur aktuell gespielten Musik geben.

Alle Komponenten und Diagramme sind nicht in Stein gemeißelt und werden mit dem Wachsen des Projekts angepasst und auf dem aktuellsten Stand gehalten.

Die API Analyse

	Spotify	Amazon Music	Apple Music	Soundcloud
Marktanteil in Deutschland (nach Statista, July 2020)	30%	30%	10%	5%
Marktanteil Weltweit (MIDIA Research, Dezember 2019)	36%	13%	18%	keine Angaben
Kostenlose, offene API?	Ja - für manche Abfragen Registrierung erforderlich	Nein - vorhanden, jedoch nicht von Amazon selbst betrieben, sondern von Drittanbieter über GitHub	Nein - Für Zugriff auf die API ist ein bezahlter Apple Developer Account nötig	Ja
Architekturstil	REST	Nicht definiert - Zugriff via Python	REST	REST
API stellt erforderliche Daten zur Verfügung	Ja - Zugriff auf gespeicherte Lieder, zuletzt gehörte Lieder, meist gehörte Lieder, Playlists, Lieblingssongs usw. uneingeschränkt möglich	Nein - lediglich gespeicherte Alben können eingesehen werden	Ja - Zugriff auf gespeicherte Playlists und Songs sowie auf Musikvorschläge für Nutzer von Apple Music möglich	Nein - Playlists und Songs der Nutzer einsehbar jedoch keine vorgeschlagenen Lieder, zuletzt gehört oder meist gehörten Songs
API ermöglicht schnelles Zugreifen auf erforderliche Nutzerdaten	Ja - wenn Zugriff vom Smartphone u. App installiert findet redirect statt und Zugriff kann in App erteilt werden. Wenn App nicht installiert Freigabe nach Anmeldung mit E- Mail/Passwort.	Nein - Anmeldung mit E- Mail/Passwort nötig.	Nein - requestUserToken ermöglicht Freigabe ohne Anmeldung nur auf iOS Geräten, sonst Freigabe nach Anmeldung mit E-Mail/Passwort	Nein - Anmeldung mit E- Mail/Passwort nötig.
Dokumentation	Sehr gut strukturiert und verständlich - Nutzung von wohldefinierten Standards, bspw. Statuscodes nach RFCs 2616 u. 6585	Sehr kurz und nur wenige Operationen ausreichend dokumentiert	Grundsätzlich übersichtlich und umfänglich jedoch teilweise sehr kurz u. ohne konkrete Ausführungen	Grundsätzlich übersichtlich und umfänglich jedoch Fokus eher auf Streamen u. Uploaden von Musik, weniger auf Präferenzen beim Hören
Zuverlässigkeit	Gut - Die ausführliche Dokumentation und der große Aufwand, der in das Bereitstellen der API investiert wird, sprechen klar für Zukunftsperspektive. Frühe Beta Versionen ermöglichen, wenn nötig, koordinierten Umstieg auf neue Version.	Nicht gegeben - Da keine API von Amazon zur Verfügung gestellt wird ist die Unsicherheit sehr groß und häufige Änderungen erwartbar.	Gut - Durch Anbindung an bezahlten Developer Account sollte Zuverlässigkeit gewährleistet sein	Voraussichtlich gut - es bestehen keine Gründe zur Annahme von Unzuverlässigkeit
Datenschutz	Der User wird ausführlich über die geteilten Daten aufgeklärt und kann seine Zustimmung jederzeit über seinen Spotify Account widerrufen	Keine Angaben	Der User wird ausführlich über die geteilten Daten aufgeklärt	Der User wird bei der Anmeldung darauf hingewiesen, dass seine Daten geteilt werden
Pluspunkte	API ermöglicht das Abspielen von Songs direkt in der WebApp	-	Ermöglicht auch Zugriff auf die iTunes Mediathek der Nutzer	Führt sehr viel lizenzfreie Musik u. unbekanntere Künstler

Bei der API Analyse wurden die Schnittstellen der drei nach Marktanteil größten Musik-Streaming-Anbieter Deutschlands sowie die des Anbieters SoundCloud verglichen. SoundCloud wurde trotz des vergleichsweise geringen Marktanteils mit in die Analyse aufgenommen, da der Anbieter einen anderen Ansatz wählt und Künstlern eine einfache und schnelle Möglichkeit bietet ihre Musik zu verbreiten. Somit findet sich auf der Plattform viel lizenzfreie Musik sowie Musik weniger verbreiteter Subgenre.

Die Analyse Kategorien wurden anhand der Leitfragen, aufgeführt im Web Development Schwerpunkt des Veranstaltungsleitfadens, sowie entscheidender Anforderungen zur Realisierung des geplanten Systems gewählt. In der Analyse geht die API des Streaming Anbieters Spotify klar als beste Alternative für die geplante Projektidee hervor. Somit wird der PoC im Folgenden auf dieser API aufbauen. Im Backlog lässt sich vermerken, dass eine spätere Erweiterung des Systems die Anbieter Apple Music und SoundCloud einschließen sollte um mehr Nutzern die teilhabe an demokratischer Musikkwahl zu ermöglichen. Von einer Einbindung des Dienstes Amazon Music ist aufgrund der unzuverlässigen API und den fehlenden Hinweisen zum Datenschutz abzusehen.

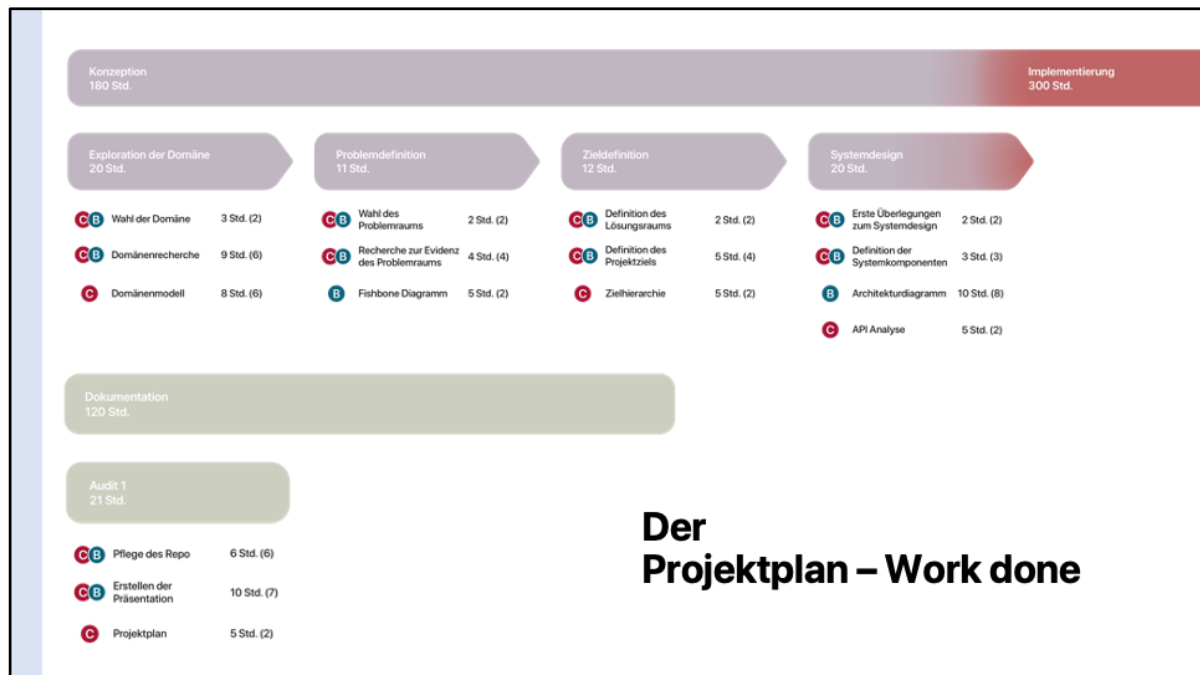
Der Projektplan – Ebene 1

Konzeption
180 Std.

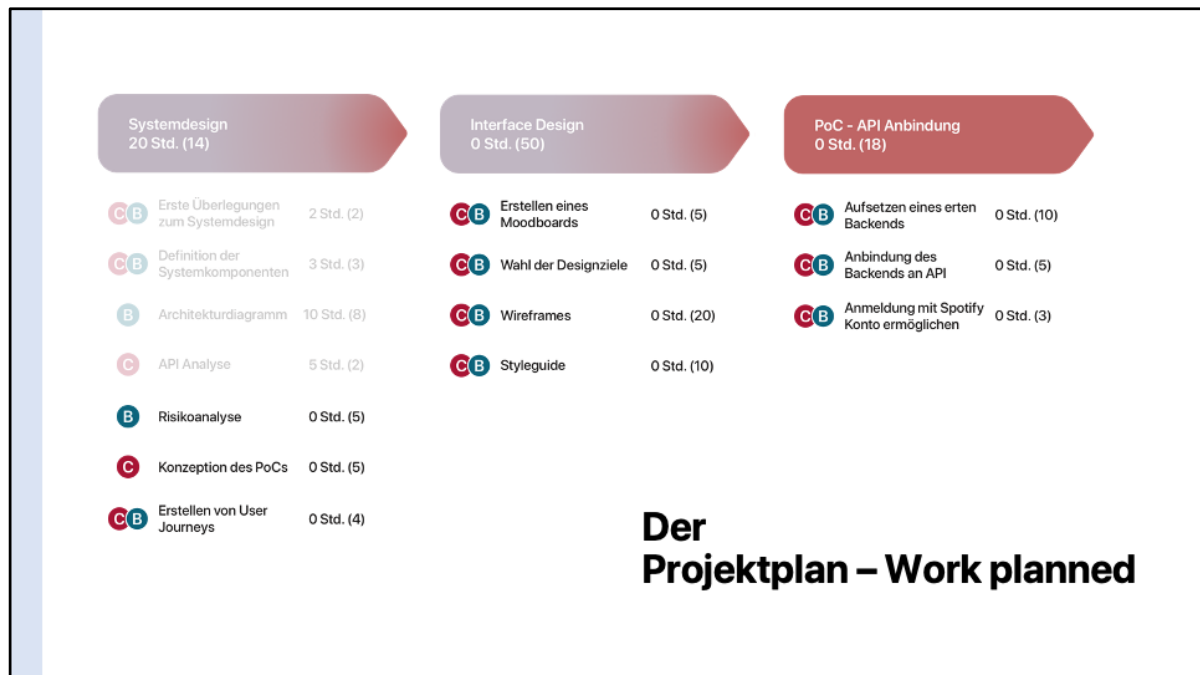
Implementierung
300 Std.

Dokumentation
120 Std.

Wie in den Anforderungen definiert wurde der Projektplan in drei Ebenen gegliedert. Die erste Ebene zeigt die grundsätzliche Einteilung der Zeit (in Personen Stunden) verteilt auf die Projektphasen der Konzeption, Implementierung sowie Dokumentation.



Auf der zweiten Ebene des Projektplans werden die jeweiligen Projektphasen in Subziele unterteilt. Hierbei findet ein fließender Übergang von der Konzeption in die Implementierung statt, während dessen wird die Dokumentation kontinuierlich fortgeführt. Die dritte Ebene des Projektplans zeigt die jeweils erledigten beziehungsweise geplanten Aufgaben. Die Badges links zeigen wer die Aufgabe erledigt hat beziehungsweise wem sie zu gewiesen ist (C steht für Carlos Bystron, B für Bogdan Krenz). Hinter der Aufgabe stehen die jeweils aufgebrauchten Stunden. In Klammern werden die Stunden aufgeführt, die ursprünglich für die Aufgabe eingeplant wurden.



Für das nächste Audit planen wir einerseits den Proof of Concept zu erarbeiten, andererseits wollen wir uns einer ersten Design Konzeption des User Interfaces widmen. Ein Übergang von der Konzeptionsphase in die Implementierung ist angedacht.



In diesem Sinne werden wir weiter mit Hochdruck an der Realisierung unseres Projektes arbeiten um zukünftig nur noch gute Musik auf Partys hören zu müssen 😊