



Delegator

Kenan Tabinas

delegator



HFINF

Klasse 21b / Praktische Diplomarbeit 2025

ID: 2558

1. Management Summary

Delegator ist eine App, die speziell für wenig technikaffine Musiker entwickelt wurde. Ziel ist es, Bands eine einfache, intuitive Plattform zu bieten, mit der sie Termine, Aufgaben, Ressourcen und Kommunikation an einem Ort verwalten können. Damit soll ein Problem gelöst werden, das viele kleine Bands betrifft. Fehlende Organisation ohne Zugang zu teuren Management-Dienstleistungen.

Die Diplomarbeit umfasst die vollständige Konzeption, Entwicklung und erste Testphase der App. Sie wurde nach dem Prinzip Design First gestaltet und mit modernen Technologien wie Django (Backend), Flutter (Frontend) und PostgreSQL umgesetzt. Der Betrieb erfolgt containerisiert, ursprünglich auf einem Raspberry Pi, später auf einem MacBook, mit einem geplanten Wechsel zu einem Managed Service (z. B. Exoscale) vor dem produktiven Einsatz.

Die Zielgruppe wurde bewusst auf Bands eingeschränkt, da der Familien- und KMU-Markt als zu gesättigt eingeschätzt wurde. Die App deckt typische Bandbedürfnisse von Kalender-Synchronisation über Rollen- und Rechteverwaltung bis zu Chatfunktionen. Die technische Architektur basiert auf einer Cross-Plattform-App mit Fokus auf einfacher UX.

Eine Marktanalyse und ein Businessplan zeigen auf, dass die App mit einem Freemium-Modell wirtschaftlich betrieben werden kann. Ein Break-even ist bei realistischer Nutzergewinnung nach ca. fünf Jahren möglich. Die App wurde erfolgreich getestet. Die Grobanforderungen und Ziele wurden grösstenteils erreicht.

Die Arbeit zeigt nicht nur ein funktionierendes Produkt, sondern auch das unternehmerische Potenzial dahinter. Ob das Projekt weitergeführt wird, hängt von Ressourcen, Marktinteresse und persönlichen Prioritäten ab.

Text Generiert von Claude.

2. Inhaltsverzeichnis

Contents

1.	Management Summary	2
2.	Inhaltsverzeichnis	3
3.	Initialisierung.....	6
3.1.	Ausgangslage.....	6
3.2.	Projektziele	7
3.3.	Projektorganisation	7
3.4.	Projektplan.....	8
3.5.	Lieferergebnisse	9
3.6.	Ressourcenplan.....	9
3.7.	Risiken.....	11
3.8.	Abgrenzung	12
3.9.	Studie	13
3.9.1.	Konkurrenzanalyse.....	13
3.9.2.	Zielgruppenanalyse	14
3.10.	Technologie	15
3.10.1.	Varianten	15
3.10.2.	Entscheidung.....	16
3.10.3.	Bewertung	16
3.10.4.	Fazit	16
3.11.	Nicht Entscheidungen.....	Error! Bookmark not defined.
3.12.	Service Stack.....	17
3.12.1.	Mögliche Varianten	17
3.12.2.	Break Even.....	18

3.12.3. Fazit	18
3.13. Wirtschaftlichkeit.....	19
3.13.1. Konkurrenz	19
3.13.2. Betriebskosten	19
3.13.3. Instagram Ads	20
3.13.4. Errechnete Neu-User	20
3.13.5. Preismodell.....	20
3.13.6. Breakeven	21
3.13.7. Mögliches Risiko	21
4. Konzept	22
4.1. Designkonzept.....	22
4.2. Berechtigungskonzept.....	24
4.3. Testkonzept.....	22
4.4. Datenschutzkonzept.....	22
4.5. Sicherheitskonzept	24
4.6. Datenbankkonzept	23
4.6.1. Projekt	23
4.6.2. User	23
5. Realisierung.....	25
5.1. Ausführung.....	25
5.1.1. Backend Umgebung.....	Error! Bookmark not defined.
5.1.2. Datenbank.....	Error! Bookmark not defined.
5.1.3. Projekt Rechte.....	25
5.2. Testprotokoll	26
6. Einführung	29

7. Schlussbetrachtung.....	31
8. Authentizität	32
9. Anhang.....	33

Das Inhaltsverzeichnis wird für das ganze Dokument (und nicht für jeden Phasenbericht einzeln) erstellt: Titelnummer (ca. 3 Ebenen); Titeltext; Seitennummer.

3. Vorwort

Organisation ist in vielen Lebensbereichen entscheidend – auch in der Musik. Gerade für kleinere Bands, die ohne professionelles Management auskommen müssen, stellt die Koordination von Terminen, Aufgaben und Ressourcen eine besondere Herausforderung dar. Aus eigener Erfahrung weiß ich: Kreativität gedeiht nur, wenn der organisatorische Rahmen stimmt.

Diese Diplomarbeit entstand aus dem Wunsch, genau hier anzusetzen. Ziel war es, eine App zu entwickeln, die wenig technikaffine Musikerinnen und Musiker im Alltag unterstützt – pragmatisch, intuitiv und ohne überladene Funktionen. Das Projekt verbindet technisches Know-how mit persönlicher Motivation und zeigt, wie aus einer Idee ein funktionierendes Produkt entstehen kann.

Natürlich lief nicht alles nach Plan. Wie sagt man so schön: Wenn ein Plan reibungslos funktioniert, war es kein IT-Projekt.

Text generiert von Claude.

4. Initialisierung

4.1. Ausgangslage

Das Leben in einer Band ist schwierig. Gerne möchte man vor allem Zeit in kreative Prozesse stecken. Große Bands haben einen Manager, der solche Aufgaben übernimmt. Der nimmt jedoch gerne 15% des Umsatzes. Leider sind kleine Bands weit von grünen Zahlen entfernt. Dazu kommt noch, dass eine kleine Band ohne Organisation klein bleibt.

Es existieren bereits Tools, welche in diese Richtung gehen. Diese sind aber eher für kleinere Unternehmen und Technik affine Menschen. Daher haben wir eine Lösung für Technikuninteressierte Musiker. Nach dem Motto «Weniger Admin mehr Musik». Eine Band ohne Organisation funktioniert nicht. Wenn wir schon nicht darauf verzichten können, soll es wenigstens so einfach wie möglich sein.

Musiker wollen nicht jedes Mal, um Termine zu sehen, in eine andere App wechseln müssen. Musiker wollen nicht für Dateien, Chats, Aufgaben und Kalender vier Separate Apps benutzen. Familien Mitglieder sollen Zugriff auf Kalender haben und über alles Wichtige informiert werden, ohne von unnötigen Informationen bombardiert zu werden.

Gäbe es doch nur etwas, welche diese Probleme zielgruppengerecht lösen könnte.

4.2. Projektziele

Folgende Ziele sollen bis Ende Projekt erreicht werden.

ID	Ziel	Beschreibung
Z001	Projekt	Ich kann Projekte erstellen
Z002	Aufgaben	Ich kann verschiedene Aufgaben in einem Projekt erfassen
Z003	Template	Ein Projekt Template erstellt mir Aufgaben
Z004	Termin	Ich kann ein Termin bei der Aufgabe definieren
Z005	Zuweisung	Ich kann Aufgaben einem User zuweisen
Z006	Kalender 01	Ich kann pro User ein Aufgaben Kalender definieren
Z007	Kalender 02	Ich kann auf einem Kalender der Zeit Rahmen des ganzen Projektes sehen
Z008	Kalender 03	Ich kann auf einem Kalender den Zeitrahmen der ganzen Unternehmung sehen
Z009	Kalender 04	Ich kann auf einem Kalender die Ressourcen eines Users bearbeiten
Z010	Kalender 05	Ich kann auf meiner Kalender APP meinen Kalender synchronisieren

Tabelle 1: Ziele

4.3. Projektorganisation

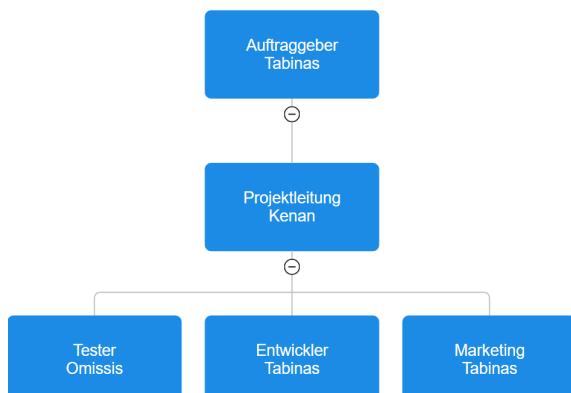


Abbildung 1: Projekt Organisation

4.4. Projektplan

Der Zeitplan befindet sich im Anhang.

	Datum	Datum	Soll	Ist	Abw.		Januar	Februar	März	April	Mai	Juni
Initialisierung							01	02	03	04	05	06
1.1 Projektwahl	06.01.2025	13.01.2025	8	8	0	erledigt	TAK					
1.2 Projektantrag	06.01.2025	13.01.2025	8	8	0	erledigt	TAK					
1.3 Aufbau einer Test Umgebung	13.01.2025	20.01.2025	16	16	0	erledigt	TAK					
1.4 Projektantrag	13.01.2025	20.01.2025	4	4	0	erledigt	TAK					
1.5 Projektinitialisierungsauftrag	13.01.2025	20.01.2025	4	4	0	erledigt	TAK					
1.6 Kick off Vorbereiten	27.01.2025	17.02.2025	16	16	0	erledigt	TAK					
1.7 Projektplan	03.02.2025	10.02.2025	4	4	0	erledigt	TAK					
1.8 Kick Off Meeting	10.02.2025	17.02.2025	2	2	0	erledigt	TAK					
2.1 Projektziele Definieren	17.02.2025	03.03.2025	8	8	0	erledigt	TAK					
2.2 Projektplan erstellen	03.03.2025	10.03.2025	8	8	0	erledigt	TAK					
2.3 Wirtschaftlichkeit Prüfen	03.03.2025	10.03.2025	8	8	0	erledigt	TAK					
2.4 Lösungsvarianten	10.03.2025	17.03.2025	8	8	0	erledigt	TAK					
2.5 Präsentation Vorbereiten	17.03.2025	24.03.2025	4	4	0	erledigt	TAK					
2.6 Zwischen Meeting	24.03.2025	31.03.2025	2	2	0	erledigt	TAK					
Konzept												
3.1 UX / Mockup	31.03.2025	07.04.2025	8	16	8	erledigt	TAK					
3.2 Security by Design	31.03.2025	07.04.2025	8	3	-5	geplant	TAK					
3.3 Architektur und Deployment	07.04.2025	14.04.2025	8	3	-5	geplant	TAK					
3.4 Datenbank Struktur	16.04.2025	23.04.2025	8	8	0	erledigt	TAK					
3.6 Berechtigungen			0	8	8	erledigt	TAK					
3.7 Testkonzept			0	8	8	erledigt	TAK					
3.8 Konzept Abgeschlossen	16.04.2025	23.04.2025	0	0	0	geplant	TAK					
Realisierung												
4.1 Backend Endpoints bereitstellen	21.04.2025	28.04.2025	16	16	0	erledigt	TAK					
4.2 Entwicklung von der Kalender Synchronisation	21.04.2025	28.04.2025	16	16	0	erledigt	TAK					
4.3 Erfassen von Projekten	28.04.2025	05.05.2025	16	16	0	erledigt	TAK					
4.4 Erfassen von Aufgaben	29.04.2025	06.05.2025	16	16	0	erledigt	TAK					
4.5 Chat	30.04.2025	07.05.2025	8	16	8	erledigt	TAK					
4.6 User Verwaltung	01.05.2025	08.05.2025	8	16	8	erledigt	TAK					
4.7 Realisierung Abgeschlossen	02.05.2025	09.05.2025	0	0	0	erledigt	TAK					
Einführung												
5.1 Release im Appstore	19.05.2025	26.05.2025	8	0	-8	geplant	TAK					
5.2 Dokumentation	26.05.2025	02.06.2025	8	36	28	erledigt	TAK					
5.3 Testdurchführung	02.06.2025	09.06.2025	8	8	0	erledigt	TAK					
5.4 Einführung Abgeschlossen	02.06.2025	09.06.2025	8	8	0	geplant	TAK					
Abschluss												
6.1 Drucken der Arbeit	13/06/2025	10/15/10	3	3	0	geplant	TAK					
6.2 Vorbereiten Präsentation	13/06/2025	10/15/10	8	8	0	geplant	TAK					
6.3 Abschluss Meeting	13/06/2025	10/20/10	4	4	0	geplant	TAK					
Total / bereits benötigt / Restliche Stunden:			259	309	50							

Abbildung 2: Projektplan

4.5. Lieferergebnisse

Aufzeigen der Ergebnisse, die während der Diplomarbeit erarbeitet werden bzw. wurden und einen massgeblichen Beitrag zur Erfüllung der Projektziele leisten.

ID	Ziel	Beschreibung
L001	Projekt	Ich kann Projekte erstellen
L002	Kanban Board	Ich kann verschiedene Aufgaben in einem Projekt erfassen
L003	Kalender	Ein Projekt Template erstellt mir Aufgaben

Tabelle 2: Lieferergebnisse

4.6. Ressourcenplan

4.6.1. Externe Kosten

Die externen Kosten wurden auf 0 geschätzt.

4.6.2. Interne Kosten

Basierend auf dem IST Zustand des Zeitplanes. Bei 120 CHF pro Stunde.

Phase	Zeit in h	Geplant in CHF
Initialisierung	100	12000
Konzept	32	3840
Realisierung	80	9600
Einführung	32	3840
Abschluss	15	1800
Total	259	31080

Tabelle 3: Interne Kostenschätzung

4.6.3. Umgebung

Damit wir die nötigen Ressourcen verstehen müssen wir die Architektur verstehen.

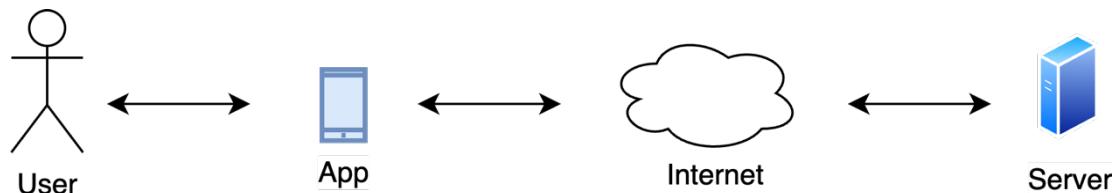


Abbildung 3: Einfache Darstellung

Am Anfang der Entwicklung lief der Server auf dem Raspberry Pi 5. Im Verlauf des Projektes bin ich auf ein M4 Pro Mac-Book gewechselt. Das System bleibt aber das gleiche.

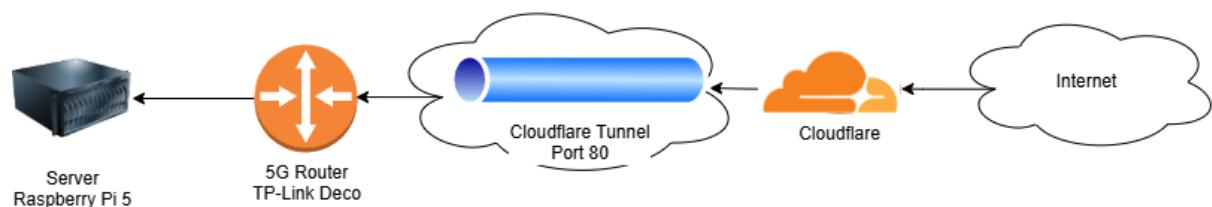


Abbildung 4: Server zu Internet

Es wurde sich zu diesem Zeitpunkt, keine Gedanken zu den Kosten der Umgebung gemacht.

Es laufen fünf verschiedene Container die je einen Zweck erfüllen.

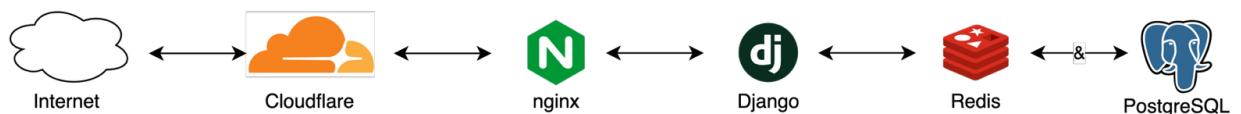


Abbildung 5: Darstellung der Container

4.7. Risiken

Bei der folgenden Auflistung gibt es einige der möglichen Risiken, um sich auf das Wesentliche zu begrenzen.

NR	Beschreibung	Eintritts-wahrscheinlichkeit	Auswirkung	Risiko
1	Backend System Ausfall	2	2	4
2	Cyber Angriffe	3	3	9
3	Fehler in der Mobileapp	2	3	6
4	Ausfall von Plattform Services	2	2	3
5	Konkurrenzverhalten	1	3	6
6	Datenschutzverletzung	2	2	4

Tabelle 4: Risiko Analyse

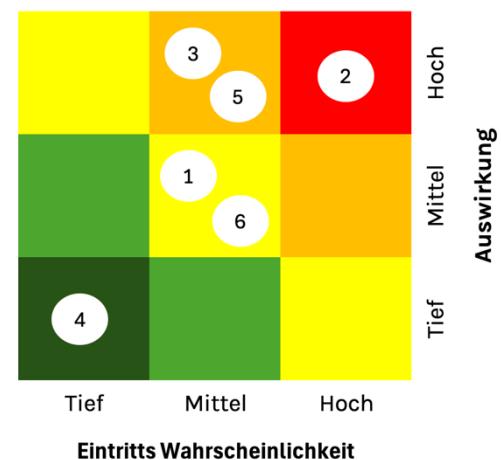


Abbildung 6: Risiko Matrix

4.8. Abgrenzung

In der Folgenden Grafik wird die Grenze des Projektes visualisiert. Alle Elemente im grünen Zylinder sind Teil vom Projekt. Der Fokus lag auf Sicherheit, der Userexperience, Design, Konzepten und einem proof of concept. Der tatsächliche Release der App und Vermarktung, kommt nicht nur mit einem hohen Zeitaufwand, sondern auch mit einem finanziellen Risiko einher welches, während dem Projekt nicht eingegangen wurde.

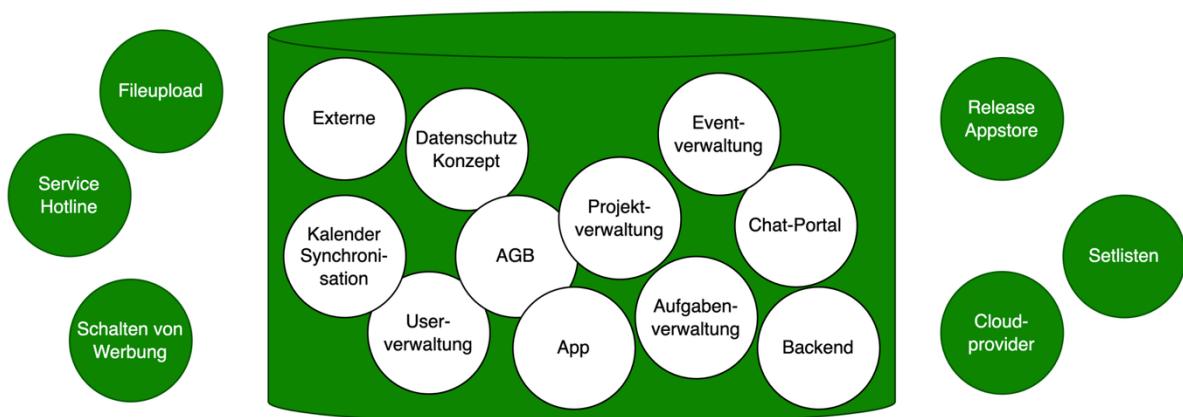


Abbildung 7: Abgrenzung

4.9. Studie

Im Folgenden wird nur auf die wesentlichen Teile der Studie eingegangen.

4.9.1. Konkurrenzanalyse

Die Frage, die sich am Anfang stellt, steige ich bereits in einen übersättigten Markt ein oder ist noch was zu holen. Bei der Analyse hat sich ergeben, dass es bereits etablierte Player auf dem Markt gibt. Hier folgt eine stark heruntergebrochene Zusammenfassung und das Fazit davon.

App	Preis	Feature Abdeckung
Trello	Kostenlos (begrenzte Features) 5 CHF pro User pro Jahr	Mittel
Asana	Kostenlos (bis 15 User) 11 CHF pro User pro Jahr	Hoch
Clickup	Kostenlos (begrenzte Features) 7 CHF pro User pro Jahr	Hoch
Google Calender	Kostenlos	Tief

Tabelle 5

Asana und Clickup erschienen als Dorn im Auge. Jedoch ist Clickup komplett überladen. Die Einstellungsmöglichkeiten scheinen schier unendlich. Das mag zuerst wie ein Vorteil klingen, jedoch ist es auch ein Nachteil. Daraus haben sich folgende Erkenntnisse ergeben:

- Positionierung: Informatik Uninteressierte haben andere Bedürfnisse als der Entwickler. Die Applikation muss mit Einfachheit herausstechen.
- Die Zielgruppe «kleine Unternehmen» ist bei dieser Konkurrenz komplett aus dem Rennen.

4.9.2. Zielgruppenanalyse

Bei der Vorstellung des Projektes war die Primäre Zielgruppe Musiker. Marc Abby und Ralph Maurer sprachen davon, dass dieses Produkt einen Nerv trifte. Dieser wird auch für Familien und kleinere Unternehmen interessant sein.

Diese Anmerkungen haben sich in Studie wieder gefunden. Ohne auf alle Details einzugehen hat sich durch die Zielgruppen Analyse folgendes ergeben.

In der Studie stehen Quellen und Rechenwege für folgende Berechnungen. Diese Zahlen beschränken sich auf den DACH-Raum (Deutschland, Österreich und Schweiz)

Zielgruppe	Möglicher Markt
Bands	ca. 148 000 Bands
Familien	ca. 39 000 000 Mehrpersonen Haushalte

Tabelle 6

Auf den ersten Blick sind Familien. Respektive Mehrpersonen Haushalte, als Zielgruppe interessanter. Bei einer weiteren Analyse hat sich aber folgendes ergeben:

- Die Zielgruppe «Familien» ist 24-mal grösser.
- Es gibt bereits über 10 Apps im Familien Bereich.
- Die App OurHome ist kostenlos und hat eine hohe Features Abdeckung.
- Im Band Bereich gibt es nur 2 mit tiefer Feature Abdeckung

Folgende Entscheidung wurde getroffen:

Zielgruppe	Hinweis
Bands	Die Zielgruppe der App
Familien	Übersättigter Markt
Kleine Unternehmen	Ist bei der Konkurrenz Analyse herausgeflogen

Tabelle 7

4.10. Technologie

In der Studie musste noch geklärt werden, welchen Technologie Stack gefahren wird. Dank der beiden vorgängigen Analyse ergaben sich Punkte, die bei der Entscheidung massgebend waren.

Folgende Punkte wurden beachtet:

- Die Userexperience muss überzeugen
- User möchten nicht das Notebook zücken, um ein Konzert einzutragen

4.10.1. Varianten

Folgende Technologien Varianten standen zur Auswahl:

- V1: Web (keine App)
- V2: Web-Viewer-App
- V3: X-Plattform App
- V4: Native App

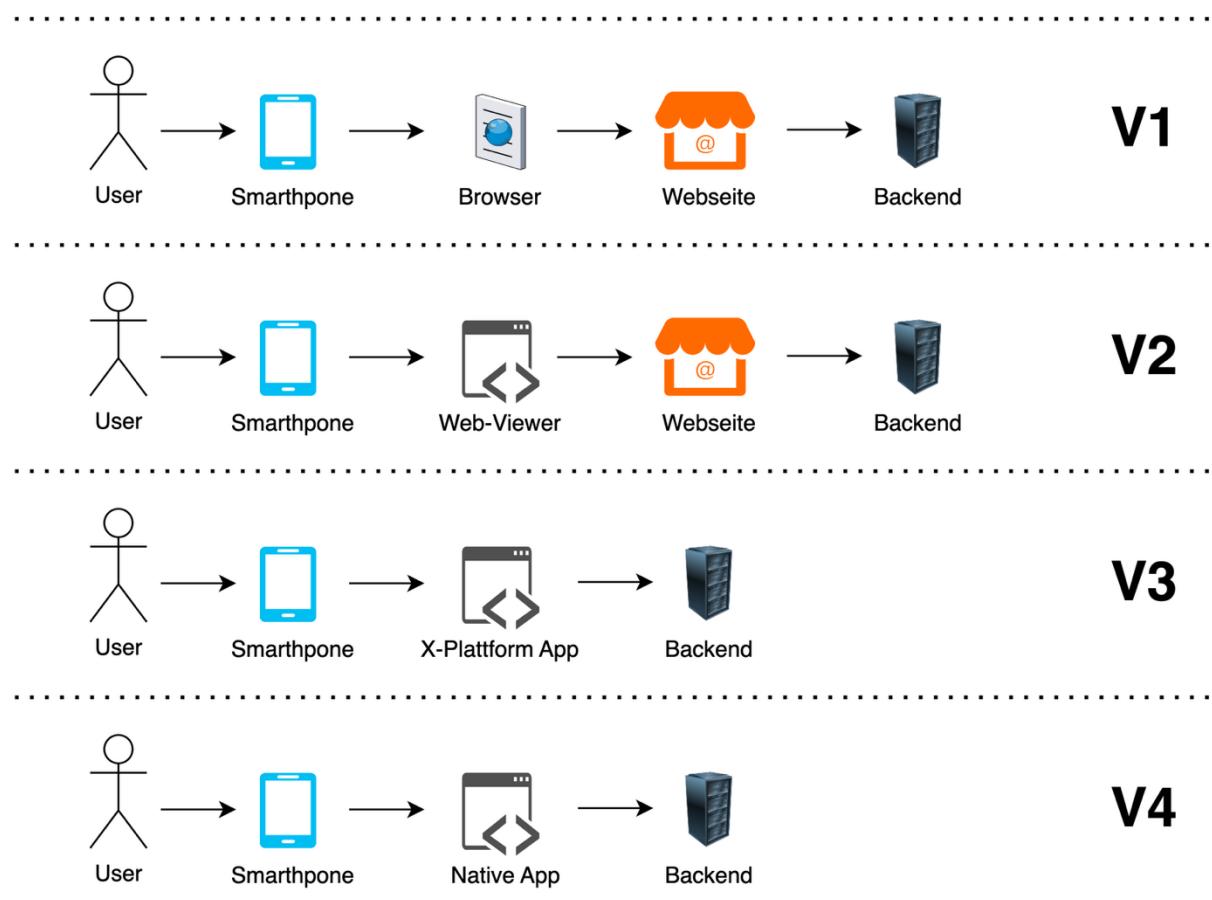


Abbildung 8: Technologievarianten

4.10.2. Entscheidung

Der gesamte Variantentscheid befindet sich im Anhang Studie. Um die Entscheidung nachvollziehen zu können muss man wissen, dass Web-View Apps zwar schnell in der Entwicklung sind und Vorteile der Nativen Entwicklung haben, jedoch verstossen sie etwa gegen Appstore Richtlinien vom Goolge Playstore und Apple Appstore. Zusammenfassend kann man sagen wegweisend waren die vorgängigen Analysen. Diese haben der Userexperience einen hohen Stellenwert zugesprochen.

4.10.3. Bewertung

Alle Kriterien werden von 1 bis 3 bewertet. Wobei 3 am besten ist und 1 am schlechtesten.

Kriterium	WebOnly	WebView	Native	X-Plattform
Entwicklung	3	3	1	2
Updates	3	3	1	3
Appstore	1	2	3	3
UX	1	2	3	3
Total	8	9	8	10

Tabelle 8: B2 Bewertung

4.10.4. Fazit

Die WebOnly ist wegen der fehlenden App rausgeflogen. Bandmitglieder möchten nicht jedes Mal das Notebook aufklappen, um zu sehen, wann die nächste Probe ist.

Die Native App ist wegen dem Entwicklungsaufwand rausgeflogen. Es braucht zu viel Zeit und Geld mehrere Apps zu entwickeln und zu Unterhalten.

Bei genauerem Hinsehen sind die WebViewApp und die Cross-Plattform nahe beieinander. Für die Entwicklung von WebView Apps bietet sich die Verwendung von Cross-Plattform-Framework an.

Die User Experience liefert die finale Entscheidung. Die User Experience auf einer Browser ähnlichen App ist nicht dieselbe wie bei einer Nativen App. Somit können wir die UX und Qualität steigern, indem wir uns auf ein Segment limitieren.

4.11. Deployment Architektur

In der Studie wurden folgende Deployment-Architekturen (CD/CI) und Anwendungs-Architekturen verglichen. Im Wesentlichen ging aus der Nutzwertanalyse hervor, dass sich **Hybridarchitektur mit Rolling Deployment** am besten für das Projekt eignet. Stark gewichtet wurden die Entwicklungskosten und Betriebskosten, da es sich um ein Ein-Mann-Projekt handelt.

Architektur	Beschreibung
Monolith	Alle Komponenten in einem einzigen Deployment-Paket.
Microservices	Aufgeteilt in Services (User-Service, Project-Service usw.)
Orchestrierung	Container-Lösung mit Kubernetes Management
Serverless	Function-as-a-Service (Cloud-Provider)
Hybrid	Monolith mit klarer modularer Trennung, Container-basiert

Tabelle 9: Anwendungs-Architekturen

Deployment	Beschreibung
Blue-Green	Zwei Umgebungen welche sich Abtauschen
Rolling	Mehrere Container werden schrittweise geupdated
Canary	Zwei Container, z.B: 5%, läuft bereits auf Version B.

Tabelle 10: Deployment-Architekturen

4.12. Service Stack

Wie bereits in der Abgrenzung illustriert wurde, ist die Bereitstellung des Services nicht ein Teil der Arbeit. Trotzdem wurden dazu ausführlich Gedanken gemacht. Diese hier aufzulisten, würde den Rahmen sprengen.

Es wurde ausgiebig verschiedene Provider wie auch Hardwarekombinationen angeschaut. Vom High-Tech Serverrack mit Rauchmelder bis zu managend all-in-one Lösung war alles dabei.

4.12.1. Mögliche Varianten

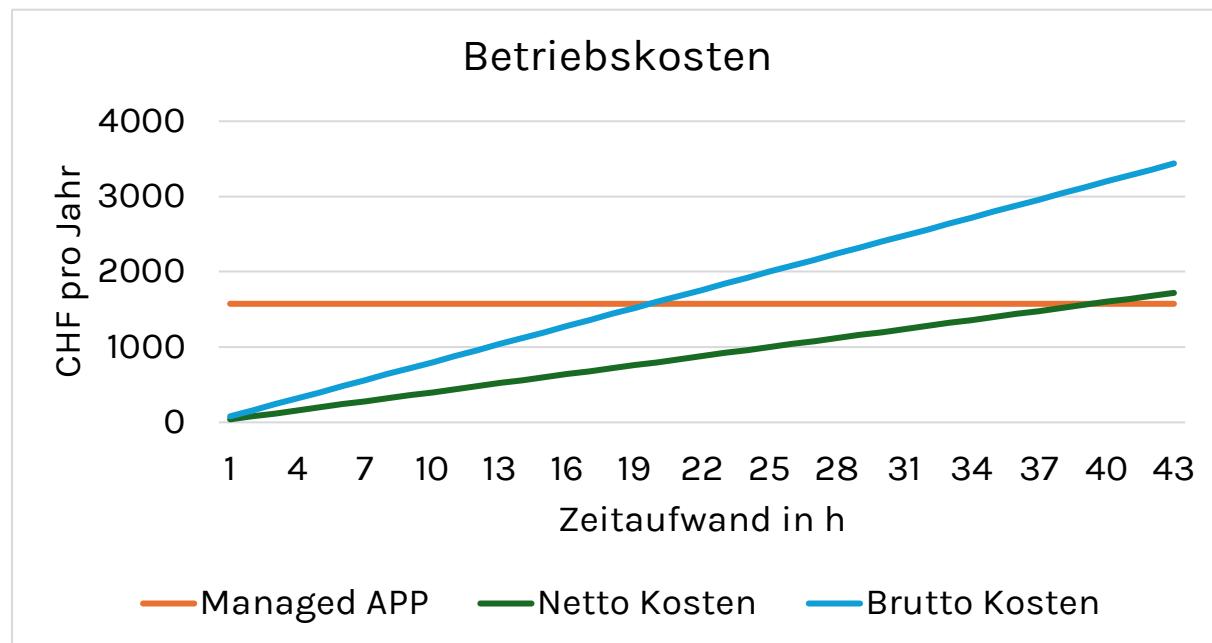
Variante	Bezeichnung
V1	On-Premise
V2	VPS
V3	CaaS

Tabelle 11: Service Stack

4.12.2. Break Even

Die Berechnung basierte auf folgendem Gedanken. Es besteht die Möglichkeit, dass ich dieses Projekt ausserhalb der Diplomarbeit weiterverfolge. Nehmen wir an ich bleibe zu 80% festangestellt. Die anderen 20% investiere ich in den Betrieb und Weiterentwicklung dieser Software.

Wenn ich mit meinem Lohn rechne, bin ich bei 40 CHF pro Stunde. Wenn ich aber Abzüge selbst bezahlen muss bin ich eher bei 80 CHF pro Stunde.



4.12.3. Fazit

Bisher war ich davon überzeugt, dass nichts günstiger ist als On-Premise. Da ein Managed Container bei 1500 CHF pro Jahr anfängt. Wenn man aber einrechnet, dass ich auch alles selbst machen muss. Logs, Backup, Sicherheit usw. Darf ich nur 20 - 40h Pro Jahr aufwenden, um noch günstiger zu sein. In Studie habe ich mehrere Szenarien aufgelistet, wie der Support meiner eigenen Software aussehen könnte. 20h im Jahr ist utopisch. Daher ist die Entscheidung klar auf einen Managed Service gefallen.

4.13. Wirtschaftlichkeit

4.13.1. Konkurrenz

Damit wir Zahlen zur Berechnung haben, habe ich die Preise der Konkurrenz angeschaut. Dabei ergaben sich die 7 CHF pro Monat

Modell	Preis pro Monat in CHF	Preis im Jahr in CHF	Anzahl Kunden für Betriebskosten
CoziFamily	5	60	27
Band Helper Pro	9	108	15
Schnitt	7	84	19

Tabelle 12

4.13.2. Betriebskosten

In der Studie ist detailliert aufgelistet und beschrieben wie es zu folgenden Zahlen gekommen ist. Relevant für das Verständnis, sind die 10% für das Marketing. Diese kommen aus einer Empfehlung für startende Unternehmen.

Folgende Zahlen beziffern die Kosten der Software im ersten Jahr.

Position	CHF	%
Diplomarbeit	16400	29
Aufwand	33280	59
Marketingbudget	6000	10
Server	1570	2
Total	57250	100

Tabelle 13

4.13.3. Instagram Ads

Im Folgenden ist eine Aufstellung basierend auf dem neuen Budget von 10%. Leider gibt es 16 CHF nicht als Tages Rate bei Instagram. Daher habe ich mich für 18 CHF entschieden. Daher ist das Budget auf 11% gestiegen.

Budget pro Jahr (CHF)	Budget pro Tag (CHF)	User Reach (Min)	User Reach (Max)
6570	18	1200	3200

4.13.4. Errechnete Neu-User

In der Studie ist genau erklärt, wie es zu folgenden Zahlen kommt. Es gibt noch andere Plattformen als Instagram. Diese Berechnung gibt uns ein Gefühl dafür, wie lange es dauert.

Case	Reach pro Jahr	Clicks	Neue User
Worst	438'000	6570	131
Best	1'168'000	23360	467

Tabelle 14: Instagram CPA bei 11%

4.13.5. Preismodell

Der Gedanke war, der User hat die Möglichkeit die App zu Testen und zu schauen, wie sich die App anfühlt. Das Preismodell bietet einen Rahmen, ohne alle Features gleich weg zu geben. Es ist ein Premium Feature für welttourende Bands, welche eine hohen Wert auf Sicherheit legen. Diese würden auf einem separaten Server einen Zugriff erhalten.

Allein Benutzen Kostenlos	Zusammen pro Jahr 72 CHF pro Band (Ein Monat Rabatt)	Zusammen pro Monat 7 CHF pro Band
------------------------------	--	--------------------------------------

Tabelle 15: Preimodell

4.13.6. Breakeven

Basierend auf den Betriebskosten, dem Marketingplan und dem Preismodell besteht die Möglichkeit das die Applikation innerhalb von 5 Jahren grüne Zahlen schreibt.

Jahr	Kosten (TCHF)	User	Gewinn (TCHF)	Saldo
1	58	131	-48	-48
2	41	263	-22	-71
3	41	394	-13	-36
4	41	526	-4	-17
5	41	657	6	2
6	41	788	15	21
7	41	920	25	40

Tabelle 16: Breakeven Studie

4.13.7. Mögliches Risiko

Nehmen wir an, ich rechne meine Zeit nicht mit ein. Ich rechne nur Kosten, welche ich an für Werbung und Server zahle. Im Worstcase bezahle ich im ersten Jahr 7000 CHF ohne etwas zurückzubekommen.

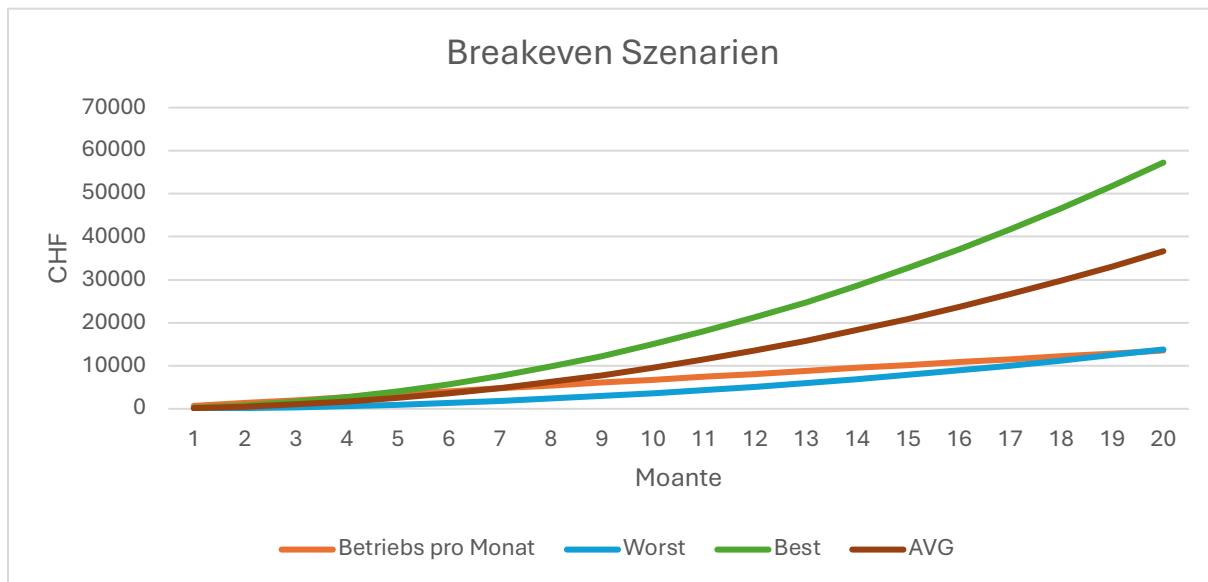


Abbildung 9: Breakeven Kosten

5. Konzept

5.1. Designkonzept

Der Design Prozess ist vollständig dokumentiert und im Anhang zu finden. Folgende drei Punkte sind relevant.

Es wurden alle möglichen Features, welche die App einmal haben könnte, geplant und haben einen festen Platz im Design bekommen. Welche Features umgesetzt wurden folgt später im Realisierung Teil.

Es wurde ein «Design First» Ansatz gewählt. Das Design war der erste Schritt und hat alle zukünftigen Schritte definiert.

Die Wireframes aus Design haben sehr bei der Entwicklung geholfen. Positiv anzumerken ist, wie nahe dem Endresultat am Design ist. Darauf wird aber noch später im Realisierung Part eingegangen.

5.2. Datenschutzkonzept

Für die Planung des Releases wurde ein Datenschutz Konzept erstellt. Dieses ist im Anhang zu finden ist. Die Wesentliche Punkte daraus sind:

- Betroffene haben Rechte (Einsicht, Deaktivierung und Vergessenwerden)
- US-Cloud Prodivder sind nach Schermen II nicht Rechtskonform.
- Ich habe eine Verantwortung und hafte auch dafür.
- Es braucht eine Roadmap, damit es nicht verloren geht.

5.3. Testkonzept

Für das Testen wurde ein Testkonzept erstellt. Dieses ist im Anhang zu finden ist. Die Wesentliche Punkte sind die Verschiedenen Teststrategien.

Nr.	Teststrategie	Beschreibung
01	Contract Testing	Test der API mit Postman
02	Integrationstest	Verknüpfung von Frontend und Backend mit Flutter-Unit-Tests
03	Anforderungsabdeckung	Testfälle basieren auf Anforderungen aus der Initialisierungsphase
04	Usability-Test	Ungeübte Benutzer*innen führen Use-Cases ohne Anleitung aus
05	Security-Test	Überprüfung der Endpoints auf Schwachstellen

Tabelle 17: Teststrategien

5.4. Datenbankkonzept

Im Anhang gibt es ein Marmaid generiertes Diagramm von Claude. Das ist aber keine Hilfe bei der Verständlichkeit. Um die Tabellen zu verstehen, beschränken wir uns auf das Wichtigste und unterteilen sie in zwei Kategorien.

5.4.1. Projekt

Der Kern der Datenbank ist das Projekt. Alle Detaileinträge sind modular und besitzen die Projekt_Id. So braucht es keinen Typ. Das Generalisieren / Spezialisieren wurde so umgangen.

Tabellennamen	Hinweis
Setlist	Song und Reihenfolge.
Timetable	Ein einzelner Eintrag z.B. 17:00 Ausladen.
Task	Kann beliebig viele zum Projekt hinzugefügt werden.
Chat	Wird automatisch erstellt, sobald das Projekt erstellt wird.
Recording	Ist eine einzelne Audio-Datei.
Mood	Ein Mood ist ein Eintrag in einem Moodboard. Also wie eine Impression in Form von meistens Fotos.

Tabelle 18: Projekt Datenbank

5.4.2. User

Die Zweite Hälfte sind die User. Die folgenden Tabellen sind auch per 1-n verbunden.

Tabellennamen	Hinweis
Externe	User welche zu einzelnen Projekten hinzugefügt wurden
UserOrganisation	Zwischentabelle zwischen User und Organisation
Message	Jeder Message hat auch einen User
History	Jede Aktivität hat auch einen User
Task	Jede Aufgabe kann einem User zugewiesen werden

Tabelle 19: User Datenbank

5.5. Sicherheitskonzept

Im Anhang wird im Dokument «Design by Security» ausführlich darauf eingegangen. Im Wesentlichen ergab es Konkrete Massnahmen, geplante Massnahmen und einige Aha Momente:

- Bei Gewisse Massnahmen reicht die Verwendung von Middleware
- Nur auf Cloud Provider setzen verstösst gegen 3-2-1-1-0

5.5.1. Konkret Dokumentierte Massnahmen

Layer	Massnahmen
APP	Certificate Pinning (via Middleware) Flutter Secure Storage (via Middleware) Input Validation (via Regex)
Cloudflared	DDoS Protection (automatisch) Bot Management (automatisch) Rate Limiting (automatisch)
Django	CSRF Protection (via Middleware) SQL Injection Prevention (via Django ORM) Secure Cookie (via Middleware)
PostgreSQL	Field-level encryption (für z.B. Passwörter) Service User mit Least Privilege

Tabelle 20

5.5.2. Geplante Massnahmen gekürzt

Security Controls	Geplante Umsetzung
Angriffsfläche minimieren	Cloudflared Zero Trust Tunnel ermöglicht Verbindung, ohne einen einzigen Port zu öffnen.
Verschlüsselung	Inter-Container Verschlüsselung mit z.B. Isito
Authentifizierung	MFA für Logins Risk-based Auth (Standort-Anomalien, Gräte Profiling) z.B. loginradius
Monitoring	Zentralisiertes Log mit z.B: ELK Stack
Compliance	Externer Penntest ins Budget einfließen lassen Jährliche Audits

Tabelle 21: Security Controls

6. Realisierung

6.1. Ausführung

Das Projekt ist nicht ohne Rückschritte passiert. Zu einem Zeitpunkt wurde 2 Tage Arbeit gelöscht. Nicht weil kein Backup existierte, sondern weil mit sämtlichem Hilfsmittel es nicht möglich war einen Fehler zu finden. Das Animierte zu einer Professionellen Methodik, welche verhalf reibungsfrei am Projekt weiterzuentwickeln. Im Folgenden wird auf die Berechtigungen und die drei Hürden welche, zu drei grossen Aha-Momente geführt haben.

6.1.1. Projekt Rechte

Was am Anfang zur Verwirrung beigetragen hat, ist die Tabelle UserProject. Man könnte meinen diese listet alle Projektteilnehmer auf. Jedoch wurde sich während der Design Phase gegen eine kompliziertes Rollen- und Berechtigungssystem entschieden. Lustigerweise ist auch eines daraus geworden, jedoch nicht für den User.

Wenn alle Band-Mitglieder zu jedem Projekt hinzugefügt werden müssen, füllt es die Tabelle UserProjekt mit immer gleichen Einträgen. Diesen Overhead wurde durch ein anderes System ersetzt.

Alle User werden via Tabelle User Organisation der Organisation hinzugefügt. Sie können mit 5 Verschiedenen Rollen hinzugefügt werden. Sobald sie Teil der Organisation sind, erhalten sie automatische Zugriffe.

ID	Rolle	Automatischer Zugriff
1	Admin	Alles
2	Long-Term member	Projekte, Verträge, Geld, User Verwaltung
3	Member	Projekte
4	Familie	Kalender Sychronisation
5	Fans	Zugriff auf öffentliche Kalender
6	Externer	Keine (Mixer*in, Videograph*in, Photograph*in.)

Tabelle 22: Zugriff

Die Tabelle UserProject ist nur für externe. Dies hat lange für Verwirrung gesorgt und wurde daher namensgebend zu Externe unbenannt. Dies war eine weitere Lektion dieses Mal im Thema Namensgebung.

Die geübte Datenbank Spezialist*in fragt sich jetzt wieso es eine UserChat Tabelle gibt, wenn alle Rechte über das Projekt laufen. Diese Frage habe ich mir auch ausgiebig überlegt. UserChat ist nicht zum Hinzufügen, sondern zum Entfernen. So kann ich mich selbst aus dem Chat entfernen oder den Chat Stumm stellen.

6.1.2. Backend

Das Backend wurde mit Python und dem Framework Django umgesetzt. Die Struktur ist daher im Voraus definiert und bietet keine grossen Freiheiten. Hier ist eine gekürzte Auflistung der wichtigsten Files. Es wurde ein Code First Ansatz gewählt. Also die Datenbank wird anhand der Modelle definiert.

File	Inhalt
models.py	Hier werden alle Datenbank Tabelle als Klasse definiert.
serializers.py	Hier wird definiert, wie das Model verarbeitet wird.
views.py	Hier wird unter anderem Zugriff eines Serialisierst geregelt.
urls.py	Hier wird die View als Endpoint bereitgestellt.

Tabelle 23

Folgende zwei Ergebnisse sind nennenswert aus diesem Teil:

- Bei der Entwicklung wurden fortlaufend alle Endpoints mit Postman getestet. Dieses Vorgehen hat sehr gut funktioniert. Dazu mehr im Testprotokoll.
- Erst während der Entwicklung ist mir Aufgefallen, dass ich gar kein Berechtigungssystem eingeplant habe. Wieder ein grosser Lernmoment.

6.1.3. Frontend / App - Pattern

Ohne klarem Vorgehen hier begonnen. Das hat sich gerächt. So musste mit einer besseren Methodik neu begonnen werden. Im Anhang Realisierung ist eine Nutzwert Analyse verschiedener Design Patterns. Diese hat ergeben, dass das MVVM-Pattern in diesem Projekt sich am besten eignet.

Directory	Inhalt
services	Hier sind alle Operationen zwischen Endpoints und Models
models	Alle Klassen basierend auf den Endpoint JSONs
widget	Hier wird Darstellung der Models festgelegt. (ViewModel)
views	Hier werden die ViewModels in Pages dargestellt.

Tabelle 24

6.1.4. Frontend / App - Fazit

Beim zweiten Anlauf wurde für jedes Model ein Service und ein Integration Unit-Test geschrieben. So konnte sichergestellt werden das alle Services funktionieren, bevor es weiter geht.

Folgende zwei Ergebnisse sind nennenswert aus diesem Teil:

- Unit Test haben sich als sehr praktisch erwiesen
- Methodisches Vorgehen hat sich bewährt und ergab ein Aha-Erlebnis

6.1.5. Kalender Synchronisation

Bei der Kalender Synchronisation gab es Probleme, welche aber nicht unlösbar waren. Google Calender synchronisiert nur alle 24h externe Kalender, dies kann nicht manuell angeworfen werden. Sie verweisen auf die API, welche sie bereitstellen. Anders als bei Apple Calender, hier lässt sich hier das Intervall auf 5min einstellen. Das Problem liess sich lösen mit einer App namens ICSx5. Hier kann man zu einem Freien Intervall die Synchronisation anstoßen.

Damit ich dies auf dem Smartphone testen kann, reicht es nicht mehr die App lokal zu Testen. Spätestens jetzt erreicht die Container Landschaft seine aktuelle Form. Denn es gibt ein Problem mit 5G Routern, GCNAT und DDNS. Die Lösung ist detailliert in der Studie beschrieben. Die Lösung, welche genutzt wurde, ist der Cloudflare Tunnel und nginx Server.



Abbildung 10: Finale Container Architektur

6.1.6. Kanban

Der Rest der Realisierung ist reibungslos abgelaufen. Alle Aufgaben wie auch Notizen habe ich auf dem «Konkurrenten» Clickup festgehalten.

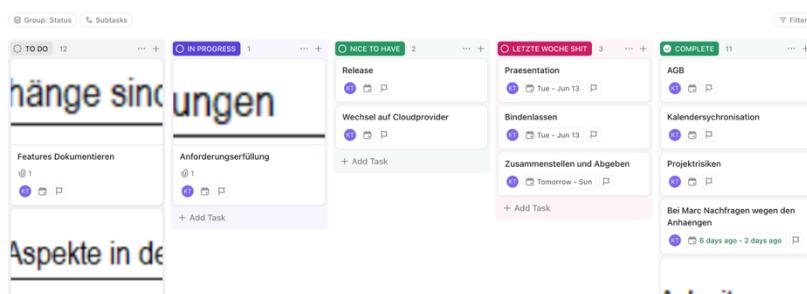


Abbildung 11: Kanban Board

6.2. Testprotokoll

Es gab mehrere Testdurchführungen. Diese wurden im Testprotokoll im Anhang festgehalten.

6.2.1. User- und Abnahmetest

Basierend auf Anforderungen der Applikation, wurde bereits in der Studie Abnahmekriterien definiert. Basierend darauf wurden Testfälle entwickelt. Diese Testfälle wurden nun einem User ohne Vorkenntnisse der Applikation auferlegt. Das Ziel war auch zu sehen, ob die Applikation selbsterklärend ist. Dies ist gegückt. Der User konnte alle Ergebnisse, ohne die Testschritte zu kennen ausführen.

Alle Muss Anforderungen wurden umgesetzt und konnten erfolgreich getestet werden. Es gab Abnahmekriterien, welche bis zum Ende nicht mehr gepasst haben. Die Templates sind aufgrund der Datenbank Architektur herausgeflogen.

6.2.2. Penetrationstests

Es wurden mehrere Penetration Test ausgeführt. Hier ist nur ein Auszug. Die Abdeckung ist auch noch zu klein. Es zeigt sich bisher, dass der Design by Security hat sich bewährt. Der Test auch noch gezeigt, beim einem Produktiven Start muss noch das Debug Modus beendet werden.

Test	Mängelklasse
Token - Geschützte Endpunkte ohne Token	0
Token - JWT Token Manipulation	0
Token - Zugriff auf fremde Daten	0
Premium - Admin-Funktionen für normale User	0
SQL Injection Tests	0
Grenzwerte und Datentypen	0
iCal Token Sicherheit	2
Error Message Information Leakage	4

Tabelle 25

7. Einführung

7.1. Wechsel auf Cloudprovider

Basierend auf dem Datenschutzkonzept und den Überlegungen aus der Studie wäre es für die Applikation am besten auf eine Managed-DB zu wechseln. Respektive auf einen DBaaS wie es bei Exoscale.com heisst. Ein Schweizer Cloud Provider mit Geo-Redundanz, Backups, Monitoring und eine Uptime von 99.9%. Das kann ich nicht für 1100 CHF bieten, dafür hätte ich länger als 13h pro Jahr. Der Wechsel muss einfach vor dem Release geschehen, um eine mühselige Migration zu umgehen. Sobald die Verbindung steht, können die Credentials im settings.py geändert werden und mit einem Command alle Migrationen wiederholt werden.

7.2. Betriebliche Umsetzung

Für den Produktionsbetrieb der Wechsel auf einen Managed Service (z. B. Exoscale) vorgesehen. Folgende Massnahmen ergaben sich aus dem Security by Design Konzept:

- 4h Backup der Datenbank (DBaaS bei Exoscale) mit 24-h-Retention.
- Disaster-Recovery-Strategie mit Backup-Restore-Test alle 6 Monate.
- 3-2-1-1-0 Strategie mit Scaleway Objectstorage und Air gapped Backup.
- Pentesting-as-a-Service jährlich durch externe Stelle (siehe Controlling).

Diese Massnahmen stellen sicher, dass im Falle eines Systemausfalls oder Datenverlusts der Betrieb innerhalb von Stunden wiederhergestellt werden kann.

7.3. Appstore Release

Das Endziel ist der Release in den Appstores. Dieses Ziel ist in den Hintergrund gerückt als ich mich zwei Wochen vor Abschluss der App von meiner Band trennte. Beim Versuch ein Account bei Goolge Console zu erstellen, muss man leider eine «richtige» Kreditkarte besitzen. Ich mit meiner Postfinance Mastercard konnte hier den Schritt noch nicht wagen. Den Aufwand ein neues Bankkonto zu erstellen, ist eine Aufgabe nach der Diplomarbeit. Der Applestore zwingt eine Feature List zu erstellen. Deshalb ist diese auch in der Realisierung gelandet.

7.4. Controlling

7.4.1. Nachkalkulation

Neu eingeflossen ist ein Jährliches Pentesting (Pentesting as a service) von einer Externen stelle. Das Marketingbudget ist bei 10% geblieben

Ist Kosten	CHF	%
Diplomarbeit	37080	34
Aufwand	49920	46
Marketingbudget	11000	10
Backup	43	0
Server	1570	1
Pentesting	10000	9
Total	109570	100

Tabelle 26: Nachkalkulation

Der grösste Unterschied, hier wurde mit 80 CHF anstelle von 120 CHF in der Stunde gerechnet. Die Diplomarbeit war noch 250 anstelle von 300. Backup und Pentesting war noch nicht dabei.

Soll Kosten	CHF	%
Total	57250	100

Tabelle 27

7.4.2. Break Even

Break Even ist immer noch im 5ten Jahr. Liegt am 10% Marketing hebel.

Jahr	Kosten (TCHF)	User	Umsatz (TCHF)	Saldo (TCHF)
1	109	438	31	-78
2	182	876	94	-87
3	254	1314	189	-65
4	327	1752	315	-11
5	399	2190	473	73
6	472	2628	662	190
7	544	3066	883	338

Tabelle 28

8. Schlussbetrachtung

Ich habe sehr viel gelernt. Ich froh habe ich das gemacht. Wenn ich nicht auf Marc gehört hätte, hätte ich noch gar nicht angefangen. Ich bin froh kann ich sagen, das nächste Mal bin ich klüger. Bei der Realisierung haben sich die «Fehler» als tolle Aha-Erlebnisse herausgestellt. Die Konzeptphase erscheint mir im Nachhinein am wichtigsten, war aber von der Aufteilung her etwas zu kurz. Die ist in der Planung etwas kurz gekommen. Datenschutzkonzept und Sicherheit-Konzept haben mich sehr bereichert. Testen hat mich auch positiv überrascht. Gefallen hat mir auch, eine Planung zu machen und am Ende die Unterschiede zu vergleichen. Sehr hilfreich war es, dass ich für das Thema Band und Musik brenne.

Dokumentweise sind sicher 95% von mir selbst. Die 5% habe ich immer deklariert. Beim Code ist es etwa umgekehrt. Code besteht etwa 95% aus Claude. Jedoch habe ich meine 5% nicht deklariert. Abbildungen sind auch alle von mir selbst, Urheberrecht gehört dem entsprechend mir.

Ob das Projekt weiter geht, steht noch in den Sternen. Ich möchte das Finanzielle Risiko nur eingehen, wenn ich an die App glaube. Dank der Arbeit weiß ich, gibt es noch diverse Baustellen. Die Zukunft zeigt noch, ob ich Lust habe das Gebäude fertig zu stellen. Reizen würde mich es schon. Jedoch würde ich gerne noch Zeit mit Menschen verbringen, welche mich in dieser Zeit unterstützt haben.

Grossen Dank an meine Grossmutter und meine Mutter, welche zu meiner Tochter geschaut habe. Dank an meine Frau, welche mir Raum gelassen hat, mich daran erinnert und motiviert hat. Danke an Tim der mich mental unterstützt hat. Danke an Dominic mit dem ich mich Austauschen konnte. Danke an meine Band welche mir diese Idea gegeben hat. Danke an meine Familie und Freunde, die noch mir Reden, auch wenn ich schon ewig nicht gemeldet habe. Danke an meine Experten und den Ausschuss, die das mitgemacht haben.

9. Authentizität

Mit meiner Unterschrift bestätige ich die vorliegende Diplomarbeit selbstständig, ohne Hilfe Dritter und nur unter Benutzung der angegebenen Quellen ohne Copyright-Verletzung, erstellt zu haben.

Bern, 09.06.2025

Kenan Tabinas

10. Anhang

Anhang A: Literaturverzeichnis, Quellenangaben

Anhang B: Diplomeingabe / Projektinitialisierungsauftrag

Anhang C: Projektauftrag

Anhang D: Terminplan

Anhang E1, E2 - En: Zusätzliche Konzeptdokumente

Anhang F: Zusätzliche Informationen / Beilagen

Anhang G: Protokolle / Präsentationen

Anhang H: Logbuch

Anhang I: Abkürzungsverzeichnis, Glossar

Anhang A

Literaturverzeichnins

keine

Quellenverzeichnis

Srf abgerufen am 26.03.2026 von Kenan Tabinas

<https://www.srf.ch/radio-srf-3/musik/mx3-feiert-geburtstag-die-groesste-musikplattform-der-schweiz-was-steckt-dahinter#:~:text=%C3%9Cber%2030'000%20Schweizer%20Bands,Geburtstag>.

USC Scv abgerufen am 26.03.2026 von Kenan Tabinas

<https://www.usc-scv.ch/index.php?p=was-ist-die-scv&l=de>

Wikipedia abgerufen am 26.03.2026 von Kenan Tabinas

https://de.wikipedia.org/wiki/Musik_in_Deutschland

Europeanchoralassocian.org abgerufen am 26.03.2026 von Kenan Tabinas

https://europeanchoralassociation.org/wp-content/uploads/2019/01/SingingEurope_DE.pdf

Wikipedia abgerufen am 26.03.2026 von Kenan Tabinas

<https://de.wikipedia.org/wiki/Deutschland>

Wikipedia abgerufen am 26.03.2026 von Kenan Tabinas

<https://de.wikipedia.org/wiki/Schweiz>

Wikipedia abgerufen am 26.03.2026 von Kenan Tabinas

<https://de.wikipedia.org/wiki/%C3%96sterreich>

Chorverband abgerufen am 26.03.2026 von Kenan Tabinas

<https://www.chorverband.at/chorverband/ueber-den-verband>

Anhang B

Freigabe Diplomarbeit

ID: 2558
Name/n: Kenan Tabinas
Freigabe: Ja, mit Vorbehalten/Rahmenbedingungen

Vorbehalte/Rahmenbedingungen:

Folgende Punkte müssen in der Diplomarbeit aufgezeigt und umgesetzt werden:

- Unterschiedliche Deployment Architekturen CI/CD müssen in der Studie evaluiert werden (Mikrosegmentierung)
- Plattform muss skalierbar und redundant aufgebaut werden
- Aufbau des Hostings mit Datenhaltung muss betrieblich umgesetzt werden (Disaster Recovery, Backup)
- Im Diplombericht muss aufgezeigt werden wie «Security by Design» umgesetzt wurde
- Ein Security Test auf die Applikation muss ausgeführt werden

Prüfungskommission, Bern 23. Januar 2025





Projektinitialisierungsauftrag

Auftraggeber Kenan Tabinas
Projektleiter Kenan Tabinas
Autor Kenan Tabinas
Klassifizierung Intern
Status Genehmigt

Änderungsverzeichnis

Datum	Version	Änderung	Autor
22.12.2024	0.1	Erster Draft	Kenan Tabinas
22.12.2024	1	Abgabe	Kenan Tabinas

Inhaltsverzeichnis

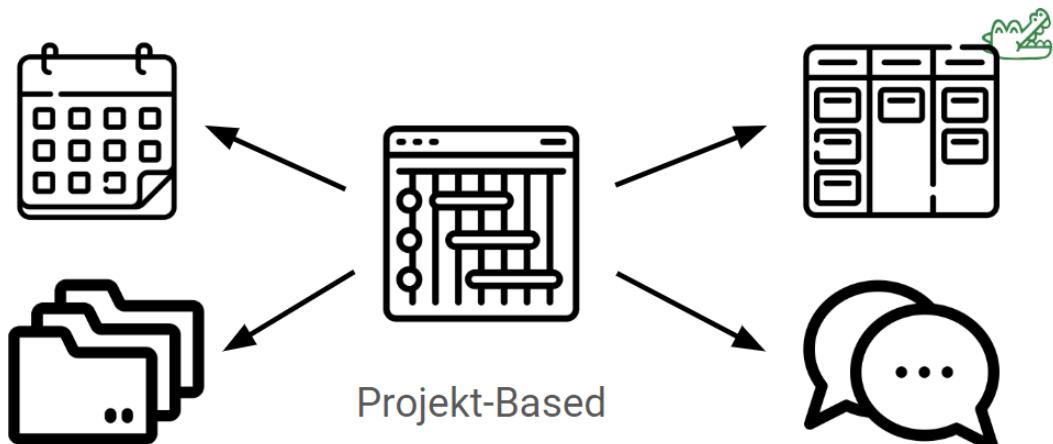
1	Ausgangslage	2
2	Ziele	3
3	Rahmenbedingungen.....	4
4	Ergebnisse und Termine	4
5	Aufwand.....	4
6	Kosten	4
7	Ressourcen.....	5
8	Kommunikation	5
9	Risiken.....	6

1 Ausgangslage

Das Organisieren von Aufgaben ist unerlässlich, sobald mehrere Personen involviert sind.

Das Unternehmen Delegator hat sich zum Ziel gesetzt, eine innovative Lösung zu entwickeln, die die Zusammenarbeit für Einzelpersonen, Familien, Vereine und Unternehmen effizienter und einfacher gestaltet. Der Fokus liegt darauf, bestehende Lücken zwischen beliebten Kollaborationstools wie Microsoft Teams, ClickUp und Trello zu schliessen. Diese Werkzeuge bieten zwar viele nützliche Funktionen, stoßen jedoch an ihre Grenzen, wenn es darum geht, Kalender und Kanban-Boards zu synchronisieren oder Ressourcen einzutragen.

Die aktuelle Marktsituation zeigt, dass Alternativen wie Asana und Jira zwar leistungsstark sind, jedoch für viele Zielgruppen zu kostspielig. OpenProject wäre zwar kostenlos, wirkt aber etwas altmodisch und wenig benutzerfreundlich.



SOLL

Delegator

Delegator möchte daher ein Tool entwickeln, das nicht nur erschwinglich und intuitiv ist, sondern auch die Anforderungen an

2 Ziele

Folgende Projektziele müssen erreicht werden:

Nr.	Kategorie	Beschreibung	Messgrösse	Priorität
1	Lieferobjekt	Sicherstellen, dass Ressourcen pro Woche pro User festgelegt werden können.	Prüfung, ob Ressourcenplanung pro User in der Software möglich ist.	M
2	Technisches Ziel	Ermöglichen, dass einzelne Ressourcen hinzugefügt oder entfernt werden können.	Funktion zur Ressourcenzuordnung und -entfernung erfolgreich getestet.	1
3	Lieferobjekt	Sicherstellen, dass alle Tasks geschätzt, terminiert und zugeordnet werden können.	Prüfung, ob Aufgaben mit geschätztem Aufwand, Terminen und Zuständigkeiten versehen werden können.	1
4	Technisches Ziel	Synchronisation aller Tasks mit dem Kalender und automatische Darstellung darin.	Funktion zur Kalendersynchronisation und Darstellung erfolgreich geprüft.	2
5	Betriebliches Ziel	Sicherstellen, dass Kalender in andere Apps importiert werden können.	Test, ob Kalenderdaten in gängige Kalender-Apps (z. B. Google Calendar, Outlook) importierbar sind.	1
6	Betriebliches Ziel	Eine User Erfassung und Verwaltung.	Funktion der Userverwaltung erfolgreich getestet.	3

* Priorität: M = Muss / 1 = hoch, 2 = mittel, 3 = tief

Folgende Ziele sind die Vorgaben für die Phase Initialisierung:

Nr.	Kategorie	Beschreibung	Messgrösse	Priorität*
1	Technisches Ziel	Abklären ob es Open Source Technologien gibt, welche man Verwenden kann damit nicht alles selber entwickelt werden muss.	Prüfung der Möglichkeiten	1
2	Technisches Ziel	Abklären der Hosting-Anbieter und Ausbaubarkeit in der Zukunft.	Prüfung der Möglichkeiten	2
3	Lieferobjekt	Bedarf von Kalender fähigkeiten bei bereits festgelegte Test usern	Prüfen ob Umfrage erfolgt ist	3

* Priorität: M = Muss / 1 = hoch, 2 = mittel, 3 = tief

3 Rahmenbedingungen

Rahmenbedingungen Phase Initialisierung

- Anwendung von HERMES Projektmethodik die auf das Projekt angepasst werden muss
- Die Projektleitung wünscht explizit, dass ein „Make or Buy“ Empfehlung aus der Initialisierung hervorgeht

4 Ergebnisse und Termine

Folgende Ergebnisse werden in der Phase Initialisierung vom Projekt erstellt.

Nr	Ergebnis	Termin
1	<i>Studie</i>	31.01.2024
2	<i>Projektauftrag</i>	31.01.2024
3	<i>Projektplan</i>	31.01.2024

5 Aufwand

Geschätzter interner Personalaufwand beläuft sich für die Phase Initialisierung auf ca. 32 Stunden.

6 Kosten

Geschätzter interner Personalaufwand beläuft sich auf ca. 3850 CHF. Der Aufwand wurde mit einem durchschnittlichen internen Ansatz von 120sFr berechnet.

Phase	Geplant
Initialisierung	3840

7 Ressourcen

Personalressourcen

Rolle / Person	Dezember	Januar	Februar	März	April	Mai	Juni	Total	Bestätigung Vorgesetzter
Kenan	32h	32h	32h	32h	32h	32h	32h	224h	Erteilt

Sachmittel

Es werden keine Sachmittel wie Räume, IT-Infrastruktur, Spezifische Software, etc. benötigt die externe Kosten verursachen.

8 Kommunikation

Reporting während der Phase Initialisierung, Information Auftraggeber,

Information der Betroffenen Stellen und Stakeholder

Adressat der Information	Verantwortlich für die Kommunikation	Inhalt	Ziel	Mittel / Medium	Termin
Abteilungsleiter	Auftraggeber Kenan Tabinas	Ziel und Planung der Projektinitialisierung	Die Abteilungen kennen den Auftrag an den Projektleiter	Abteilungsmeeting	22.12.2024
Auftraggeber	Projektleiter	Monatlicher Statusbericht	Vorschritt des Projekt an AG kommunizieren	Status Report	Monatlich am 1. Arbeitstag im Monat

9 Risiken

Für die Erarbeitung der Studie werden auch interne Ressourcen kaum belastet. Die Studie kann neben dem „Daily Business“ durchgeführt werden. Somit sind keine wirklichen Risiken für die Phase Initialisierung zu erwarten.

Nr.	Risiko- beschreibung	EW	AG	RZ	Massnahmen	Verantw.	Termin

Legende: EW=Eintretenswahrscheinlichkeit: 1 Niedrig / 2 Mittel / 3 Hoch;
AG=Auswirkungsgrad: 1 Gering / 2 Mittel / 3 Gross;
RZ=Risikozahl: RZ = EW x AG

Anhang C

Studie

Delegator

Klassifizierung	intern
Status	in Arbeit
Programmname	Delegator
Projektnummer	1
Projektleiter	Tabinas Kenan
Version	0.1
Datum	26. März 2025
Auftraggeber	Tabinas Kenan
Autor/Autoren	Tabinas Kenan
Verteiler	

Änderungsverzeichnis

Version	Datum	Änderung	Autor
0.1	19.03	Erstellt	TAK
1.0	06.06	Farblich überarbeitet	TAK

Tabelle 1: Änderungsverzeichnis

Beschreibung

Die Studie beschreibt die angestrebte Lösung, in dem sie die groben Ziele definiert, möglichen Lösungsvarianten aufführt und diese dann bewertet. Sie bildet die Grundlage für die Entscheidung, ob ein Projekt freigegeben wird oder nicht. Sie ist die Voraussetzung für die Erarbeitung des Projektmanagementplans und des Projektauftrags.

1 Informationsbeschaffung

1.1 Konkurrenz Analyse

Um mir ein Überblick zu verschaffen habe ich mir diverse Konkurrenz Produkte angeschaut:

Programm	Pricing	Unterschiede zu meiner Applikation
Trello	Kostenlos (Reduzierten Features) 5 CHF pro User pro Jahr	Aufgaben sind nicht zuteilbar auf einzelne Personen in der Grund Version
Asana	Kostenlos bis zu 15 Personen 11 CHF pro User pro Jahr	Zielgruppe sind eher kleinere Unternehmen
Google Calendar	Kostenlos	Es ist möglich Aufgaben zu Terminieren. Es ist aber nicht anpassbar auf bestimmte Abläufe. Subaufgaben sind nicht zuweisbar.
ClickUp	Kostenlos (Reduzierte Features) 7 CHF pro User Pro Jahr	Clickup kann sehr viel von dem ich auch einbauen möchte. Meine Applikation muss sicher Kosten günstiger sein. Nicht so intuitiv. Zielgruppe sind eher kleinere Unternehmen.

Tabelle 2: Informationsbeschaffung Konkurrenz

Zuerst hat es mir etwas sorgen gemacht. Was wenn ClickUp alles besser kann als meine Applikation. Ich bin meine ich probiert meine Features durch zugehen.

Ich versuchte Grobanforderung G3 zu Testen. Das Synchronisieren mit bestehenden Kalender Apps. Das Feature wird angeboten, in den Einstellung wird man regelrecht überfahren. Der Funktionsumfang von ClickUp ist gigantisch.

Hier habe ich festgestellt, der grösste Unterschied muss Usability sein. Viele Features mag für IT-Gurus genau das richtige sein. Für eine Band oder Familie ist es jedoch ein Killer Kriterium.

Daher habe ich auch realisiert, dass die Software zwingen Narren sicher sein muss und Qualität entscheidend ist. Wenn die Kunden die Software nicht verstehen oder zu lange warten müssen, verliere ich das Vertrauen.

Das hat auch gerade in die Frage Service Hotline eingespielt. Es muss Tutorials geben, wie die App bedient werden kann. Such Option und ein FAQ. Eine Hotline für Hilfe zu bekommen.

Es braucht ein Feedback Möglichkeit und ein Bug-Reporting. Mit z.B. einem Goodie. Wenn ein Bug basierend auf ihrem Ticket gelöst wird, gibt es erlass auf eine Monatsbeitrag.

2 Ziele

Im Gegensatz zu den später im Projekt konkret definierten und priorisierten Zielen, sind die Ziele in der Studie erst grob formuliert und nach Muss, Soll und Kann gewichtet. Sie dienen als Grundlage für die Ausarbeitung der Lösungsvarianten.

2.1 Projektziele

Nr.	Kategorie	Beschreibung	Messgrösse	Gewicht*
P1	Lieferobjekt	Ressourcenplannung	Ressourcen pro Woche und User können geplant werden. Planbare Ressourceneinträge in der Oberfläche	M
P2	Lieferobjekt	Aufgabenmanagement	Aufgaben können erstellt, geschätzt, terminiert und zugewiesen werden. Funktion getestet: Aufgabe enthält Aufwand, Termin, User	M
P3	Lieferobjekt	Kalendersynchronisation	Aufgaben werden automatisch mit Kalender synchronisiert und angezeigt. Termine erscheinen korrekt im Kalender	M
P4	Lieferobjekt	Kalender Integration	Kalender können mit externen Tools (z. B. Google Calendar) synchronisiert werden. iCal-Export getestet mit Outlook, Google Kalender	M
P5	Lieferobjekt	Userverwaltung	Benutzerkonten können erstellt und verwaltet werden. Neue Benutzer können angelegt und berechtigt werden	S
P6	Betriebliches Ziel	Front und backend Stack gewählt	Die geeigneten Technologien werden identifiziert. Bewertungsmatrix von Technologievarianten	M

Nr.	Kategorie	Beschreibung	Messgrösse	Gewicht*
P7	Leistungsziel	Organisation vereinfacht	Priorisierte Featureliste nach Muss/Soll/Kann	M
P8	Betriebliches Ziel	Marketing	In einem Zeitraum von Wöchentliche Sozial Media Beiträge erhöhen die Awareness. Ich schalte Werbung mit einem Budget von 100 CHF pro Monat	S
P9	Technisches Ziel	Plattform bereitgestellt	Software ist Verfügbar im AppStore und die User können sich einloggen und ein Account erstellen	K
P10	Technisches Ziel	Skalierungsplan	Es ist nicht Teil des Projektes dass ich in die Cloud Wechsle oder mein System Update. Trotzdem wäre es toll wenn ich Vorräus Denke und ein Konzept mache. Idealerweise beachte ich die Fallstricke bereits während der Entwicklung um bei einer Migration der Job so einfach wie möglich zu machen.	S
P11	Technisches Ziel	Ressourcen Einschätzung	Ich muss im Verlauf das Projektes Mock Raquets machen, um herauszufinden, wie lange mein System funktioniert. Ab wann muss ich auf ein grösseres System Updaten.	M

* Gewicht: M = Muss / S = Soll / K = Kann

Tabelle 3: Ziele

2.2 Projektvorgehensziele

Beispiele von Kategorien:

Nr.	Kategorie	Beschreibung	Messgrösse	Gewicht*
T1	Strategie-analyse	Technologien und Stack müssen evaluiert werden	Verantenscheid mit Begründung	M
T2	Infrastrukturwahl	Hosting- und Skalierbarkeit prüfen.	Vergleich für ein Upgrade auf Cloud/VPS/Self-hosting	S
T3	Zielgruppenbedarf	Kalenderbedarf und Funktionalitätsbedürfnisse der Zielgruppe validieren (Umfrage).	Ergebnisse von mindestens 5 Testnutzern	K

* Gewicht: M = Muss / S = Soll / K = Kann

Tabelle 4: Projektvorgehensziele

3 Grobanforderungen

Hier ist eine Liste der Grobanforderungen. Die Vollständige Liste ist im Anhang unter ID2558_TabinasKenan_Studie_Anforderungen.xlsx

ID	Anforderungen	Art ¹	Abnahmekriterium	Wichtigkeit ²	Dringlichkeit ³
G1	Projekte erfassen & verwalten	F	Projekt-CRUD getestet	5	5
G2	Aufgaben erstellen & zuweisen	F	Aufgaben-Zuweisung & Terminierung getestet	5	5
G3	Synchronisation mit gängigen Kalender-Apps	Q	Synchronisiert mit bestehenden Apps	4	4
G4	Setlisten erstellen & exportieren	F	PDF-Export erfolgreich getestet	3	3
G5	Projekttypen Auswahl	F	Verschiedene Projekttypen mit Abläufen sind Auswählbar	5	5

1) Art = Anforderungsart: G = Geschäftsorganisation, F = Funktional, Q = Qualität, S = Sicherheit, M= Migration, A= Architektur, B = Betrieb, K = Konformität (Gesetzgebung, Weisungswesen, Normen und Richtlinien)

2) Wichtigkeit: 5 = muss zwingend umgesetzt werden; 4 = sehr wichtig, 3 = wichtig, 2 = normal, 1 = nicht wichtig

3) Dringlichkeit: 5 = muss sofort umgesetzt werden, 4 = sehr dringend, 3 = dringend, 2 = normal, 1 = nicht dringend

Tabelle 5: Grobanforderungen

4 Grundsatzentscheidungen

Es ich habe kein Offizielles Vorgehen für Übergeordnete Variantenentscheidungen gefunden. Jedoch gab es in der Planung diverse Entscheidungen welche Treffen musste. Ich habe daher diese Grundsatzentscheidungen genannt. Hier ist eine Aufleistung mit allen Varianten Entscheidungen.

4.1 Grundsatzentscheidungen Übersicht

Grundsatzentscheidungen	Bezeichnung
B1	Evaluation der Zielgruppe. Bands, Musiker und Familien.
B2	Evaluation der Frontend Technologien. WebOnly, WebView, Native App oder Cross Plattform.
B3	Evaluation der Plattform. Selfhostet oder VPS

Tabelle 6: Grundsatz Entscheidungen

5 Lösungsvarianten von B1

5.1 Variantenübersicht

Variante	Bezeichnung
V1	Weniger Admin mehr Musik.
V2	Familien und Bands

Tabelle 7: Variantenübersicht von B1

Methoden zur Generierung von Lösungsvarianten: Kreativitätstechniken (z.B. Brainstorming, Methode 6-3-5), Ursache-Wirkungs-Grafiken (-Matrizen), Morphologischer Kasten, Analogie-Methode, u.a.

5.2 Variante V1 «Weniger Admin mehr Musik»

Die Erste Variante «Weniger Admin mehr Musik» beschränkt sich vor allem auf Bands.

5.2.1 Kurzbeschreibung

Delegator ist eine moderne Organisationsplattform, die speziell für Bands entwickelt wurde. Weniger Admin und mehr Musik. Mit einer einfachen Bedienung und nur den wichtigen Funktionen. Kalender, Aufgaben, Setlisten und Ressourcen. Delegator bietet alles, was es braucht, um den Probenraum zu organisieren, ganz ohne Management Knebelvertrag. Kostenlos Testen und jetzt loslegen.

Beschreibung und Grafik

5.2.2 Zielegruppe

Herauszufinden wie viele Bands es gibt es etwas schwierig es gibt leider keine Zahlen von Bundesamt für Statistik. Ich habe trotzdem versucht Informationen zu finden. Ich habe mich mal im DACH Raum umgesehen.

Laut der Webseite mx3.ch, hier können sich Bands Registrieren. Es sind über 33'000 Bands Angemeldet.¹ Die Zahl ist weit höher, da es auch lokal Musiker gibt, sogenannte «gugge musig». Laut dem Schweizerischen Chor Vereinigung gibt es 1400 Chöre und etwa 38000 Mitglieder.²

Laut Wikipedia gibt es in Deutschlang 740.000 Personen in Musikgruppen.³ Interessant könnte auch sein dass es 1.3 Million Personen in Chören unterwegs sind. Damit wir mit Zahlen Rechnen können haben wir Durchschnitte angeschaut. Laut dem Europäischen Chor Verband «singineurope.org» besteht der Durchschnittliche Chor aus 36 Personen.⁴ Der Schnitt in der Schweiz beträgt 26. Ich habe leider keine Zahlen zu Durchschnittlichen Bands gefunden. Wir nehmen mal an es beträgt 5.

Einwohneranzahl von Deutschland beträgt 85 Mio.⁵ Die Einwohner Anzahl der Schweiz beträgt 9 Mio.⁶ Österreich beträgt 9. Mio.⁷

Laut dem Österreichischen Chorverband, gibt es 4000 Chöre und 115000 Mitglieder.⁸ Das entspricht einer Chor Grösse von 28 Personen. In der Österreich gibt es kein Verband für Bands und keine Zahlen. Scheint kein Thema zu sein.

Land	Chor dichte pro 10000	Band dichte pro 10000
CH	4.2	17.4
DE	1.5	36.6
AT	4.4	N/A

Tabelle 8: B1 V1 Zielgruppe

Fazit:

- Wir haben in der Schweiz eine enorm grosse Band Dichte auf unseren kleinen Raum.
- Chöre sind eine mögliche Zielgruppe. Davon gibt es einfach 4-5 mal weniger als Chöre.

¹ <https://www.srf.ch/radio-srf-3/musik/mx3-feiert-geburtstag-die-groesste-musikplattform-der-schweiz-was-steckt-dahinter#:~:text=%C3%9Cber%2030'000%20Schweizer%20Bands,Geburtstag>.

² <https://www.usc-scv.ch/index.php?p=was-ist-die-scv&l=de>

³ https://de.wikipedia.org/wiki/Musik_in_Deutschland

⁴ https://europeanchoralassociation.org/wp-content/uploads/2019/01/SingingEurope_DE.pdf

⁵ <https://de.wikipedia.org/wiki/Deutschland>

⁶ <https://de.wikipedia.org/wiki/Schweiz>

⁷ <https://de.wikipedia.org/wiki/%C3%96sterreich>

⁸ <https://www.chorverband.at/chorverband/ueber-den-verband>

5.2.3 Konkurrenz Analyse

Ich habe im Android App Store mir angeschaut welche Produkte bereits auf dem Markt sind im Thema Organisation und Band.

Folgende Apps sind in der Suche Auftaucht und wurden aus der Analyse ausgeschlossen.

- Aufnahme Tools.
- Apps zum Bands und Konzerte in der Stadt suchen
- Apps für das lernen von Tonleitern
- Smartphone Tools wie Gitarren Tuner
- GuitarHero ähnliche Videospiele
- Apps für Sport mit Bändern zu machen
- Setlist Manager

Folgende zwei Apps habe ich mir im Detail angeschaut:

App	Pricing	Features
BandHelper	Basic: 5 pro Monat pro Band Plus: 7 pro Monat pro Band Pro: 9 pro Monat pro Band	Noten Kalender Kontakte Finanzen Checklisten
Band Mule	Pricing nicht ersichtlich. Die Applikation ist in den ersten zwei Monaten frei. Anschliessend ist es nicht klar.	Kalender Chat Abstimmungen Setliste online

Tabelle 9: B1 V1 Konkurrenzanalyse

Fazit:

- Es gibt fast keine Konkurrenz
- Setlisten und Abstimmungen scheint etwas Wichtiges zu sein

5.3 Variante V2 «Familien und Bands»

Die zweite Variante «Familien und Bands» wir schauen uns an ob es möglich ist auch für Familien selbe Funktionalität zu bieten.

5.3.1 Kurzbeschreibung

Delegator ist eine moderne Organisationsplattform, die speziell für Familien und kleine Organisationen. Mit einer einfachen Bedienung und nur den wichtigen Funktionen. Kalender, Aufgaben, Ressourcen und das nötige. Delegator bietet alles, was es braucht, mit angepassten Einkaufslisten und Essensplanung.

Beschreibung und Grafik

5.3.2 Zielegruppe

Laut Statistika gibt es in der Schweiz 4.02 Millionen Privat Haushalte.⁹ Dabei beträgt die Durchschnittliche 2.18 Personen.¹⁰

In Deutschland sind es 41.3 Millionen.¹¹ In Österreich 4.16 Millionen.

Personen welche alleine Wohnen wollen wir Entfernen, da der Faktor Zusammen fehlt.

In der Schweiz wohnen 37% der Personen allein. Also 1.4 Millionen Menschen sind wahrscheinlich nicht an der App interessiert.¹² In Deutschland sind es 20.1% und in Österreich 17.4%.¹³

Mehr zu den Berechnungen finden sie im Anhang.

Land	Mehrpersonen Haushalte	Personen pro Mehrpersonen Haushalt
DE	33 Mio	2.3
AT	3.4 Mio	2.4
CH	2.4 Mio	3.0

Tabelle 10: B1 V2 Zielgruppe

Fazit:

- Wir haben in der Schweiz eine hohe allein Haushalts Dichte.
- Mehrpersonen Haushalte bestehen aus 2-3 Personen

⁹ <https://de.statista.com/statistik/daten/studie/710925/umfrage/privathaushalte-in-der-schweiz/>

¹⁰ <https://de.statista.com/statistik/daten/studie/75588/umfrage/durchschnittliche-haushaltsgroesse-in-der-schweiz/>

¹¹ <https://de.statista.com/statistik/daten/studie/156950/umfrage/anzahl-der-privathaushalte-in-deutschland-seit-1991/>

¹² <https://www.tagesanzeiger.ch/einzelhaushalte-schweiz-warum-menschen-alleine-wohnen-404606747517>

¹³ <https://www.destatis.de/Europa/DE/Thema/Bevoelkerung-Arbeit-Soziales/Bevoelkerung/Alleinlebende.html#:~:text=Jeder%20f%C3%BCnfte%20Mensch%20in%20Deutschland%20lebt%20allein&text=Mit%202020%2C3%20%25%20lag%20der,Vergleich%20noch%20mehr%20Menschen%20allein.>

5.3.3 Konkurrenz Analyse

Ich habe im Android App Store mir angeschaut, welche Produkte bereits auf dem Markt sind im Thema Organisation und Familien. Eingeschlossen habe ich Apps welche eher in Richtung.

Folgende Apps habe ich mir nicht Angeschaut weil der Fall klar wurde:

- FamilyWall
- TimeTree
- FamilyNest
- Maple: Family Organizer
- MyFamiliz
- Octave Family Organizer
- Fam Cal: Family Shared Calender
- Share(d)
- the Family App

Folgende zwei Apps habe ich mir im Detail angeschaut:

App	Pricing	Features
Cozi Family Organizer	5 CHF pro Monat pro Familie	Einkauflisten Kalender
OurHome	Kostenlos	Aufgaben Belohnungssystem

Tabelle 11: B1 V2 Konkurrenz Analyse

OurHome hat mich etwas überrascht, bin mir nicht sicher wie das Rentabel ist. Erst war ich etwas eingeschüchtert von OurHome. Ich suchte nach Reviews. Bis mir aufgefallen ist, es existiert nicht im AppleStore. Eine Familien App hat strenge Konkurrenz.

Fazit:

- Apple App Store ist zwingend.
- Eine Familien App hat strenge und gute Konkurrenz

5.4 Analyse und Bewertung der Varianten

5.4.1 Zielerreichung

Nr.	Beschreibung	Gewicht *	V1	V2
1	Der Markt Kapital ist noch nicht ausgeschöpft.	Muss	Ja	Nein

Tabelle 12: Zielerreichung B1

5.5 Variantenwahl

Ich habe mich für die Variante V1 «weniger Admin mehr Musik» entscheiden. Der Markt bietet zwar 24-mal Kunden. Also Mehrpersonenhaushalte als Bands. Jedoch der Markt an Lösung in den Familien und WG Umfeld ist mehr als ausgeschöpft.

Hier eine Disruptive Lösung auf den Markt zu bringen ist utopisch. Da es auch kostenlose Lösungen gibt, welche unschlagbar ist.

In der Schweiz haben wir noch die höchste Band Dichte und die höchste Einzelpersonen Dichte im ganzen DACH-Raum.

6 Lösungsvariantenwahl von B2

Hier geht es um Frontend Technologien. Dabei sind jedoch nicht die Technischen Frameworks im Vordergrund. Im Focus ist eine Design Entscheidung. Setzen wir eher auf Mobile Apps, nur aufs Web oder etwas dazwischen.

6.1 Variantenübersicht

Variante	Bezeichnung
V1	WebOnly
V2	WebView
V3	NativeApp
V4	CrossPlattform

Tabelle 13: Variantenübersicht von B2

6.2 Variante V1 «WebOnly»

6.2.1 Kurzbeschreibung

Diese Variante ist basiert auf einem Webauftritt. Die Möglichkeit die Applikation über eine Native Smartphone App zu bedienen, wird komplett weggelassen.

6.2.2 Vorteile

Thema	Erklärung
Codebasis	Die Menge an Code wird massiv reduziert, da nur eine Plattform entwickelt und gewartet werden muss.
Entwicklung	Die Entwicklungszeit verringert sich erheblich, da keine separaten Apps für verschiedene Betriebssysteme programmiert werden müssen.
Wartung & Updates	Änderungen können zentral durchgeführt und sofort für alle Nutzer verfügbar gemacht werden, ohne dass App-Updates verteilt werden müssen.
Plattformunabhängig	Die Anwendung ist von jedem Gerät mit Internetzugang und einem Browser erreichbar. Die höchste Unabhängigkeit von allen Varianten.

Tabelle 14: B2 V1 Vorteile

6.2.3 Nachteile

Thema	Erklärung
UX	Smartphone-Nutzer sind eher daran gewöhnt, native Apps zu verwenden. Eine Weblösung kann als weniger intuitiv empfunden werden.
Performance	Eine Webapplikation ist in der Regel langsamer und als eine native App.
Appstore	Ohne eine Präsenz in den App Stores ist es schwieriger, neue Nutzer zu gewinnen, da viele Nutzer direkt im Store nach Lösungen suchen.
Geräte Funktionen	Native Apps haben direkten Zugriff auf Funktionen Push-Benachrichtigungen und Offline-Speicherung.
Abhängigkeit	Die Webseite erfordert eine permanente Internetverbindung

Tabelle 15: B2 V1 Nachteile

6.3 Variante V2 «WebViewApp»

6.3.1 Kurzbeschreibung

Diese Variante basiert auf einer WebView-App, also einer nativen Anwendung, die im Grunde nur eine Webbrowser-Komponente enthält und eine Webanwendung darstellt.

6.3.2 Vorteile

Thema	Erklärung
Entwicklung	Da die eigentliche Anwendung eine Webseite ist, muss nur eine einfache native Hülle entwickelt werden.
Appstore	Die App läuft auf verschiedenen Betriebssystemen und Geräten, verglichen zu Native Apps.
Wartung & Updates	Änderungen an der Webanwendung werden direkt übernommen. Die App muss nicht jedes Mal im AppStore geupdated werden
Features	Mehr Features verglichen zu einer Web-Only Anwendung

Tabelle 16: B2 V2 Vorteile

6.3.3 Nachteile

Thema	Erklärung
Entwicklung	Verglichen zur WebOnly Anwendung muss, mehr entwickelt werden.
UX	Nutzer erwarten eine native Benutzerführung und Performance, die eine WebView-App oft nicht bieten kann.
Abhängigkeit	Da die App eine Webseite lädt, ist eine permanente Internetverbindung erforderlich.
Performance	WebView-Apps sind oft langsamer als native Apps.
App Store Policy	Apple und Google lehnen manchmal einfache WebView-Apps ab, wenn sie nicht genügend native Funktionen bieten.

Tabelle 17: B2 V2 Nachteile

6.4 Variante V3 «Native App»

6.4.1 Kurzbeschreibung

Bei dieser Variante verwendet man 3 Verschiedene Codesbases. Ein WebFramework, Swift für iOS, Kotlin für Android.

6.4.2 Vorteile

Thema	Erklärung
Performance	Die Individuelle Entwicklung macht die Performance 5-10 mal schneller als z.B React Native. ¹⁴
UX	Native Apps sind intuitiv und schnell. Das Design passt genau zum jeweiligen Betriebssystem.
Features	Zugriff auf alle Kamera, GPS, Bluetooth, Push-Benachrichtigungen etc.
Offline	Die App ist grösstenteils offline verwendbar.
App Store Policy	Native Apps entsprechen auch den Store-Richtlinien von Apple und Google

Tabelle 18: B2 V3 Vorteile

6.4.3 Nachteile

Thema	Erklärung
Entwicklung	Drei verschiedene Code-Bases ist ein Alptraum.
Wartung	Die Erstellung und Wartung von drei separaten Apps ist astronomisch.

Tabelle 19: B2 V3 Nachteile

¹⁴ <https://medium.com/swlh/flutter-vs-native-vs-react-native-examining-performance-31338f081980>

6.5 Variante V4 «Cross Plattform»

6.5.1 Kurzbeschreibung

Diese Variante basiert auf einer Cross-Plattform-App, die mit einer einzigen Codebasis für mehrere Betriebssysteme (Android, iOS, Windows, macOS, Linux) entwickelt wird.

6.5.2 Vorteile

Thema	Erklärung
Entwicklung	Eine einzige Codebasis spart Zeit und Geld.
Plattformübergreifend	Die App läuft auf verschiedenen Betriebssystemen und Geräten, verglichen zu Nativen Apps.
Wartung und Updates	Änderungen und Bugfixes müssen nur einmal gemacht werden. Der Wartungsaufwand reduziert sich massiv.
Performance	Moderne Frameworks bieten vergleichbare Performance verglichen zu Nativen Apps.
Features	Mehr Features verglichen zu einer Web-Only Anwendung
App Store	Die Anwendung ist verfügbar im App Store und macht sie zugänglicher als nur eine Webseite.

Tabelle 20: B2 V4 Vorteile

6.5.3 Nachteile

Thema	Erklärung
Performance	Die Performance ist zwar gut. Jedoch nicht so gut wie bei Nativen Apps.
Individuelle Anpassungen	Da jede Plattform anders ist, müssen im Zweifel Anpassungen gemacht werden.
Abhängigkeit	Die App ist von der Weiterentwicklung der gewählten Cross-Plattform-Frameworks abhängig.
Speicherverbrauch	Cross-Plattform-Frameworks benötigen zusätzliche Laufzeitbibliotheken. Welche die App etwas grösser machen.

Tabelle 21: B2 V4 Nachteile

6.6 Bewertungskriterien

6.6.1 Kriterien

Nach Evaluation der Möglichkeiten wurden folgende Vor und Nachteile in den Kriterien berücksichtig:

Thema	Erklärung
Entwicklung	Kosten beim Start
Updates	Kosten bei Wartung und weiter Entwicklung
App Store	Verfügbarkeit im Appstore
UX	Zugänglichkeit und Anwendbarkeit

Tabelle 22: B2 Kriterien

6.6.2 Nicht Kriterien

Folgende Vor- und Nachteile haben es nicht in die Kriterien Geschaft:

Thema	Begründung
Performance	Die Performance ist vernachlässigbar.
Features	Push Up Benachrichtigungen sind ein Nice to have. Mehr auch nicht.
Speicherverbrauch	Speicherverbrauch ist vernachlässigbar
Abhängigkeit von Frameworks	Die Verwendung von Frameworks ist über all der Fall und kann daher Ignoriert werden.
Offline	Offline Features sind ein Nice to have, aber kein Break or Make Kriterium.

Tabelle 23: B2 Nicht Kriterien

6.7 Analyse und Bewertung der Varianten

6.7.1 Bewertung

Alle Kriterien werden von 1 bis 3 Bewertet. Wobei 3 am besten ist und 1 am schlechtesten.

Kriterium	WebOnly	WebView	Native	X-Plattform
Entwicklung	3	3	1	2
Updates	3	3	1	3
Appstore	1	2	3	3
UX	1	2	3	3
Total	8	9	8	10

Tabelle 24: B2 Bewertung

6.7.2 Fazit

Die WebOnly ist wegen der Fehlenden App rausgeflogen. Bandmitglieder möchten nicht jedes Mal das Notebook aufklappen, um zu sehen, wann die nächste Probe ist.

Die Native App ist wegen dem Entwicklungsaufwand rausgeflogen. Es braucht zu viel Zeit und Geld mehrere Apps zu Entwickeln und Unterhalten.

Eine Frage welche noch nicht geklärt ist. Welchen Ansatz der Zielführende ist. Bei genauerem Hinsehen ist die WebViewApp und Cross-Plattform nahe beieinander. Für die Entwicklung von WebView Apps bietet sich die Verwendung von Cross-Plattform-Framework an.

Die User Experience liefert den Finale Entscheidung. Die User Experience auf einem Browser ähnlichen App ist nicht dieselbe wie bei einer Nativen App. Somit können wir die UX und Qualität steigern, indem wir uns auf ein Segment limitieren.

7 Lösungsvariantenwahl von B3

Die Frage der Frontend Technologie ist geklärt. Jetzt schauen wir uns Backend Technologien an.

7.1 Ausgangslage

Als Vorbereitung der Entwicklung habe ich für eine Temporäre und Kostengünstige Lösung Entscheiden.

Zuhause habe ich keine Kabel Verbindung zu meinem ISP. Ich nutze einen 5G Router und eine zweite Sim-Karte. Der Nachteil ist, ich kann DDNS nicht nutzen.

Wingo verwendet CGNAT (Carrier-Grade NAT). Alle Teilnehmer beim selben Cellular Tower erhalten dieselbe IP. Für jeden Request wird ein neuer Port geöffnet. Da ich keine Firewall-Rules editieren kann, kann ich auch kein Port-Forwarding einrichten.

Es gibt eine Art wie ich das umgehen kann, ohne mehr Geld zu bezahlen. In dem ich ein Tunnel mache zu einer Public IP. Ich habe mir diverse VPS (Virtual Private Server) angeschaut. Es gibt Kostengünstig (3 CHF pro Monat) eine Public IP. Die Rechtleistung die man erhält ist niedrig, das stört mich nicht. Leider erhält man auch langsames Internet.

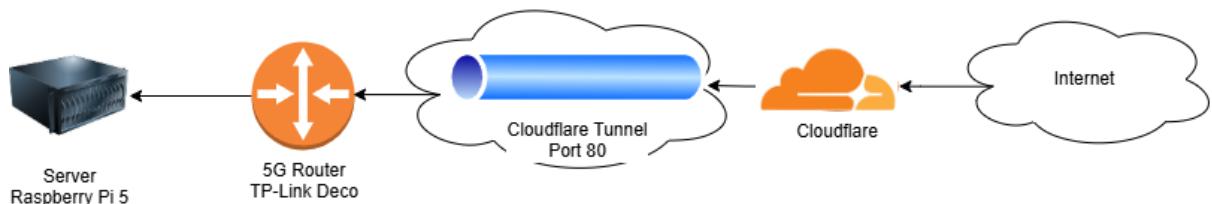


Abbildung 1: Cloudflare Tunnel

Ich bin nach langer suche und ausprobieren einiger VPS, auf eine Tolle Lösung gestossen. Ein Tunnel zu Cloudflare. Cloudflare bietet den Dienst Kostenlos an. Sie beschreiben in der AGB ein Acceptable Use.¹⁵ Einen Fileserver zu betreiben wäre zu viel, ein Service zu hosten ist in Ordnung.

Zu einem Späteren Zeitpunkt ist ein Update unumgänglich. Entweder weil die Performance des Raspberry Pi 5 nicht mehr reicht oder ich dienste Anbieten möchte welche nicht mehr mit Cloudflare Tunnel umsetzbar sind.

¹⁵ <https://www.cloudflare.com/terms/>

7.2 Variantenübersicht

Im Folgenden sieht

Variante	Bezeichnung
V1	On-Premise
V2	VPS
V3	CaaS

Tabelle 25: B3 Variantenübersicht

7.3 Soll Situation

Um ein Update auf eine andere Plattform zu vereinfachen, braucht es Planung. Etwas zu beachten ist, ist die Verwendung einer Skalierbaren Umgebung.

Folgende Anforderungen habe ich daher festgehalten bei der Umsetzung.

- Die Verwendung von Container
- Konfigurationen und Einstellungen in readme.md festhalten
- Gute Dokumentation

7.4 Variante V1 «On-Premise»

Selbst On-Premise bietet einige Optionen. Die Wahl habe ich eingeschränkt auf ARM.

7.4.1 Hardware

Modell	Price	Features	Storm kosten pro Jahr bei (0.29 pro kWh)
Raspberry Pi 5	100\$	Broadcom A76 (4x2.4 GHz) 4W – 12W ¹⁶	10 - 30 CHF
Apple Mac Mini	500\$	M4 (10x2.89 GHz) Hardware nicht Updatebar 4W – 65W ¹⁷	10 - 152 CHF
Nvidia Jetson	280\$	Rockchip A76 (4x1.8 GHz) Mehr GPU Power. M.2 4W – 40W ¹⁸	10 – 101 CHF
SolidRun HonyComb	919\$	A72 (16 x 2.2 GHz) 21W – 60W ¹⁹	53 – 152 CHF

Tabelle 26: B3 V1 Hardware

Fazit:

- Ein Raspberry Pi bietet ein sehr solides Preisleistungsverhältnis.
- Nvidias Modell würde sich eher für die Verwendung von AI lohnen
- Das SolidRun Modell bietet eine echte Alternative bei einer Aufrüstung
- Die Kosten sind Trotzdem nicht zu unterschätzen

¹⁶ <https://bret.dk/raspberry-pi-5-review/>

¹⁷ <https://support.apple.com/de-ch/103253>

¹⁸ <https://www.macnica.co.jp/en/business/semiconductor/manufacturers/nvidia/products/139797/#:~:text=Jetson%20Orin%20NX%20Module,%C2%AE%20Jetson%20Xavier%E2%84%A2%20NX.>

¹⁹ SolidRun HonyComb

7.4.2 Verantwortung

Verglichen zur Cloud habe ich gewisse Aufgaben welche mir sonst abgenommen werden.

Thema	Beschreibung	Pricing in CHF	Zeit
Sicherheitsupdates	Zertifikate können Abläufen. Updates müssen Überprüft werden.	N/A	3h pro Woche
Skalieren	Bei einem VPS kann das Per Knopfdruck gemacht werden	100-1000	4h pro mal
Hardware	Initalkosten	100-1000	1h einmalig
Redundanz	Um Ausfall zu verhindern. Brauchen wir ein Identisches System.	100-1000	1h einmalig
USV	Ein USV ist einmalig.	300	1h einmalig
Backup	Muss einmalig eingerichtet werden und Regelmässig überprüft	N/A	20h pro Jahr
Monitoring	Sollte idealerweise regelmässig Updated und verbessert werden.	N/A	3h pro Woche
Physische Sicherheit	Klimatisierung, Staubschutz, Feuer und Rauchmelder. Netzwerkschrank mit Rauchmelder und Lüftung	ab 600 CHF	1h einmalig
Storm	Annahme bei zwei Gräten	20 - 300 CHF	N/A

Tabelle 27: B3 V1 Verantwortung

7.4.3 Vorschlag mit SolidRun für mögliches Update

Folgenden Aufgelistet die Einmaligen Kosten:

Typ	Modell	Pricing in CHF
Server	SolidRun Hony-Comb x2	2000
USV	Powerwalker VI 2200	240
Netzwerkschrank	Digitus 19 Zoll Rack	485
Rauchmelder	Gude Rauchmelder	40
Backup	WD Red 4 TB x2	200
Total		3000

Tabelle 28: B3 V1 Initial Kosten

Folgenden Aufgelistet die Fortlaufenden Kosten:

Typ	Menge	Pricing pro Jahr
Strom	120 W @ 0.29 CHF	300

Tabelle 29: B3 V1 Fortlaufende Kosten

7.5 Variante V2 «VPS»

Ein VPS steht für Virtual Private Server. Um im selben Bereich zu bleiben wie On-Premise und Caas. Wäre ein VPS ein IaaS, Infrastruktur as a Service.

Damit wir VPS verstehen brauchen wir ein neues Wissen. Als Beispiel habe ich mir DigitalOcean angeschaut.

7.5.1 Terminologie

Begriff	Erklärung	Vergleich
Shared oder Dedicated	Bei einer Dedicated CPU verwendest du alleine die CPU. Weniger Schrankungen Ist etwa 2-3x teurer	Dedicated startet bei 31\$ pro Monat Shared vergleichbar 12\$ pro Monat
vCPU	Entspricht einer CPU z.B. 3GHz	
Manged vs Unmanged	Der Provider übernimmt Backups, Updates, Monitoring und Skalierung.	Datenbank unmanged ab 4\$ pro Monat Datenbank managed ab 15\$ pro Monat

Tabelle 30: B3 V2 Terminologie

7.5.2 DigitalOcean

Ich habe mir mal im Detail Digital Ocean angeschaut. Als vergleich habe ich ein Raspberry PI 5 mit 8 GB Ram und 4 x 2.5 GHz.

Unter folgenden Annahmen:

- Bei einem Wechselkurs von 1\$ = 0.88 CHF²⁰
- Strompreis von 0.29 CHF pro kWh²¹

Produkt	Specs	Preis pro Jahr in CHF
Raspberry PI		30
Droplet (shared)	Internet: 5TB SSD: 160GB	465
Droplet (dedicated)	Internet: 5TB SSD: 50GB	887
App (managed)	Internet: 800GB	1647
DB (managed)	Disk: 15GB	158

Tabelle 31: B3 V2 DigitalOcean

Fazit:

- Preisleistung ist etwas niedrig
- Managend Datenbank finde ich noch interessant

²⁰ Stand 28.03.2025, <https://wise.com/ch/currency-converter/usd-to-chf-rate?amount=1000>

²¹ <https://de.statista.com/statistik/daten/studie/329740/umfrage/haushaltstrompreis-in-der-schweiz/>

7.5.3 Hetzner

Ich habe mir mal im Detail Hetzner angeschaut. Als vergleich habe ich ein Raspberry PI 5 mit 8 GB Ram und 4 x 2.5 GHz.

Unter folgenden Annahmen:

- Bei einem Wechselkurs von 1\$ = 0.88 CHF²²
- Strompreis von 0.29 CHF pro kWh²³

Produkt	Specs	Preis pro Jahr in CHF
Raspberry PI		30
Server Shared (CX32)	Internet: 20TB SSD: 80GB	80
Server Dedicated (CCX23)	Internet: 20TB SSD: 80GB RAM 16 GB	285
Managed App	SSD: 80GB RAM 16 GB	400
Managed DB	N/A	N/A

Tabelle 32: B3 V2 Hetzner

Fazit:

- Günstiger als DigitalOcean
- Leider keine Managed Datenbanken

²² Stand 28.03.2025, <https://wise.com/ch/currency-converter/usd-to-chf-rate?amount=1000>

²³ <https://de.statista.com/statistik/daten/studie/329740/umfrage/haushaltstrompreis-in-der-schweiz/>

7.5.4 Contabo

Ich habe mir mal im Detail Contabo angeschaut. Als vergleich habe ich ein Raspberry PI 5 mit 8 GB Ram und 4 x 2.5 GHz.

Unter folgenden Annahmen:

- Bei einem Wechselkurs von 1€ = 0.95 CHF²⁴
- Strompreis von 0.29 CHF pro kWh²⁵

Produkt	Specs	Preis pro Jahr in CHF
Raspberry PI		30
Server Shared (Cloud VPS 4C)	RAM: 4GB SSD: 100GB	45.6
Server Shared (Cloud VPS 6C)	RAM: 12GB SSD: 150GB	108.3
Server Dedicated (DS1)	CPU: 12 x 3.7 GHz RAM: 32GB SSD: 1TB Internet: 32TB	1242.6
Server Backup (Cloud VPS 4C)	RAM: 4GB SSD: 100GB	59.85

Tabelle 33: B3 V2 Contabo

7.5.5 Fazit:

- Die Preise skalieren rasant
- Die Tiers hier machen grosse Sprünge
- Du erhältst beim günstigsten Dedicated auch ordentlich Leistung.

²⁴ Stand 28.03.2025, <https://wise.com/ch/currency-converter/usd-to-chf-rate?amount=1000>

²⁵ <https://de.statista.com/statistik/daten/studie/329740/umfrage/haushaltstrompreis-in-der-schweiz/>

7.6 Variante V3 «CaaS»

Nur Spasseshalber habe ich mir Container angeschaut. Bei Google, Amazon und Digital Coean. Als vergleich habe ich ein Raspberry PI 5 mit 8 GB Ram und 4 x 2.5 GHz.

Unter folgenden Annahmen:

- Bei einem Wechselkurs von 1\$ = 0.88 CHF
- Strompreis von 0.29 CHF pro kWh
-

Produkt	Specs	Preis pro Jahr in CHF
Raspberry PI		30
Digital Ocean		1647
Google (n1-standard-4)	RAM: 15GB	1561
Amazon (EC2)		1516

Tabelle 34: B3 V3 Caas

Fazit:

- Preise wirken im ersten Moment in keinem Verhältnis

7.6.1 Zeitersparnis

Folgenden aufgelistet die Fortlaufenden Kosten in Zeit und Geld.

Im Best Case Szenario wende ich Pro Monat 1 Stunde.

Im Geschätzten Case bin ich jede Woche ein Tag für alles Dran.

Bei der Annahme:

- 80 CHF pro Stunde.

Grund	h pro Jahr	CHF
Sicherheitsupdates	13	1040
Backup	20	1600
Monitoring	13	1040
Total	46	3680

Tabelle 35: B3 V1 Fortlaufende Zeit Kosten Best Case

Grund	h pro Jahr	CHF
Sicherheitsupdates	156	12480
Backup	40	3200
Monitoring	156	12480
Total	352	28160

Tabelle 36: B3 V1 Fortlaufende Zeit Kosten geschätzt

7.7 Variantenentscheid

7.7.1 Wirtschaftliche Kriterien

Hier folgen zwei verschiedene Perspektiven. Entweder meine Zeit die verliere und anstelle an einem Ort nicht Arbeiten kann. Es gibt hier auch abstreiche wie Pensionskasse, Steuern usw. Daher gibt es noch die Perspektive mit Abzügen.

Unter der Annahme:

- Brutto: 80 CHF pro Stunde.
- Netto: 40 CHF pro Stunde

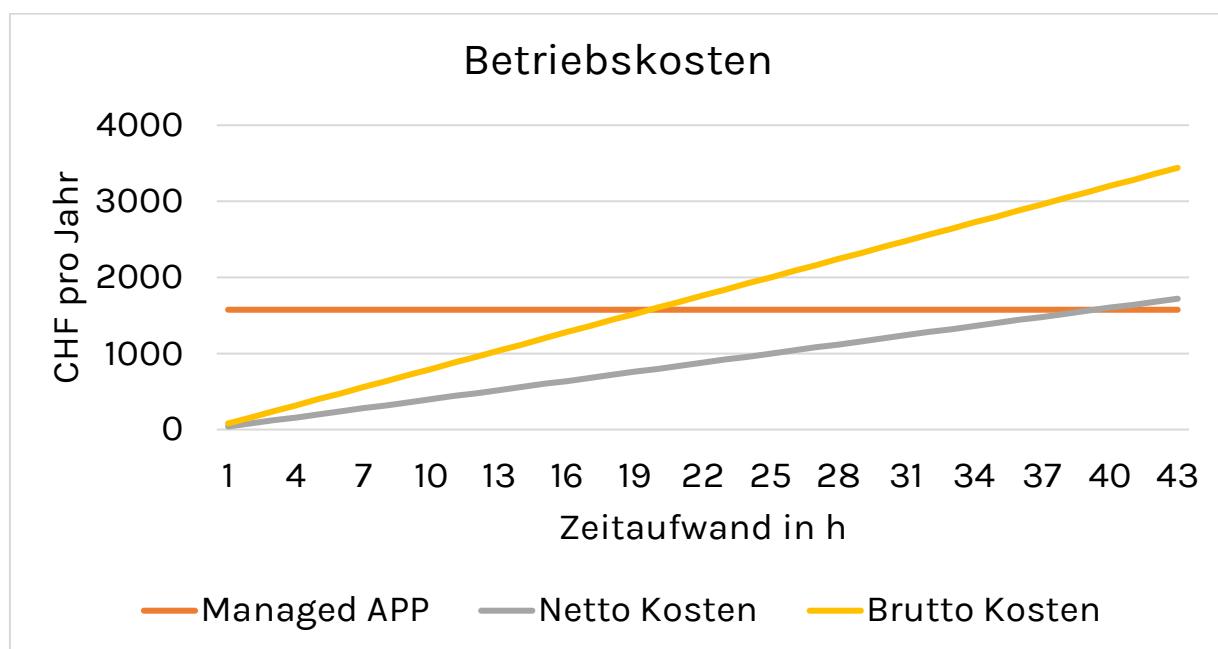


Abbildung 2: Breakeven bei Verwendung von CaaS

Fazit:

- Sobald ich mehr als 20-40h investiere lohnt sich der Managed Service bereits.

7.7.2 Nutzerbasis für Breakeven Kurve

Im Folgenden sind die Vergleiche mit ähnlichen Diensten.

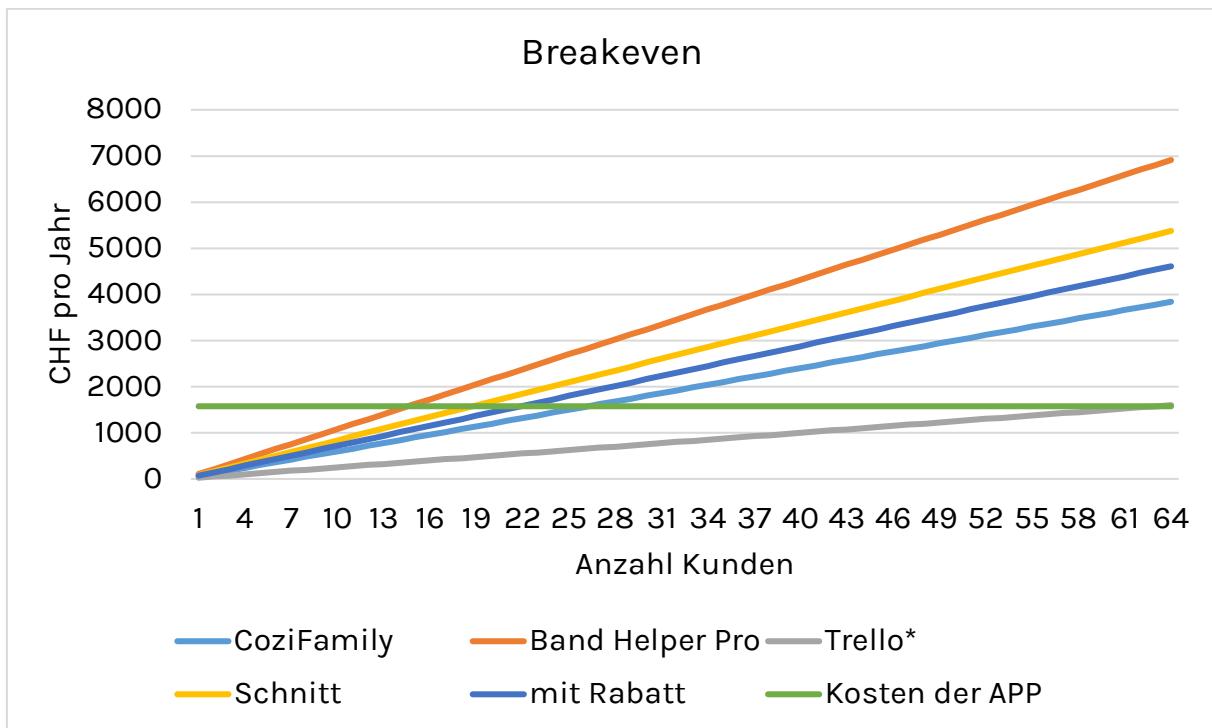


Abbildung 3: Breakeven nach Nutzer bei CaaS

* Diese Dienste haben ein Preismodell X CHF pro User pro Jahr. Unter der Annahme, dass eine Band aus 5 Personen besteht.

7.7.3 Nutzerbasis Breakeven Aufstellung

Vergleichs Modell	Preis pro Jahr	Anzahl Kunden
CoziFamily & Band Helper Basic	60	27
Band Helper Pro	108	15
Trello	25	65
Clickup	35	47
Asana	55	30
Schnittt	56	37

Tabelle 37: B3 V3 Nutzerbasis für CaaS nötig

7.7.4 Variantenwahl

Im folgenden sehen sie die Kosten im ersten Jahr:

Unter der Annahme:

- 80 CHF pro Stunde
- Verwendung von Raspberry Pi 5 und vergleichbare Performance

Model	On-Premise	VPS	Caas
Initalkosten	100		0
Unterhalkosten	28000	28000	0
Betriebskostem	30	108	1570
Total	28130	28108	1570

Tabelle 38: B3 Variantenwahl Zeitvergleich

7.7.5 Varianten Wahl B3

Die Entscheidung ist aus Wirtschaftlichen Gründen auf die Managend App gefallen. Bis zum Schluss war ich überzeugt, Self hostet ist günstiger. Jedoch ein Blick auf die Zahlen genügen und die Entscheidung ist klar. 20h entsprechen weniger als 2 Stunden pro Monat. Das halte ich nicht für möglich. Erstaunt hat mich das On-Premise sogar das Teuerste ist.

8 Wirtschaftlichkeit

Bei der Wirtschaftlichkeit hat dieses Projekt enorme Vorteil.

Es ist ausschliesslich Open-Source Technologien geplant in der Verwendung.
Also keine Lizenz kosten.

Die Serverkosten ist fast 0. Durch die Verwendung von ARM CPU sind auch die Stromkosten minimal.

Es keine grossartigen Lizenzkosten geplant.

Einzig das Marketing braucht ein gewisses Kapital.

8.1 Basierend auf der Konkurrenz

Im Folgenden ist eine Aufleistung der Konkurrenz.

Model	Prei pro Monat in CHF	Preis im Jahr in CHF	Anzahl Kunden für Betriebskosten
CoziFamily	5	60	27
Band Helper Pro	9	108	15
Schnitt	7	84	19

Tabelle 39: Wirtschaftlichkeit Preisvergleich

Fazit:

- Ein Preis bei 7 CHF scheint möglich zu sein.
- Es könnte sich ein Rabatt bei Jahre Bezahlungen Lohnen

8.2 Mögliche Preismodelle

Allein Benutzen Kostenlos	Zusammen pro Jahr 72 CHF pro Band	Zusammen pro Monat 7 CHF pro Band
------------------------------	--------------------------------------	--------------------------------------

Tabelle 40: Wirtschaftlichkeit möglichen Preismodell

Users	7 CHF	6 CHF
1	84	72
2	168	144
3	252	216
4	336	288
5	420	360
6	504	432
7	588	504
8	672	576
9	756	648
10	840	720
11	924	792
12	1008	864
13	1092	936
14	1176	1008
15	1260	1080
16	1344	1152
17	1428	1224
18	1512	1296
19	1596	1368
20	1680	1440
21	1764	1512
22	1848	1584
23	1932	1656
24	2016	1728

Tabelle 41: Wirtschaftlichkeit Umsatz pro User

8.3 Marketing

Wie viel Geld braucht?

Laut Leipziger Schule für Medien empfiehlt etwa 10% des Umsatz in Marketing zu Investieren.²⁶ Diese Webseite Spricht von fast 12%.²⁷

Wir befinden uns so in einem ähnlichen Bereich.

8.3.1 Instagram

Auf Instagram wird täglich ein Werbe Ansatz verlangt. 2 CHF ist das Niedrigste.

Plattform	CHF pro Tag	CHF Pro Monat	CHF Pro Jahr
Instagram	2	60	730

Tabelle 42: Marketing Werbung auf Instagram

Unter Folgenden Annahmen:

- Laut advetace klicken 1.5 – 2% der User auf eine Werbung²⁸. Sogenanntes CTR (Click-Trough-Ratio).
- Bei einem Versuch eine Werbung zu schalten in der Region Bern habe ich ein User Reach pro Tag Vorgeschlagen bekommen von ca. 400.

User pro Tag	User Pro Jahr	Clicks (Bei 1.5%)	Clicks (Bei 2%)
400	146000	2190	2920

Tabelle 43: Marketing Instagram Ad Reach

Im Folgenden ist eine Auflistung. Der Erreichbaren Personen Unter Folgenden Annahmen:

- Laut advetace beträgt der CPA (Click-per-Acquisition) 2-4%²⁹. Also es braucht im Schnitt 50-25 Clicks für einen neuen Kunden.

Marketing Kosten pro Jahr	Neukunden pro Jahr (bei 2%)	Neukunden pro Jahr (bei 4%)
730	44	58

Tabelle 44: Marketing Instagram Ad CPA

²⁶ <https://www.leipzigschoolofmedia.de/blog/marketingbudget/#:~:text=Wie%20viel%20ausgeben%3F,nach%20Branche%20kann%20das%20variieren.>

²⁷ <https://www.o8.agency/de/blog/marketing-strategy/marketing-budget-for-a-tech-startup>

²⁸ <https://www.advertace.de/blog/instagram-werbung-kosten>

²⁹ <https://www.advertace.de/blog/instagram-werbung-kosten>

8.3.2 Betriebskosten

In der Folgenden Aufstellung wird der Gewinn veranschaulicht. Unter Folgenden Annahmen:

- Potentieller Kundenzuwachs nach einem Jahr
- Worstcase entspricht CTR 1.5%, CPA 2% und 6 CHF pro Monat
- Bestcase entspricht CTR 2%, CPA 4% und 7 CHF pro Monat
- Gewinn = Umsatz - (Server 1570 CHF und Marketing 730 CHF)
- Vernachlässigung der Zeiten

Szenario	Umsatz (CHF)	Kosten (CHF)	Gewinn (CHF)
Worst Case	3679	2300	853.6
Best Case	4905	2300	2605.6

Tabelle 45: Marketing Werbekosten verglichen mit Einnahmen

8.4 Zeiten

Im Folgenden ist eine Aufstellung der Arbeitszeiten. Angenommen:

- 80 CHF pro Stunde
- 20% (Ein Tag pro Woche)

Aufgabe	h pro Woche	CHF	CHF pro Jahr
Marketing	3	240	12480
Neue Features	3	240	12480
Support	2	160	8320
Total	8	640	33280

Tabelle 46: Aufteilung bei 20% Pensem

Wenn wir jetzt noch die Zeit der Diplomarbeit bei der Entwicklung dazu zählen.

Aufgabe	h	CHF
Initialisierung	45	3600
Konzept	60	4800
Realisierung	100	8000
Einführung	25	2000
Abschluss	20	1600
Total	250	16400

Tabelle 47: Kosten bei der Diplomarbeit

8.5 Kosten nach dem ersten Jahr

Wenn wir annehmen der Prototyp wurde auf dem Raspi 5 zu vernachlässigbaren Kosten entwickelt. Es ist direkt nach Abschluss der Diplomarbeit Online gegangen in einem Managend Container und es wurde erst dann die Werbung im Marketing gestartet.

Position	CHF	%
Diplomarbeit	16400	31
Aufwand	33280	64
Marketingbudget	730	1
Server	1570	3
Total	51980	100

Tabelle 48: Kostenaufstellung nach dem ersten Jahr

Selbst bei 7 CHF pro Monat entspricht das:

Jahr	Kosten in CHF	Nötige Anzahl User
1.	52000	620
ab 2.	36000	430

Tabelle 49: Vergleich mit nötigen Usern

8.5.1 Wachstum Strategie

Annahme das Wachstum ist linear und die Server Kosten steigen nicht an.

Wir nehmen den Schnitt aus Best und Worst Case im Zuwachs.

Jahr	Kosten (TCHF)	User	Gewinn (TCHF)	Saldo
1	52	51	-48	-48
2	36	102	-27	-75
3	36	153	-23	-50
4	36	204	-18	-41
5	36	256	-14	-33
6	36	307	-10	-24
7	36	358	-6	-15
8	36	409	-1	-7
9	36	460	3	2
10	36	511	7	10
11	36	562	12	19
12	36	613	16	28
13	36	664	20	36

Tabelle 50: Wachstum Strategie bei 1%

Fazit:

- Mit dieser Wachstumsstrategie oder mit diesen Preisen dauert es 9 Jahre bis ich Grüne Zahlen schriebe

8.6 Kosten mit 10% Marketing

Wenn wir uns zurückbesinnen, sollten wir 10% des Umsatzes in Marketing investieren. Somit steigen auch die Gesamtkosten:

Position	CHF	%
Diplomarbeit	16400	29
Aufwand	33280	59
Marketingbudget	6000	10
Server	1570	2
Total	57250	100

Tabelle 51: Kostenaufstellung nach dem ersten Jahr bei 10%

Im Folgenden ist eine Aufstellung basierend auf dem neuen Budget von 10%. Leider gibt es 16 CHF nicht als Tages Rate bei Instagram. Daher habe ich mich für 18 CHF entschieden. Daher ist das Budget auf 11% gestiegen.

Budget pro Jahr (CHF)	Budget pro Tag (CHF)	User Reach (Min)	User Reach (Max)
6570	18	1200	3200

Tabelle 52: Instagram Ad Reach bei 11%

Unter denselben Voraussetzungen wie zuvor.

- Worstcase entspricht CTR 1.5%, CPA 2%
- Bestcase entspricht CTR 2%, CPA 4%

Case	Reach pro Jahr	Clicks	Neue User
Worst	438'000	6570	131
Best	1'168'000	23360	467

Tabelle 53: Instagram CPA bei 11%

8.6.1 Wachstum Worstcase

Annahme das Wachstum ist linear, und die Server Kosten steigen nicht an.

Jahr	Kosten (TCHF)	User	Gewinn (TCHF)	Saldo
1	58	131	-48	-48
2	41	263	-22	-71
3	41	394	-13	-36
4	41	526	-4	-17
5	41	657	6	2
6	41	788	15	21
7	41	920	25	40
8	41	1'051	34	59
9	41	1'183	44	78
10	41	1'314	53	97
11	41	1'445	63	116
12	41	1'577	72	135
13	41	1'708	82	154

Tabelle 54: Wachstum Strategie bei 11% Worstcase

Fazit:

- Es dauert immer noch 5 Jahre bis das Projekt Grüne Zahlen schreibt.

8.6.2 Wachstum Bestcase

Annahme das Wachstum ist linear, und die Server Kosten steigen nicht an.

Jahr	Kosten (TCHF)	User	Gewinn (TCHF)	Saldo
1	58	467	-19	-19
2	41	934	37	18
3	41	1'402	76	113
4	41	1'869	116	192
5	41	2'336	155	270
6	41	2'803	194	349
7	41	3'270	233	427
8	41	3'738	273	506
9	41	4'205	312	584
10	41	4'672	351	663
11	41	5'139	390	741
12	41	5'606	430	820
13	41	6'074	469	898

Tabelle 55: Wachstum Strategie bei 10% Bestcase

Fazit:

- Bereits Ende 2. Jahres schreibt das Projekte Grüne Zahlen.
- Ab dem 4. Jahr könnte ich die Vollzeit machen

8.6.3 Reelle Ausgaben

Wenn das Projekt wirklich Fliegen lasse, gehe ein Risiko ein. Die Zeit Investition ist eines, das andere sind kosten welche ich an Giganten wie Instagram und Google verschenke.

Hier die Verschiedenen Szenarien als Kurve.

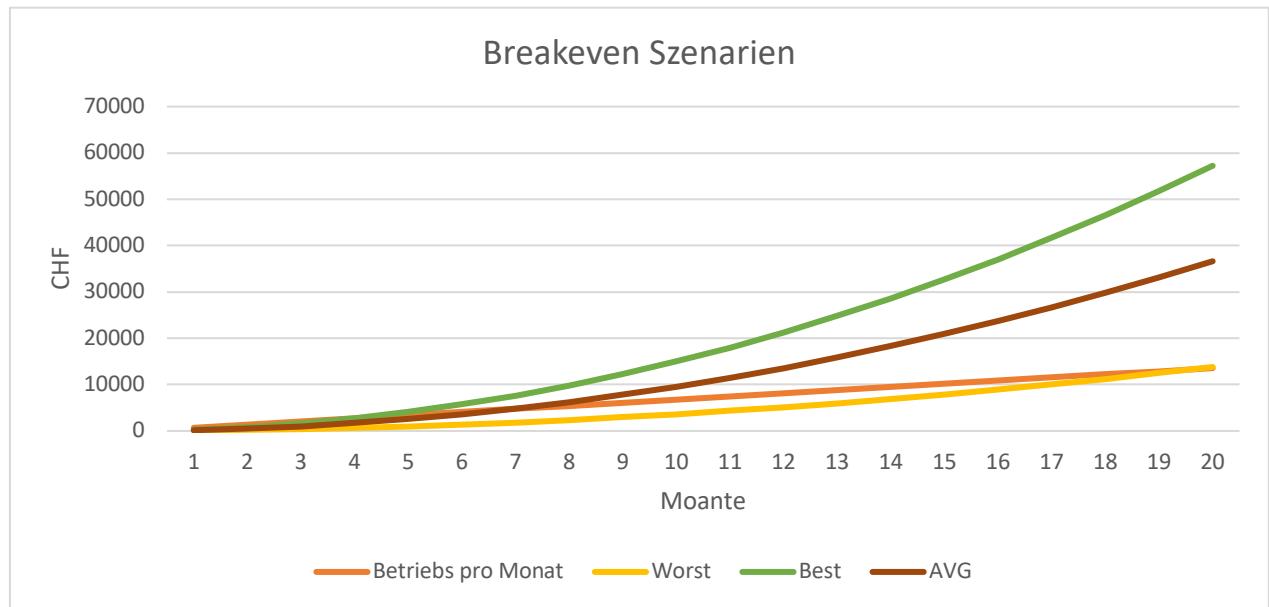


Abbildung 4: Reelle Ausgaben

Szenario	Breakeven Monat	Verlust Potential in CHF
Worst	19	12210
Best	5	2700
AVG	7	4070

Tabelle 56: Verlustpotential verschiedener Szenarien

Fazit:

- Es ist immer noch ein Grosses Risiko ich könnte mehrere Tausende CHF Verpulvern. Um herauszufinden, dass es nicht funktioniert.

9 Anwendungsarchitekturen

Die Anwendung muss

9.1 Optionen

Folgende Möglichkeiten stehen zu Option:

Architektur	Beschreibung
Monolith	Alle Komponenten in einem einzigen Deployment-Paket.
Microservices	Aufgeteilt in Services (User-Service, Project-Service usw.)
Orechstrierung	Container-Lösung mit Kubernetes Management
Serverless	Function-as-a-Service (Cloud-Provider)
Hybrid-Architektur	Monolith mit klarer modularer Trennung, Container-basiert

Tabelle 57

9.2 Gewichtung

Basierend auf den Zahlen von Zuvor. Also etwa Breakeven nach 5 Jahren und mit 650+ Usern.

Kriterium	Gewichtung	Begründung
Entwicklungsaufwand	25%	Kritisch für Ein-Personen-Projekt
Betriebskosten	20%	Budget-Constraint
Wartbarkeit	15%	Langfristige Entwicklung geplant
Skalierbarkeit	15%	650 User in 5 Jahren
Ausfallsicherheit	10%	Wichtig für Reputation
Security	10%	DSGVO-Compliance erforderlich
Performance	5%	Nicht kritisch

Tabelle 58

9.3 Bewertung

Basierend auf den Kriterien werden die vorhandenen folgendermassen gewichtet.

Bewertungsskala: 1 = sehr schlecht, ..., 5 = sehr gut

Kriterium	Monolith	Microservices	Kubernetes	Serverless	Hybrid
Entwicklungsaufwand	5	2	3	3	4
Betriebskosten	4	3	2	4	4
Wartbarkeit	3	4	4	3	4
Skalierbarkeit	2	5	5	5	3
Ausfallsicherheit	2	4	5	5	3
Security	3	3	4	5	4
Performance	4	3	4	3	4

Tabelle 59

9.4 Gewichtete Bewertung

Basierend auf der Bewertung werden die jetzt die Kriterien gewichtet, um einen Sieger zu erkennen.

Architektur	Gewichtung	Ranking
Hybrid	3.65	1
Monolith	3.55	2
Serverless	3.50	3
Kubernetes	3.45	4
Microservices	3.15	5

Tabelle 60

9.5 Fazit

Die Hybride Architektur hat sich hier am besten bewährt. Sie ist zwar nicht so skalierbar oder speziell Ausfallsicher, hat hier aber am konstantesten Performend und eignet sich für ein Ein-Mann-Projekt dieser grösste am besten.

Die Applikation ist zu wenig Komplex für Micro-Services und hat einen zu kleinen Nutzen an der Serverless Architekturen um diese Komplexität zu rechtfertigen.

10 Deployment Architekturen

10.1 Optionen

Deployment	Beschreibung
Blue-Green	Zwei Umgebungen welche sich Abtauschen
Rolling	Mehrere Container werden schrittweise geupdated
Canary	Zwei Container, z.B: 5%, läuft bereits auf Version B.

Tabelle 61

10.2 Kriterien

Kriterium	Gewichtung	Begründung
Implementierungsaufwand	25 %	Limitierte Ressourcen bei Ein-Mann-Projekt
Kosten	20 %	Budget-Constraint
Risikominimierung	20 %	Keine Ausfälle wegen Reputation
Wartungsaufwand	15 %	Möglichst tief
Rollback-Fähigkeit	10 %	Wichtig bei Problemen
Testing-Möglichkeiten	5 %	Feature-Validierung bei bestimmten Bands
Skalierbarkeit	5 %	Langsames Wachstum erwartet

Tabelle 62

10.3 Bewertung

Bewertungsskala: 1 = sehr schlecht, ..., 5 = sehr gut

Kriterium	Blue-Green	Rolling	Canary
Implementierungsaufwand	2	4	3
Kosten (Infrastruktur)	1	5	4
Risikominimierung	5	3	5
Wartungsaufwand	2	4	3
Rollback-Fähigkeit	5	2	4
Testing-Möglichkeiten	2	2	4
Skalierbarkeit	4	5	4

Tabelle 63

10.4 Gewichtung

Basierend auf der Bewertung werden, die jetzt die Kriterien gewichtet, um einen Sieger zu erkennen.

Deployment-Strategie	Gewichteter Score	Ranking
Rolling	3.751	1
Canary	3.702	2
Blue-Green	2.853	3

Tabelle 64

10.5 Fazit

Die Rolling Deployment Strategie hat in diesem Projekt gewonnen. Die Einfachheit in einem Ein-Mann Projekt wurde stark gewichtet. Die Kosten einer Doppelten Umgebung bei Blue Green ist sehr unattraktiv. Das Ausrollen für einen Teil der User scheint mir kompliziert. Kann in Zukunft auch eine interessante alternative sein.

Abkürzungen und Glossar

Abkürzung / Fachwort	Erläuterung
HERMES	Vorgehensmethodik für Projekte und Programme HERMES 5 ist ein eCH Standard

Tabelle 65: Abkürzungen und Glossar

Inhaltsverzeichnis

1 Informationsbeschaffung	2
1.1 Konkurrenz Analyse	2
2 Ziele	3
2.1 Projektziele.....	3
2.2 Projektvorgehensziele	5
3 Grobanforderungen.....	6
4 Grundsatzentscheidungen.....	7
4.1 Grundsatzentscheidungen Übersicht	7
5 Lösungsvarianten von B1.....	8
5.1 Variantenübersicht	8
5.2 Variante V1 «Weniger Admin mehr Musik».....	8
5.2.1 Kurzbeschreibung.....	8
5.2.2 Zielegruppe	9
5.2.3 Konkurrenz Analyse	10
5.3 Variante V2 «Familien und Bands»	11
5.3.1 Kurzbeschreibung.....	11
5.3.2 Zielegruppe	12
5.3.3 Konkurrenz Analyse	13
5.4 Analyse und Bewertung der Varianten.....	14
5.4.1 Zielerreichung.....	14
5.5 Variantenwahl.....	14
6 Lösungsvariantenwahl von B2.....	15
6.1 Variantenübersicht	15
6.2 Variante V1 «WebOnly»	16
6.2.1 Kurzbeschreibung.....	16
6.2.2 Vorteile	16
6.2.3 Nachteile	16
6.3 Variante V2 «WebViewApp».....	17
6.3.1 Kurzbeschreibung.....	17
6.3.2 Vorteile	17
6.3.3 Nachteile	17
6.4 Variante V3 «Native App».....	18
6.4.1 Kurzbeschreibung.....	18
6.4.2 Vorteile	18
6.4.3 Nachteile	18
6.5 Variante V4 «Cross Plattform».....	19
6.5.1 Kurzbeschreibung.....	19
6.5.2 Vorteile	19

6.5.3	Nachteile	19
6.6	Bewertungskriterien	20
6.6.1	Kriterien.....	20
6.6.2	Nicht Kriterien.....	20
6.7	Analyse und Bewertung der Varianten.....	21
6.7.1	Bewertung.....	21
6.7.2	Analyse	Error! Bookmark not defined.
7	Lösungsvariantenwahl von B3.....	22
7.1	Ausgangslage.....	22
7.2	Variantenübersicht.....	23
7.3	Soll Situation.....	23
7.4	Variante V1 «On-Premise»	24
7.4.1	Hardware.....	24
7.4.2	Verantwortung.....	25
7.4.3	Vorschlag mit SolidRun für mögliches Update.....	26
7.5	Variante V2 «VPS»	27
7.5.1	Terminologie	27
7.5.2	DigitalOcean.....	28
Fazit:	28	
7.5.3	Hetzner.....	29
Fazit:	29	
7.5.4	Contabo	30
Fazit:	30	
7.6	Variante V3 «CaaS».....	31
Fazit:	31	
7.6.1	Zeitersparnis.....	32
7.7	Variantenentscheid	33
7.7.1	Wirtschaftliche Kriterien.....	33
7.7.2	Nutzerbasis für Breakeven Kurve	34
7.7.3	Nutzerbasis Breakeven Aufstellung.....	34
7.7.4	Variantenwahl.....	35
7.7.5	Varianten Wahl B3.....	35
8	Wirtschaftlichkeit.....	36
8.1	Basierend auf der Konkurrenz.....	36
8.2	Möglichen Preismodell.....	37
8.3	Marketing	38
8.3.1	Instagram	38
8.3.2	Betriebskosten.....	39
8.4	Zeiten	39
8.5	Kosten nach dem ersten Jahr	40
8.5.1	Wachstum Strategie	41
8.6	Kosten mit 10% Marketing	42

8.6.1	Wachstum Worstcase	43
8.6.2	Wachstum Bestcase	44
8.6.3	Reelle Ausgaben	45

Abbildungsverzeichnis

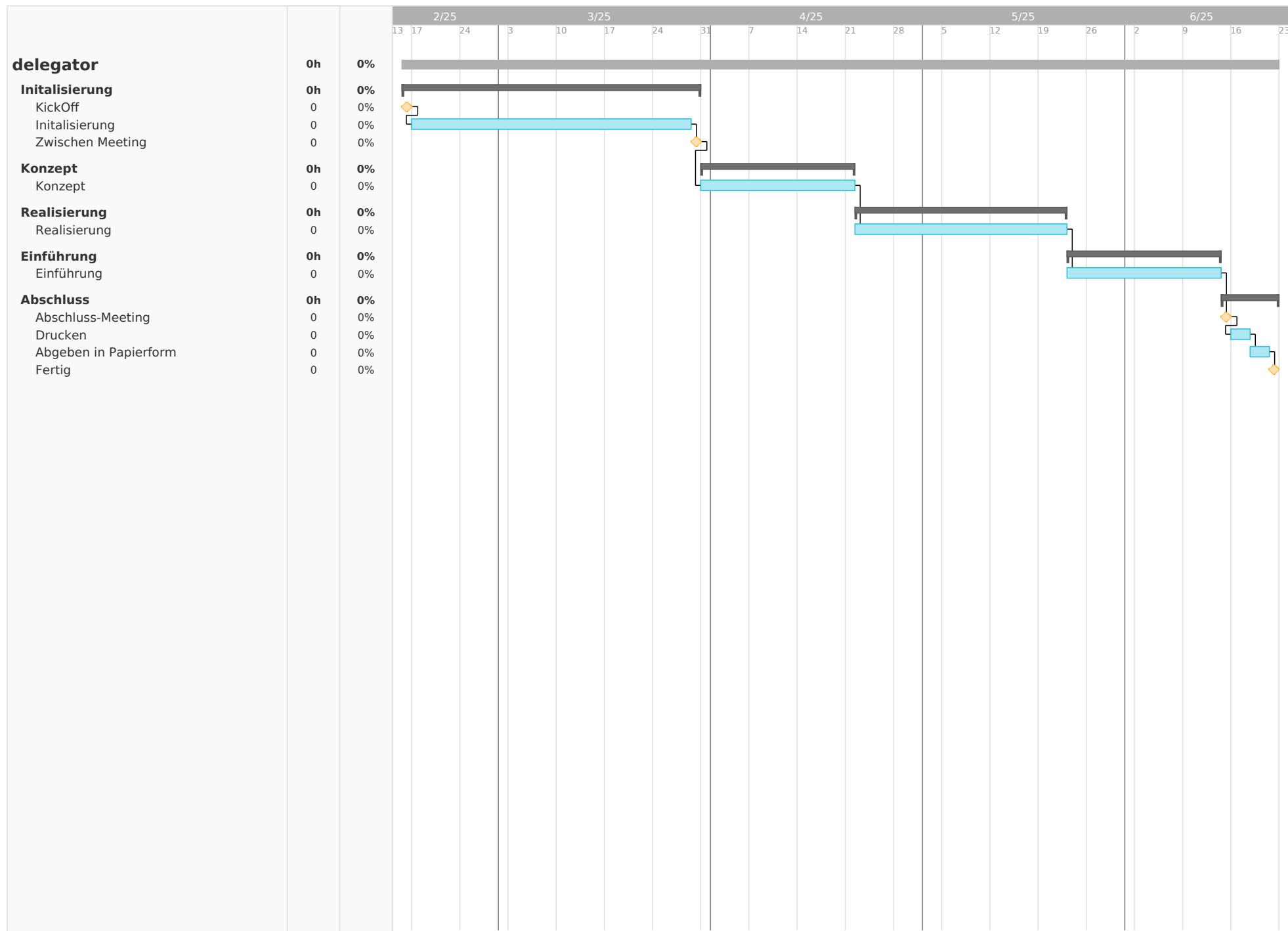
Es konnten keine Einträge für ein Abbildungsverzeichnis gefunden werden.

Tabellenverzeichnis

Tabelle 1: Änderungsverzeichnis.....	1
Tabelle 2: Informationsbeschaffung Konkurrenz.....	2
Tabelle 3: Ziele	4
Tabelle 4: Projektvorgehensziele	5
Tabelle 5: Grobanforderungen.....	6
Tabelle 6: Grundsatz Entscheidungen.....	7
Tabelle 7: Variantenübersicht von B1.....	8
Tabelle 8: B1 V1 Zielgruppe.....	9
Tabelle 9: B1 V1 Konkurrenzanalyse.....	10
Tabelle 10: B1 V2 Zielgruppe.....	12
Tabelle 11: B1 V2 Konkurrenz Analyse	13
Tabelle 12: Zielerreichung B1.....	14
Tabelle 13: Variantenübersicht von B2.....	15
Tabelle 14: B2 V1 Vorteile	16
Tabelle 15: B2 V1 Nachteile	16
Tabelle 16: B2 V2 Vorteile.....	17
Tabelle 17: B2 V2 Nachteile	17
Tabelle 18: B2 V3 Vorteile	18
Tabelle 19: B2 V3 Nachteile.....	18
Tabelle 20: B2 V4 Vorteile	19
Tabelle 21: B2 V4 Nachteile.....	19
Tabelle 22: B2 Kriterien	20
Tabelle 23: B2 Nicht Kriterien	20
Tabelle 24: B2 Bewertung.....	21
Tabelle 25: B3 Variantenübersicht.....	23
Tabelle 26: B3 V1 Hardware.....	24
Tabelle 27: B3 V1 Verantwortung.....	25
Tabelle 28: B3 V1 Initial Kosten	26
Tabelle 29: B3 V1 Fortlaufende Kosten	26
Tabelle 30: B3 V2 Terminologie	27
Tabelle 31: B3 V2 DigitalOcean.....	28
Tabelle 32: B3 V2 Hetzner.....	29
Tabelle 33: B3 V2 Contabo	30
Tabelle 34: B3 V3 Caas	31
Tabelle 35: B3 V1 Fortlaufende Zeit Kosten Best Case	32
Tabelle 36: B3 V1 Fortlaufende Zeit Kosten geschätzt	32
Tabelle 37: B3 V3 Nutzerbasis für CaaS nötig	34

Tabelle 38: B3 Variantenwahl Zeitvergleich	35
Tabelle 39: Wirtschaftlichkeit Preisvergleich.....	36
Tabelle 40: Wirtschaftlichkeit mögliches Preismodell.....	37
Tabelle 41: Wirtschaftlichkeit Umsatz pro User.....	37
Tabelle 42: Marketing Werbung auf Instagram	38
Tabelle 43: Marketing Instagram Ad Reach	38
Tabelle 44: Marketing Instagram Ad CPA	38
Tabelle 45: Marketing Werbekosten verglichen mit Einnahmen.....	39
Tabelle 46: Aufteilung bei 20% Pensum	39
Tabelle 47: Kosten bei der Diplomarbeit.....	40
Tabelle 48: Kostenaufstellung nach dem ersten Jahr.....	40
Tabelle 49: Vergleich mit nötigen Usern	40
Tabelle 50: Wachstum Strategie bei 1%.....	41
Tabelle 51: Kostenaufstellung nach dem ersten Jahr bei 10%	42
Tabelle 52: Instagram Ad Reach bei 10%	42
Tabelle 53: Instagram CPA bei 10%.....	42
Tabelle 54: Wachstum Strategie bei 10% Worstcase.....	43
Tabelle 55: Wachstum Strategie bei 10% Bestcase	44
Tabelle 56: Verlustpotential verschiedener Szenarien	45
Tabelle 57: Abkürzungen und Glossar	50

Anhang D



Anhang E1

Datenschutz

Delegator

Klassifizierung intern
Status in Arbeit
Programmname Delegator
Projektnummer 1
Projektleiter Tabinas Kenan
Version 1
Datum 06. Juni 2025
Auftraggeber Tabinas Kenan
Autor/Autoren Tabinas Kenan
Verteiler

Änderungsverzeichnis

Version	Datum	Änderung	Autor
1	06.06	Erstellt	TAK

Tabelle 1: Änderungsverzeichnis

1 Übersicht und Zweck

Es folgt ein Managementsummary. Damit ein gemeinsames Verständnis gegenüber der Applikation und dessen Inhalt aufgebaut werden kann.

1.1 Management Summary

1.2 Zweck dieses Konzeptes

Es ist nicht nur etwaige datenschutzrechtliche Anforderungen zu erfüllen wie z.B. die der DSGVO oder das Schweizerische Datenschutzgesetz (nDSG).

Sondern auch die Daten der Kunden zu Respektieren. Es besteht die Möglichkeit das die Applikation Informationen von Persönlichkeiten der Öffentlichkeit verwendet wird. Es ist im gemeinsamen Interesse, dass diese Daten mit höchster Sorgfalt behandelt werden.

2 Rechtliche Grundlagen

2.1 Bestehendes Recht

Folgende Bereits erwähnte Rechtsgrundlagen sind bereits in Kraft. Dabei beschränken wir uns auf den DACH (Deutschland, Austria, Schweiz) Raum.

- DSGVO: Allen EU-Bürgern
- nDSG: Allen Schweizern-Bürgern

2.2 Verantwortlichkeiten

- Delegator: Kenan Tabinas
- Service: CaaS Provider

2.3 Wichtigkeit

Jetzt stellt sich die Frage: Wieso ist das für uns Relevant.

- Aus Respekt gegenüber dem Kunden
- DSGVO-Busgelder bis zu 4% des Jahreseinkommen.¹
- nDSG-Busgelder bis zu 250'000 CHF.²
- Ein Muss für Investoren³

¹ <https://www.datenschutz.org/dsgvo-bussgeld/>

² https://haerting.ch/wissen/strafbestimmungen_des_neuen_datenschutzgesetzes/

³ <https://www.datenschutzelxperte.de/blog/die-rolle-der-dsgvo-beim-unternehmenskauf-due-diligence>

3 Datenarten und Zweck

3.1 Personenbezogene Daten

3.1.1 Stammdaten

- E-Mail-Adresse
- Benutzername
- Passwort (verschlüsselt)
- Profilbild (optional)
- Zugehörigkeit zu Organisation

3.1.2 Nutzungsdaten

- Termine
- Kalenderdaten
- Aufgaben
- Projekte
- Setlisten
- Songs
- Chat Nachrichten
- Zugriffszeiten (Sicherheit)
- Speicherbedarf (Monitoring von Wachstum)

3.2 Zweck

Alle diese Daten dienen nur der Verfügbarkeit des Dienstes.

Sie sollte nicht für Marketing oder andere Zwecke missbraucht werden.

4 Rechtsgrundlagen der Verarbeitung

Ohne die Rechtliche Grundlage

4.1 Art. 6 DSGVO / Art. 13 nDSG

4.1.1 Vertragserfüllung (Art. 6 Abs. 1 lit. b DSGVO):

- Benutzerkonto-Verwaltung
- Bereitstellung der Kernfunktionen
- Kalender- und Aufgabenverwaltung

4.1.2 Berechtigtes Interesse (Art. 6 Abs. 1 lit. f DSGVO):

- Systemsicherheit und Betriebsstabilität
- Fehleranalyse und technische Optimierung
- Schutz vor Missbrauch

4.1.3 Einwilligung (Art. 6 Abs. 1 lit. a DSGVO):

- Marketing-E-Mails
- Erweiterte Analytics
- Nicht-essenzielle Cookies

Text stammt aus Claude.

5 Wiederrufsrecht

Damit die Applikation funktioniert werden gewisse Daten benötigt. Die zwingenden Cookies die man von Webseiten kennt. Man macht sich Strafbar, wenn die Cookies ablehnt und die Applikation nicht mehr funktioniert.

Basierend auf den Artikel von Zuvor (Art. 6 Abs. 1 lit. b DSGVO und Art. 13 Abs. 2 lit. a nDSG) haben wir für das Projekt eine klare Abgrenzung zwischen vertragserfüllungsrelevanter und einwilligungspflichtiger Datenverarbeitung vorgenommen.

5.1 Vertragserfüllung

5.1.1 Kalenderfunktion

- Termine speichern Nötig für versprochenen Service
- Erinnerungen senden Teil der Kalenderfunktion
- Mit anderen Bandmitgliedern teilen Kern-Feature

5.1.2 Aufgabenverwaltung

- Aufgaben erstellen/zuweisen Versprochener Service
- Status-Updates Nötig für Funktionalität
- Benachrichtigungen Teil des Services

5.1.3 Benutzerkonto

- Name, E-Mail speichern Nötig für Account
- Login-Daten Nötig für Zugang
- Gruppenzugehörigkeit Nötig für Band-Features

5.2 Nicht Vertragserfüllung fällt:

5.2.1 Marketing:

- Newsletter über neue Features Nicht für Service nötig
- Werbung für andere Produkte Nicht vertraglich vereinbart
- Marktforschung Nicht Teil des Services

5.2.2 Analytics:

- Detailliertes Nutzerverhalten Nicht nötig für Service
- A/B-Tests Nicht vertraglich vereinbart
- Performance-Optimierung Kann auch ohne erfolgen

Text aus Claude

6 Technische Massnahmen

6.1 Wichtigkeit

Ein Leak von Privaten Daten zerstört nicht nur das Vertrauen, Existenz und kann auch Schadensersatzforderungen kommen.

6.2 Verschlüsselung

- **In Transit:** TLS 1.3 für alle Datenübertragungen
- **At Rest:** AES-256 Verschlüsselung für sensible Daten
- **Passwörter:** bcrypt mit Salt, mindestens 12 Rounds

Vorschläge von Claude.

6.3 Zugriffskontrolle

- Multi-Faktor-Authentifizierung
- Rollenbasierte Zugriffsrechte
- Kurze Timeout Intervalle
-

6.4 Einstellungen für die Organisation

Folgende Einstellungen könnten in der Organisation gewählt werden. Damit folgende Richtlinien vom Organisation Admin auferlegt wird anstelle vom Unternehmen.

- Multi-Faktor zwingend für alle User der Organisation
- Passwort des Users wird Generiert (Passwort Manager zwang)

7 Massnahmen

7.1 US Server

Weshalb ist die Wahl des Cloud Providers Relevant? Ich bin kein Jurist, aber laut dem Scherben Urteil II scheint der Fall klar zu sein. Das Schnutzniveau von dem US Privacy Shield nicht ausreichend.

«Mit Urteil vom 16. Juli 2020 (Rechtssache C 311/18 – „Schrems II“) hat der EuGH diesen Durchführungsbeschluss zum Privacy Shield für unwirksam erklärt.»⁴

«Bezüglich der Standarddatenschutzklauseln (Standardvertragsklauseln) hat der EuGH im Schrems II-Urteil entschieden, dass diese grundsätzlich weiterhin genutzt werden können. Allerdings muss tatsächlich ein Schutzniveau für die personenbezogenen Daten sichergestellt sein, das dem in der Europäischen Union entspricht»⁵

Das heisst nicht Schützenswerte Daten dürfen auch auf US Servern liegen, alle anderen müssen in der mindestens in der EU bleiben.

7.2 US Providers

Leider endet es hier nicht. In den US gilt der CLOUD Act. US Behörden können die Herausgabe von Daten von Unternehmen mit Sitz in USA fordern. Dies trifft auch zu wenn die Server in beispielsweise in der Schweiz liegen.

«Datenschutz-Experten sehen hier einen klaren Konflikt mit der Datenschutzgrundverordnung, die Unternehmen die Übergabe von innerhalb der EU gesicherten Daten ohne Rechtshilfeabkommen verbietet (vgl. Artikel 48 DSGVO). Ein Verstoß gegen die in Artikel 48 aufgeführten Pflichten kann nach Art. 83 DSGVO mit Bußgeldern in Höhe von bis zu 20 Millionen Euro bzw. vier Prozent des weltweiten Jahresumsatzes geahndet werden.»⁶

7.3 Kalender Integration

Bei der Verwendung von Kalender Integration wie z.B. Google Calender muss auch darauf geachtet werden, das möglich wenig Daten übertragen werden.

⁴ https://www.lfd.niedersachsen.de/startseite/themen/internationaler_datenverkehr/das_schrems_ii_urteil_des_eugh_und_seine_bedeutung_fur_datentransfers_in_drittlander/das-schrems-ii-urteil-des-europaischen-gerichtshofs-und-seine-bedeutung-fur-datentransfers-in-drittlander-194085.html

⁵https://www.lfd.niedersachsen.de/startseite/themen/internationaler_datenverkehr/das_schrems_ii_urteil_des_eugh_und_seine_bedeutung_fur_datentransfers_in_drittlander/das-schrems-ii-urteil-des-europaischen-gerichtshofs-und-seine-bedeutung-fur-datentransfers-in-drittlander-194085.html

⁶ <https://blog.idgard.com/de/us-cloud-act-vs-datenschutz/>

8 Betroffenenrecht

Das Betroffenenrecht, anders als ich gedacht habe, hat nichts mit Verstorbenen zu tun. Sondern beschreibt Rechte, welche ich habe, wenn ich in der EU meine Daten einem Unternehmen gebe.

8.1 Auskunftsrecht

Alle Daten müssen jeder Zeit einsehbar sein.

«Mit dem Auskunftsrecht garantiert Ihnen Art. 15 der Datenschutz-Grundverordnung (DSGVO) ein bedeutsames Betroffenenrecht. Danach können Sie als betroffene Person von dem für die Datenverarbeitung Verantwortlichen Auskunft darüber verlangen, welche Daten dort über Sie gespeichert sind bzw. verarbeitet werden.»⁷

8.2 Recht auf Berichtigung

Alle Daten müssen jeder Zeit änderbar sein.

«Das Recht auf Berichtigung ist sehr klar und schnell zusammengefasst: Wenn Ihnen auffallen sollte, dass Sie betreffende Daten unrichtig sind, so können Sie deren unverzügliche Berichtigung verlangen.»⁸

8.3 Einschränkung auf Verarbeitung

Der Account muss einfrierbar / deaktivierbar sein.

«Während der Einschränkung dürfen Ihre Daten nur noch gespeichert, aber nicht mehr auf andere Weise verarbeitet werden. Die Einschränkung dient dazu, Ihre Rechte in Ausgleich mit den Rechten des für die Verarbeitung Verantwortlichen zu bringen.»

8.4 Recht auf Vergessenwerden

Der User hat recht das seine Daten gelöscht werden.

«Das Recht auf Löschung ist eines der zentralen Werkzeuge zur Durchsetzung Ihrer datenschutzrechtlichen Selbstbestimmung. Mit diesem Recht können Sie die restlose Entfernung Ihrer personenbezogenen»⁹

Dabei gibt es eine Ausnahme, wenn ich z.B. die Aufbewahrungspflicht für Steuererklärung.

«Nach Art. 17 Abs. 3 DSGVO gibt es Ausnahmen vom Recht auf Löschung, die direkt in der DSGVO geregelt sind Betroffenenrechte der DSGVO - Das Recht auf Löschung / "Recht auf Vergessenwerden" (Art. 17 DSGVO). Eine wichtige Ausnahme ist, wenn andere Rechtsvorschriften eine Aufbewahrung verlangen.»

⁷ https://www.bfdi.bund.de/DE/Buerger/Basiswissen/Betroffenenrechte/BetroffenenRechte_node.html

⁸ https://www.bfdi.bund.de/DE/Buerger/Inhalte/Allgemein/Betroffenenrechte/Betroffenenrechte_Berichtigung.html

⁹ https://www.bfdi.bund.de/DE/Buerger/Inhalte/Allgemein/Betroffenenrechte/Betroffenenrechte_L%C3%B6schung_Vergessenwerden.html

9 Incident Response

9.1 Datenschutzverletzungen

9.1.1 Erkennungsmaßnahmen

- **Monitoring:** Automatische Erkennung ungewöhnlicher Zugriffe
- **Logging:** Umfassende Protokollierung sicherheitsrelevanter Ereignisse
- **Alerting:** Sofortige Benachrichtigung bei kritischen Ereignissen

9.1.2 Response-Prozess

- **Sofortmaßnahmen:** Eindämmung der Verletzung (binnen 1 Stunde)
- **Bewertung:** Risikoanalyse und Betroffenen-Impact (binnen 4 Stunden)
- **Meldung:** An Aufsichtsbehörde binnen 72 Stunden
- **Benachrichtigung:** Der Betroffenen bei hohem Risiko
- **Dokumentation:** Vollständige Aufzeichnung des Vorfalls

9.2 Meldeverfahren

- **Interne Meldung:** Klare Eskalationswege
- **Behördenmeldung:** Template für einheitliche Meldungen
- **Betroffenen-Information:** Verständliche Kommunikation

Erstellt von Claude

10 Roadmap

10.1 Grundlagen

Zeitraum: Monat 1-2

- Datenschutzerklärung erstellen
- Cookie-Consent-System implementieren
- Grundlegende Sicherheitsmaßnahmen
- Betroffenenrechte-Interface

10.2 Erweiterte Massnahmen

Zeitraum: Monat 3-4

- Verschlüsselung at Rest implementieren
- Audit-Logging einrichten
- Incident Response Prozesse
- Mitarbeiter-Schulungen

10.3 Optimierung

Zeitraum: Monat 5-6

- Automatisierte Löschung
- Erweiterte Analytics (privacy-compliant)
- Penetration Testing
- Compliance-Review

11 Fazit

11.1 Weniger ist mehr

Weniger ist mehr. Also nein weniger ist nie mehr. Aber weniger Daten zu sammeln hat mehrere Vorteile:

- Weniger Speicherplatz
- Weniger Verpflichtungen

11.2 Rechte

Jeder User muss folgende Rechte haben:

- Recht auf Dateneinsicht
- Recht auf Deaktivierung
- Recht auf Vergessenwerden

11.3 Verantwortung

Ich bin verpflichtet die Daten zu schützen.

- Monitoring
- Logging
- Alarmierung
- Pentesting

11.4 Reaktion

Sobald etwas passiert ist muss ich folgende Schritte Einleiten:

- Sofortmassnahmen
- Betroffenen User Informieren
- Behörden Informieren
- Dokumentieren

11.5 Zeitplan

Damit diese nicht gewinneinbringenden Massnahmen nichts in Vergessenheit geraten und geschäftsschädigend werden ist es wichtig einen Plan zu haben.

11.5.1 Vor der Veröffentlichung

- Kurzfristig ist unter anderem folgende Punkte zu beachten:
- Oberfläche, um Rechte auszuüben
- AGB erstellen
- Sicherheitsmassnahmen
- Rechtskonformer Cloud Provider finden

11.6 Zukunft

Langfristig sind unter anderem folgende Punkte zu beachten:

- Datenschutz Audit mit Externen
- Penetration Test von Externen
- Incidents Report Tests
- Datenschutz Themen im Budget einplanen

Abkürzungen und Glossar

Abkürzung / Fachwort	Erläuterung
DSGVO	Datenschutzgrundverordnung
nDSG	National (Schweiz) Datenschutz Gesetz
DACH	(Deutschland, Austria, Schweiz)
bcrypt	Eine Art Passwort hashing.
API	Steht für Application Programming Interface. Ist eine Schnittstelle, die in Form von Endpoints, Daten via JSON bereitstellen oder verarbeiten.
Bearer Token	Theoretisches Konzept für Authentifizierung in diesem Fall umgesetzt mit JWT in Django
Django REST	Ein Framework zur Erweiterung Django für die Entwicklung von REST-APIs
Docker	Plattform für bereitstellen von Container. Ähnlich wie ein VMWare aber für Container anstelle von Containern.
Flutter	Crossplattform Framework. Analog zu .Net Maui und React Native
Gunicorn	Python HTTP Server. Als Brücke zwischen Django und nginx.
Hot-Reload	Ermöglicht Neuladen von Code-Änderungen ohne Neustart
IoT	Steht für Internet of Things. Beispiel Kühlschrank, welcher dir sagt, was du noch zuhause hast.
JWT	Steht für JSON Web Token. Ist ein Alphanumerischer Code, welcher bei der Authentifizierung genutzt wird.
Wireframe	Visueller Entwurf einer Benutzeroberfläche. Verglichen zum Mockup keine Farbe und keine Bilder.
Postman	Software zum Testen und Entwickeln von APIs. In diesem Fall als VS Code Extension.
Raspberry Pi (Raspi)	Singleboardcomputer. In diesem Fall Raspi 5.

Tabelle 2: Glossar 1

Vorschläge der Wörter kommen von Claude. Erläuterungen nicht.

Abkürzung / Fachwort	Erläuterung
CLOUD Act	US Gesetz Behörden den Zugriff auf Daten von US-Unternehmen ermöglicht.
Schrems II	Urteil vom EU-Parlament von 2020. In dem wurde Privacy Shield als ungültig erklärt.
Privacy Shield	Ehemaliges Datenschutzabkommen zwischen EU und USA.
Standardvertragsklauseln	Vertraglich vereinbarte Datenschutzstandards für internationale Datenübertragungen
Betroffenenrechte	Rechte von Personen bezüglich ihrer Daten
TLS 1.3	Transport Layer Security, aktueller Standard für verschlüsselte Datenübertragung
AES-256	Advanced Encryption Standard mit 256-Bit Schlüssellänge für Datenverschlüsselung
Salt	Zufällige Daten, die beim Passwort-Hashing zur Erhöhung der Sicherheit verwendet werden
Multi-Faktor-Authentifizierung (MFA)	Sicherheitsverfahren mit mehreren Authentifizierungsfaktoren
Incident Response	Strukturierter Prozess zur Behandlung von Sicherheitsvorfällen
Penetration Testing	Sicherheitstest durch simulierte Angriffe auf IT-Systeme
CaaS Provider	Container-as-a-Service Anbieter für Cloud-Infrastruktur

Tabelle 3: Glossar 2

Inhaltsverzeichnis

1 Übersicht und Zweck.....	2
1.1 Management Summary	2
1.2 Zweck dieses Konzeptes	2
2 Rechtliche Grundlagen.....	2
2.1 Bestehendes Recht	2
2.2 Verantwortlichkeiten	2
2.3 Wichtigkeit.....	2
3 Datenarten und Zweck	3
3.1 Personenbezogene Daten	3
3.1.1 Stammdaten.....	3
3.1.2 Nutzungsdaten	3
3.2 Zweck	3
4 Rechtsgrundlagen der Verarbeitung	4
4.1 Art. 6 DSGVO / Art. 13 nDSG	4
4.1.1 Vertragserfüllung (Art. 6 Abs. 1 lit. b DSGVO):	4
4.1.2 Berechtigtes Interesse (Art. 6 Abs. 1 lit. f DSGVO):	4
4.1.3 Einwilligung (Art. 6 Abs. 1 lit. a DSGVO):.....	4
5 Wiederrufsrecht	5
5.1 Vertragserfüllung.....	5
5.1.1 Kalenderfunktion	5
5.1.2 Aufgabenverwaltung.....	5
5.1.3 Benutzerkonto	5
5.2 Nicht Vertragserfüllung fällt:.....	5
5.2.1 Marketing:	5
5.2.2 Analytics:.....	5
6 Technische Massnahmen	6
6.1 Wichtigkeit.....	6
6.2 Verschlüsselung.....	6
6.3 Zugriffskontrolle	6
6.4 Einstellungen für die Organisation.....	6
7 Massnahmen	7
7.1 US Server.....	7
7.2 US Providers	7

7.3	Kalender Integration	7
8	Betroffennenrecht.....	8
8.1	Auskunftsrecht.....	8
8.2	Recht auf Berichtigung	8
8.3	Einschränkung auf Verarbeitung	8
8.4	Recht auf Vergessenwerden.....	8
9	Incident Response	9
9.1	Datenschutzverletzungen.....	9
9.1.1	Erkennungsmaßnahmen.....	9
9.1.2	Response-Prozess	9
9.2	Meldeverfahren.....	9
10	Roadmap	10
10.1	Grundlagen.....	10
10.2	Erweiterte Massnahmen	10
10.3	Optimierung.....	10
11	Fazit	11
11.1	Weniger ist mehr.....	11
11.2	Rechte	11
11.3	Verantwortung.....	11
11.4	Reaktion.....	11
11.5	Zeitplan	12
11.5.1	Vor der Veröffentlichung.....	12
11.6	Zukunft.....	12

Abbildungsverzeichnis

Es konnten keine Einträge für ein Abbildungsverzeichnis gefunden werden.

Tabellenverzeichnis

Tabelle 1: Änderungsverzeichnis	1
Tabelle 2: Glossar 1.....	13
Tabelle 3: Glossar 2.....	14

Anhang E2

Design Konzept

Delegator

Klassifizierung intern
Status in Arbeit
Programmname Delegator
Projektnummer 1
Projektleiter Tabinas Kenan
Version 0.1
Datum 26. März 2025
Auftraggeber Tabinas Kenan
Autor/Autoren Tabinas Kenan
Verteiler

Änderungsverzeichnis

Version	Datum	Änderung	Autor
1	26.03.2025	Erstellt	TAK

Tabelle 1: Änderungsverzeichnis

1 UX

Mir ist und bleibt die User Experience ein wichtiger Teil. Das Design der App muss gut geplant werden.

Wichtig für das Konzept ist ein Design der APP. Um die App optimal zu planen müssen alle Features und Poentielle Features Platz haben.

2 Dos and Donts

Es gibt einen Bereich, wo sich Elemente eignen, dementsprechend gibt einen Bereich der sich gar nicht eignet. Aus dem können wir schliessen, das für die UX die Elemente möglichst weit unten sein sollten.

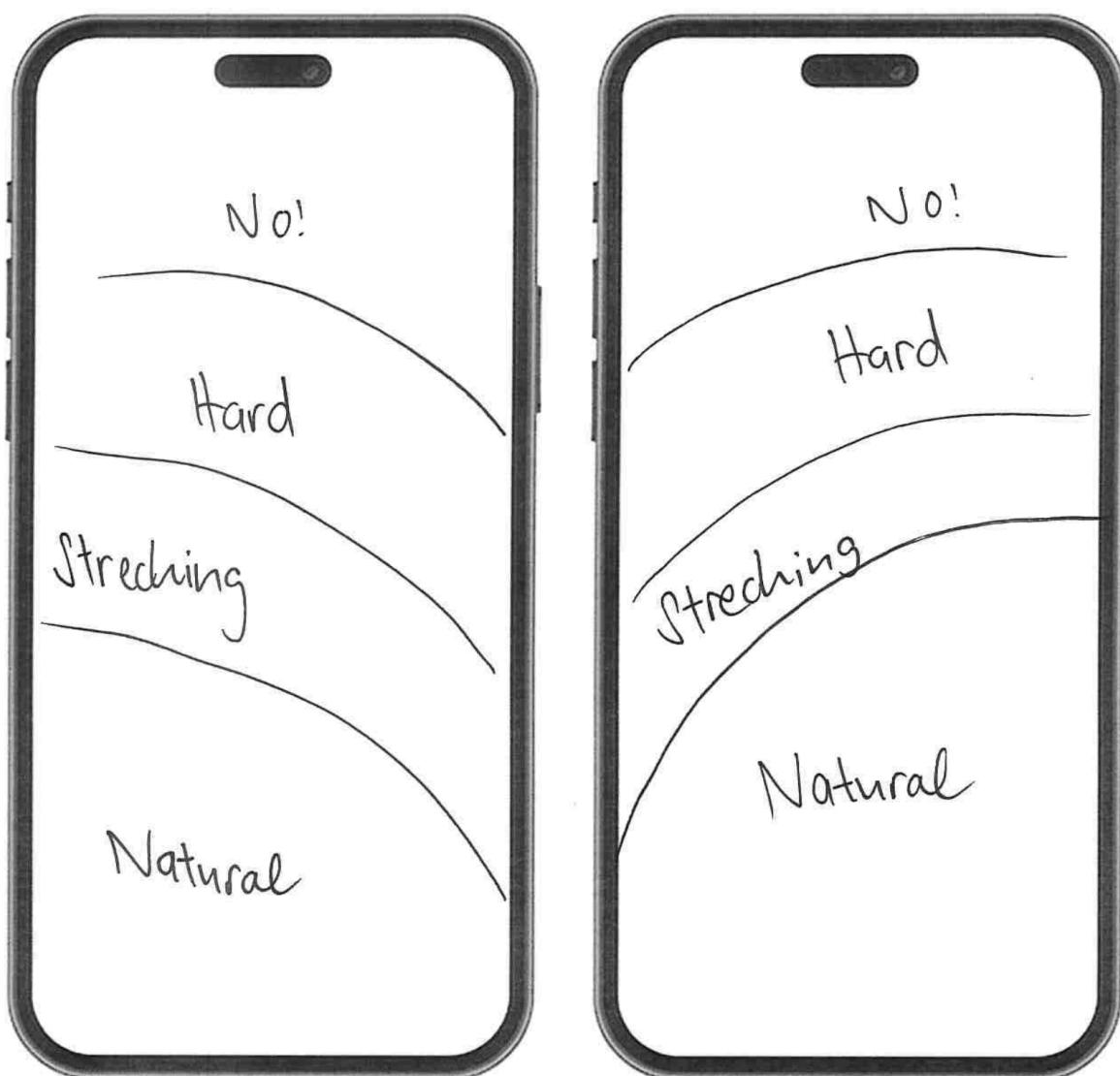


Abbildung 1: Dos and Donts

3 Navigation Optionen

Zuerst müssen wir klären welche Navigation Elemente wir überhaupt zu Verfügung haben.

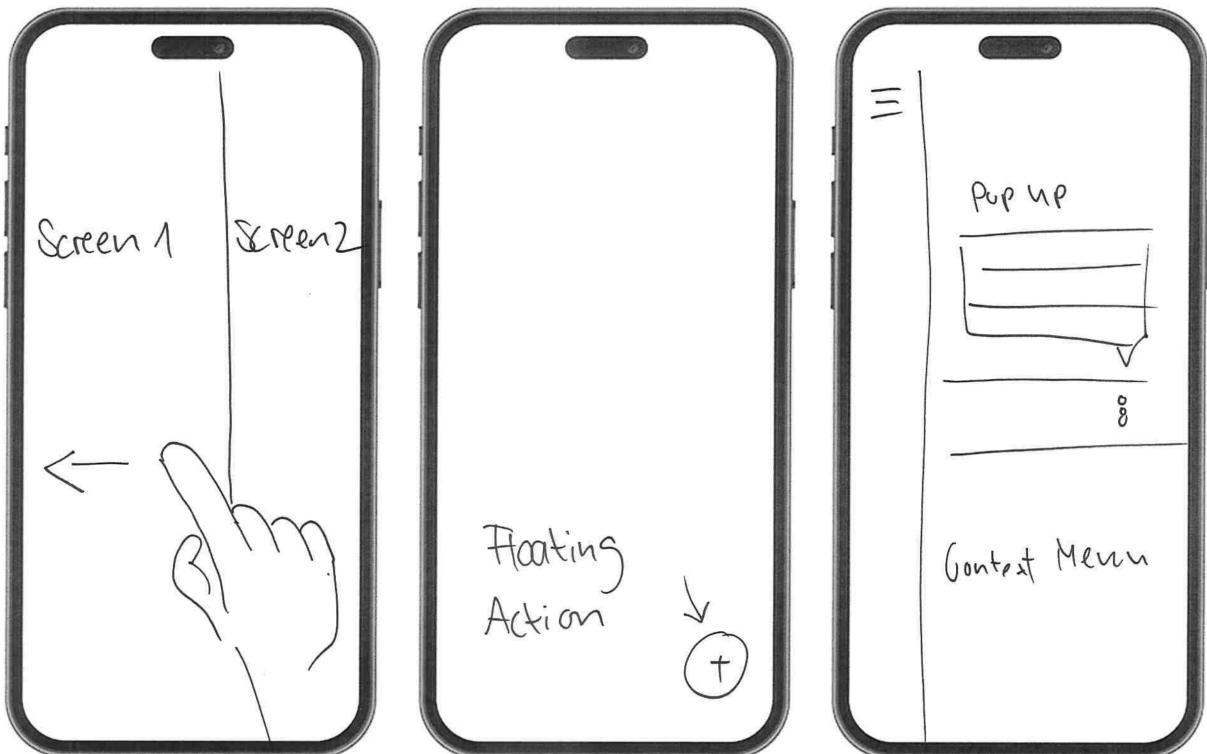


Abbildung 2: Navigations Optionen 1

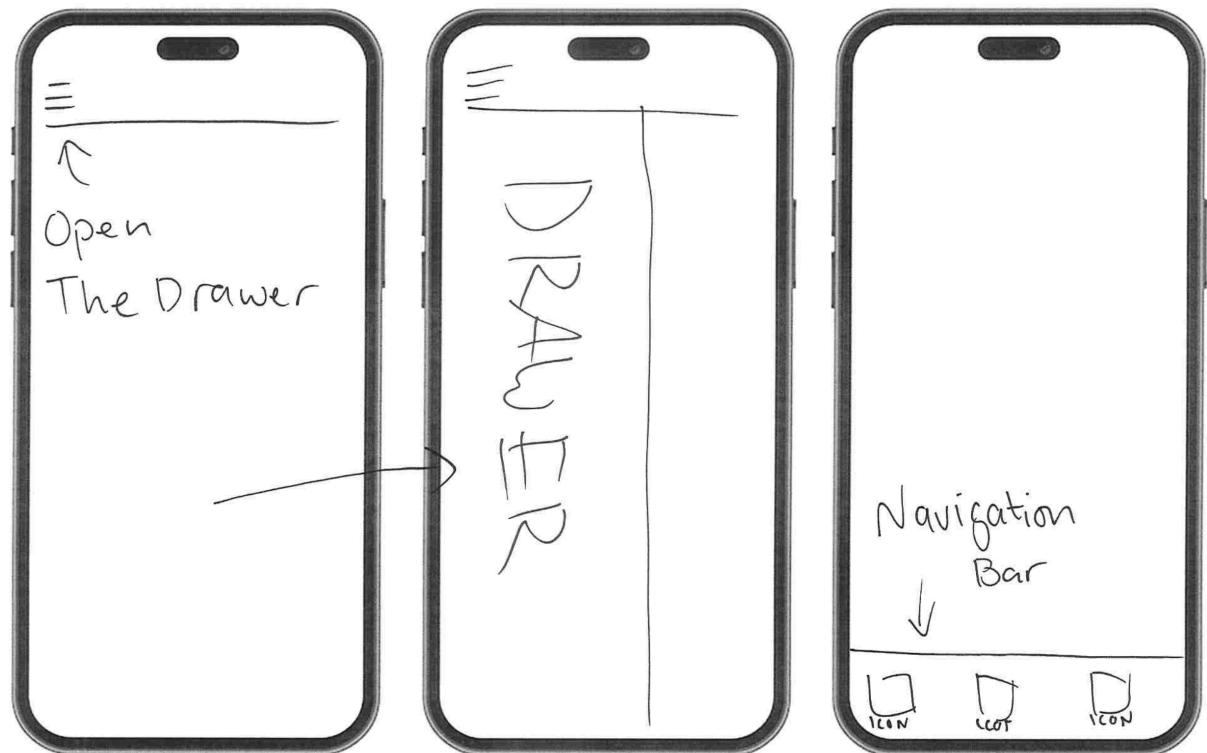


Abbildung 3: Navigations Optionen 2

4 Features

Es muss auch noch geklärt welche Features könnten interessant sein. Unabhängig davon, ob sie umgesetzt werden.

4.1 Übersicht

Es folgt eine Aufzählung aller möglichen Features. Auch wenn sie nicht realisiert werden, muss trotzdem ein Platz definiert werden und wie sie mit anderen Sachen interagieren.

Feature	Hinweis
Projekte	Konzerte, Medien, Songs
Songs Live	Länge (Studio Länge & Live Länge), Hinweise Live, Backing Track, Passt vor und zu.
Songs Studio	Aufnahme, Diskussionen mit Externen, Aufnahme Go, Versionsverlauf, Chat, History, Tasks, Termine
Gigs	Wann, wo? Werbung? Alle Aufgaben dazu? Setlist?
Retro	Definierbare Punkte wie war (das Set, was das nächste mal anders)
Band Meetings	Auf Retros eingehen, Release Planung, Social Media, Budget planning. Zielgruppe, Merch. Konflikte
Merch	Kauf, menge, Varianten, Design, Release, Store
Social Media	Upload Plan, Scheduled Content und Abhängigkeiten
Tour	Weiss ich selber noch zu wenig.
Strategie	Positionierung, Identität, Werte, Story, Ziele, USP, Ziel Gruppe, Meilensteine
Verträge	Weiss ich selber noch zu wenig.
Rechte	GEMA / Suissa
Videos	Video Konzept, Storyboard, Dreh Tage, Props. Miet Material, Externe
Budget	Konzerte, Risiko bei Veranstaltungen, geplante Investitionen
Abstimmungen	Proben, Entscheidungen

Abbildung 4: Komplette Feature Liste

4.2 Feature Design “Gigs”

ID	Feature	Hinweis
FD0001	Gig Timetable	Load In, Arrival, Setup, Soundcheck, Change over, Playtime, Loadout,
FD0002	Gig Information	Playtime, Location, Change Over before, Change over after, food time, money this gig gives
FD0003	Gig Checklist	Send Stageplan, Techrider, who brings the drums, miks needed, is pa there, mixer available, is car organized, did someone check the food times
FD0004	Custom Gig Template	Diese Informationen sind wie nicht fix. Vielleicht braucht es ein Manager, welcher erlaubt die vorhandenen punkte zu ergänzen entfernen oder ab ändern
FD0005	Gig Promo	Mention a week before, mention a day before, mention on that day, updated on the website, flyer?
FD0006	Gig Conversaion	Linked E-Mails, uploaded screenshot from chats

Tabelle 2: Features Design Gigs

4.3 Feature Design “Social Media”

ID	Feature	Hinweis
FD0101	Abhängigkeit	Ein Post muss eine Verbindung zu einem Event haben. Wenn das Event z.B. ein Release verschoben wird, wird auch der Post dazu verschoben.
FD0102	Content	Es gibt verschiedene Versionen, welche pro Post hochgeladen werden können. Dazu gibt es Kommentare. Es gibt einen Upload Plan.
FD0103	Sceduler	Es gibt einen Upload Rhythmus der erstellt wird. Alle Post haben Platz dort.

Tabelle 3: Features Design Social Media

4.4 Feature Design «Merch»

ID	Feature	Hinweis
FD0201	Lagerbestand	Aktueller Bestand
FD0202	Merch Planung	Zukünftige Designes, Trend Verkauf damit geplant werden kann wann es neuen Merch brauch.
FD0203	Kaufverträge und Produkt Informationen	Lieferfristen und Budget welches gebraucht werden um eine Investition zu machen

Tabelle 4: Features Design Merch

4.5 Feature Design «Budget»

ID	Feature	Hinweis
FD0301	Einkommen pro Gig	
FD0302	Geplante Investitionen	Geräte wie Racks, Licht und andere Stage Elemente, back Up Guitaren usw
FD0303	Mögliches Risiko	Geht einer mit FD0001.
FD0304	Merchverkäufe	Geht einher mit FD0202.
FD0304	Merch Roadmap	Geht einher mit FD0203.

Tabelle 5: Features Design Budget

4.6 Feature Design «Songs Live»

ID	Feature	Hinweis
FD0401	Live Länge	
FD0402	Live Variationen	Wenn es ein längeres Outro gibt macht es nicht es zu spielen wenn der nächste Song ein längere Intro hat.
FD0403	Hinweise	Zum Beispiel Informationen zu Crowd Interaction
FD0404	Backing Track Versionen	File Upload und Versionierung. Kommentare und eine History.
FD0405	Verbindung zu Studio Songs.	Ein Link zu einem anderen Song.

Tabelle 6: Features Design Songs Live

4.7 Feature Design «Songs Studio»

ID	Feature	Hinweis
FD0501	Länge	
FD0502	Variationen	
FD0503	Files	Destination
FD0504	Commenting Plattform	Eine Plattform um mit Mixer zu Kommentieren
FD0505	Aufgaben	
FD0506	Moodboard	
FD0507	Versionen Musik	
FD0508	Versionen Art	
FD0509	Chat	

Tabelle 7: Features Design Songs Live

4.8 Feature Design «Strategie»

ID	Feature	Hinweis
FD0601	Identität	
FD0602	Bio	
FD0603	Press Kit	
FD0604	Langzeit Strategie und Ziele	Beispielweise Verbinden mit Zahlen aus Spotify und gesetzten Zielen
FD0605	Rules	
FD0606	Basic Release Plan	

Tabelle 8: Features Design

4.9 Feature Design «Contacts»

ID	Feature	Hinweis
FD0701	Liste mit Kontakten	
FD0702	Eigenschaften	E-Mail, Name, Spitzname, Hinweise, Preise aus der Vergangenheit

Tabelle 9: Features Design

4.10 Feature Design «Meetings»

ID	Feature	Hinweis
FD0801	Zukunft	
FD0802	Planung bis zum Nächsten Meeting	
FD0803	Aufgaben Stand	Schauen, ob wir den plan Anpassen müssen oder ob wir gut fahren
FD0804	Retro	Siehe Retro
FD0805	Song-Guess Einstieg	Es gibt ein Einstieg wobei Songs erraten werden können
FD0806	Liste mit Entscheidungen	
FD0807	Feedback	Eine Art Feedback Runde welche man Vorbereitet Feedback gebe kann ohne einander zu Verletzten.

Tabelle 10: Features Design Meetings

4.11 Feature Design «Videos»

ID	Feature	Hinweis
FD0901	Moodboard	Downlaod und Hochladen? Eine Direkte Suche nach Goolge Bilder?
FD0902	Arbeit mit anderen	DP, Direktor einalden
FD0903	Props	
FD0904	Ideen (Mind Map)	
FD0905	Location	
FD0906	Time Table	
FD0907	Drehtag Organisation	Essen Alergen?

Tabelle 11: Features Design Videos

4.12 Feature Design «Retro»

ID	Feature	Hinweis
FD1001	Feedback	
FD1002	Anpassbare Vorlage	
FD1003	Checkmark	Damit jede Retro auch

Tabelle 12: Features Design Retro

4.13 Feature Design «Abstimmungen & Prozesse»

ID	Feature	Hinweis
FD1101	Eine Art Aktivät	Eine Art Bell wo es eine übersicht gibt mit allen änderungen. Aber auch Filterbar oder Kategorierbar,
FD1102	Eine Art von einer Liste mit Entscheidungen	Sobald eine Entscheidung getroffen wird, wird es in eine Liste aufgenommen.
FD1103	Checklisten	Noch nicht klar wo sich das mit den Aufgaben trifft. Aber es gibt eine gewisse Überschneidung.

Tabelle 13: Features Design Abstimmungen und Prozesse

4.14 Fazit

Daraus lässt sich schliessen, dass gewisse UI-Elemente müssen Dynamisch generiert werden.

5 Dashboard

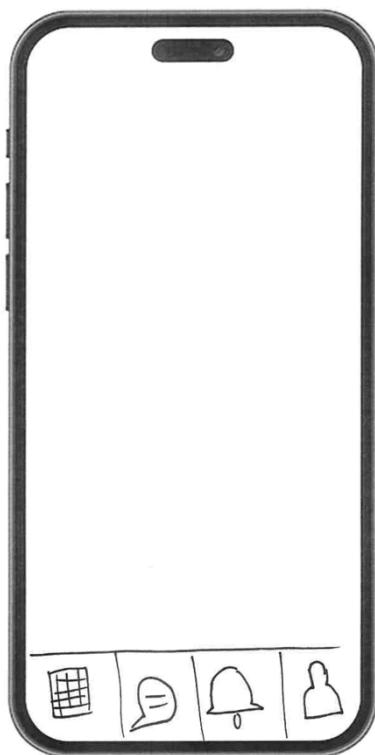
Das Dashboard ist wie die übergeordnete Navigation und Ordnung. Dabei habe ich 5 Verschiedene Varianten erstellt und mich am Ende für eine Entschieden.

5.1 Dashboard V1

Wichtige Elemente in der NavBar:

- Kalender
- Chat
- Aktivitäten
- Band Informationen

Dashboard v1



5.2 Dashboard V2

Wichtige Elemente in der NavBar:

- Kalender
- Aufgaben
- Chat
- Aktivitäten
- Band Informationen

Persönlich ist es mir etwas zu voll.

Abbildung 5: Dashboard 1

Dashboard v2



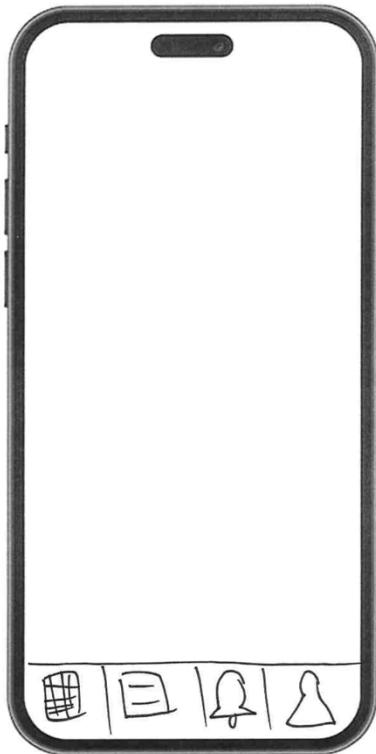
Abbildung 6: Dashboard 2

Dashboard V3

5.3 Dashboard V3

Wichtige Elemente in der NavBar:

- Kalender
- Aufgaben
- Aktivitäten
- Band Informationen



5.3.1 Dashboard V4

Wichtige Elemente in der NavBar:

- Kalender
- Aufgaben
- Chat
- Band Informationen

Aktivitäten sind in der Band Informationen

Abbildung 7: Dashboard 3

Dashboard V4



Abbildung 8: Dashboard 4

5.3.2 Dashboard V5

Wichtige Elemente in der NavBar:

- Chat
- Home
- Band Informationen

Aktivitäten sind in der Band Informationen

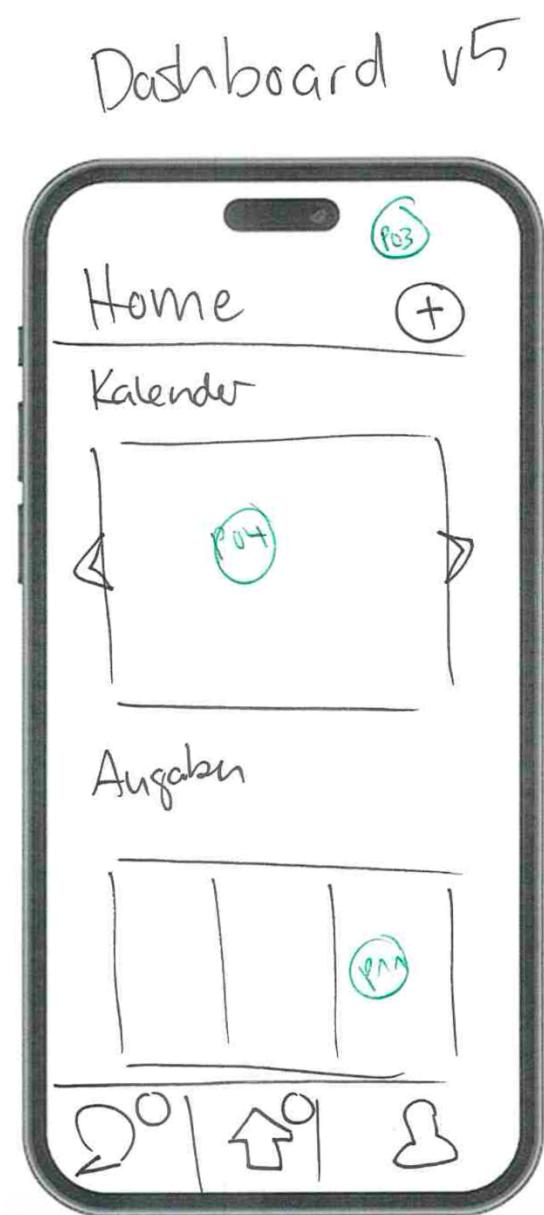


Abbildung 9: Dashboard 5

6 Dashboard

Ich habe mich für Dashboard 5 entschieden und dort gibt es 3 Seiten.

6.1 Chat

- Alle Chats aus allen Projekten.
- Ein Filter damit eingeschränkt werden kann.
- Ein Such Funktion

6.2 Aktuell (ehemalig Home)

- Plus Button für Jetzt Aktionen
- Kalender
- Aufgaben
- Projekte
- On going Abstimmungen
- Ungesehene Aktivitäten (mit direkt Link)

6.3 Info

- Eine Such Funktion
- Eine Typ zur Einschränkung

Inhalt	Typ
Finished Songs	Operativ
Contacts	Operativ
E-Mails	Operativ
Merch	Operativ
Budget	Strategisch
Merch Statistik	Strategisch
Vision1	Strategisch
Alte Projekte	Archiv
Alte Retros	Archiv
Alte Gigs	Archiv
Alte Meetings	Archiv
Alte Entscheidungen	Archiv

Tabelle 14: Dashboard 5 beschreibung

6.4 Trennung

Es gibt gewisse Überschneidungen. Daher habe ich am folgenden Design Leitsatz.

Im Aktuell erscheinen auch nur aktuell relevant Inhalte. Im Info Teil gehören ältere und spätere Inhalte.

6.5 Jetzt Aktionen

Etwas, was mir in den Sinngekommen ist, sind Jetzt Aktionen. Wie Team-Meeting Starten, jetzt Setlist bearbeiten, jetzt Projekt erstellen

- Meeting Starten
- Setlist Drucken
- Neuer Song
- Neues Projekt
- Neuer Termin
- Neuer Kontakt
- Lager Bestand ändern
- Neue Abstimmung

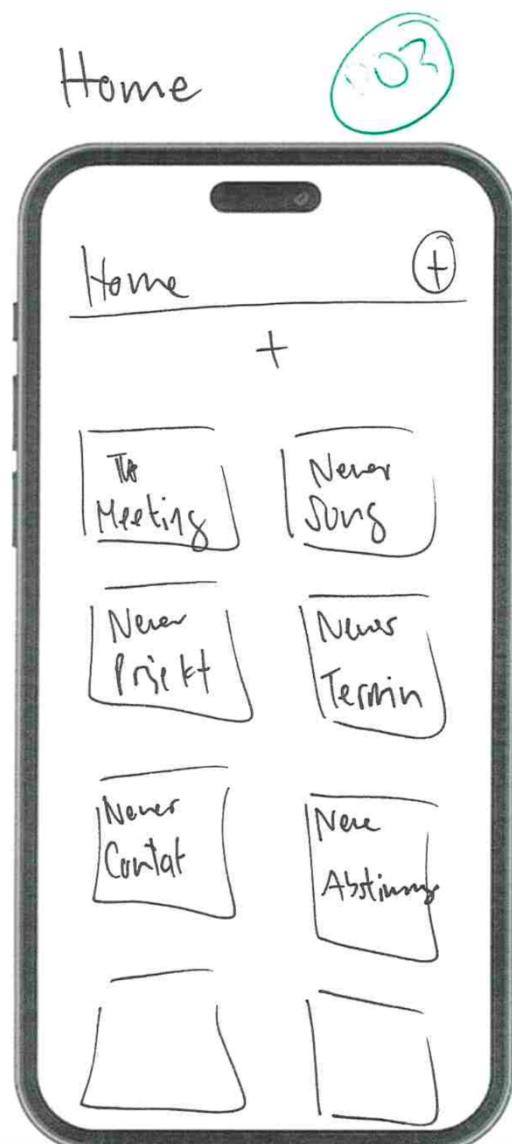


Abbildung 10: Jetztaktionen

7 Pages

ID	Name	Hinweise
P01	Chats	
P02	Chat Details	
P03	Home	
P04	Calender	
P05	Eintrag erstellen	
P06	Gig Information	Timetable, Location, Contacts, Emails
P07	Gig Mails	Habe ich später dagegen Entscheidgen. Dafür kann E-Mail attachen.
P08	Gig Setlist	
P09	Gig Board	
P10	Aufgaben (Kanban)	Solltes ein Universelles Board geben? Oder eine Liste mit allen Boards?
P11	Terminierte Aufgaben	
P12	Projekte	
P13	Aufgaben Liste	
P14	Aufgaben Element	
P15	Song Studio Übersicht	
P16	Aktivitäten	
P17	Termin Liste	
P18	Retro	
P19	Retro Editor	Aktuell noch nicht gezeichnet
P20	Abstimmungen	
P21	Abstimmung Detail	
P22	Abstimmungs Editor	
P23	Infos	
P24	Song List	
P25	Contacts	
P26	E-Mails	
P27	Budget	
P28	Merch Stats	
P29	Vision	
P30	Alte Retros	

P31	Alte Gigs	
P32	Alte Meetings	
P33	Merch Stock	
P34	Alte Entscheidungen	

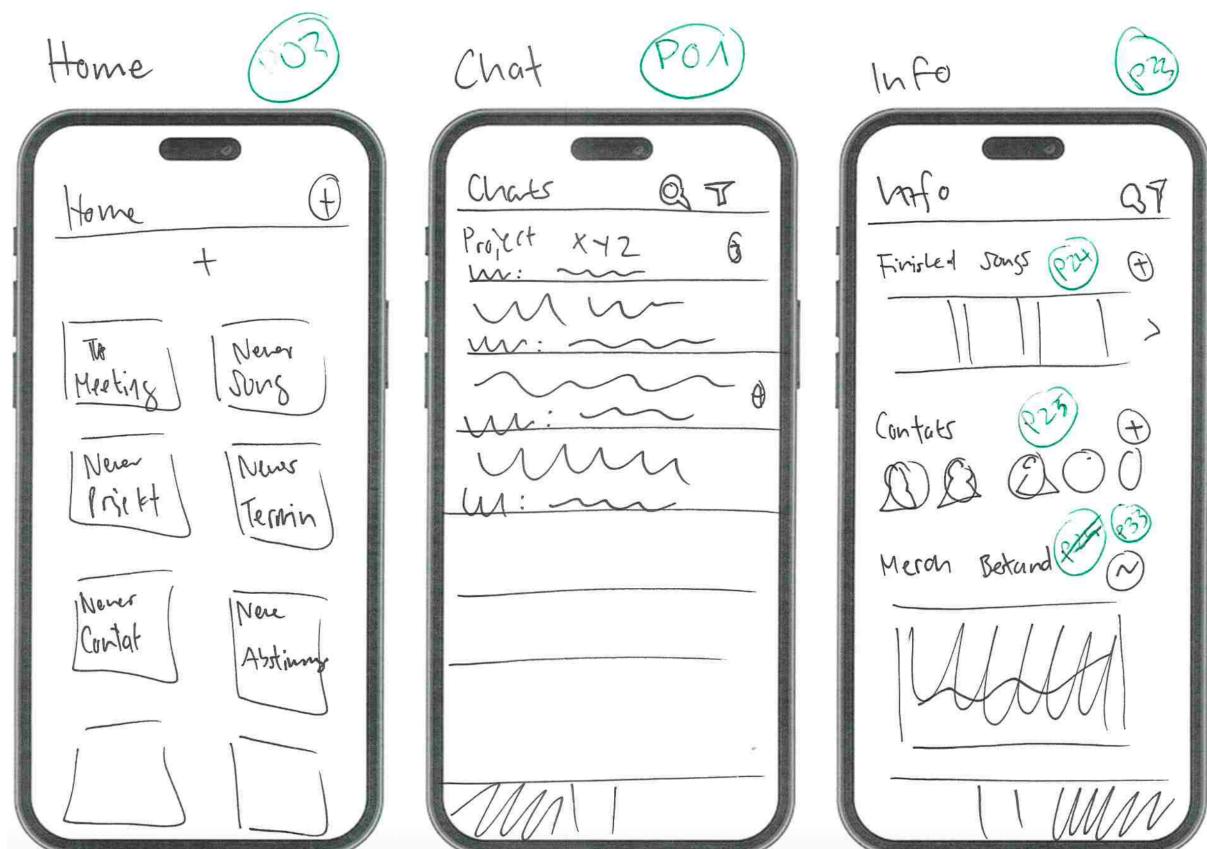


Abbildung 11: Screens 1

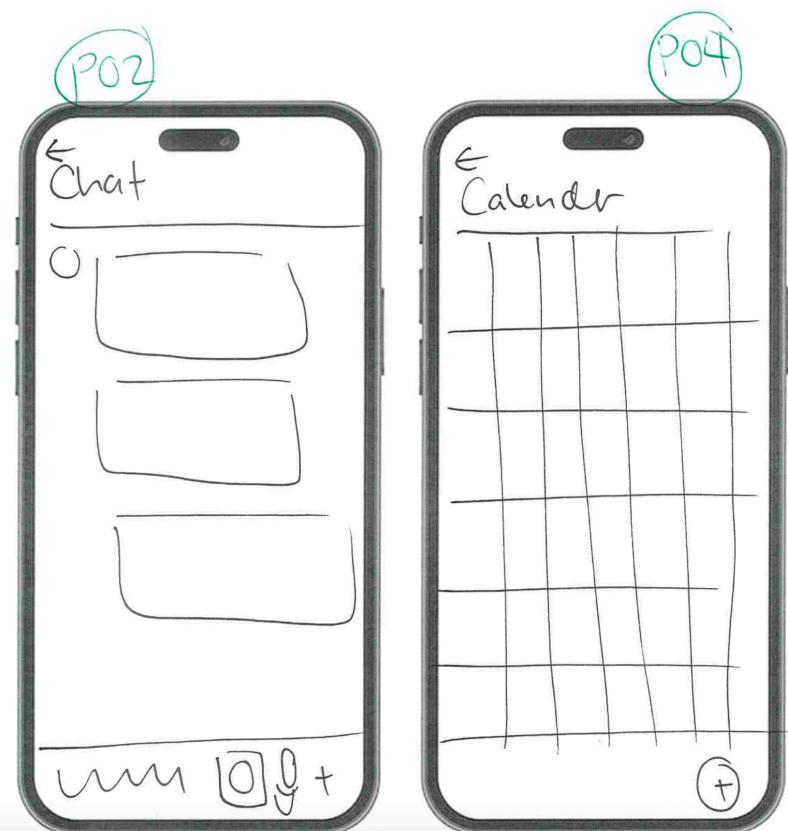


Abbildung 12: Screens 2



Abbildung 13: Screens 3

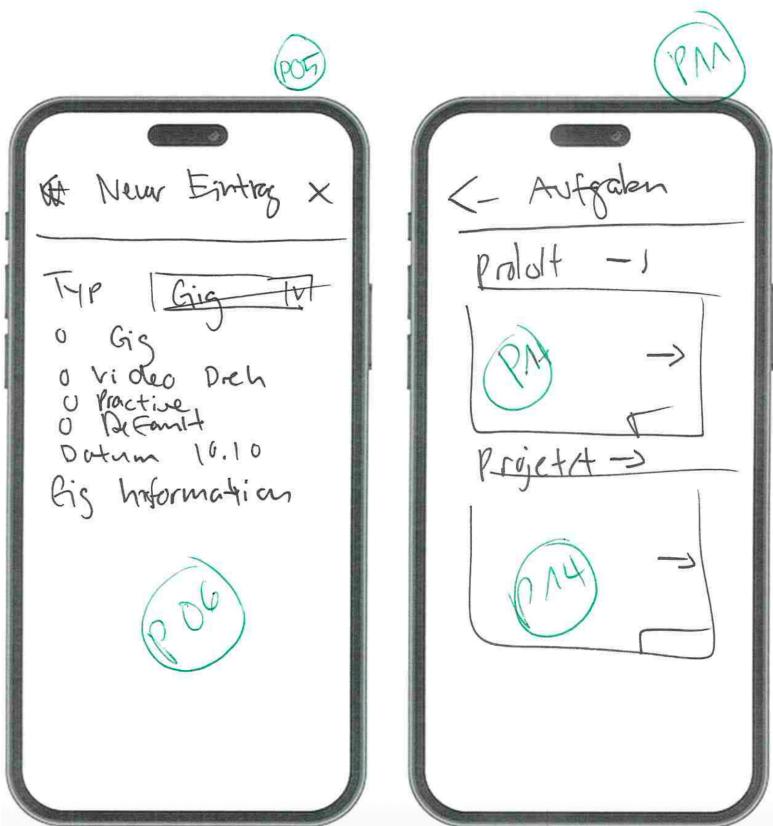


Abbildung 14: Screens 4



Abbildung 15: Screens 5



Abbildung 16: Screens 6

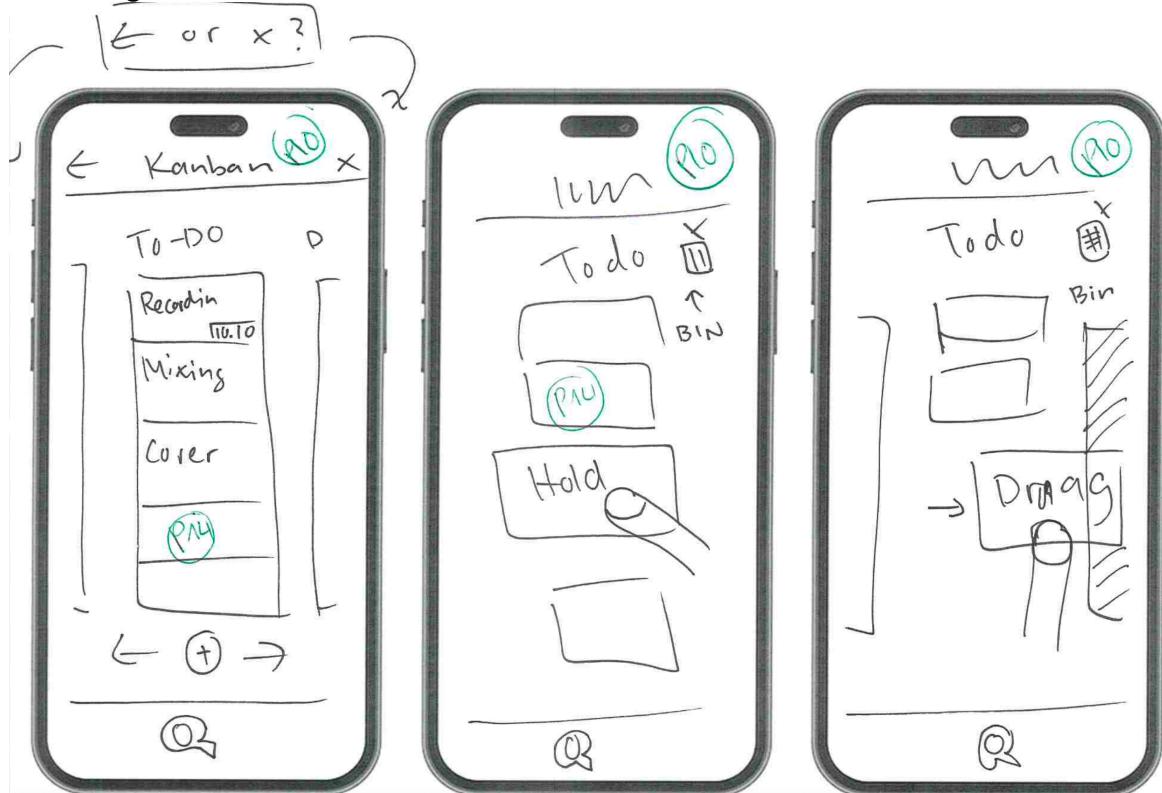


Abbildung 17: Screens 7

Geordnet nach Updates

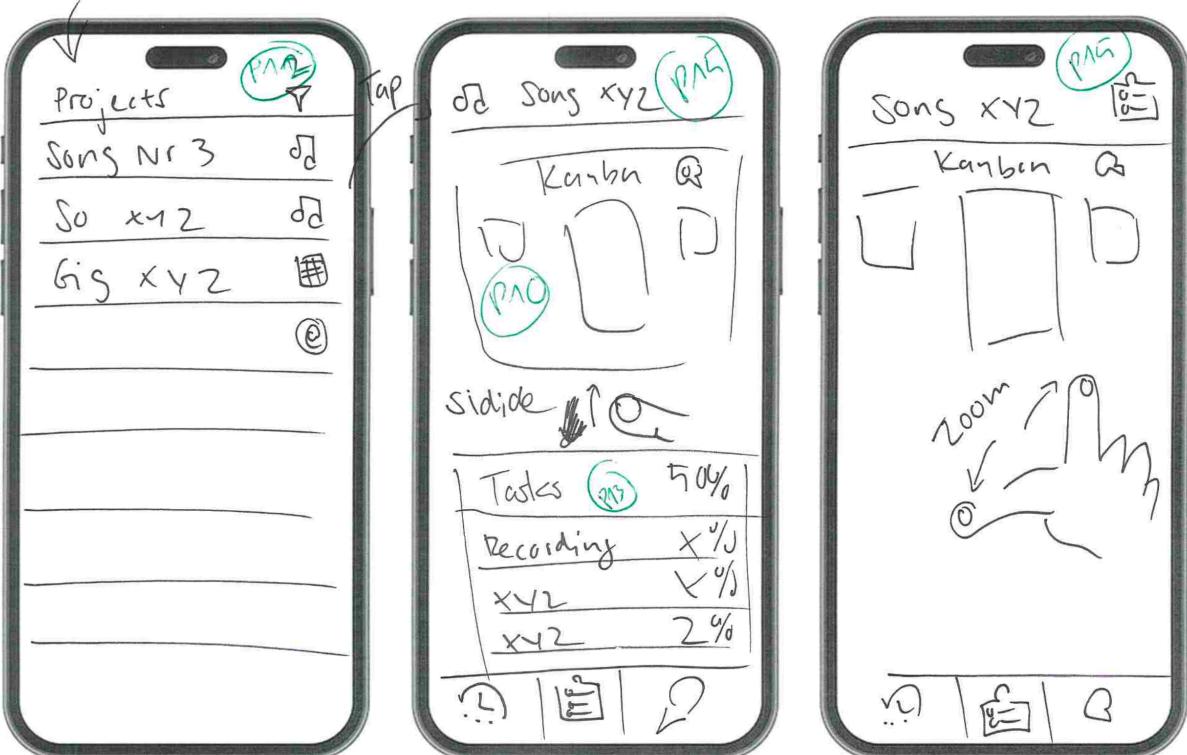


Abbildung 18: Screens 8



Abbildung 19: Screens 9

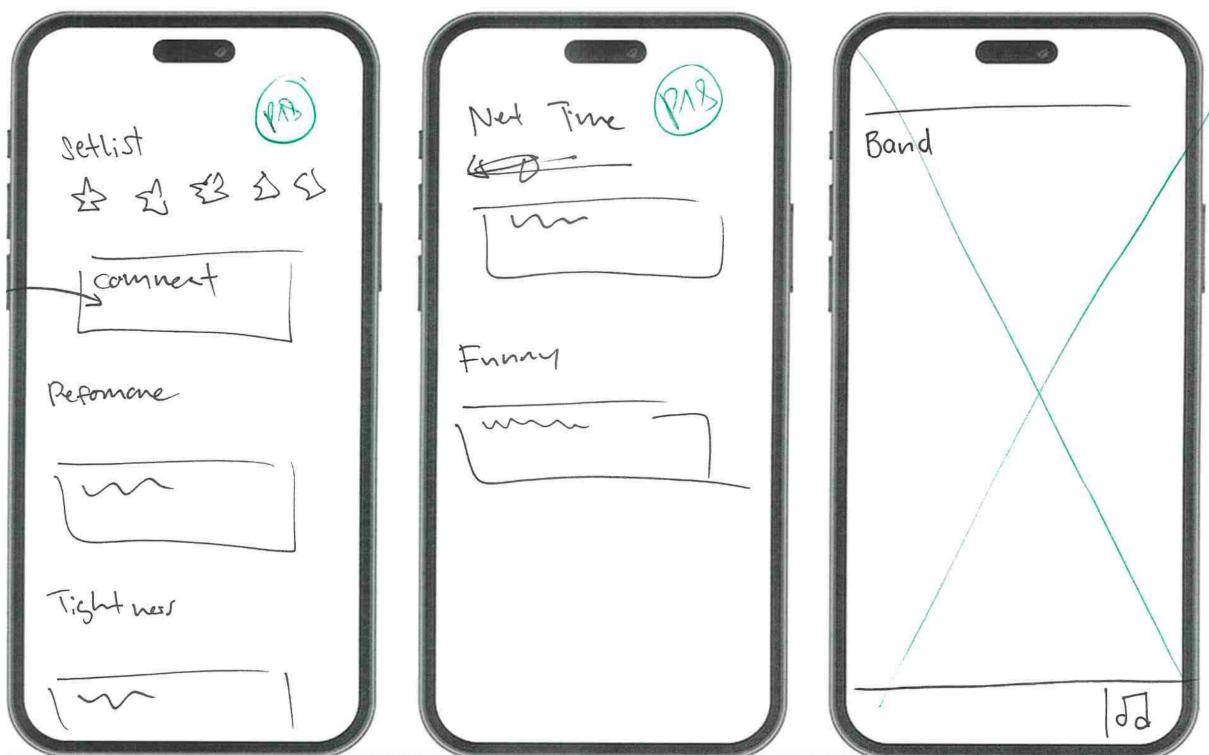


Abbildung 20: Screens 10

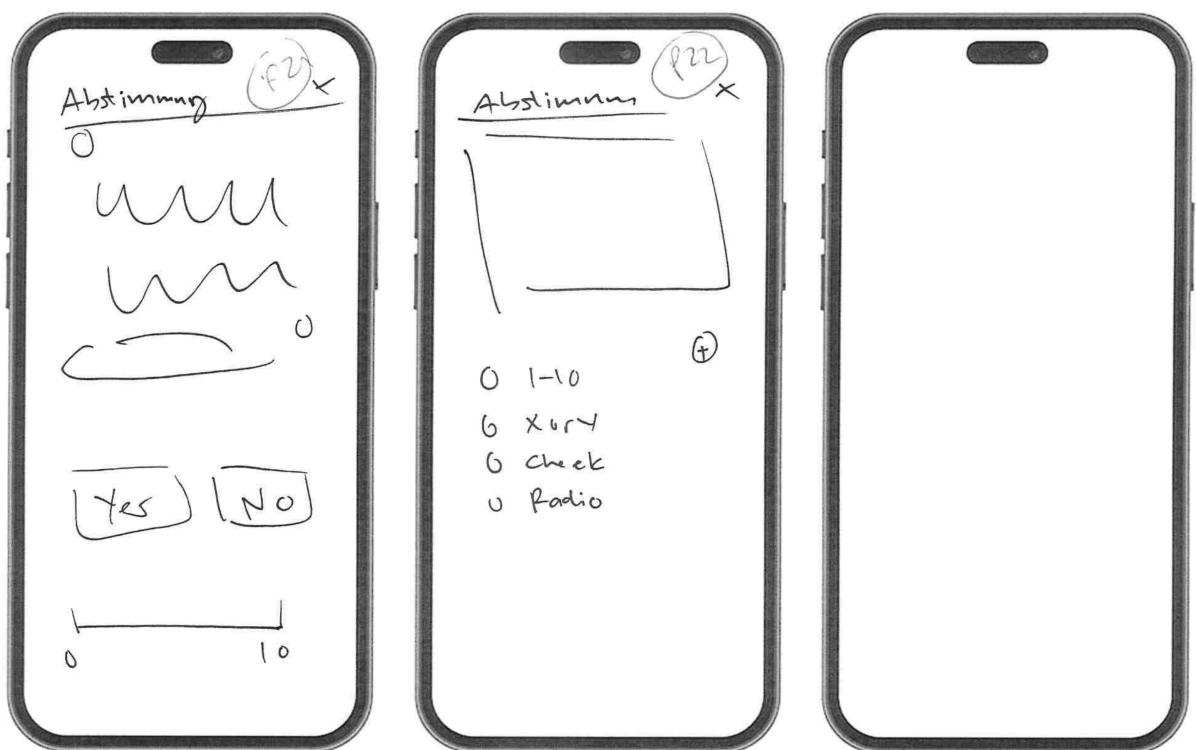


Abbildung 21: Screens 11



Abbildung 22: Screens 12

Abkürzungen und Glossar

Abkürzung / Fachwort	Erläuterung
HERMES	Vorgehensmethodik für Projekte und Programme HERMES 5 ist ein eCH Standard

Tabelle 15: Abkürzungen und Glossar

Inhaltsverzeichnis

1 UX	2
2 Dos and Donts	2
3 Navigation Optionen	3
4 Features.....	4
4.1 Übersicht.....	4
4.2 Feature Design “Gigs”	5
4.3 Feature Design “Social Media”.....	5
4.4 Feature Design «Merch»	5
4.5 Feature Design «Budget»	6
4.6 Feature Design «Songs Live»	6
4.7 Feature Design «Songs Studio»	6
4.8 Feature Design «Strategie»	7
4.9 Feature Design «Contacts»	7
4.10 Feature Design «Meetings».....	7
4.11 Feature Design «Videos»	8
4.12 Feature Design «Retro».....	8
4.13 Feature Design «Abstimmungen & Prozesse»	8
4.14 Fazit.....	8
5 Dashboard	9
5.1 Dashboard V1	9
5.2 Dashboard V2	9
5.3 Dashboard V3	10
5.3.1 Dashboard V4	10
5.3.2 Dashboard V5	11
6 Dashboard	12
6.1 Chat	12
6.2 Aktuell (ehemalig Home).....	12
6.3 Info.....	12
6.4 Trennung	13
6.5 Jetzt Aktionen	13
7 Pages	14

Abbildungsverzeichnis

Abbildung 1: Dos and Donts	2
Abbildung 2: Navigations Optionen 1.....	3
Abbildung 3: Navigations Optionen 2	3
Abbildung 4: Komplette Feature Liste.....	4
Abbildung 5: Dashboard 1	9
Abbildung 6: Dashboard 2	9
Abbildung 7: Dashboard 3	10
Abbildung 8: Dashboard 4.....	10
Abbildung 9: Dashboard 5.....	11
Abbildung 10: Jetztaktionen.....	13
Abbildung 11: Screens 1.....	15
Abbildung 12: Screens 2	16
Abbildung 13: Screens 3	16
Abbildung 14: Screens 4.....	17
Abbildung 15: Screens 5.....	17
Abbildung 16: Screens 6.....	18
Abbildung 17: Screens 7.....	18
Abbildung 18: Screens 8	19
Abbildung 19: Screens 9	19
Abbildung 20: Screens 10	20
Abbildung 21: Screens 11	20
Abbildung 22: Screens 12	21

Tabellenverzeichnis

Tabelle 1: Änderungsverzeichnis	1
Tabelle 2: Features Design Gigs.....	5
Tabelle 3: Features Design Social Media	5
Tabelle 4: Features Design Merch	5
Tabelle 5: Features Design Budget.....	6
Tabelle 6: Features Design Songs Live.....	6
Tabelle 7: Features Design Songs Live	6
Tabelle 8: Features Design	7
Tabelle 9: Features Design	7
Tabelle 10: Features Design Meetings	7
Tabelle 11: Features Design Videos	8
Tabelle 12: Features Design Retro.....	8
Tabelle 13: Features Design Abstimmungen und Prozesse.....	8
Tabelle 14: Dashboard 5 beschreibung.....	12
Tabelle 15:Abkürzungen und Glossar	22

Anhang E3

Realisierung

Delegator

Klassifizierung intern
Status in Arbeit
Programmname Delegator
Projektnummer 1
Projektleiter Tabinas Kenan
Version 0.1
Datum 26. März 2025
Auftraggeber Tabinas Kenan
Autor/Autoren Tabinas Kenan
Verteiler

Änderungsverzeichnis

Version	Datum	Änderung	Autor
0.1			

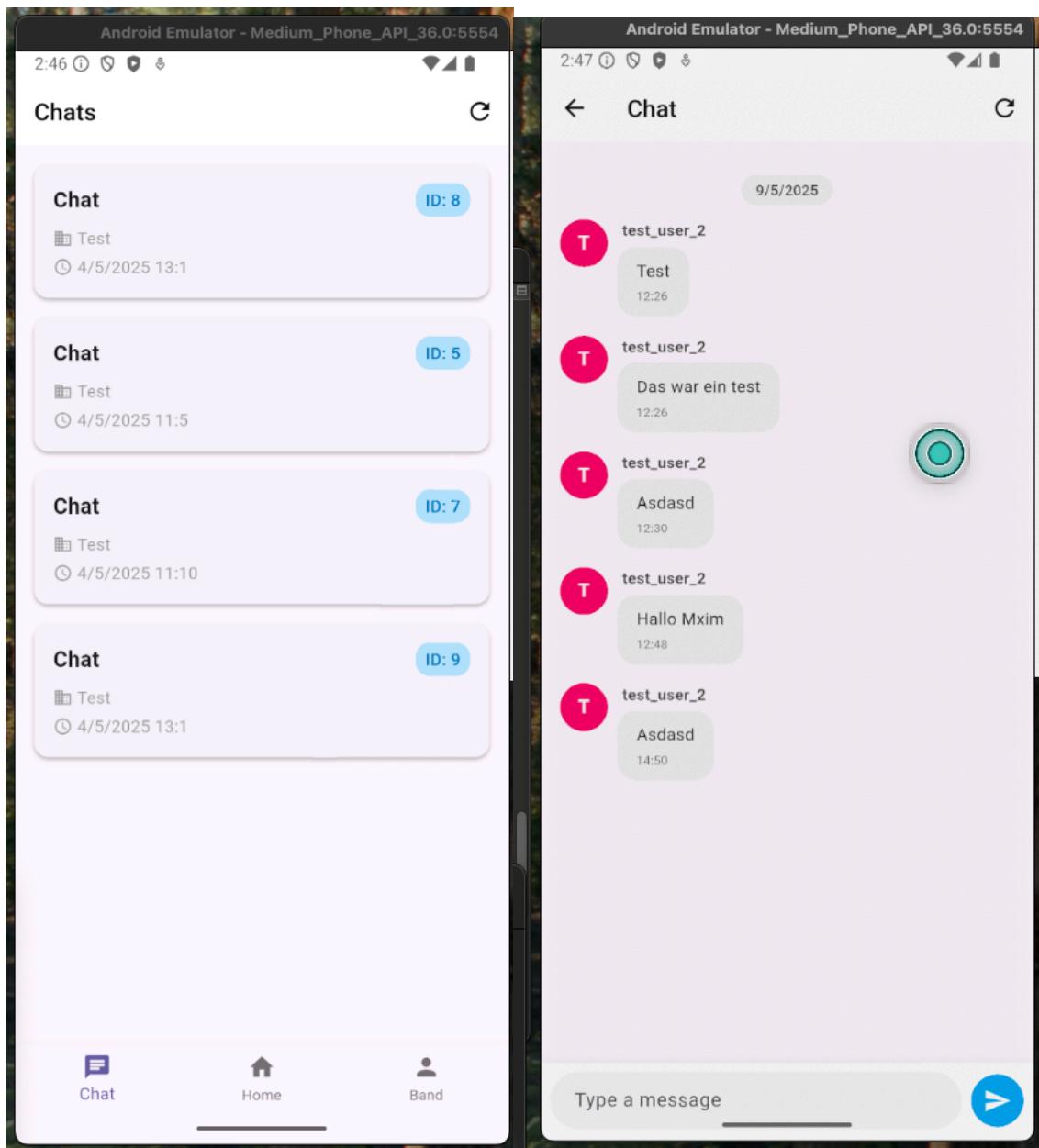
Tabelle 1: Änderungsverzeichnis

Beschreibung

Die Studie beschreibt die angestrebte Lösung, in dem sie die groben Ziele definiert, möglichen Lösungsvarianten aufführt und diese dann bewertet. Sie bildet die Grundlage für die Entscheidung, ob ein Projekt freigegeben wird oder nicht. Sie ist die Voraussetzung für die Erarbeitung des Projektmanagementplans und des Projektauftrags.

1 Designpattern in Flutter

Zuerst habe einfach angefangen, weil ich wenig Erfahrung mit Flutter hatte. Ich habe vieles ChatGPT / Claude machen lassen. Das zu einem gewissen Grad funktioniert. Der Chat Teil hat wunderbar funktioniert.



1.1 Design Pattern

Im Verlauf hat sich das leider nicht tragbar gezeigt. Beim erstellen von Tasks bin ich an die grenzen gekommen. Die Language Modells konnten mir nicht mehr helfen. Selbst mit Zurgiff auf das Repository, keine Chance. Der Code scheint gut auszusehen, sagten sie. Irgendwelche Lösung Ansätze die nicht weiter brachten.

Ich habe Claude nach Design Patterns gefragt folgende Vorschläge gab es:

Aspekt	Feature-Based	Clean Architecture	MVVM
Grundprinzip	Organisation nach Funktionalitäten	Strenge Schichtentrennung mit Abhängigkeitsregeln	Trennung von Ansicht und Ansichtslogik
Ordnerstruktur	Nach Features gruppiert, jedes Feature enthält alle relevanten Schichten	Konzentrische Kreise: Domain (innerste), Data, Presentation (äußerste)	Models, Views, ViewModels als Hauptgruppen
Stärken	Leichte Navigation innerhalb eines Features, bessere Isolation von Features, gut für parallele Teamarbeit	Sehr klare Abhängigkeitsrichtung, Domain-Schicht völlig unabhängig und Langfristige Wartbarkeit	Intuitive Struktur, einfacher zu erlernen und Gute Testbarkeit der ViewModels
Schwächen	Mögliche Code-duplizierung zwischen Features, kann bei gemeinsamen Komponenten unübersichtlich werden	Steilere Lernkurve, Mehr Boilerplate-Code und Überkompliziert für kleinere Apps	Weniger rigide Regeln als Clean Architecture, ViewModels können zu groß werden
Skalierbarkeit	Sehr gut	Exzellent	Gut
Komplexität	Mittel	Hoch	Mittel
Eignung für	Mittelgroße bis große Apps mit klar abgegrenzten Features	Komplexe, langlebige Enterprise-Anwendungen	Apps mit vielen UI-Zusständen
State Management	Flexibel (BLoC, Provider, Riverpod, usw.)	Meist Use-Cases und Repositories	Typischerweise Provider oder Riverpod
Testbarkeit	Gut	Sehr gut	Gut

Code-Isolation	Feature-Isolation	Schichten-Isolation	Zuständigkeits-Isolation
Lernkurve	Mittel	Steil	Flach bis mittel
Teamarbeit	Sehr gut (Feature-Teams)	Gut (Schichten-Teams)	Gut
Änderungen an Geschäftslogik	Auf Features beschränkt	Einfach durch isolierte Domain-Schicht	Auf ViewModels beschränkt
Änderungen an UI	Auf Features beschränkt	Auf Presentation-Schicht beschränkt	Nur Views betroffen
Wiederverwendbarkeit	Mäßig zwischen Features	Sehr gut (besonders Domain-Schicht)	Gut (besonders ViewModels)
Code-Organisation	Nach Business-Funktionen	Nach technischen Schichten	Nach Presentation-Pattern
Flutter-Spezifisch	Gut integrierbar	Erfordert mehr Anpassung	Natürlich passend

1.2 Nutzwertanalyse

Habe anschliessend eine Wertung gemacht. Basierend auf mein Projekt und folgender Wert:

Wert	Bedeutung
0	Irrelevant
1	Schlecht
2	Mittel
3	Gut

Tabelle 2: Wertung

Aspekt	Feature-Based	Clean Architecture	MVVM
Grundprinzip	0	0	0
Ordnerstruktur	0	0	0
Stärken	1	2	3
Schwächen	2	1	2
Skalierbarkeit	3	3	2
Komplexität	2	1	2
Eignung für	3	1	2
State Management	0	0	0
Testbarkeit	2	3	2
Code-Isolation	0	0	0
Lernkurve	2	1	2
Teamarbeit	0	0	0
Änderungen an Geschäftslogik	0	0	0
Änderungen an UI	0	0	0
Wiederverwendbarkeit	1	3	3
Code-Organisation	0	0	0
Flutter-Spezifisch	2	1	3
TOTAL	18	16	21

Tabelle 3: Pattern Nutzwertanalyse

1.3 Fazit

Nach der Wertung hat klar MVVM gewonnen und so fahre ich fort. Beim ersten Anlauf ist mir Aufgefallen, das der Service, also die Schnittstelle zum Backend am Problematischstens ist. Daher habe ich mich folgenden Plan erstellt.

- Services
- Models
- Views
- Integrationstest

Inhaltsverzeichnis

1 Designpattern in Flutter	2
1.1 Design Pattern.....	3
1.2 Nutzwertanalyse.....	5
1.3 Fazit.....	6

Abbildungsverzeichnis

Es konnten keine Einträge für ein Abbildungsverzeichnis gefunden werden.

Tabellenverzeichnis

Tabelle 1: Änderungsverzeichnis	1
Tabelle 2: Wertung.....	5
Tabelle 3: Pattern Nutzwertanalyse.....	5

Anhang E4

Security by Design

Delegator

Klassifizierung intern
Status in Arbeit
Programmname Delegator
Projektnummer 1
Projektleiter Tabinas Kenan
Version 0.1
Datum 26. März 2025
Auftraggeber Tabinas Kenan
Autor/Autoren Tabinas Kenan
Verteiler

Änderungsverzeichnis

Version	Datum	Änderung	Autor
0.1	26.03.2025	Erstellung	Kenan Tabinas
0.2	16.04.2025	Rollenkonzept hinzugefügt	Kenan Tabinas

Tabelle 1: Änderungsverzeichnis

1 Prinzip

Folgende Entscheidung halfen bei der Sicherstellung von Sicherheitsrelevanten Features.

1.1 Grundprinzip

Ganz nach dem Motto «Sicherheit noch vor der ersten Code-Zeile» sind während jedem Schritt die Sicherheitsrelevanten Parameter beachtet worden.

Folgende 3 Grundprinzipien wurden beachtet:

«Angriffe erwarten: Security by Design basiert auf der Annahme, dass Cyber-Angriffe, Sicherheitslücken und Benutzerfehler erfolgen. Deshalb gilt es, deren Auswirkungen zu minimieren.

Security by Obscurity vermeiden: In der Praxis und in Studien haben sich offene Codes gegen geschlossene durchgesetzt: Leaks, Unfälle und Reverse Engineering gehören zur Realität. Deshalb haben sich Offenheit und Transparenz als sicherer erwiesen als die sogenannte Security by Obscurity, bei der die Codes geheim bleiben sollen.

Privilegien einschränken: Das Prinzip der eingeschränkten Privilegien (principle of least privilege, PoLP) basiert darauf, Benutzenden, Prozessen und Programmen nur Zugriff auf die Informationen und Ressourcen zu gewähren, die für ihre Arbeit unbedingt erforderlich sind.»¹

¹ <https://www.nexusgroup.com/de/security-by-design>

2 Zero Trust Architektur

Im Rahmen der Möglichkeiten des Projekts orientiert sich die Architektur an Zero Trust Architektur. Nach dem Motto «Nein zu implizitem Vertrauen». Jeder Zugriff, jede Verbindung und jede Kommunikation muss authentifiziert, autorisiert und verschlüsselt erfolgen. Im Folgenden ist eine Liste mit den Prinzipen welche berücksichtigt wurden.

Prinzip	Beschreibung
Cloudflare Tunnel	Schliessen aller Ports
Endpoint Authentication	Jeder Knoten muss sich Authentisieren. (mutual TLS, API Keys, usw.)
Least Privilege Access	Minimale Berechtigungen für jeden User und Service

Tabelle 2: Zero Trust

2.1 Darstellung

Es folgt eine Darstellung der Kommunikationswege innerhalb der Umgebung.

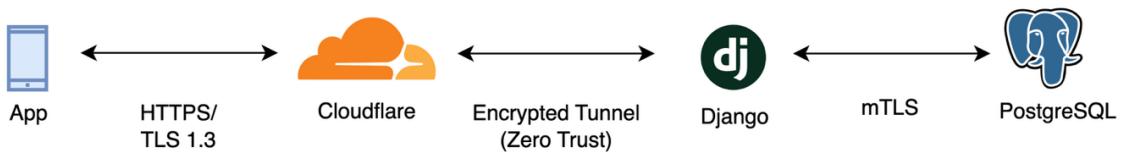


Abbildung 1: Zero Trust

2.2 Massnahmen Dokumentation

Zero Trust schreibt auch vor jeden Layer der Verbindung muss Dokumentiert werden.

Layer	Massnahmen
APP	Certificate Pinning (via Middleware) Flutter Secure Storage (via Middleware) Input Validation (via Regex)
Cloudflared	DDoS Protection (automatisch) Bot Management (automatisch) Rate Limiting (automatisch)
Django	CSRF Protection (via Middleware) SQL Injection Prevention (via Django ORM) Secure Cookie (via Middleware)
PostgreSQL	Field-level encryption (für z.B. Passwörter) Service User mit Least Privilege

Tabelle 3: Zero Trust Massnahmen

3 Stride

“STRIDE ist ein Modell von Sicherheitsrisiken. STRIDE unterscheidet folgende 6 Kategorien an Sicherheitsrisiken»²

Buchstabe	Beschreibung	Beispiel
S	Spoofing	Email Spoofing, IP Spoofing
T	Tampering	Man-in-the-middle, Database Tampering (Ändern von Fremden Daten)
R	Repudiation	Steht für «Abstreitbarkeit». Wenn ich z.B einen «Einbruch» nicht beweisen kann durch z.B: Fehlende Audit Logs
I	Information Disclosure	SQL Injection, Directory auflisten, Banner grabbing,
D	Denial of Service	DDoS-Angriff, Resource Exhaustion
E	Elevation of Privilege	Token Manipulation, Buffer Overflow

Tabelle 4: Stride Auflistung

² <https://de.wikipedia.org/wiki/STRIDE>

3.1 Stride Analyse

	Bezug zum Projekt Delegator	Gegenmaßnahmen
S	Stehlen von Credentials oder Tokens. Durch unsichere Verbindung oder fehlende Validierung	MFA Tokens mit kurzer Lebensdauer Risk-based Authentifizierung
T	Manipulation von fremden Aufgaben, Projekten oder Kalenderdaten Durch unsichere Endpoints oder direkten Datenbank zugriffen	Inter-Container-Verschlüsselung Zugriffskontrolle Least Privilege CSRF-Schutz SQL-ORM
R	Ohne Log sind Manipulationen nicht beweisbar	Logging mit zentralisiertem Log-system (z. B. ELK Stack) SIEM-System (z. B. Wazuh)
I	Offenlegung sensibler Nutzerdaten Durch unsichere Endpoints oder unsichere Verbindungen	Zero Trust Tunnel (Cloudflare) HTTPS zwang Field-Level-Verschlüsselung in PostgreSQL Secure Cookies
D	Cloud-Service oder App ist nicht mehr erreichbar Durch Sperrung oder eigenes Versäumnis.	Cloudflare DDoS-Schutz Rate Limiting Redundanz Befolgen von Store Richtlinien
E	Ein regulärer Nutzer könnte unerlaubte Aktionen ausführen. Durch unsichere Endpoints.	Rollenbasiertes Zugriffssystem Keine Container auf Root Secrets Management Zero Trust auf API-Ebene

Tabelle 5: Stride Analyse

4 Rollenkonzept

Hier folgen die Berechtigungen welche die User erhalten.

ID	Rolle	Automatischer Zugriff
1	Admin	Alles
2	Long-Term member	Projekte, Verträge, Geld, User Verwaltung
3	Member	Projekte
4	Familie	Kalender Sychronisation
5	Fans	Zugriff auf öffentliche Kalender
6	Externer	Keine (Mixer*in, Videograph*in, Photograph*in.)

Tabelle 6: Rollenkonzept

4.1 Berechtigungsmatrix

Endpoint	Read	Create	Update	Delete
users	Dein eigener User. User deiner Org	Ja	Nur dein eigener User	Nur dein eigener User
organisations	Via user-organisation	Nur als Premium User	Eigene Org durch user-organisation	Via user-organisation
roles	Alle	x	x	x
user-organisations	Eigene Org durch user-organisation	Als Admin von Org. Automatisch bei erstellung von Org.	Als Admin von Org	Als Admin von Org
calendars	Via Projects oder Personelle calender via user_id	Automatisch bei erstellung von Projekten	x	Automatisch bei löschen von Projekten. Automatisch bei entfernung von Mitgliedern
events	Via Kalender	Via Kalender	Via Kalender	Via Kalender
projects	Via Org. Via user-projects	Via org. (Min-Role: 3)	Via Org. Via user-projects	Via Org. Via user-projects
chats	Via Org. Via Projekte.	Via org. (Min-Role: 3)		Automatisch bei löschen von Projekten.
chat-users	Via Chat	Via Chat	Via Chat	Via Chat
messages	Via Chat	Via Chat	Via Chat. Via User (eigene)	Via Chat. Via User (eigene)
songs	Via org. (Min-Role: 3)	Via org. (Min-Role: 3)	Via org. (Min-Role: 3)	Via org. (Min-Role: 3)

Tabelle 7: Berechtigungsmatrix 1

Endpoint	Read	Create	Update	Delete
timetables	Via org. (Min-Role: 3). Via Projekt.	Via org. (Min-Role: 3). Via Projekt.	Via org. (Min-Role: 3). Via Projekt.	Via org. (Min-Role: 3). Via Projekt.
setlists	Via org. (Min-Role: 3). Via Projekt.	Via org. (Min-Role: 3). Via Projekt.	Via org. (Min-Role: 3). Via Projekt.	Via org. (Min-Role: 3). Via Projekt.
history	Via org. (Min-Role: 3)	Automatisch	x	x
statuses	Alle	x	x	x
tasks	Via org. (Min-Role: 4). Via Projekt.	Via org. (Min-Role: 3). Via Projekt.	Via org. (Min-Role: 3). Via Projekt.	Via org. (Min-Role: 3). Via Projekt.
recordings	Via org. (Min-Role: 3). Via Projekt.	Via org. (Min-Role: 3). Via Projekt.	Via org. (Min-Role: 3). Via Projekt.	Nur dein eigener User
user-projects	Via org. (Min-Role: 3). Via eigener User	Via org. (Min-Role: 3)	Via org. (Min-Role: 3)	Via org. (Min-Role: 3)

Tabelle 8: Berechtigungsmatrix 2

5 Backup und Restore

5.1 GFK

Auch beim Wechsel auf einen Managed Service, darf ein Backup and Restore Konzept nicht fehlen. Wieso wird später erklärt. Das (GFK) Grand-Father-Son Backup Prinzip hat überzeugt. Es gibt jedoch kein Zeitraum vor. Hier wird jetzt 7-4-12 dargestellt.

Woche	Mo	Di	Mo	Do	Fr	Sa	So
1	Yellow						
2	Yellow						
3	Yellow						
4	Yellow						Orange

Tabelle 9: GFK Darstellung

Farbe	Backup Intervall
Blue	Nach 24h wird ein Backup erstellt. Tägliches Backup mit 7-Tage-Retention.
Yellow	Nach 7 Tagen wird ein Wochen Backup erstellt. Wochen Backup mit 4-Wochen-Retention.
Orange	Nach 4 Wochen wird der Monat Backup erstellt. Monats Backup mit 12-Monats-Retention.

Tabelle 10: GFK-Legende

5.2 Backup 3-2-1-1-0

Die bekannte 3-2-1 wurde durch weitere Ziffern ergänzt. Das System sieht folgendes vor. Dabei gibt 3-2-1 nicht vor in welcher Frequenz Backups gemacht werden sollten. Sondern auf Was und wo sie gespeichert werden soll.

5.2.1 3-2-1-1-0 Erklärung

Ziffer	Erklärung
3	Dieselbe Information sollte dreimal Existieren. Einmal Live und z.B. zweimal als Backup. Man kann bei hoch Sensiblen Daten auch eine Datenbank Replikation erstellen.
2	Backups sollten auf Verschiedenen Medien gespeichert werden.
1	Ein Backup muss an einem anderen Orten sein. Nur einem Cloud-Provider vertrauen verstößt gegen diesen Punkt.
1	Ein Backup muss offline und nicht beschreibbar sein
0	Keine Fehler bei Wiederherstellung. Dies erfordert regelmässige Tests.

Tabelle 11: 3-2-1-1-0

5.3 Parameter

Damit wir die Frequenz bestimmen können orientieren wir uns an bestimmten «Kennzahlen».³

5.3.1 Parameter Erklärung

Abkürzung	Parameter	Erklärung
RPO	Recovery Point Objective.	Zeit Abstand zwischen Backups
RTO	Recovery Time Objective.	Zeit vom Incident bis zum Rollback
WRT	Work Recovery Time	Zeitraum vom Rollback bis Freigabe
MTD	Maximum Tolerable Downtime	Maximal erlaubter Zeitraum zwischen Incident und Freigabe

Tabelle 12: Parameter Erklärung

³ https://www.cms.gov/tra/Infrastructure_Services/IS_0410_DR_Capability_Considerations.htm

5.3.2 RTO-Vergleichswerte

Diese Zeiträume definieren ist noch schwer. Es folgen Wert aus der Praxis als vergleich. Laut darwinsdata.com sehen die RTO Werte so aus.⁴

Branche	From (h)	To (h)
Finanz	1	4
E-commerce	1	24
Gesundheit	1	72
Produktion	24	48
Einzelhandel	24	72
Bund	24	72

Tabelle 13: RTO-Vergleichswerte

5.3.3 Tiers und Dynamische Backups

Selbst bei der Bank, erhalten nicht alle Daten dieselbe RTO. Zahlungen dürfen nach Bank Richtlinen wie z.B: SEC nur eine minimale Downtime haben. Hingegen Administrative Daten, für die Bank selbst, sind nicht so streng.

Eine Variante wäre es auch, Backups nicht nur nach einer gewissen Zeit zu machen, sondern auch nach Datenfluss. So wird werden bei einem Peak, auch häufiger Backups gemacht.

5.3.4 Parameter

Offlinesynchronisierung kommt definitiv auf den Plan. Es reduziert den Backup kosten massiv. Die Faustregel ist «halbe so lange ist doppelt so teuer.»

Abkürzung	Zeitraum (h)	Begründung
RPO	2-4	4 Stunden sind tolerierbar.
RTO	4-8	Als Ein-Mann-Unternehmen ist kürzer unrealistisch
WRT	8-12	So ein drittel bis einen halben Tag kann man Warten
MTD	24	Mehr als ein ganzer Tag ist zu lang

Tabelle 14: Delegator Parameter

⁴ <https://darwinsdata.com/what-is-industry-standard-rpo-rto/>

6 Fazit

6.1 Security Controls

Basierend auf den Prinzipen, der Stride Analyse und der Zero Trust Architektur ergeben sich folgende 8 Security Controls. Diese kommen jeweils mit konkreten planen zur Umsetzung.

Security Controls	Geplante Umsetzung
Angriffsfläche minimieren	Cloudflared Zero Trust Tunnel ermöglicht Verbindung, ohne einen einzigen Port zu öffnen. Zugriff auf Server ist nur im privaten LAN erreichbar. (Zero Trust) Regelmässiges Container Image scanning
Verschlüsselung	Cloudflared Zero Trust Tunnel und eigenem Trusted CA zu Cloudflare Inter-Container Verschlüsselung mit z.B. Isito Backup Verschlüsslung
Authentifizierung	MFA für Logins Tokens mit kurzer Lebensdauer Automatisches Session Timeout Risk-based Auth (Standort-Anomalien, Gräte Profiling) z.B. loginradius API Access Rate limiting
Least Privilege	Keine Passwörter Secrets Management Container laufen nie auf Root. Zugriffe werden grundsätzlich abgelehnt.
Getrennte Systeme	Container Trennen die Kommunikation via VLAN
Monitoring	Zentralisiertes Log mit z.B: ELK Stack Security Information and Event Management (SIEM) z.B: Wazuh
Incident Reponse	Incident Reponse Framework erstellen und daraus konkrete Playbooks als Prozess ableiten
Compliance	Externer Penntest ins Budget einflieessen lassen Jährliche Audits

Tabelle 15: Security Controls

6.2 Backup Plan

Kombinieren wir das Wissen aus den vorgängigen Konzepten und Prinzipien ergibt sich folgender Backup Plan.

6.2.1 GFK-Strategie

Kind: 4h mit 24h Retention

Vater: 1 Tag mit 7 Tag Retention

Grossvater: 1 Woche mit 4 Wochen Retention

Tag	0	4	8	12	16	20
1	Yellow	Blue	Blue	Blue	Blue	Blue
2	Yellow	Blue	Blue	Blue	Blue	Blue
3	Yellow	Blue	Blue	Blue	Blue	Blue
4	Yellow	Blue	Blue	Blue	Blue	Blue
5	Yellow	Blue	Blue	Blue	Blue	Blue
6	Yellow	Blue	Blue	Blue	Blue	Blue
7	Yellow	Blue	Blue	Blue	Blue	Orange

Tabelle 16: GFK Strategie

6.2.2 3-2-1-1-0 Strategie

Ziffer	Strategie
3	1. Exoscale DBaaS 2 Exoscale backup service 3 Export zu Scaleway
2	1 Exoscale Cloudstorage 2 Scaleway Objectstorage
1	1 Backup bei Scaleway (Frankreich) (0.1752€ pro GB pro Jahr)
1	1 Raspi mit Script (Physical Air gapped Backup)
0	0 Fehler durch Test siehe mehr bei Disaster Recovery

Tabelle 17: 3-2-1-1-0 Strategie

6.2.3 Disaster Recovery Plan

Je nach Regulatorium muss es mehr oder weniger häufig getestet werden. 3 Monaten, 6 Monate und 12 Monaten sind mögliche Zeiträume. Es bietet sich jedoch an, vor grossen Updates ein Recovery Test durchzuführen.

Halbjährlich scheint vernünftig zu sein.

6.2.4 3-2-1-1-0 Zeitplan

Aus all diesen Informationen ergibt sich folgender Zeitplan.

Strategie	Zeitplan
1. Exoscale DBaaS	-
2 Exoscale backup service	04h
3 Export zu Objectstorage	24h
1 Exoscale Cloudstorage	04h
2 Scaleway Objectstorage	24h
1 Backup bei Scaleway	24h
1 Raspi mit Script	4 Tage
Disaster Recovery Plan	6 Monate

Tabelle 18: 3-2-1-1-0 Zeitplan

6.2.5 Wachstum

Was kostet dieser spass? Das hängt stark vom Speicherplatz ab. Nehmen wir an pro Monat ein GB erreicht die Aktuelle Menge User. Wenn wir uns das Wachstum anschauen werden Ende 2tes Jahr die 3 Fache Menge Daten sein.

Im ersten Jahr. Bei 12 GB und somit 204 GB Back Up Storage.

Scaleway (Frankreich) (0.1752€ pro GB pro Jahr)

Exoscale Backup Service (0.2409€ pro GB pro Jahr)

Bei 1€ = 0.94 CHF

Jahr	Storage (GB)	Scaleway (GB)	Exoscale (GB)	Kosten (CHF)
1	12	48	156	43
2	36	144	468	130
3	72	288	936	259
4	120	480	1560	432
5	180	720	2340	648
6	252	1008	3276	908
7	336	1344	4368	1210
8	432	1728	5616	1556
9	540	2160	7020	1945
10	660	2640	8580	2378

Tabelle 19

Abkürzungen und Glossar

Abkürzung / Fachwort	Erläuterung
API	Application Programming Interface
CSRF	Cross-Site Request Forgery
DDoS	Distributed Denial of Service
ELK	Elasticsearch, Logstash, Kibana (Stack)
GFK	Grossvater-Vater-Sohn (Backup-Prinzip)
HTTPS	Hypertext Transfer Protocol Secure
MFA	Multi-Factor Authentication
MTD	Maximum Tolerable Downtime
ORM	Object-Relational Mapping
PoLP	Principle of Least Privilege
RPO	Recovery Point Objective
RTO	Recovery Time Objective
SEC	Securities and Exchange Commission
SIEM	Security Information and Event Management
SQL	Structured Query Language
STRIDE	Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, Elevation of Privilege
TLS	Transport Layer Security
VLAN	Virtual Local Area Network
WRT	Work Recovery Time
3-2-1-1-0 Backup	Backup-Strategie: 3 Kopien, 2 verschiedene Medien, 1 offsite, 1 offline, 0 Fehler
Air Gapped Backup	Physisch getrennte Backup-Lösung ohne Netzwerkverbindung
API Keys	Eindeutige Identifikatoren für API-Zugriff
Banner Grabbing	Sammeln von Informationen über Services durch Systemantworten
Bot Management	Automatische Erkennung und Abwehr von Bots
Certificate Pinning	Festlegung spezifischer Zertifikate für sichere Verbindungen
Cloudflare Tunnel	Sichere Verbindung ohne offene Ports
Container Image Scanning	Automatische Überprüfung von Container-Images auf Schwachstellen

Abkürzung / Fachwort	Erläuterung
Directory Listing	Unerlaubte Auflistung von Verzeichnisinhalten
Disaster Recovery	Wiederherstellungsplan nach Systemausfall
Field-Level Encryption	Verschlüsselung auf Datenfeldebene
Flutter Secure Storage	Sichere Datenspeicherung in Flutter-Apps
Incident Response	Reaktion auf Sicherheitsvorfälle
Input Validation	Überprüfung von Eingabedaten
Inter-Container Verschlüsselung	Verschlüsselung zwischen Containern
Least Privilege	Minimale Berechtigungen für Benutzer und Services
Man-in-the-Middle	Angriff durch Abfangen der Kommunikation
Mutual TLS	Gegenseitige TLS-Authentifizierung
Penetration Test	Sicherheitstest durch simulierte Angriffe
Rate Limiting	Begrenzung der Anfragerate
Resource Exhaustion	Erschöpfung von Systemressourcen
Risk-based Authentication	Risikobasierte Authentifizierung
Secrets Management	Verwaltung von Passwörtern und Schlüsseln

Tabelle 20: Abkürzungen und Glossar

Glossar erstellt bei Claude. Begriffe und Beschreibungen.

Inhaltsverzeichnis

1	Prinzip.....	2
1.1	Grundprinzip.....	2
2	Zero Trust Architektur	3
2.1	Darstellung.....	3
2.2	Massnahmen Dokumentation.....	4
3	Stride.....	4
3.1	Stride Analyse	5
4	Rollenkonzept.....	5
4.1	Berechtigungsmatrix	6
5	Backup und Restore.....	8
5.1	GFK.....	8
5.2	Backup 3-2-1-1-0	9
5.2.1	3-2-1-1-0 Erklärung.....	9
5.3	Parameter.....	9
5.3.1	Parameter Erklärung.....	9
5.3.2	RTO-Vergleichswerte	10
5.3.3	Tiers und Dynamische Backups	10
5.3.4	Parameter	10
6	Fazit	11
6.1	Security Controls.....	11
6.2	Backup Plan.....	12
6.2.1	GFK-Strategie.....	12
6.2.2	3-2-1-1-0 Strategie.....	12
6.2.3	3-2-1-1-0 Zeitplan.....	13
6.2.4	Disaster Recovery Plan.....	13

Abbildungsverzeichnis

Abbildung 1: Zero Trust.....	3
------------------------------	---

Tabellenverzeichnis

Tabelle 1: Änderungsverzeichnis.....	1
Tabelle 2: Zero Trust.....	3
Tabelle 3: Zero Trust Massnahmen	4
Tabelle 4: Stride Auflistung.....	4
Tabelle 5: Stride Analyse	5
Tabelle 6: Rollenkonzept	5
Tabelle 7: Berechtigungsmatrix 1.....	6
Tabelle 8: Berechtigungsmatrix 2	7
Tabelle 9: GFK Darstellung	8
Tabelle 10: GFK-Legende.....	8
Tabelle 11: 3-2-1-1-0.....	9
Tabelle 12: Parameter Erklärung.....	9
Tabelle 13: RTO-Vergleichswerte	10
Tabelle 14: Delegator Parameter.....	10
Tabelle 15: Security Controls.....	11
Tabelle 15: GFK Strategie	12
Tabelle 16: 3-2-1-1-0 Strategie.....	12
Tabelle 18: 3-2-1-1-0 Zeitplan	13
Tabelle 19: Abkürzungen und Glossar	16

Anhang E5

Design Konzept

Delegator

Klassifizierung intern
Status in Arbeit
Programmname Delegator
Projektnummer 1
Projektleiter Tabinas Kenan
Version 0.1
Datum 26. März 2025
Auftraggeber Tabinas Kenan
Autor/Autoren Tabinas Kenan
Verteiler

Änderungsverzeichnis

Version	Datum	Änderung	Autor
1	03.06.2025	Erstellung	Kenan Tabinas
1.1	07.06.2025	Systemtest hinzugefuegt	

Tabelle 1: Änderungsverzeichnis

1 Testkonzept

1.1 Teststrategien

Nr.	Teststrategie	Beschreibung
01	Contract Testing	Test der API mit Postman
02	Integrationstest	Verknüpfung von Frontend und Backend mit Flutter-Unit-Tests
03	Anforderungsabdeckung	Testfälle basieren auf Anforderungen aus der Initialisierungsphase
04	Usability-Test	Ungeübte Benutzer*innen führen Aufgaben ohne Anleitung
05	Security-Test	Überprüfung der Endpoints auf Schwachstellen

Tabelle 2

Tabelle 3: Teststrategien

1.2 Testobjekte

Nr.	Objekt	Beschreibung
01	Frontend	Funktionalität, Darstellung auf Verschiedenen Screens
02	Backend	Modelle wie User, Organisation auf Daten und Zugriff

Tabelle 4: Testobjekte

1.3 Testabdeckung

1.3.1 Übersicht Testfälle

Nr.	Testobjekt	Testfälle
01	User	Zugriff ohne Recht
02	User	Zugriff
03	User	Manipulation ohne Recht
04	Bewegungsdaten	Manipulation
05		

Tabelle 5: Testfälle

1.3.2 Testabdeckung

Text, grundsätzliche Einschätzung zu Zielen und Abdeckung des Testings

1.4 Testplan

Nr.	Aktivität	Verantwortlich	Mitarbeit	Termin
01	Postmann	Kenan Tabinas	Kenan Tabinas	03.06.2025
02	Integrationstest	Kenan Tabinas	Kenan Tabinas	03.06.2025
03	Abnahmetest	Kenan Tabinas	Noëmi Tabinas	03.06.2025
04	Usertest	Kenan Tabinas	Noëmi Tabinas	03.06.2025
05	Sysstemtest	Kenan Tabinas	Kenan Tabinas	07.06.2025

Tabelle 6: Testplan

Abkürzungen und Glossar

Abkürzung / Fachwort	Erläuterung
HERMES	Vorgehensmethodik für Projekte und Programme HERMES 5 ist ein eCH Standard

Tabelle 7: Abkürzungen und Glossar

Inhaltsverzeichnis

1 Testkonzept	2
1.1 Teststrategien.....	2
1.2 Testobjekte.....	2
1.3 Testabdeckung	2
1.3.1 Übersicht Tesfälle.....	2
1.3.2 Testabdeckung	3
1.4 Testplan	3

Abbildungsverzeichnis

Es konnten keine Einträge für ein Abbildungsverzeichnis gefunden werden.

Tabellenverzeichnis

Tabelle 1: Änderungsverzeichnis.....	1
Tabelle 2: Teststrategien.....	2
Tabelle 3: Testobjekte.....	2
Tabelle 4: Testfälle.....	2
Tabelle 5: Testplan.....	3
Tabelle 6: Abkürzungen und Glossar	4

Anhang E6

Datenbank ERDs

Delegator

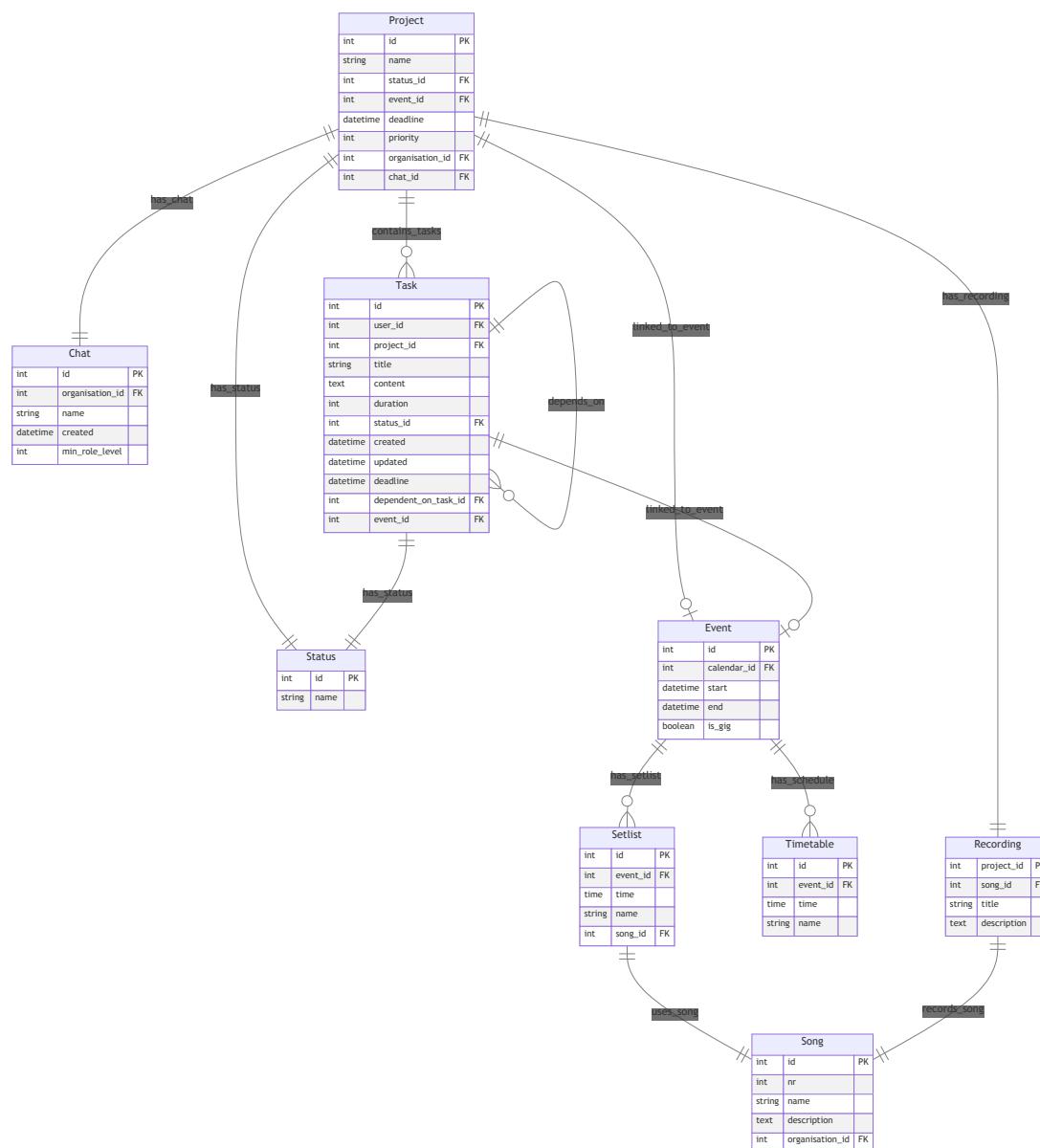
Klassifizierung intern
Status in Arbeit
Programmname Delegator
Projektnummer 1
Projektleiter Tabinas Kenan
Version 1
Datum 06. Juni 2025
Auftraggeber Tabinas Kenan
Autor/Autoren Tabinas Kenan
Verteiler

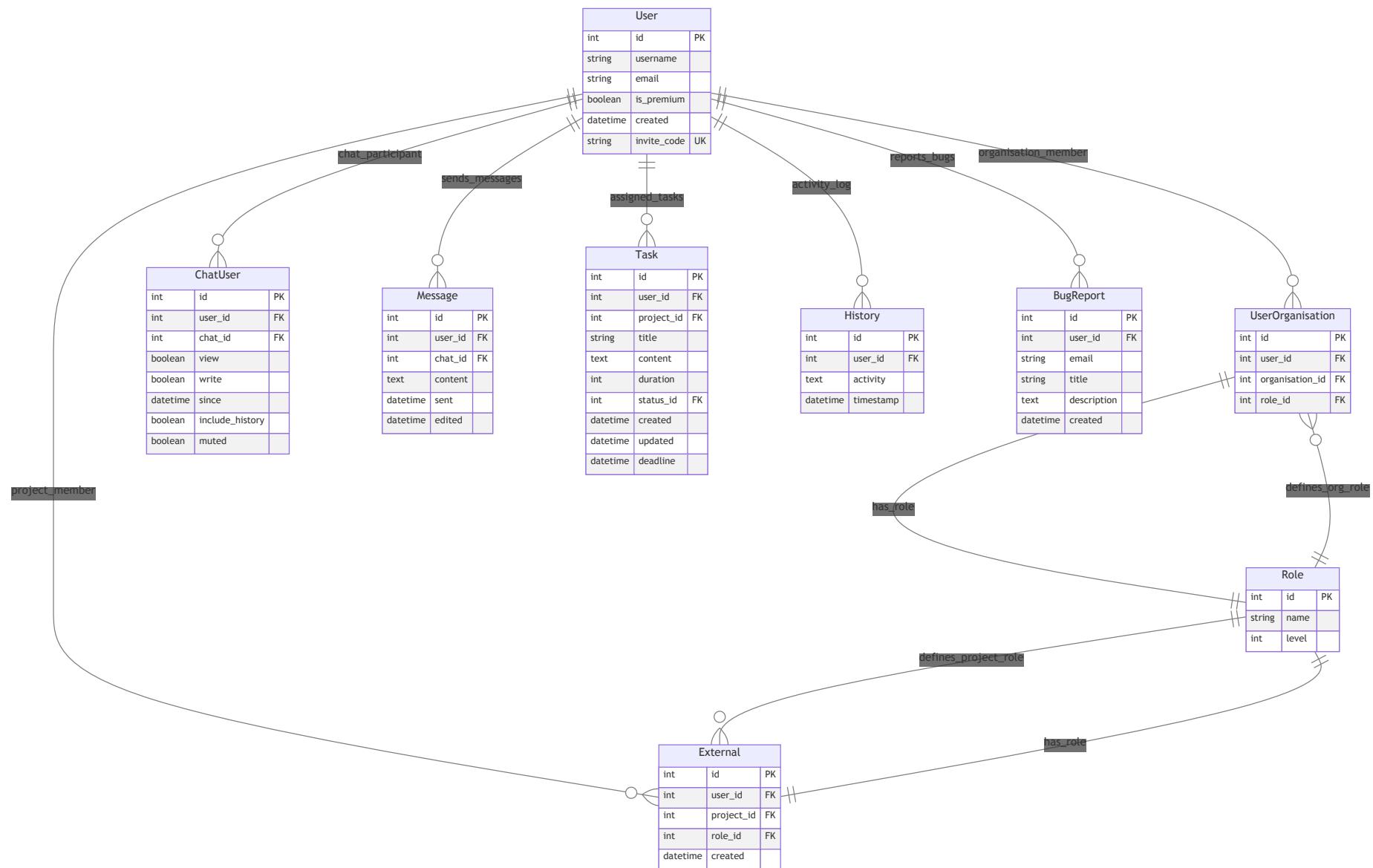
Änderungsverzeichnis

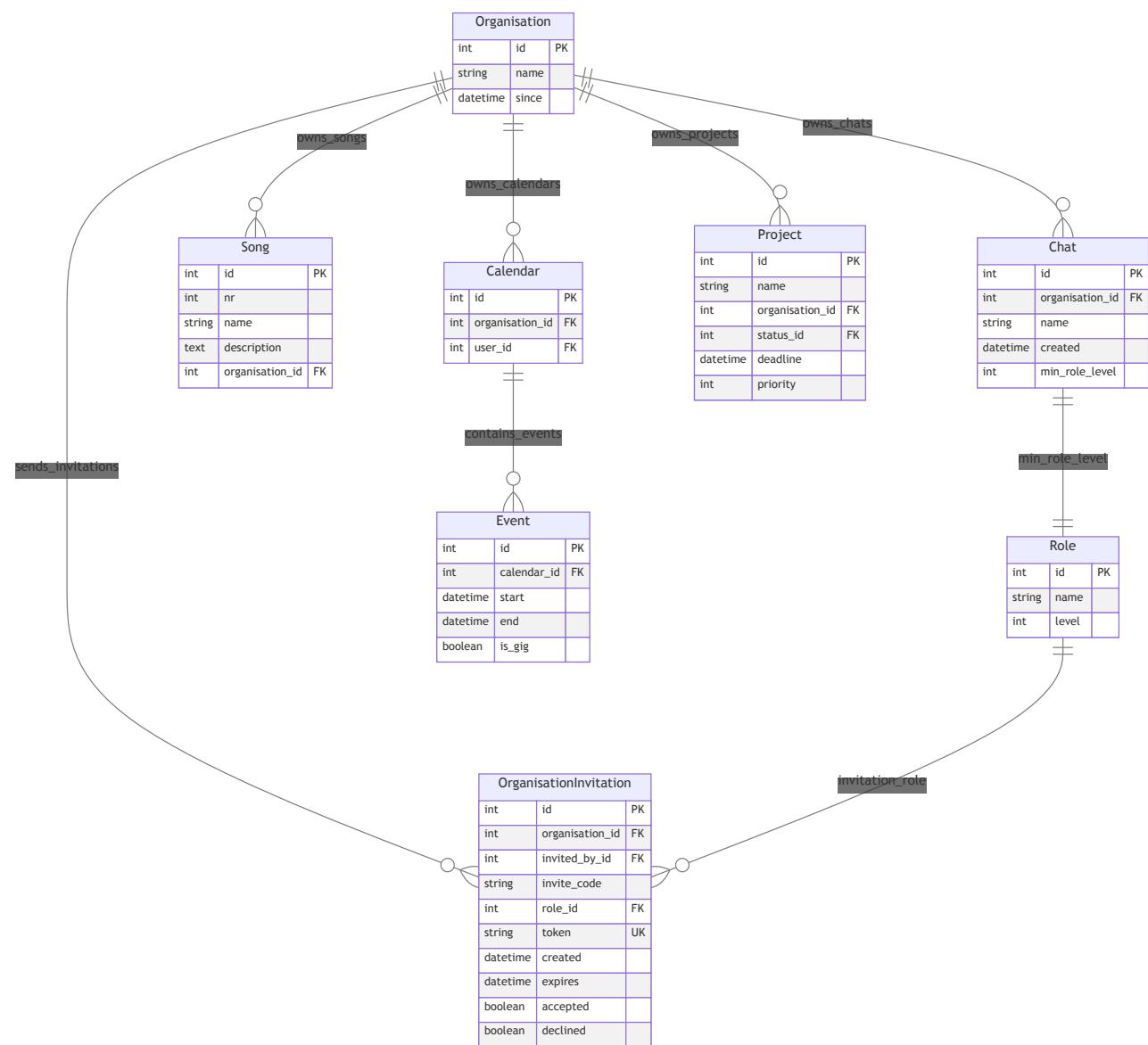
Version	Datum	Änderung	Autor
1	07.06	Erstellt	TAK

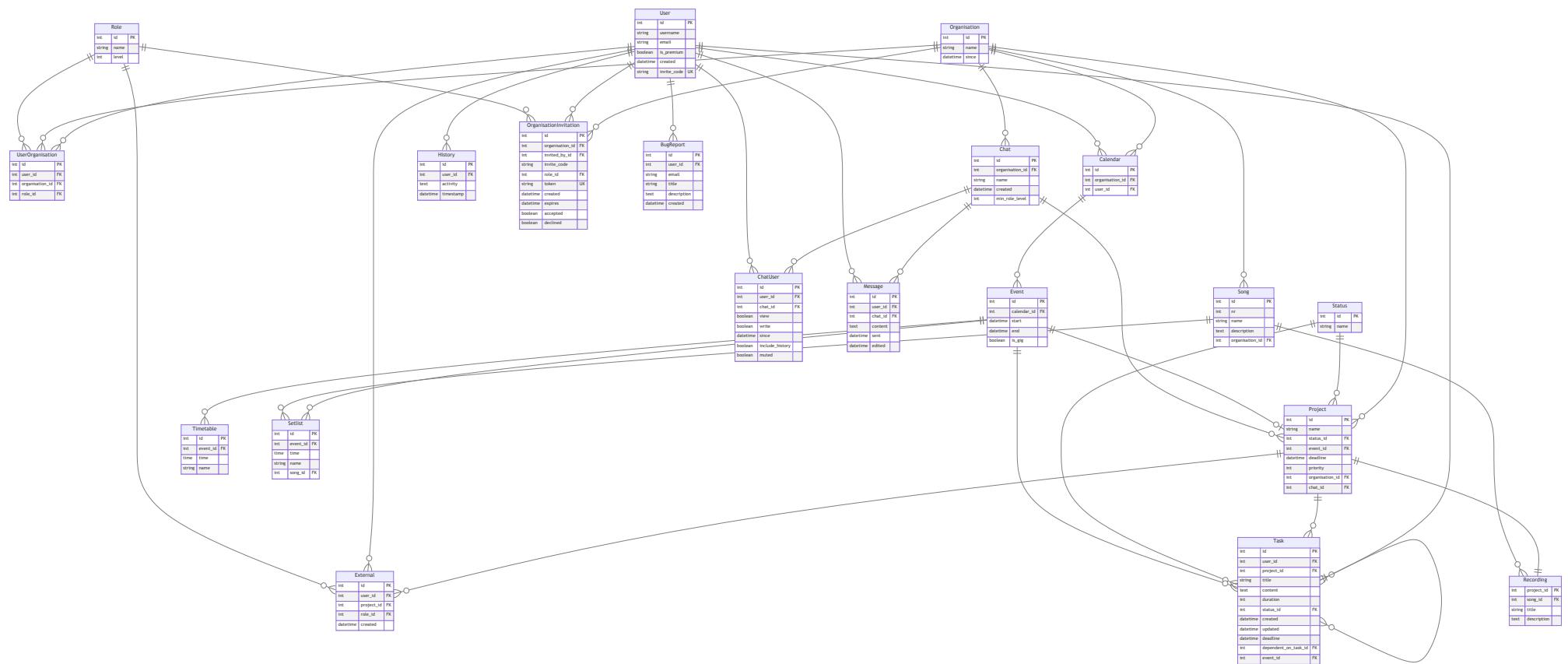
Tabelle 1: Änderungsverzeichnis

Es folgen 4 ERD. Projekt, User, Organisation und einmal alles.









Alle 4 Generiert von Claude mit Marmaid.

Anhang F

AGB

Delegator

Klassifizierung intern
Status in Arbeit
Programmname Delegator
Projektnummer 1
Projektleiter Tabinas Kenan
Version 0.1
Datum 26. März 2025
Auftraggeber Tabinas Kenan
Autor/Autoren Tabinas Kenan
Verteiler

Änderungsverzeichnis

Version	Datum	Änderung	Autor
1	05.06	Erstellt	TAK

Tabelle 1: Änderungsverzeichnis

Allgemeine Geschäftsbedingungen (AGB)

Delegator - Organisationsplattform für Bands

Stand: 05.06.2025
Anbieter: Tabinas Kenan, Melchtalstrasse 10, Schweiz
E-Mail: info@delegator.ch
Handelsregister: XXXXXXXX

§ 1 Geltungsbereich und Vertragspartner

1.1 Geltungsbereich

Diese Allgemeinen Geschäftsbedingungen (nachfolgend "AGB") gelten für die Nutzung der Delegator-Plattform (nachfolgend "Service" oder "App"), einer webbasierten und mobilen Anwendung zur Organisation und Verwaltung von Bands, Musikgruppen und ähnlichen Organisationen.

1.2 Vertragspartner

Vertragspartner ist Tabinas Kenan (nachfolgend "Anbieter", "wir", "uns"). Der Nutzer wird nachfolgend als "Sie" oder "Nutzer" bezeichnet.

1.3 Abweichende Bedingungen

Abweichende, entgegenstehende oder ergänzende allgemeine Geschäftsbedingungen des Nutzers werden nicht Vertragsbestandteil, es sei denn, ihrer Geltung wird ausdrücklich schriftlich zugestimmt.

§ 2 Leistungsbeschreibung

2.1 Service-Umfang

Delegator bietet eine Organisationsplattform mit folgenden Kernfunktionen:

2.1.1 Kalendermanagement:

- Erstellung, Bearbeitung und Verwaltung von Terminen und Proben
- Automatische Erinnerungen und Benachrichtigungen
- Synchronisation mit externen Kalenderdiensten (Google Calendar, Outlook)

2.1.2 Aufgabenmanagement:

- Erstellung und Zuweisung von Aufgaben an Bandmitglieder
- Verfolgung des Aufgabenstatus und Deadlines
- Benachrichtigungen bei Änderungen

2.1.3 Ressourcenplanung:

- Verwaltung von Equipment und Proberäumen
- Buchungsübersicht und Verfügbarkeitsprüfung

2.1.4 Setlisten-Management:

- Erstellung und Verwaltung von Setlisten
- Export-Funktionen (PDF)

2.1.5 Gruppen-/Bandverwaltung:

- Mitgliederverwaltung und Rollenzuweisungen
- Kommunikationsfunktionen innerhalb der Gruppe

2.2 Technische Verfügbarkeit

Der Service ist als Web-Anwendung und mobile App (Cross-Platform) verfügbar. Wir bemühen uns um eine Verfügbarkeit von 99% im Jahresmittel, ausgenommen geplante Wartungsarbeiten.

2.3 Leistungsänderungen

Wir behalten uns vor, den Funktionsumfang zu erweitern, zu ändern oder einzuschränken, sofern dies für den Nutzer zumutbar ist und die Kernfunktionalität erhalten bleibt.

§ 3 Vertragsschluss und Registrierung

3.1 Vertragsschluss

Der Nutzungsvertrag kommt durch die Registrierung und Bestätigung der AGB zustande. Mit der Registrierung gibt der Nutzer ein bindendes Angebot zum Vertragsschluss ab, welches wir durch Freischaltung des Accounts annehmen.

3.2 Registrierungsvoraussetzungen

- Mindestalter: 16 Jahre (bei Minderjährigen ist die Zustimmung der Erziehungsberechtigten erforderlich)
- Angabe vollständiger und wahrheitsgemäßer Daten
- Gültige E-Mail-Adresse

3.3 Account-Sicherheit

Der Nutzer ist verpflichtet:

- Sichere Passwörter zu verwenden
- Zugangsdaten vertraulich zu behandeln
- Unbefugte Nutzung unverzüglich zu melden
- Bei Verdacht auf Missbrauch sofort das Passwort zu ändern

§ 4 Nutzungsrechte und -pflichten

4.1 Nutzungsrecht

Der Nutzer erhält ein nicht-exklusives, nicht-übertragbares Recht zur Nutzung des Service für die Dauer des Vertragsverhältnisses.

4.2 Zulässige Nutzung

Der Service darf nur für legale Zwecke der Band-/Gruppenorganisation verwendet werden.

4.3 Verbotene Nutzungen

Untersagt sind insbesondere:

- Verwendung für kommerzielle Zwecke Dritter ohne Zustimmung
- Upload von rechtswidrigen, beleidigenden oder urheberrechtsverletzenden Inhalten
- Störung oder Beeinträchtigung der Systemsicherheit

- Reverse Engineering, Dekompilierung oder Disassemblierung
- Automatisierte Datenabfragen (Scraping, Crawling) ohne Zustimmung
- Weitergabe von Zugangsdaten an Dritte

4.4 Inhalte des Nutzers

- Der Nutzer bleibt Inhaber aller Rechte an seinen hochgeladenen Inhalten
 - Der Nutzer räumt uns das Recht ein, diese Inhalte zu speichern und für die Service-Bereitstellung zu verarbeiten
 - Der Nutzer ist für die Rechtmäßigkeit seiner Inhalte verantwortlich
-

§ 5 Preise und Zahlungsbedingungen

5.1 Preismodell

Einzelnutzung: Kostenlos mit eingeschränktem Funktionsumfang

Bandnutzung: 7 CHF pro Monat pro Band (bei monatlicher Zahlung)

Jahresabo: 72 CHF pro Jahr pro Band (entspricht 6 CHF/Monat)

5.2 Zahlungsmodalitäten

- Zahlung im Voraus per Kreditkarte, PayPal oder Banküberweisung
- Bei Monatsabos: Abbuchung jeweils zum Monatsanfang
- Bei Jahresabos: Abbuchung bei Vertragsschluss und jährlich wiederkehrend
- Alle Preise verstehen sich inklusive der gesetzlichen Mehrwertsteuer

5.3 Zahlungsverzug

Bei Zahlungsverzug können wir nach Mahnung den Zugang sperren. Verzugszinsen: 5% p.a. über dem Basiszinssatz.

5.4 Preisänderungen

Preiserhöhungen für Bestandskunden werden mindestens 30 Tage im Voraus angekündigt. Bei Erhöhungen über 5% jährlich steht dem Nutzer ein außerordentliches Kündigungsrecht zu.

§ 6 Laufzeit und Kündigung

6.1 Vertragslaufzeit

- **Kostenlose Accounts:** Unbefristet mit jederzeitiger Kündigungsmöglichkeit
- **Monatsabos:** Laufzeit 1 Monat, automatische Verlängerung
- **Jahresabos:** Laufzeit 12 Monate, automatische Verlängerung

6.2 Ordentliche Kündigung

- **Monatsabos:** Kündigung bis zum 25. des laufenden Monats zum Monatsende
- **Jahresabos:** Kündigung mit 30 Tagen Frist zum Ablauf der Vertragslaufzeit
- **Kündigung in Textform** (E-Mail ausreichend) an support@delegator.ch

6.3 Ausserordentliche Kündigung

Beide Parteien können aus wichtigem Grund fristlos kündigen, insbesondere bei:

- Schwerwiegenden Verstößen gegen diese AGB
- Zahlungsverzug trotz Mahnung (durch uns)
- Erheblichen Leistungsstörungen (durch den Nutzer)

6.4 Folgen der Kündigung

- Zugang wird zum Kündigungszeitpunkt deaktiviert
 - Daten werden 30 Tage zur Wiederherstellung gespeichert, danach gelöscht
 - Bereits gezahlte Beiträge werden nicht erstattet (außer bei Kündigung aus wichtigem Grund unsererseits)
-

§ 7 Datenschutz

7.1 Datenverarbeitung

Die Verarbeitung personenbezogener Daten erfolgt gemäß unserer Datenschutzerklärung und den gesetzlichen Bestimmungen (DSGVO, nDSG).

7.2 Rechtsgrundlagen

- **Vertragserfüllung** (Art. 6 Abs. 1 lit. b DSGVO): Alle zur Service-Bereitstellung erforderlichen Datenverarbeitungen
- **Einwilligung** (Art. 6 Abs. 1 lit. a DSGVO): Marketing-Kommunikation und erweiterte Analytics

7.3 Betroffenenrechte

Nutzer haben das Recht auf Auskunft, Berichtigung, Löschung, Einschränkung der Verarbeitung, Datenübertragbarkeit und Widerspruch gemäß DSGVO.

§ 8 Gewährleistung und Haftung

8.1 Gewährleistung

- Wir gewährleisten die vertragsgemäße Bereitstellung des Service
- Kleine Abweichungen oder Störungen, die die Nutzung nicht erheblich beeinträchtigen, berechtigen nicht zur Minderung oder Kündigung
- Bei erheblichen Mängeln stehen dem Nutzer zunächst Nacherfüllungsansprüche zu

8.2 Haftungsbeschränkung

Unsere Haftung ist beschränkt auf:

- **Vorsatz und grobe Fahrlässigkeit:** Unbeschränkte Haftung
- **Verletzung wesentlicher Vertragspflichten:** Haftung auf den typischen, vorhersehbaren Schaden
- **Leichte Fahrlässigkeit:** Keine Haftung, außer bei Verletzung von Leben, Körper, Gesundheit

Ausgeschlossen ist die Haftung für:

- Datenverlust durch höhere Gewalt
- Schäden durch Fehlbedienung des Nutzers
- Ausfälle durch Drittanbieter (Internet-Provider, Cloud-Services)
- Mittelbare Schäden und entgangenen Gewinn

8.3 Datensicherung

Der Nutzer ist für eigene Backups kritischer Daten verantwortlich. Wir führen regelmäßige Backups durch, übernehmen aber keine Garantie für Datenrettung.

§ 9 Höhere Gewalt

Bei aussergewöhnlichen Umständen (höhere Gewalt, Pandemie, Cyberangriffe, Stromausfälle, etc.) ruhen unsere Leistungspflichten für die Dauer der Beeinträchtigung. Bei Ausfällen über 30 Tage steht beiden Parteien ein Kündigungsrecht zu.

§ 10 Änderungen der AGB

10.1 Änderungsbefugnis

Wir können diese AGB bei sachlich gerechtfertigten Gründen ändern (neue Gesetze, Rechtsprechung, technische Entwicklungen).

10.2 Zustimmungsverfahren

- Änderungen werden mindestens 30 Tage vor Inkrafttreten per E-Mail mitgeteilt
 - Widerspruch binnen 30 Tagen in Textform möglich
 - Bei Widerspruch: Kündigungsrecht zum Zeitpunkt des Inkrafttretens
 - Schweigen gilt als Zustimmung
-

§ 11 Schlussbestimmungen

11.1 Anwendbares Recht

Es gilt ausschließlich Schweizer Recht unter Ausschluss des UN-Kaufrechts.

11.2 Gerichtsstand

Ausschließlicher Gerichtsstand für alle Streitigkeiten ist Bern, Schweiz.

11.3 Salvatorische Klausel

Sollten einzelne Bestimmungen unwirksam sein, bleibt die Wirksamkeit der übrigen Bestimmungen unberührt. Unwirksame Bestimmungen werden durch rechtlich zulässige ersetzt, die dem ursprünglich Gewollten am nächsten kommen.

11.4 Vertragssprache

Diese AGB sind in deutscher Sprache verfasst. Bei Übersetzungen ist die deutsche Fassung maßgeblich.

Kontakt für Fragen zu diesen AGB:

E-Mail: abg@delegator.ch

Adresse: Melchthalstrasse 10, 3014 Bern

Diese AGB wurden zuletzt am 05.06.2025 aktualisiert.

Generiert von Claude

Glossar

Begriff	Erklärung
AGB	Allgemeine Geschäftsbedingungen - rechtliche Regeln zwischen Anbieter und Nutzer
DSGVO	Datenschutz-Grundverordnung - EU-weites Gesetz zum Schutz personenbezogener Daten
nDSG	Neues Datenschutzgesetz der Schweiz - schweizer Pendant zur DSGVO
Salvatorische Klausel	Bestimmung, die sicherstellt, dass bei Unwirksamkeit einzelner AGB-Punkte der Rest gültig bleibt
Höhere Gewalt	Außergewöhnliche, unvorhersehbare Ereignisse (Naturkatastrophen, Pandemie, etc.)
Ausserordentliche Kündigung	Sofortige Kündigung bei schwerwiegenden Vertragsverletzungen
Verzugszinsen	Zusätzliche Kosten bei verspäteter Zahlung (hier: 5% p.a.)
Basiszinssatz	Referenzzinssatz der Europäischen Zentralbank
Cross-Platform	Software, die auf verschiedenen Betriebssystemen läuft (iOS, Android, Web)
Reverse Engineering	Rückentwicklung von Software zur Analyse des Codes
Scraping/Crawling	Automatisiertes Auslesen von Daten durch Programme
Synchronisation	Abgleich von Daten zwischen verschiedenen Systemen
Vertragspartner	Die Personen/Unternehmen, die den Vertrag abschließen
Nacherfüllung	Recht auf Reparatur/Ersatz bei Mängeln
Gerichtsstand	Ort, an dem bei Rechtsstreitigkeiten geklagt werden muss
Setlisten-Management	Funktion zur Erstellung und Verwaltung von Songlisten für Auftritte
Ressourcenplanung	Verwaltung von Equipment und Proberäumen

Rollenzuweisungen	Festlegung unterschiedlicher Berechtigungen für Bandmitglieder
Betroffenenrechte	Ihre Rechte bezüglich Ihrer gespeicherten Daten
Einwilligung	Ihre ausdrückliche Zustimmung zur Datenverarbeitung
Vertragserfüllung	Rechtsgrund für notwendige Datenverarbeitung zur Service-Bereitstellung
Minderjährige	Personen unter 18 Jahren
Textform	Schriftliche Erklärung ohne Unterschrift (E-Mail reicht)
Mittelbare Schäden	Indirekte Folgeschäden
Wesentliche Vertragspflichten	Kernleistungen, die den Vertragszweck prägen

Tabelle 2: Glossar

Text aus Claude

Tabellenverzeichnis

Tabelle 1: Änderungsverzeichnis.....	1
Tabelle 2: Glossar	11

Inhaltsverzeichnis

§ 1	Geltungsbereich und Vertragspartner.....	2
1.1	Geltungsbereich	2
1.2	Vertragspartner	2
1.3	Abweichende Bedingungen	2
§ 2	Leistungsbeschreibung.....	2
2.1	Service-Umfang.....	2
2.1.1	Kalendermanagement:	2
2.1.2	Aufgabenmanagement:.....	3
2.1.3	Ressourcenplanung:	3
2.1.4	Setlisten-Management:.....	3
2.1.5	Gruppen-/Bandverwaltung:.....	3
2.2	Technische Verfügbarkeit	3
2.3	Leistungsänderungen.....	3
§ 3	Vertragsschluss und Registrierung.....	4
3.1	Vertragsschluss	4
3.2	Registrierungsvoraussetzungen.....	4
3.3	Account-Sicherheit	4
§ 4	Nutzungsrechte und -pflichten	4
4.1	Nutzungsrecht	4
4.2	Zulässige Nutzung	4
4.3	Verbote Nutzungen	4
4.4	Inhalte des Nutzers	5
§ 5	Preise und Zahlungsbedingungen.....	5
5.1	Preismodell	5
5.2	Zahlungsmodalitäten.....	5
5.3	Zahlungsverzug	5
5.4	Preisänderungen.....	5
§ 6	Laufzeit und Kündigung	6
6.1	Vertragslaufzeit.....	6
6.2	Ordentliche Kündigung	6
6.3	Ausserordentliche Kündigung	6
6.4	Folgen der Kündigung	6
§ 7	Datenschutz	7

7.1	Datenverarbeitung.....	7
7.2	Rechtsgrundlagen.....	7
7.3	Betroffenenrechte	7
§ 8	Gewährleistung und Haftung.....	8
8.1	Gewährleistung	8
8.2	Haftungsbeschränkung.....	8
8.3	Datensicherung.....	8
§ 9	Höhere Gewalt.....	8
§ 10	Änderungen der AGB	9
10.1	Änderungsbefugnis	9
10.2	Zustimmungsverfahren	9
§ 11	Schlussbestimmungen.....	9
11.1	Anwendbares Recht.....	9
11.2	Gerichtsstand	9
11.3	Salvatorische Klausel	9
11.4	Vertragssprache.....	9

Key Partner	Key Activities	Value Propositiones	Customer Relationship	Customer Segments
Lokale Schweizer Bands & Musiker*innen	Weiterentwicklung der App	<i>Wir regeln die Organisation. Damit Musiker wieder Musik machen können.</i>	Direkter, persönlicher Support. Über eine Hotline und online Chat	Hobbybands & Semi-Profis (vor allem aus der Schweiz)
Influencer in der Musikszene	Kundensupport & Bugfixes Social Media Management & Updates Onboarding & Live-Demos	Einfache Bandorganisation (Kalender, Aufgaben, Setlisten etc.) Intuitiv & verständlich, auch für Nicht-Techies	Tutorials	Wenig technikaffin, möchten keine Zeit mit Organisation verschwenden Fokus auf Musik statt Tools
	Key Resources Entwickler (du selbst), Testnutzer (deine Band) Raspberry Pi 5 als Entwicklung Infrastruktur Frameworks: Python/Django (Backend), Flutter (Frontend)	Persönlicher Support für Vertrauen	Channels Persönlicher Kontakt bei Konzerten Social Media Live-Demos vor Ort Website mit Registrierung & Login	
Cost Structure Aktuell kaum Fixkosten (Raspberry Pi, Open Source) Später: VPS/Hosting-Kosten, Domain, evtl. Support-Aufwand Zeitaufwand für Support, Marketing und Entwicklung		Revenue Stream Freemium-Modell Basisfunktionen kostenlos, Kollaboration kostenpflichtig Preis: 20 CHF/USD pro Band pro Monat (ähnlich Netflix-Modell)		

Anhang G1

Testdurchführung

Delegator

Klassifizierung intern
Status in Arbeit
Programmname Delegator
Projektnummer 1
Projektleiter Tabinas Kenan
Version 0.1
Datum 26. März 2025
Auftraggeber Tabinas Kenan
Autor/Autoren Tabinas Kenan
Verteiler

Änderungsverzeichnis

Version	Datum	Änderung	Autor
0.1	03.06.2025	Usertest	TAK
1	06.06.2025	Pentest	TAK

Tabelle 1: Änderungsverzeichnis

1 Abnahme von Muss Kriterien

Test Bedingungen:

- Gerät: OnePlus EB2103.
- Testerin: Noëmi Tabinas, wusste nicht wie die App Funktioniert und erhielt nur das Erwartete Ergebnisse und musste die Tests Schritte absolvieren.
- Testvoraussetzungen: Diese wurde im Vorhinein sichergestellt.

1.1 Testfall

ID / Bezeichnung	T-01	User Funktionalität
Beschreibung	Erstellen eines Projektes	
Testvoraussetzung	Eingelogged, Organisation beigetreten	
Testschritte	<ul style="list-style-type: none"> • Auf neues Projekt drücken 	
Erwartetes Ergebnis	Es erscheint ein neues Projekt	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	03.06.2025
Tester	Noëmi Louise Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	Beim ersten Mal, habe ich nicht beachtet, dass es wir keine Org gegründet haben.

*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel

1.2 Testfall

ID / Bezeichnung	T-05	Aufgaben
Beschreibung	Aufgabe erstellt	
Testvoraussetzung	Eingelogged, Projekt ist erstellt	
Testschritte	<ul style="list-style-type: none"> • Klickt auf neue Aufgabe • Gibt geforderten Informationen ein 	
Erwartetes Ergebnis	Es ermöglicht dem User eine Aufgabe zu erstellen	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	03.06.2025
Tester	Noëmi Louise Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	Im Hauptmenu Verlinkung zum Task funktioniert nicht. Gibt overflow

*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel

1.3 Testfall

ID / Bezeichnung	T-06	Aufgaben
Beschreibung	Aufgabe einer anderen Person zu weisen	
Testvoraussetzung	Eingelogged, Projekt ist erstellt und es ist mehrere Personen in diesem Projekt	
Testschritte	<ul style="list-style-type: none"> • Aufgabe erstellen • Auf Zuweisung eine andere Person auswählen. • Speichern 	
Erwartetes Ergebnis	Es erscheint unter Aufgaben der Name der Ausgewählten Person bei der Aufgabe	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	03.06.2025
Tester	Noëmi Louise Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	keine

*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel

1.4 Testfall

ID / Bezeichnung	T-07	Aufgaben
Beschreibung	Aufgabe Terminieren	
Testvoraussetzung	Eingelogged, Projekt ist erstellt	
Testschritte	<ul style="list-style-type: none"> • Aufgabe erstellen • Bei Deadline Datum auswählen • Speichern. 	
Erwartetes Ergebnis	Es erscheint bei der Aufgabe ein Datum	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	03.06.2025
Tester	Noëmi Louise Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	keine

*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel

1.5 Testfall

ID / Bezeichnung	T-08	Aufgaben
Beschreibung	Kalender synchronisiert	
Testvoraussetzung	Eingelogged, Projekt ist erstellt und Kalender ist Synchronisiert	
Testschritte	<ul style="list-style-type: none"> • Aufgabe auf einen anderen Tag setzen. • Kalender App Synchronisieren 	
Erwartetes Ergebnis	Die Aufgabe erscheint an einem anderen Tag	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	03.06.2025
Tester	Noëmi Louise Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	Code ist nicht verfügbar aus der App heraus

*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel

1.6 Testfall

ID / Bezeichnung	T-10	Aufgaben
Beschreibung	Kalender zugriff	
Testvoraussetzung	Eingelogged	
Testschritte	<ul style="list-style-type: none"> • Man kann der Link von eigenen Kalender kopieren 	
Erwartetes Ergebnis	Link kann genutzt werden für die Kalender synchronisation	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	03.06.2025
Tester	Noëmi Louise Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	keine

*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel

1.7 Testfall

ID / Bezeichnung	T-11	Zugriff
Beschreibung	Projekt zugriff	
Testvoraussetzung	Eingelogged, User hat Zugriff auf existierende Projekte	
Testschritte	<ul style="list-style-type: none"> • Projekte Liste anzeigen 	
Erwartetes Ergebnis	Projekte werden angezeigt	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	03.06.2025
Tester	Noëmi Louise Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	keine

*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel

1.8 Testfall

ID / Bezeichnung	T-12	Zugriff
Beschreibung	Aufgaben anzeigen	
Testvoraussetzung	Eingelogged, User ist zugewiesen bei Projekten, Es existieren Aufgaben	
Testschritte	<ul style="list-style-type: none"> • Aufgaben liste anzeigen 	
Erwartetes Ergebnis	Aufgaben werden angezeigt	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	03.06.2025
Tester	Noëmi Louise Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	keine

*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel

1.9 Testfall

ID / Bezeichnung	T-25	Bug Report
Beschreibung	Jeder User ohne Anmeldung ein Bug Report erstellen	
Testvoraussetzung	-	
Testschritte	<ul style="list-style-type: none"> • Vor dem Login einen Bug Report ausfüllen • Titel, Inhalt und E-Mail eingeben • Absenden drücken 	
Erwartetes Ergebnis	Abstimmung verschwindet wieder.	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	03.06.2025
Tester	Noëmi Tabinas
Mängelklasse*	0
Mangelbeschreibung	
Bemerkungen	keine

*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel

1.10 Testfall

ID / Bezeichnung	T-26	Bug Report
Beschreibung	Nach dem Bug Report gibt es eine Meldung an den User	
Testvoraussetzung	-	
Testschritte	<ul style="list-style-type: none"> • Vor dem Login einen Bug Report ausfüllen • Titel, Inhalt und E-Mail eingeben • Absenden drücken 	
Erwartetes Ergebnis	User erhält E-Mail auf die angegebene E-Mail	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	
Tester	
Mängelklasse*	4
Mangelbeschreibung	Dieses Feature wurde leider nichts umgesetzt.
Bemerkungen	keine

*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel

2 Abnahme von Kann Kriterien

2.1 Testfall

ID / Bezeichnung	T-19	Chat
Beschreibung	Projekte sind Kommentierbar	
Testvoraussetzung	Eingelogged, User ist zugewiesen bei Projekten	
Testschritte	<ul style="list-style-type: none"> • Bei einem Projekt • Auf Chats drücken • Nachricht absenden 	
Erwartetes Ergebnis	Nachrichten stehen im Chat	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	03.06.2025
Tester	Noëmi Louise Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	keine

*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel

2.2 Testfall

ID / Bezeichnung	T-20	Chat
Beschreibung	Channels sind Kommentierbar	
Testvoraussetzung	Eingelogged, User ist zugewiesen bei Projekten	
Testschritte	<ul style="list-style-type: none"> • Bei Chats hin gehen • Einen Channel auswählen • Nachricht absenden 	
Erwartetes Ergebnis	Nachrichten stehen im Chat	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	03.06.2025
Tester	Noëmi Louise Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	keine

*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel

3 Penetrationstest

Um die Sicherheit der Applikation zu gewährleisten, gehen wir auf den Sicherheit Aspekt ein. Wie aus dem Sicherheitskonzept hervorgeht, sind Schwache Endpoints die grösste Schwachstelle.

3.1 Testfall

ID / Bezeichnung	PT001	Token
Beschreibung	Testen, ob geschützte Endpunkte ohne Token erreichbar sind	
Testvoraussetzung	Keine Anmeldung	
Testschritte	GET /projects/ ohne Authorization Header aufrufen GET /calendars/ ohne Token versuchen POST /messages/ ohne Authentifizierung testen	
Erwartetes Ergebnis	401 Unauthorized für alle geschützten Endpunkte	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	06.06.2025
Tester	Kenan Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	keine

*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel

3.2 Testfall

ID / Bezeichnung	PT002	Token
Beschreibung	Versuchen, JWT Tokens zu manipulieren oder zu fälschen	
Testvoraussetzung	Keine Anmeldung	
Testschritte	JWT Token dekodieren und Payload ändern (z.B. user_id) Token mit ungültiger Signatur verwenden Abgelaufenes Token testen Token mit geändertem Header verwenden	
Erwartetes Ergebnis	Alle manipulierten Tokens werden abgelehnt	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	06.06.2025
Tester	Kenan Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	keine

*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel

3.3 Testfall

ID / Bezeichnung	PT003	Token
Beschreibung	Testen, ob User auf fremde Daten zugreifen können	
Testvoraussetzung	Zwei verschiedene User-Accounts	
Testschritte	Mit User A einloggen Versuchen, Projekte von User B abzurufen (GET /projects/{userproject_id}/)	
Erwartetes Ergebnis	403 Forbidden bei Zugriff auf fremde Daten	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	06.06.2025
Tester	Kenan Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	Es kommt 404 not Found. Was mir sogar besser gefällt als 405.

*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel

3.4 Testfall

ID / Bezeichnung	PT004	Premium
Beschreibung	Testen, ob normale User Admin-Funktionen ausführen können	
Testvoraussetzung	Zwei verschiedene User-Accounts	
Testschritte	Versuchen, neue Organisation zu erstellen ohne Premium Versuchen, andere User aus Organisation zu entfernen Versuchen, auf /history/ zuzugreifen ohne Staff-Rechte Songs erstellen/bearbeiten ohne entsprechende Rolle	
Erwartetes Ergebnis	403 Forbidden bei Zugriff auf fremde Daten	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	06.06.2025
Tester	Kenan Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	

*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel

3.5 Testfall

ID / Bezeichnung	PT005	SQL Injection
Beschreibung	SQL Injection in verschiedenen Endpunkten testen	
Testvoraussetzung	Gültiger API-Zugang	
Testschritte	SQL Injection in Suchfeldern: GET /songs/?search='; DROP TABLE-- SQL Injection in Filtern: GET /users/?username=admin' OR '1'='1 SQL Injection in POST/PUT Requests in JSON-Feldern SQL Injection in URL-Parametern	
Erwartetes Ergebnis	Keine SQL-Statements werden ausgeführt, Eingaben werden escaped	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	06.06.2025
Tester	Kenan Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	

*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel

3.6 Testfall

ID / Bezeichnung	PT006	Grenzwerte und Datentypen
Beschreibung	SQL Injection in verschiedenen Endpunkten testen	
Testvoraussetzung	Gültiger API-Zugang	
Testschritte	Extrem lange Strings in Text-Feldern (>10000 Zeichen) Negative Zahlen wo positive erwartet werden Float-Werte in Integer-Feldern Leere/null Werte in Required-Feldern Unicode/Emoji in verschiedenen Feldern	
Erwartetes Ergebnis	Angemessene Validierungsfehler, keine 500-Errors	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	06.06.2025
Tester	Kenan Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	Username max 150 Chars Floating Point numbers werden Ignoriert Json Parse Error bei Emojis

*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel

3.7 Testfall

ID / Bezeichnung	PT007	iCal
Beschreibung	Sicherheit der iCal-Token testen	
Testvoraussetzung	Zugriff auf Calendar Subscription	
Testschritte	Versuchen, fremde Calendar-Tokens zu erraten/bruteforcen Token-Enumeration: /ical/calendar/AAAA.../, /ical/calendar/AAAB.../ Prüfen, ob Token in Logs oder URLs geleakt werden Testen, ob revoked Token noch funktionieren	
Erwartetes Ergebnis	Angemessene Validierungsfehler, keine 500-Errors	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	06.06.2025
Tester	Kenan Tabinas
Mängelklasse*	2
Mangelbeschreibung	Nicht Ideal. Da das Token in der URL auftaucht. Eventuell wäre Token via Body besser. Aber das liegt eventuell an der Technologie. Es gäbe noch eine Art login.
Bemerkungen	keine

*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel

3.8 Testfall

ID / Bezeichnung	PT008	Organisation Invitation Security
Beschreibung	Sicherheit des Einladungssystems testen	
Testvoraussetzung	Organisationsadmin-Rechte	
Testschritte	Versuchen, Einladungen für andere Invite-Codes zu erstellen Mehrfache Verwendung von Einladungs-Tokens Einladungen für bereits existierende Mitglieder Abgelaufene Einladungen verwenden	
Erwartetes Ergebnis	Robuste Validierung aller Einladungsschritte	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	06.06.2025
Tester	Kenan Tabinas
Mängelklasse*	0
Mangelbeschreibung	
Bemerkungen	

*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel

3.9 Testfall

ID / Bezeichnung	PT009	Project Access Control
Beschreibung	Zugriffskontrolle bei Projekten testen	
Testvoraussetzung	Mehrere User mit verschiedenen Projekt-Zugehörigkeiten	
Testschritte	User zu Projekt hinzufügen ohne Berechtigung Auf Projekt-Chat zugreifen ohne Projekt-Mitgliedschaft Tasks in fremden Projekten erstellen/bearbeiten Project-Level vs Organization-Level Berechtigungen testen	
Erwartetes Ergebnis	Strikte Trennung der Projekt-Zugriffe	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	06.06.2025
Tester	Kenan Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	

*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel

3.10 Testfall

ID / Bezeichnung	PT010	HTTP Method Testing
Beschreibung	Unerlaubte HTTP-Methoden testen	
Testvoraussetzung	Gültiger API-Zugang	
Testschritte	POST bei /chat/ DELETE an ReadOnly-Endpunkten /history/ Method Override Headers testen	
Erwartetes Ergebnis	Nur erlaubte HTTP-Methoden werden akzeptiert	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	06.06.2025
Tester	Kenan Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	

*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel

3.11 Testfall

ID / Bezeichnung	PT011	Content-Type Validation
Beschreibung	Verschiedene Content-Types und Payload-Formate testen	
Testvoraussetzung	Gültiger API-Zugang	
Testschritte	XML-Payload an JSON-Endpunkt senden Malformed JSON senden Verschiedene Charset-Encodings testen Content-Type spoofing versuchen	
Erwartetes Ergebnis	Robuste Content-Type Validierung	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	06.06.2025
Tester	Kenan Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	JSON Parse error

*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel

3.12 Testfall

ID / Bezeichnung	PT012	Error Message Information Leakage
Beschreibung	Testen, ob Fehlermeldungen sensitive Informationen preisgeben	
Testvoraussetzung	Verschiedene fehlerhafte Requests	
Testschritte	Nicht-existierende Endpunkte aufrufen Falsche Parameter-Typen senden Server-Errors provozieren Database-Errors auslösen	
Erwartetes Ergebnis	Generische Fehlermeldungen ohne technische Details	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	06.06.2025
Tester	Kenan Tabinas
Mängelklasse*	4
Mangelbeschreibung	Es ist noch Debug Level eingestellt.
Bemerkungen	

*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel

3.13 Testfall

ID / Bezeichnung	PT013	Session & Token Management
Beschreibung	Token-Lebensdauer und Session-Management testen	
Testvoraussetzung	Gültiger API-Zugang	
Testschritte	Token-Refresh-Mechanismus testen Token-Expiry Handling prüfen Concurrent Sessions testen Token-Invalidierung nach Logout prüfen	
Erwartetes Ergebnis	Sichere Token-Verwaltung	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	06.06.2025
Tester	Kenan Tabinas
Mängelklasse*	0
Mangelbeschreibung	
Bemerkungen	

*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel

Pentest Fälle sind von Claude.

4 Integrationtest

4.1 Projects

4.1.1 Project Get

```

⌚ Attempting login for user: test_user_2
📝 POST request to: https://delegator.ch/token/
📝 Request data: {"username":"test_user_2","password":"sml12345"}
📦 Response status: 200
📝 Received login response: (refresh, access)
✅ Tokens received successfully
👤 Extracted user data from token: test_user_2
📝 GET request to: https://delegator.ch/projects/
📦 Response status: 200
✅ Got 7 projects
    📁 Project ID: 20, Org ID: 5
    📁 Project ID: 15, Org ID: 5
    📁 Project ID: 14, Org ID: 5
    📁 Project ID: 16, Org ID: 5
    📁 Project ID: 13, Org ID: 5
    📁 Project ID: 21, Org ID: 5
    📁 Project ID: 17, Org ID: 5

```

4.1.2 Project Delete & Get by Id



The screenshot shows a REST client interface with the following details:

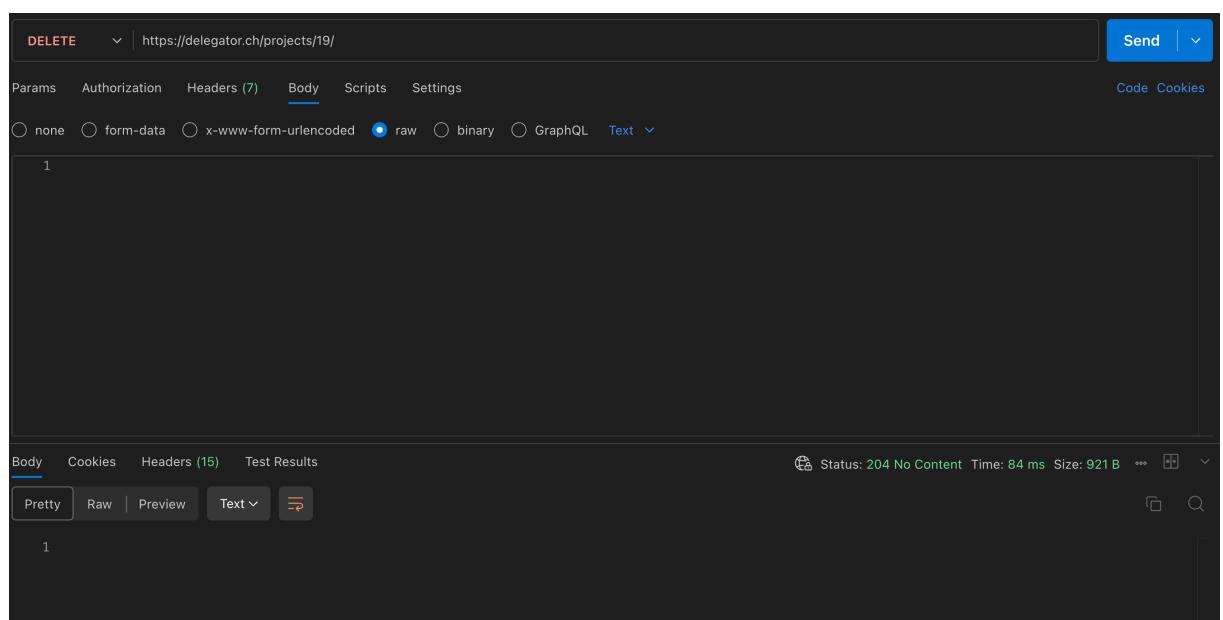
- Headers:** Headers (18) - Status: 200 OK, Time: 83 ms, Size: 1.06 KB
- Body:** JSON (Pretty) - Shows a single project object with ID 21.
- Content:**

```

1   {
2     "id": 21,
3     "event": null,
4     "deadline": null,
5     "priority": 1,
6     "event_details": null,
7     "tasks": []
8   }

```

4.1.3 Project Delete



The screenshot shows a REST client interface with the following details:

- Method:** DELETE - URL: https://delegator.ch/projects/19/
- Headers:** Headers (7) - Status: 204 No Content, Time: 84 ms, Size: 921 B
- Body:** Text (None selected)
- Content:** Shows a single digit '1'.

```

⌚ Attempting login for user: test_user_2
📝 POST request to: https://delegator.ch/token/
📝 Request data: {"username":"test_user_2","password":"sml12345"}
📦 Response status: 200
🕒 Received login response: (refresh, access)
✅ Tokens received successfully
👤 Extracted user data from token: test_user_2
📅 Creating new project for organisation 5
📝 POST request to: https://delegator.ch/projects/
📝 Request data: {"priority":1,"event_details":null,"organisation":5}
📦 Response status: 201
✅ Project created with ID: 23
🗑️ Deleting project with ID: 23
📝 DELETE request to: https://delegator.ch/projects/23/
📦 Response status: 204
✅ Project deleted successfully
📝 GET request to: https://delegator.ch/projects/23/
📦 Response status: 404
✖️ API Error: {detail: Not found.}
✖️ API Error (raw): {"detail":"Not found."}
✅ Project no longer exists (expected error): ApiException: [404] {"detail":"Not found."}

```

4.2 Events

4.3 Get Events

Body Cookies Headers (18) Test Results

Pretty | Raw | Preview | JSON | 

```

1   [
2     "count": 3,
3     "next": null,
4     "previous": null,
5     "results": [
6       {
7         "id": 3,
8         "calendar": 2,
9         "start": "2025-05-17T12:00:00Z",
10        "end": "2025-05-17T13:00:00Z",
11        "is_gig": false,
12        "calendar_details": {
13          "id": 2,
14          "organisation": 5,
15          "organisation_details": {
16            "id": 5,
17            "name": "Test",
18            "since": "2025-05-03T17:12:54.6502592"
19          }
20        },
21      },
22    ],

```

PROBLEMS 87 OUTPUT DEBUG CONSOLE TERMINAL TEST RESULTS PORTS POSTMAN CONSOLE

⌚ Attempting login for user: test_user_2
📝 POST https://delegator.ch/token/
📝 Body: {"username":"test_user_2","password":"sml12345"}
📦 Response: 200
🕒 Received login response: (refresh, access)
✅ Tokens received successfully
👤 Extracted user data from token: test_user_2
📝 GET https://delegator.ch/calendars/
📦 Response: 200
🕒 Using calendar with ID: 2
📝 GET https://delegator.ch/events/
📦 Response: 200
✅ Got 10 events.
🕒 Event ID: 3, Calendar ID: 2, Start: 2025-05-17 12:00:00.000Z
🕒 Event ID: 4, Calendar ID: 2, Start: 2025-05-18 12:00:00.000Z
🕒 Event ID: 5, Calendar ID: 2, Start: 2025-05-18 12:00:00.000Z
🕒 Event ID: 6, Calendar ID: 2, Start: 2025-05-18 16:55:15.155300Z
🕒 Event ID: 7, Calendar ID: 2, Start: 2025-05-18 17:03:17.215372Z
🕒 Event ID: 8, Calendar ID: 2, Start: 2025-05-18 17:03:17.696761Z
🕒 Event ID: 9, Calendar ID: 2, Start: 2025-05-18 17:03:17.215372Z
🕒 Event ID: 10, Calendar ID: 2, Start: 2025-05-18 17:07:00.821416Z

Dart

- ⌚ getAll should return list of events from real backend EventService Int...
- ⌚ create and delete event EventService Integration Tests < integration_te...
- ⌚ update event EventService Integration Tests < integration_test/services/...
- ⌚ getByCalendarId should return events for a specific calendar Event...
- ⌚ getGigs should return only gig events EventService Integration Tests ...
- ⌚ (tearDownAll) integration_test/services/event_service_test.dart

> 32 older results

4.4 Create and Delete

Body Cookies Headers (17) Test Results

Pretty Raw Preview JSON ↻

```

3   "calendar": 2,
4   "start": "2025-05-18T17:03:17.215372Z",
5   "end": "2025-05-18T19:03:17.215372Z",
6   "is_gig": false,
7   "calendar_details": {
8     "id": 2,
9     "organisation": 5,
10    "organisation_details": {
11      "id": 5,
12      "name": "Test",
13      "since": "2025-05-03T17:12:54.650259Z"
14    }
15  }
16

```

>Status: 201 Created Time: 111 ms Size: 1.16 KB ⚡ 🔍 Go Live 🛡

Body Cookies Headers (16) Test Results

Pretty Raw Preview Text ↻

```
1
```

>Status: 204 No Content Time: 166 ms Size: 938 B ⚡ 🔍 Go Live 🛡

PROBLEMS 87 OUTPUT DEBUG CONSOLE TERMINAL TEST RESULTS PORTS POSTMAN CONSOLE

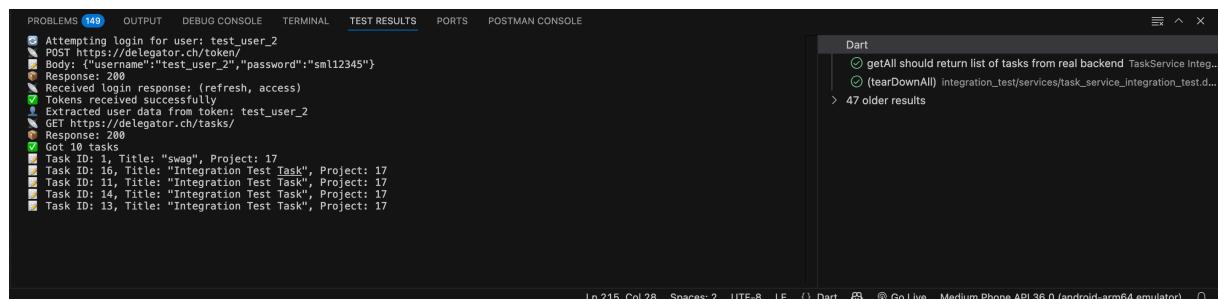
Dart

- Attempting login for user: test_user_2
- POST https://delegator.ch/token/
- Body: {"username":"test_user_2","password":"sm112345"}
- Response: 200
- Received login response: (refresh, access)
- Tokens received successfully
- Extracted user data from token: test_user_2
- GET https://delegator.ch/calendars/
- Response: 200
- Using calendar with ID: 2
- Creating new event for calendar 2
- POST https://delegator.ch/events/
- Body: {"title":"Test","calendar":2,"start":"2025-05-18T17:17:13.319863","end":"2025-05-18T19:17:13.319863","is_gig":false}
- Response: 201
- Event created with ID: 15
- Expected 2025-05-18 17:17:13.319863 and received 2025-05-18 17:17:13.319863Z
- Deleting event with ID: 15
- DELETE https://delegator.ch/events/15/
- Response: 204
- Event deleted successfully
- GET https://delegator.ch/events/15/

Ln 64, Col 18 Spaces: 2 UTF-8 LF ⌂ Dart 🔍 Go Live Medium Phone API 36.0 (android-arm64 emulator) 🔍

4.5 Tasks

4.5.1 Get



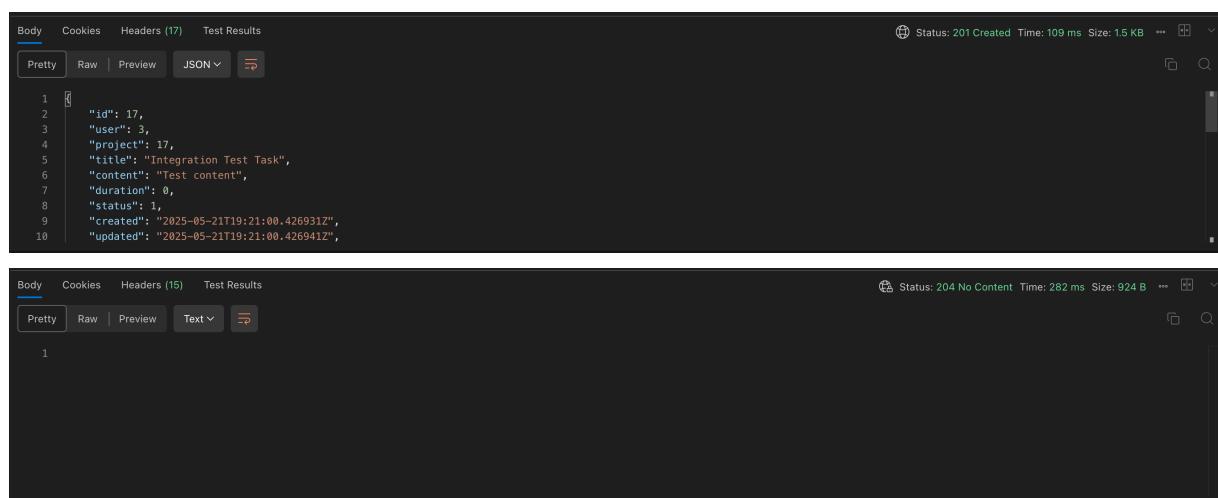
```

PROBLEMS 149 OUTPUT DEBUG CONSOLE TERMINAL TEST RESULTS PORTS POSTMAN CONSOLE
Attempting login for user: test_user_2
POST https://delegator.ch/token/
Body: {"username":"test_user_2","password":"sm112345"}
Response: 200
Received login response: (refresh, access)
Tokens received successfully
Extracted user data from token: test_user_2
GET https://delegator.ch/tasks/
Response: 200
got 10 tasks
Task ID: 1, Title: "wqag", Project: 17
Task ID: 16, Title: "Integration Test Task", Project: 17
Task ID: 11, Title: "Integration Test Task", Project: 17
Task ID: 14, Title: "Integration Test Task", Project: 17
Task ID: 13, Title: "Integration Test Task", Project: 17

```

Ln 215, Col 28 Spaces: 2 UTF-8 LF Dart Go Live Medium Phone API 36.0 (android-arm64 emulator)

4.5.2 Create and Delete



Body Cookies Headers (15) Test Results

Pretty Raw Preview JSON ↗

```

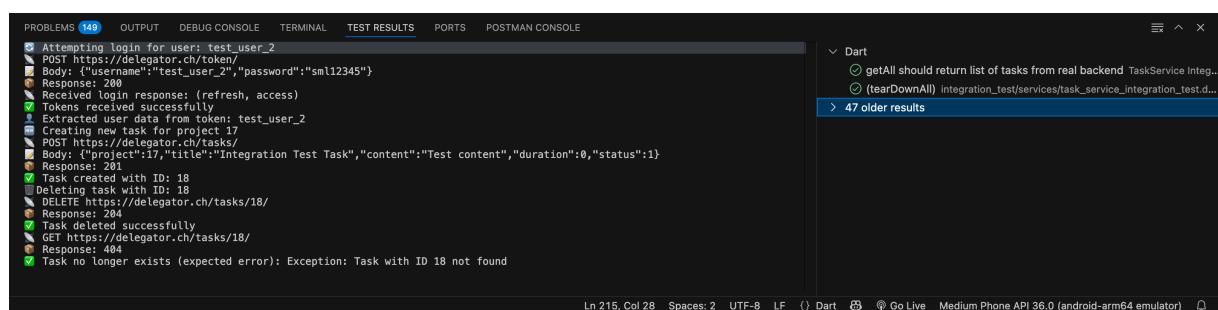
1
2   "id": 17,
3   "user": 3,
4   "project": 17,
5   "title": "Integration Test Task",
6   "content": "Test content",
7   "duration": 0,
8   "status": 1,
9   "created": "2025-05-21T19:21:00.426931Z",
10  "updated": "2025-05-21T19:21:00.426941Z",

```

Body Cookies Headers (15) Test Results

Pretty Raw Preview Text ↗

1



```

PROBLEMS 149 OUTPUT DEBUG CONSOLE TERMINAL TEST RESULTS PORTS POSTMAN CONSOLE
Attempting login for user: test_user_2
POST https://delegator.ch/token/
Body: {"username":"test_user_2","password":"sm112345"}
Response: 200
Received login response: (refresh, access)
Tokens received successfully
Extracted user data from token: test_user_2
Creating task for project 17
POST https://delegator.ch/tasks/
Body: {"project":17,"title":"Integration Test Task","content":"Test content","duration":0,"status":1}
Response: 201
Task created with ID: 18
Deleting task with ID: 18
DELETE https://delegator.ch/tasks/18/
Response: 204
Task deleted successfully
GET https://delegator.ch/tasks/18/
Response: 404
Task no longer exists (expected error): Exception: Task with ID 18 not found

```

Ln 215, Col 28 Spaces: 2 UTF-8 LF Dart Go Live Medium Phone API 36.0 (android-arm64 emulator)

4.6 Messages

4.6.1 Get

Body Cookies Headers (18) Test Results

Pretty Raw | Preview JSON ↻

```
1 {  
2     "count": 41,  
3     "next": "http://delegator.ch/messages/?page=2",  
4     "previous": null,  
5     "results": [  
6         {  
7             "id": 17,  
8             "user": 3,  
9             "chat": 8,  
10            "content": "Test",  
11            "sent": "2025-05-09T12:26:21.005145Z",  
12            "edited": null,  
13            "user_details": {  
14                "id": 3,  
15                "username": "test_user_2",  
16                "email": "",  
17                "first_name": "",  
18                "last_name": "",  
19                "created": "2025-05-02T13:30:38.372837Z"  
20            }  
21        }  
22    ]  
23}
```

The screenshot shows the VS Code interface with the Test Results tab selected. A test named "Attempting login for user: test_user_2" has passed. The test code is as follows:

```
Attempting login for user: test_user_2
POST https://delegator.ch/token/
  Body: {"username": "test_user_2", "password": "sml12345"}
  Response: 200
  Received login response: (refresh, access)
  ✓ Tokens received successfully
  Extracted user data from token: test_user_2
    GET https://delegator.ch/chats/
    Response: 200
    ✓ User data with ID: 8
    GET https://delegator.ch/messages/
    Response: 200
    ✓ Got 10 messages
      Message ID: 17, Chat ID: 8, Content: "Test"
      Message ID: 19, Chat ID: 8, Content: "Das war ein test"
      Message ID: 20, Chat ID: 8, Content: "hallo world"
      Message ID: 28, Chat ID: 8, Content: "Hallo Mxim"
      Message ID: 36, Chat ID: 8, Content: "Asdasd"
```

The status bar at the bottom indicates "Ln 246, Col 1" and "Spaces: 2".

```
PROBLEMS 219 OUTPUT DEBUG CONSOLE TERMINAL TEST RESULTS PORTS POSTMAN CONSOLE

Attempting login for user: test_user_2
POST https://delegator.ch/token/
Body: {"username":"test_user_2","password":"sm112345"}
Response: 200
Received login response: (refresh, access)
Tokens received successfully
Extracted user data from token: test_user_2
GET https://delegator.ch/chats/
Response: 200
Using chat with ID: 8
GET https://delegator.ch/messages/?chat=8
Response: 200
Got 5 messages for chat 8

Dart
create and delete message Attempting login for user: test_user_2
attempting message Attempting login for user: test_user_2
search should find messages with matching content Attempting login ...
sendMessage convenience method should work Attempting login ...
getAll should return list of messages from real backend MessageService
getByChatId should return messages for a specific ... 📁 🔍 ➔ ➕
(tearDownAll) integration_test/services/message_service_integration...
> 49 older results
```

4.6.2 Delete

The screenshot shows the Postman interface with the following details:

- PROBLEMS**: 219
- OUTPUT**
- DEBUG CONSOLE**
- TERMINAL**
- TEST RESULTS**
- PORTS**
- POSTMAN CONSOLE**

Test Results:

- Body: {"username":"test_user_2","password":"sm112345"}
 - Response: 200
 - Received login response: (refresh, access)
- Token received successfully
 - Extracted user data from token: test_user_2
 - GET https://delegator.ch/chats/
 - Response: 200
 - Using chat with ID: 8
- Creating new message for chat 8
 - POST https://delegator.ch/messages/
 - Data: {"chat":8,"content":"Integration Test Message"}
 - Response: 201
 - Message created with ID: 43
 - Deleting message with ID: 43
 - DELETE https://delegator.ch/messages/43/
 - Response: 204
 - Message deleted successfully
 - GET https://delegator.ch/messages/43/
 - Response: 404
 - Message no longer exists (expected error): Exception: Message with ID 43 not found

Dart

- create and delete message MessageService Integration Tests < integrat...
- (tearDownAll) integration_test/services/message_service_integrat...

> 50 older results

DELETE | https://delegator.ch/messages/39/ | Send | ↴

Params Authorization Headers (7) Body Scripts Settings | Code Cookies

Query Params

Key	Value
Key	Value

Body Cookies Headers (15) Test Results | Status: 204 No Content Time: 244 ms Size: 930 B | ↴ Go Live

Pretty Raw Preview Text | ↴

1

PROBLEMS (219) OUTPUT DEBUG CONSOLE TERMINAL TEST RESULTS PORTS POSTMAN CONSOLE | 🔍 ⌂ Go Live

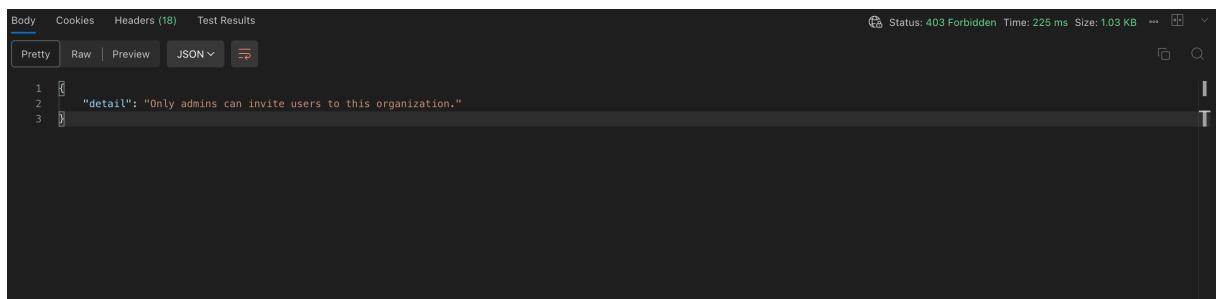
Attempting login for user: test_user_2
 POST https://delegator.ch/token/
 Body: {"username":"test_user_2","password":"sml12345"}
 Response: 200
 Received token response: (refresh, access)
 Token received successfully
 Extracted user data from token: test_user_2
 GET https://delegator.ch/chats/
 Response: 200
 Using chat with ID: 8
 Sending message to chat 8
 POST https://delegator.ch/messages/
 Body: {"id":3,"chat":8,"content":"Sent with convenience method"}
 Response: 201
 Message sent with ID: 44
 Cleaning up - deleting test message
 DELETE https://delegator.ch/messages/44/
 Response: 204

Dart | sendMessage convenience method should work | MessageService Int...
 (tearDownAll) integration_test/services/message_service_integration_t...
 > 51 older results

Ln 246, Col 1 Spaces: 2 UTF-8 LF (Dart 🔍 Go Live Medium Phone API 36.0 (android-arm64 emulator) 🔍

4.7 Invitation

4.7.1 Org im not a part of



A screenshot of a browser developer tools Network tab. The tab shows a single request with the following details:

- Status: 403 Forbidden
- Time: 225 ms
- Size: 1.03 KB

The response body is displayed in JSON format:

```
1 [{"detail": "Only admins can invite users to this organization."}]
```

Abkürzungen und Glossar

Abkürzung / Fachwort	Erläuterung
HERMES	Vorgehensmethodik für Projekte und Programme HERMES 5 ist ein eCH Standard

Tabelle 2: Abkürzungen und Glossar

Inhaltsverzeichnis

1 Testinterface	Error! Bookmark not defined.
2 Testen und so.....	Error! Bookmark not defined.
2.1 Projects.....	27
2.1.1 Project Get.....	27
2.1.2 Project Delete & Get By Id.....	27
2.1.3 Project Delete.....	27
2.2 Events	28
2.3 Get Events	28
2.4 Create and Delete.....	29
2.5 Tasks.....	30
2.5.1 Get.....	30
2.5.2 Create and Delete.....	30
2.6 Messages.....	31
2.6.1 Get.....	31
2.6.2 Delete.....	31
2.7 Invitation.....	33
2.7.1 Org im not a part of	33

Abbildungsverzeichnis

Es konnten keine Einträge für ein Abbildungsverzeichnis gefunden werden.

Tabellenverzeichnis

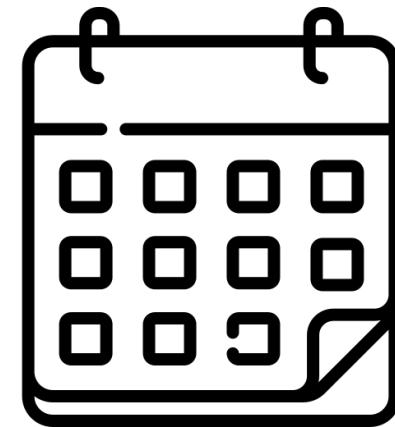
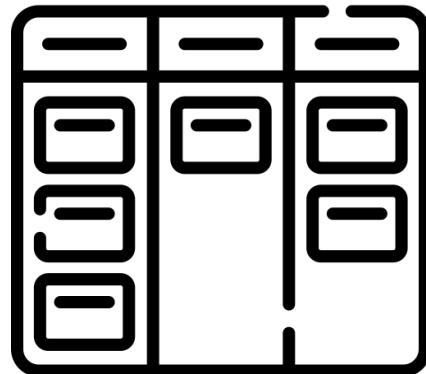
Tabelle 1: Änderungsverzeichnis.....	1
Tabelle 4: Abkürzungen und Glossar	34

Anhang G2



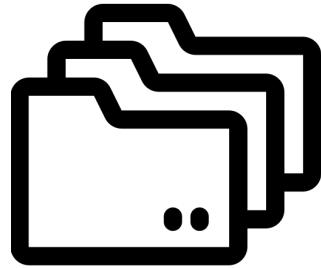
Delegator

Etwas was nicht mal Teams kann

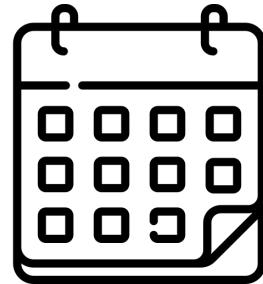


Das Problem

Delegator



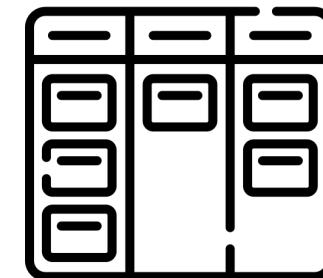
G-Drive



TimeTree



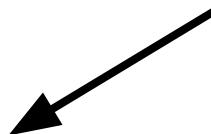
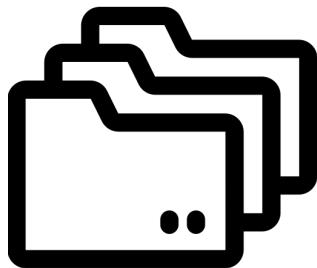
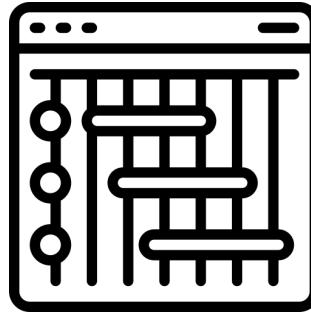
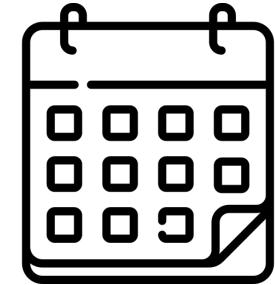
Signal



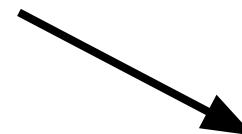
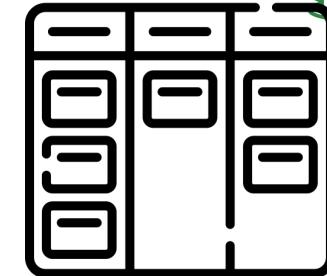
ClickUp /
Trello

IST

Delegator

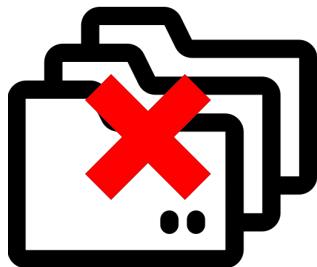
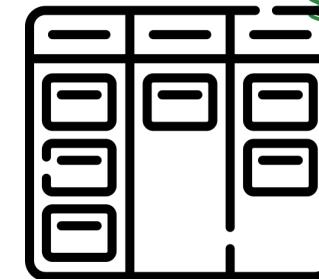
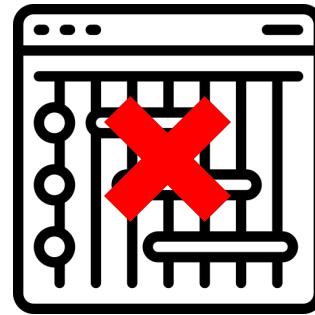
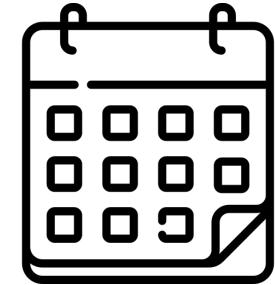


Projekt-Based

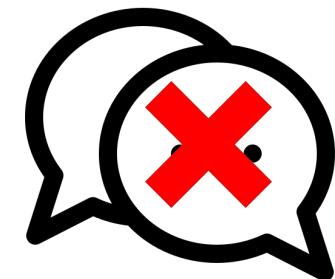
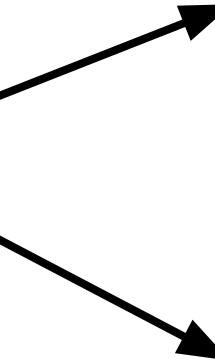


SOLL

Delegator



Projekt-Based



SDA

Delegator



Ziele

Man kann Ressourcen pro Woche pro User festlegen

Es können einzelne Ressourcen hinzugefügt oder entfernt werden

Alle Tasks können geschätzt, terminiert und zugeteilt werden

Alle Tasks werden auto. im Kalender synchronisiert und dargestellt

Die Kalender müssen Importierbar in andere Apps sein



Grenzen

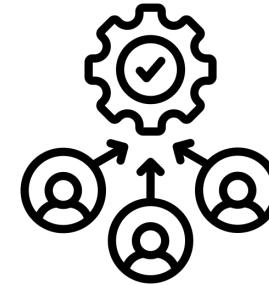
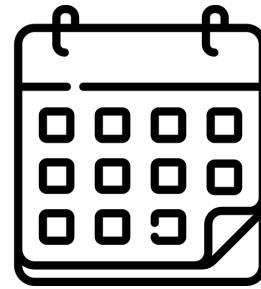
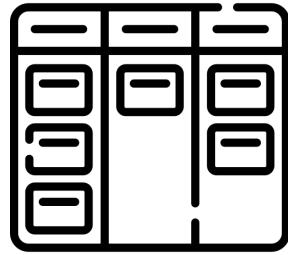
Verfügbarkeit: Web-Based only

Schnittstellen: Andere Kalender Apps

User: Zugriff auf Kanban, Tasks, und Kalender

Nicht Teil: Chat, Files, Apps, User-Management, Roadmap, Gantt

Integration: Ja



Lieferobjekte

Delegator

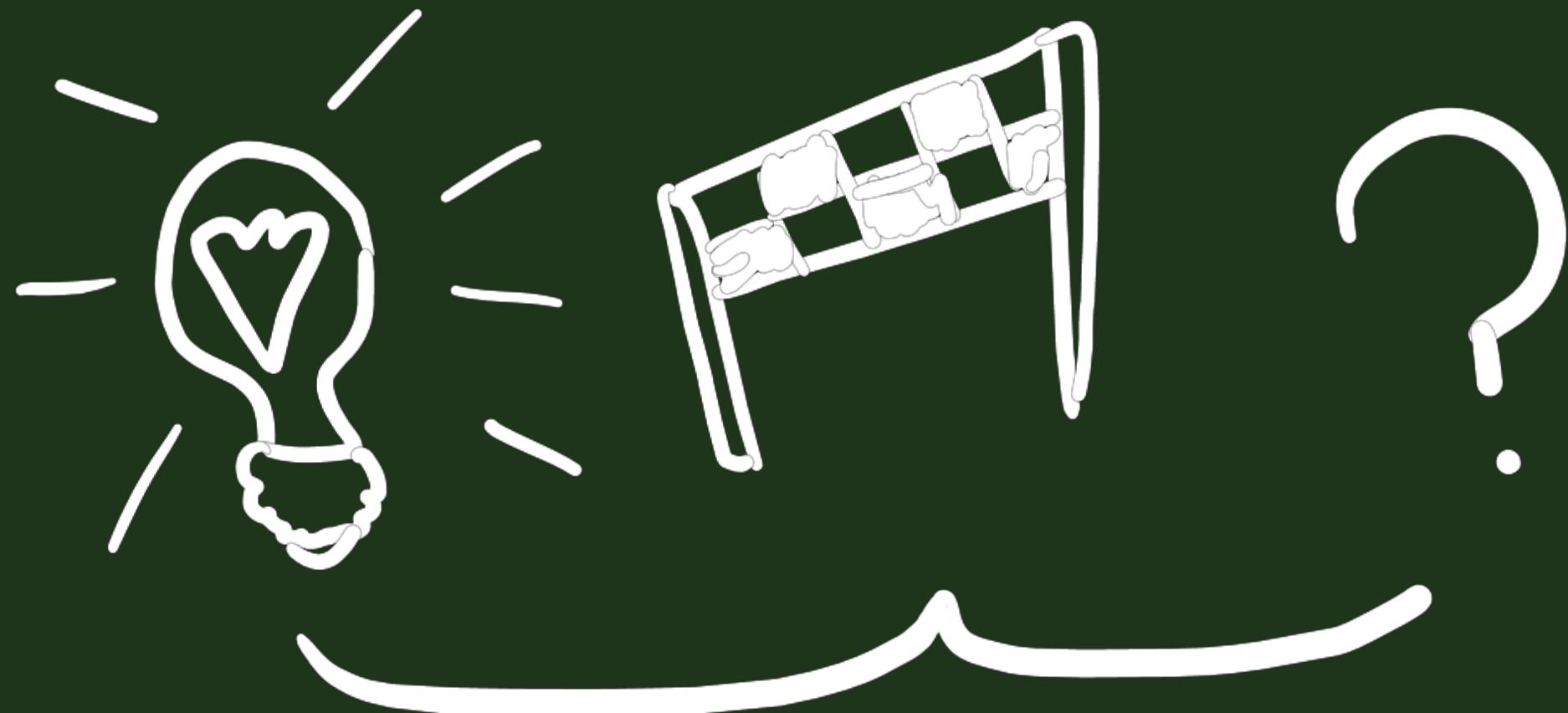


Fragen?

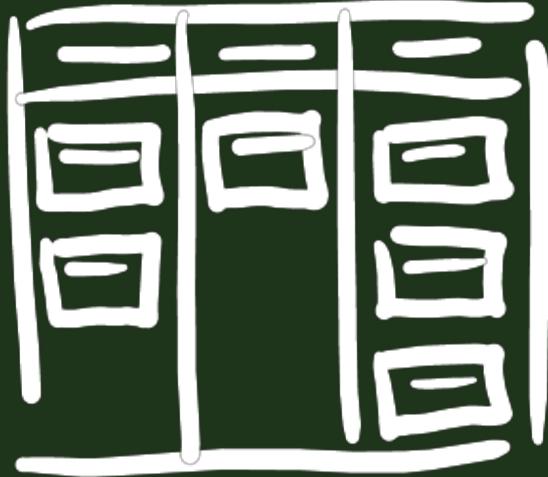
Anhang G3

delegator





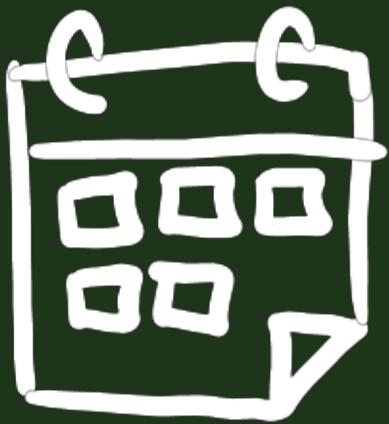
projektablauf



ausgangslage



g-drive



time-tree



g-drive



trello

ist



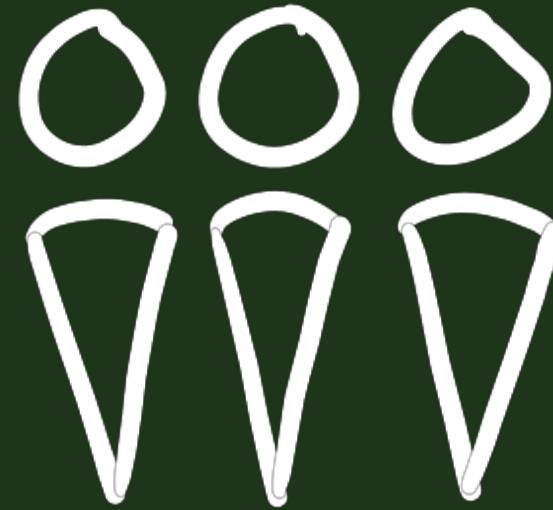
projektbasiert



projektbasiert



Dev



Test

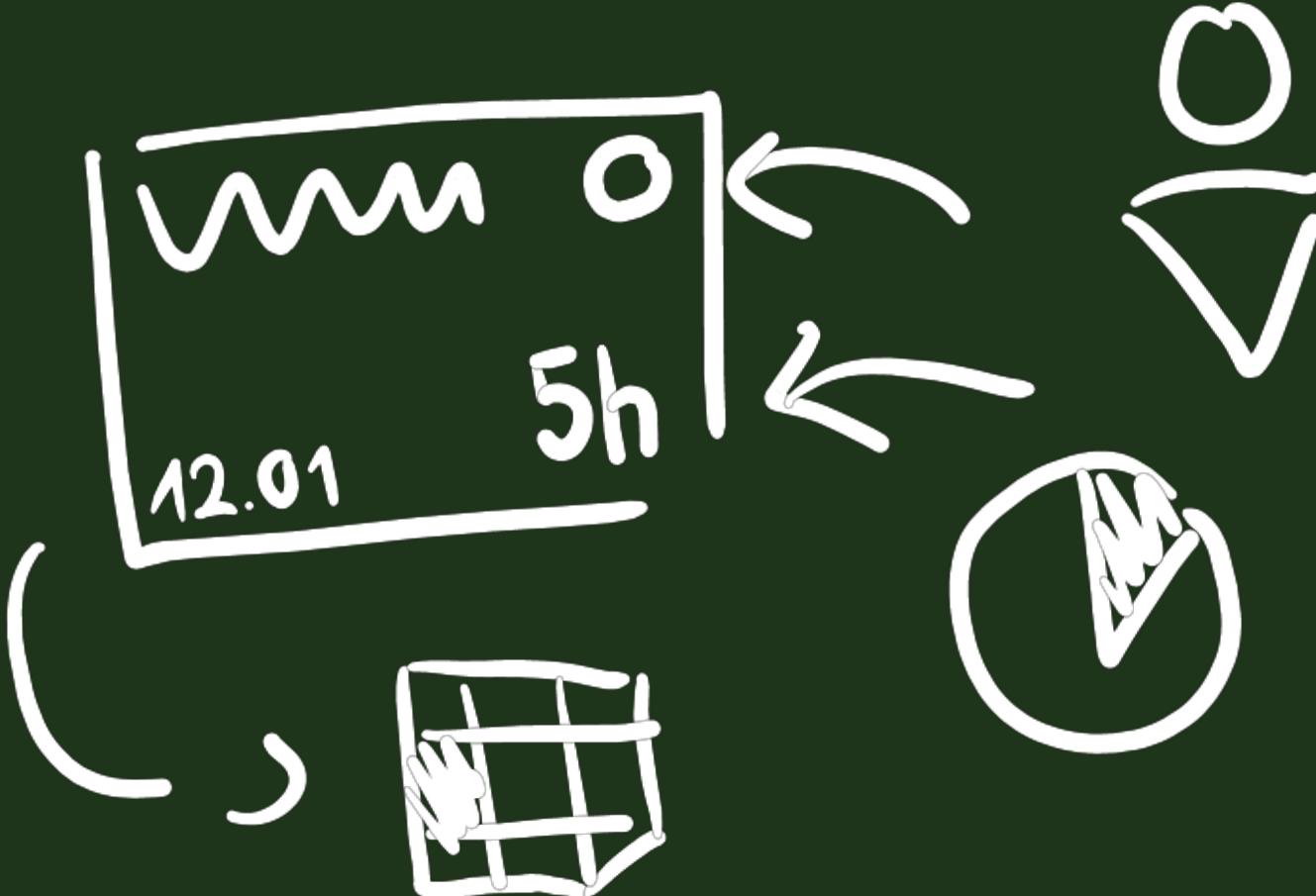
projektorganisation

Man kann Ressourcen pro Woche pro User festlegen



projektziele

Alle Tasks können geschätzt, terminiert und zugeteilt werden



projektziele

Es können einzelne Ressourcen hinzugefügt oder entfernt werden

KW03



KW04



KW05



projektziele

Die Kalender müssen Importierbar in andere Apps sein

Microsoft



Google



Apple



projektziele

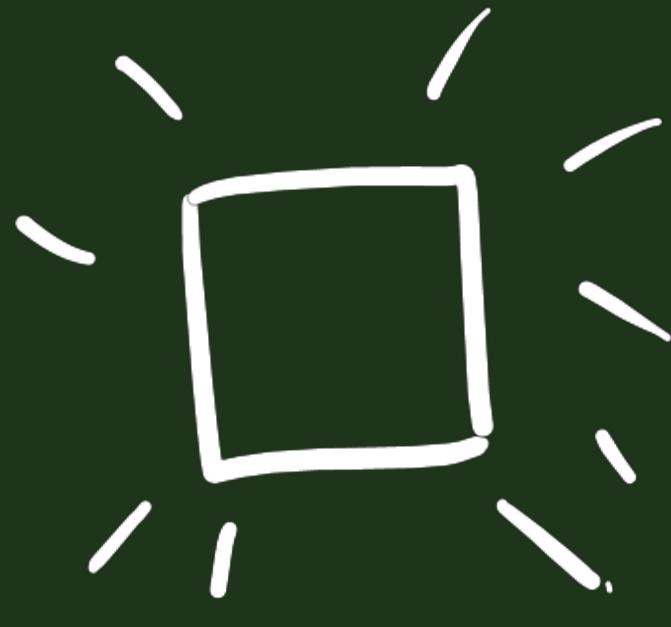
Alle Tasks werden auto. im Kalender synchronisiert und dargestellt



projektziele

Man kann Ressourcen pro Woche pro User festlegen
Es können einzelne Ressourcen hinzugefügt oder entfernt werden
Alle Tasks können geschätzt, terminiert und zugeteilt werden
Alle Tasks werden auto. im Kalender synchronisiert und dargestellt
Die Kalender müssen Importierbar in andere Apps sein

projektziele



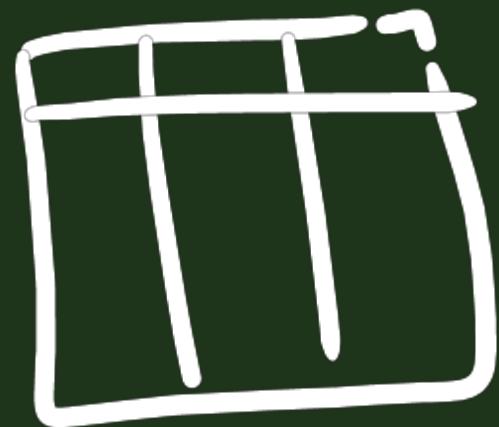
UX



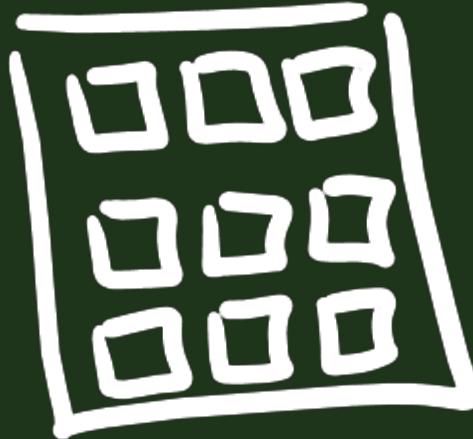
focus



abgrenzungen



Kanban

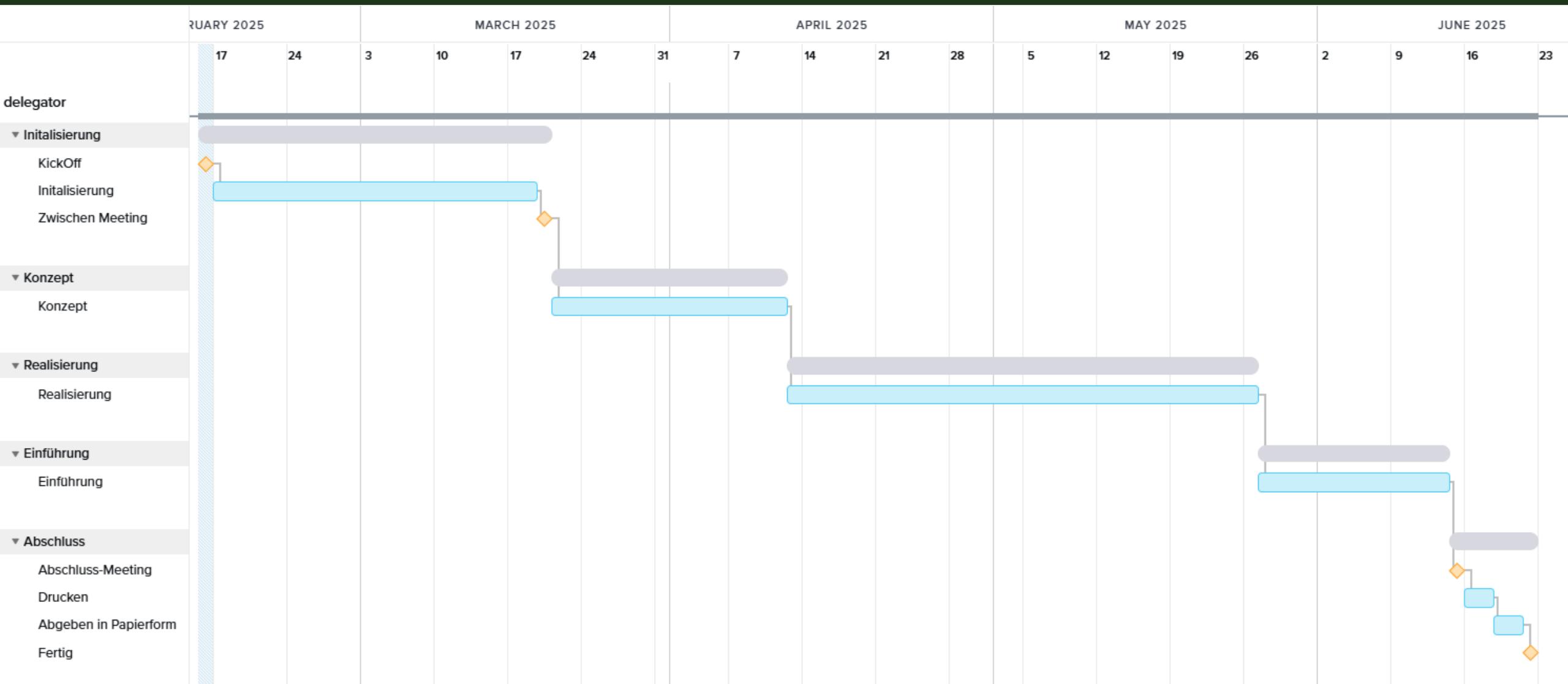


Kalender

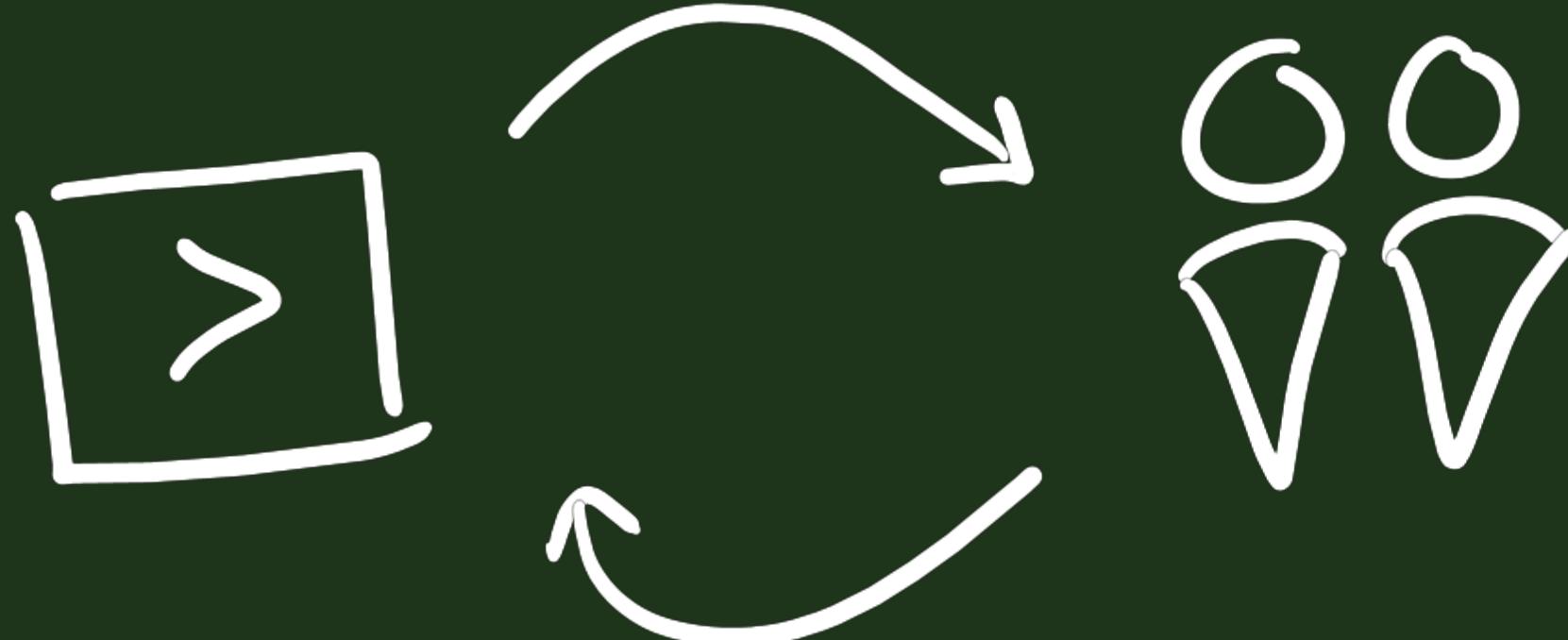


Projekte

arbeitspakte



projektablauf



projektablauf



fragen

Anhang H

Arbeitslogbuch

Delegatorx

Klassifizierung intern
Status in Arbeit
Programmname Delegator
Projektnummer 1
Projektleiter Tabinas Kenan
Version 0.1
Datum 26. März 2025
Auftraggeber Tabinas Kenan
Autor/Autoren Tabinas Kenan
Verteiler

Änderungsverzeichnis

Version	Datum	Änderung	Autor
0.01	17.02.	Weitere Einträge hinzufügt	TAK
0.02	28.02	Weitere Einträge hinzufügt / angepasst	TAK
0.03	01.04	Weitere Einträge hinzufügt / angepasst	TAK
0.04	16.04	Weitere Einträge hinzufügt / angepasst	TAK
0.05	30.04	Weitere Einträge hinzufügt / angepasst	TAK
0.06	02.05	Weitere Einträge hinzufügt / angepasst	TAK
0.07	03.05	Weitere Einträge hinzufügt / angepasst	TAK
0.08	06.05	Weitere Einträge hinzufügt / angepasst	TAK
0.09	10.05	Weitere Einträge hinzufügt / angepasst	TAK
0.10	16.05	Weitere Einträge hinzufügt / angepasst	TAK
0.11	18.05	Weitere Einträge hinzufügt / angepasst	TAK
0.12	21.05	Weitere Einträge hinzufügt / angepasst	TAK
0.13	23.05	Weitere Einträge hinzufügt / angepasst	TAK
0.14	26.05	Weitere Einträge hinzufügt / angepasst	TAK
0.15	29.05	Weitere Einträge hinzufügt / angepasst	TAK
0.16	01.06	Weitere Einträge hinzufügt / angepasst	TAK
0.17	02.06	Weitere Einträge hinzufügt / angepasst	TAK
0.18	03.06	Weitere Einträge hinzufügt / angepasst	TAK
0.19	05.06	Weitere Einträge hinzufügt / angepasst	TAK
0.20	06.06	Glossar & Tabellen Verzeichnis	TAK

Tabelle 1: Änderungsverzeichnis

1 Initialisierung

1.1 KW02

Ereignisse	
Gedanken	Ich habe diverse Fragen zu der Diplomarbeit. Vieles ist mir noch klar. Ich habe das Gefühl es wird besser, wenn es mal begonnen hat. Ich kann es mir noch nicht so vorstellen.
Projektplan	Bereits vor dem Plan

Tabelle 2: KW02

Ergebnis	Zeit h
Alle Dokumente und die Aufzeichnung angeschaut. In einem Dokument wichtiges festgehalten	3
Struktur aus Richtlinien-Dokument übernommen	1
Fragen im Teams gestellt	0.5
Themenwahl	2
Projekt Antrag ausgefüllt	2

Tabelle 3: KW02

1.2 KW03

Ereignisse	<p>Ich habe Hosting bei IoTSwitch ausprobiert. Das hat leider mega probleme geben. Die Partition war nur 2 GB obwohl ich 25GB hatte. Ab einem Punkt ist gar nichts mehr gegangen, weil apt abgebrochen hat. Hatte ewig Sachen zu löschen wegen demselben Problem.</p> <p>Marc erinnert wegen der Abgabe. Beantwortet mir die Frage zu meinem Experten aber nicht. Ich sehe seine Erinnerung nicht.</p>
Gedanken	Ich habe mal gelernt, there is no Free Lunch. Ich habe die 3\$ pro Monat jetzt ganz anders bezahlt.
Projektplan	Bereits vor dem Plan

Tabelle 4: KW03

Ergebnis	Zeit h
Hosting probleme bei IoTSwitch	6
Django Restframework ausprobiert	2
Suche nach Experten	1
Kommunikation mit Marc	
Projektantrag	4
Projektinitialisierungsantrag	4

Tabelle 5: KW03

1.3 KW04

Ereignisse	Mein Raspi ist angekommen. IoT Switch gekündigt.
Gedanken	Mir ist IoTswitch Fiasco entschieden zu weit. Ich habe mich für eine private Lösung entschieden.
Projektplan	Existiert zu diesem Zeitpunkt nicht

Tabelle 6: KW04

Ergebnis	Zeit h
Raspi 5 aufsetzen und Umgebung Einrichten	4
Weiter Djangorestframework erforschen und kleine Webseite einrichten	4

Tabelle 7: KW04

1.4 KW05

Ereignisse	Ich sehe die Erinnerung von Marc wegen der Abgabe. Erinnere Marc an meine Frage wegen des Experten. Marc lehnt ab. Bietet mir Adressen für einen neuen Experten. Ich habe Experten gefunden und reiche provisorisch ein.
Gedanken	Fängt schonmal toll an.
Projektplan	Existiert zu diesem Zeitpunkt nicht

Tabelle 8: KW05

Ergebnis	Zeit h
Suche nach Experten	2
Vorbereitung Kickoff	6

Tabelle 9: KW05

1.5 KW06

Ereignisse	Erhalte die Offizielle Bestätigung von meinem Experten.
Gedanken	Ich habe im Kopf Marc hat gesagt, wir sollen den Kickoff gut vorbereiten. Hier können wir gratis Feedback holen.
Projektplan	Existiert zu diesem Zeitpunkt nicht

Tabelle 10: KW06

Ergebnis	Zeit h
E-Mail und Terminfindung	0.5
Vorbereitung Kickoff	4
Projektplan erstellen	4

Tabelle 11: KW06

1.6 KW07

Ereignisse	Kick-Off
Gedanken	<p>Der Kick-Off hat mir geholfen, das Projekt mir besser vorzustellen. Ich habe ein Gefühl für die Erwartungen der Experten bekommen. Auch einige Positive Zusprüche, ich habe das Gefühl, das es so gut kommt.</p> <p>Ich werde die nächste Zeit nutzen, um mir auch andere Technologien anzuschauen und schauen welche Frameworks ich verwende.</p>
Projektplan	Existiert zu diesem Zeitpunkt nicht

Tabelle 12: KW07

Ergebnis	Zeit h
Kick-Off	2
Vorbereitung Kickoff	6

Tabelle 13: KW07

1.7 KW12

Ereignisse	Familiäre Konflikte
Gedanken	<p>Ich habe die Praktische Arbeit etwas schleifen lassen, ich musste gerade andere Sachen Priorisieren.</p> <p>Ich habe mich stark verschätzt bei der Kostenberechnung. Ich war zum Schluss sehr überzeugt nichts schlägt den Raspi 5. Sobald ich angefangen habe alle Aufgaben zu schätzen und meine Zeit aufzurechnen, bin ich etwas auf die Weltgekommen.</p>
Projektplan	Stark hinter dem Zeitplan

Tabelle 14: KW12

Ergebnis	Zeit h
Projektziele definieren	6
Projektplan erstellen	6
Konkurrenzanalyse und Zielgruppenanalyse	4
Wirtschaftlichkeit prüfen und verschiedene Hostings	8

Tabelle 15: KW12

1.8 KW13

Ereignisse	Ein Experte ist nicht zum Zwischen Meeting aufgetaucht.
Gedanken	Ich bin wieder auf Kurs. Ich habe mich gut auf das Zwischenmeeting vorbereitet. Ich fühle freue mich auf das Feedback. Wir haben ein neues Datum abgemacht, ich hoffe der Experte Antwortet mit früh genug.
Projektplan	Etwas in Verzögerung, da Kick Off jetzt erst nächste Woche ist.

Tabelle 16: KW13

Ergebnis	Zeit h
Zwischenmeeting	0.5
Lösungsvarianten	6
Zwischenmeeting Vorbereiten	4

Tabelle 17: KW13

1.9 KW14

Ereignisse	Zwischenmeeting hat stattgefunden
Gedanken	Der Experte hat sich gemeldet und wir konnten kurzfristig das Zwischenmeeting Organisieren und halten. Ich habe positives Feedback erhalten im Zwischenmeeting. Die Experten und ich sind gute Dinge, ich habe ein gutes Gefühl.
Projektplan	Etwas in Verzögerung

Tabelle 18: KW14

Ergebnis	Zeit h
Zwischenmeeting	1

Tabelle 19: KW14

2 Konzept

2.1 KW15

Ereignisse	Ich habe mit der Konzeptphase begonnen.
Gedanken	<p>Damit ich den Zeitplan nicht mehr so aus den Augen verlieren plane ich mir einen festen Tag der Woche ein. Mit Familie ist das immer ein wenig schwierig. Ich hoffe ich kann beibehalten.</p> <p>Mir hat das Mockup Zeichen sehr gefallen. Beim Prozess habe ich mir viele Fragen gestellt, dabei bin ich froh habe ich mir diese jetzt gestellt. Die App wird immer wie realer.</p> <p>Es braucht um einiges mehr Zeit als ich gedacht habe.</p> <p>Zuerst wollte ich es digital machen. Ich habe festgestellt mit Stift auf Papier, fiel mir am leichtesten.</p>
Projektplan	Etwas in Verzögerung

Tabelle 20: KW15

Ergebnis	Zeit h
UX / Wireframes	8

Tabelle 21: KW15

2.2 KW16

Ereignisse	
Gedanken	<p>Ich habe Fortschritte gemacht und die nächsten Schritte sind klar.</p> <p>Ich muss schauen, dass ich mich nicht übernehme. Dass ich alle möglichen Features der App geplant habe, fand ich aber gut.</p> <p>Das Design des UI ist ausbaubar bis alle möglichen Zukünftigen Features einen Platz haben. Zudem ist dadurch auch bekannt, wo die Reise hingeht wenn ich dieses Projekt außerhalb der Diplomarbeit weiter gehe.</p>
Projektplan	Etwas in Verzögerung

Tabelle 22: KW16

Ergebnis	Zeit h
UX / Wireframes	8
Features definiert	2

Tabelle 23: KW16

3 Realisierung

3.1 KW17

Ereignisse	Begonnen mit Teilen der Realisierung.
Gedanken	<p>Ich habe viel zu spät festgestellt das ich dringend ein Rollenkonzept benötige. Dies habe ich während der Planung nicht berücksichtigt.</p> <p>Das ist etwas, was mir an der Arbeit gefällt. Ich war noch nie von Anfang bis zum Schluss bei einer Applikation bei allem beteiligt.</p> <p>Ich werde immer besser mit Docker. Hotreload mit Gunicorn hilft sehr bim entwickeln.</p>
Projektplan	Etwas in Verzögerung

Tabelle 24: KW17

Ergebnis	Zeit h
Datenbankkonzept erstellt	8
Modelle im Backend entwickeln	4
Umgebung angepasst	0.5

Tabelle 25: KW17

3.2 KW18

Ereignisse	
Gedanken	<p>Ich habe ein gutes Gefühl. So wie ich vorgehe also Postman und dem Konzept komme ich gut voran.</p> <p>Mir gefällt, dass ich Zeit mir Gedanken zu solchen Sachen zu stellen. So muss ich später weniger Änderungen machen.</p> <p>Das Backend so sicher zumachen entspricht auch dem Sicherheit Gedanken, der mir am Anfang auferlegt wurde.</p>
Projektplan	Ein paar Wochen im Verzug

Tabelle 26: KW18

Ergebnis	Zeit h
Rollen- und Zugriffskonzept	8
Konzept im Backend durchsetzen	4
Testen	4
JWT Bearer Token eingebaut	4

Tabelle 27: KW18

3.3 KW19

Ereignisse	Mit Frontend begonnen Calender Synchronisierung angefangen
Gedanken	<p>Ich lerne immer wieder neue Sachen wie über Seed values helfen bei der Umsetzung.</p> <p>Diese Woche habe ich das erste Mal mit Flutter begonnen. Das hat zu einem gewissen Zeitpunkt gut funktioniert. Leider hatte ich später mit Tasks. Welche ich nicht wusste wie lösen. Habe jedes Language Modell der Welt und andere Entwickler um Rat geholfen, bis ich festgestellt habe, dass mein Vorgehen so nicht weiter geht.</p> <p>Leider kann ich ohne Tasks die Kalendersynchronisierung nicht Testen, da ich ja sonst keine Termine habe. Ich kann es daher nicht wirklich Testen.</p> <p>Für das Projekt ein Rückschritt, für mich als Entwickler ein Schritt in die richtige Richtung. Diese Lektion hat mir viel gezeigt.</p>
Projektplan	Ein paar Wochen im Verzug

Tabelle 28: KW19

Ergebnis	Zeit h
Flutter Projekt begonnen (Chat und Kalender umgesetzt)	4
Vergeblich Fehler gesucht	4
Kalender Synchronisierung soweit umgesetzt	2

Tabelle 29: KW19

3.4 KW20

Ereignisse	Die Domain delegator.ch eingerichtet mit meinen Containern. Neu mit dem Frontend begonnen Ich verliess meine Band
Gedanken	Mein neuer Ansatz gefällt mir sehr. Wenn ich zuerst Integrationstest schreibe, ermöglicht es mir meine Applikation besser zu testen. Ich habe leider festgestellt, dass Google nur alle 24h Synchronisierte Kalender Updated. Daher muss ich mein über das Internet verfügbar machen. Anders kann ich sonst nicht die Kalender Synchronisierung auf dem Smartphone Testen. Zum Glück habe ich am Anfang bereits mir Gedanken zu der Onlineschaltung gemacht. Jetzt habe ich ein Großartiges Mobiles Setup. Solange mein Computer läuft ist meine Webseite online verfügbar. Dazu nutze ich den Cloudflare Tunnel. Zudem ich habe mich von meiner Band getrennt, was es nicht mehr möglich macht die App zu testen. Die Priorität diese auf den Appstore zu bringen ist so gesunken.
Projektplan	Ein paar Wochen im Verzug

Tabelle 30: KW20

Ergebnis	Zeit h
Cloudflared Tunnel	4
Recherchiert zu Kalender Synchronisierung	4
Mobile App Entwicklung	8

Tabelle 31: KW20

3.5 KW21

Ereignisse	
Gedanken	<p>Mittlerweile investiere ich jeden Moment meiner Freizeit in die Entwicklung. Mir gefällt meine App immer wie mehr. Ich komme auch schnell voran.</p> <p>Ich nahm mir auch Zeit wieder die Anforderungen aufzuschreiben und habe mir Fragen ausgeschrieben, welche ich nächste Woche stellen möchte.</p> <p>Ich habe mir die Anforderungen für App release angeschaut. Leider habe ich nur eine Prepaid Kreditkarte und kann leider die App nicht releasen. Trotzdem kann ich mich auf den App Release mit den Anforderungen vorbereiten.</p> <p>Als ich mich auf das Testkonzept vorbereitet habe, habe ich mir trotzdem überlegt, dass eine Userverwaltung trotzdem sinnvoll ist und habe hier viele Features eingebaut.</p>
Projektplan	Je nach Punkt bin ich nur noch eine oder zwei Wochen im Verzug

Tabelle 32: KW21

Ergebnis	Zeit h
Mobile App Entwicklung	16
Userverwaltung	7
Testkonzept	3
Bug Report Funktion eingebaut	2
App release vorbereitet	1

Tabelle 33: KW21

4 Abschluss

4.1 KW22

Ereignisse	Der Abschluss steht an
Gedanken	<p>Letzte Woche jetzt noch mal Vollgas.</p> <p>Ich habe gutes Feedback erhalten von Marc. Wenn ich das Coaching nicht genutzt hätte, hätte ich irgend etwas ganz komisches abgegeben.</p> <p>Die Testdurchführung fand ich unheimlich gut, viele Sachen sind mir aufgefallen, welche sonst komplett untergangen während. Hat mich positiv überrascht.</p>
Projektplan	Sonntag respektive Montag um 06:00 muss ich abgeben.

Tabelle 34: KW22

Ergebnis	Zeit h
Testdurchführung	3
Coaching	0.5
Fact Sheet	
Betriebliche Sicht festhalten	
Diplombericht	
Anhänge Vorbereiten	
Risiko Analyse	

Tabelle 35: KW22

5 Abkürzungen und Glossar

Abkürzung / Fachwort	Erläuterung
API	Steht für Application Programming Interface. Ist eine Schnittstelle, die in Form von Endpoints, Daten via JSON bereitstellen oder verarbeiten.
Bearer Token	Theoretisches Konzept für Authentifizierung in diesem Fall umgesetzt mit JWT in Django
Django REST	Ein Framework zur Erweiterung Django für die Entwicklung von REST-APIs
Docker	Plattform für bereitstellen von Container. Ähnlich wie ein VMWare aber für Container anstelle von Containern.
Flutter	Crossplattform Framework. Analog zu .Net Maui und React Native
Gunicorn	Python HTTP Server. Als Brücke zwischen Django und nginx.
Hot-Reload	Ermöglicht Neuladen von Code-Änderungen ohne Neustart
IoT	Steht für Internet of Things. Beispiel Kühlschrank, welcher dir sagt, was du noch zuhause hast.
JWT	Steht für JSON Web Token. Ist ein Alphanumerischer Code, welcher bei der Authentifizierung genutzt wird.
Wireframe	Visueller Entwurf einer Benutzeroberfläche. Verglichen zum Mockup keine Farbe und keine Bilder.
Postman	Software zum Testen und Entwickeln von APIs. In diesem Fall als VS Code Extension.
Raspberry Pi (Raspi)	Singleboardcomputer. In diesem Fall Raspi 5.

Tabelle 36: Abkürzungen und Glossar

Vorschläge der Wörter kommen von Claude. Erläuterungen nicht.

Inhaltsverzeichnis

1 Initialisierung.....	2
1.1 KW02.....	2
1.2 KW03	3
1.3 KW04.....	4
1.4 KW05.....	4
1.5 KW06.....	5
1.6 KW07	6
1.7 KW12	7
1.8 KW13	8
1.9 KW14	8
2 Konzept.....	9
2.1 KW15	9
2.2 KW16	10
3 Realisierung	11
3.1 KW17.....	11
3.2 KW18.....	12
3.3 KW19	13
3.4 KW20.....	14
3.5 KW21.....	15
4 Abschluss	16
4.1 KW22.....	16
5 Abkürzungen und Glossar.....	17

Tabellenverzeichnis

Tabelle 1: Änderungsverzeichnis.....	1
Tabelle 2: KW02.....	2
Tabelle 3: KW02	2
Tabelle 4: KW03.....	3
Tabelle 5: KW03.....	3
Tabelle 6: KW04.....	4
Tabelle 7: KW04	4
Tabelle 8: KW05	4

Tabelle 9: KW05.....	4
Tabelle 10: KW06.....	5
Tabelle 11: KW06	5
Tabelle 12: KW07	6
Tabelle 13: KW07	6
Tabelle 14: KW12	7
Tabelle 15: KW12	7
Tabelle 16: KW13	8
Tabelle 17: KW13.....	8
Tabelle 18: KW14	8
Tabelle 19: KW14	8
Tabelle 20: KW15.....	9
Tabelle 21: KW15	9
Tabelle 22: KW16.....	10
Tabelle 23: KW16.....	10
Tabelle 24: KW17.....	11
Tabelle 25: KW17	11
Tabelle 26: KW18.....	12
Tabelle 27: KW18	12
Tabelle 28: KW19.....	13
Tabelle 29: KW19.....	13
Tabelle 30: KW20	14
Tabelle 31: KW20.....	14
Tabelle 32: KW21.....	15
Tabelle 33: KW21.....	15
Tabelle 34: KW22	16
Tabelle 35: KW22	16
Tabelle 36: Abkürzungen und Glossar.....	17

Anhang I

Abkürzungen und Glossar

Abkürzung / Fachwort	Erläuterung
DSGVO	Datenschutzgrundverordnung
nDSG	National (Schweiz) Datenschutz Gesetz
DACH	(Deutschland, Austria, Schweiz)
bcrypt	Eine Art Passwort hashing.
API	Steht für Application Programming Interface. Ist eine Schnittstelle, die in Form von Endpoints, Daten via JSON bereitstellen oder verarbeiten.
Bearer Token	Theoretisches Konzept für Authentifizierung in diesem Fall umgesetzt mit JWT in Django
Django REST	Ein Framework zur Erweiterung Django für die Entwicklung von REST-APIs
Docker	Plattform für bereitstellen von Container. Ähnlich wie ein VMWare aber für Container anstelle von Containern.
Flutter	Crossplattform Framework. Analog zu .Net Maui und React Native
Gunicorn	Python HTTP Server. Als Brücke zwischen Django und nginx.
Hot-Reload	Ermöglicht Neuladen von Code-Änderungen ohne Neustart
IoT	Steht für Internet of Things. Beispiel Kühlschrank, welcher dir sagt, was du noch zuhause hast.
JWT	Steht für JSON Web Token. Ist ein Alphanumerischer Code, welcher bei der Authentifizierung genutzt wird.
Wireframe	Visueller Entwurf einer Benutzeroberfläche. Verglichen zum Mockup keine Farbe und keine Bilder.
Postman	Software zum Testen und Entwickeln von APIs. In diesem Fall als VS Code Extension.
Raspberry Pi (Raspi)	Singleboardcomputer. In diesem Fall Raspi 5.

Tabelle 1: Glossar 1

Vorschläge der Wörter kommen von Claude. Erläuterungen nicht.

Abkürzung / Fachwort	Erläuterung
CLOUD Act	US Gesetz Behörden den Zugriff auf Daten von US-Unternehmen ermöglicht.
Schrems II	Urteil vom EU-Parlament von 2020. In dem wurde Privacy Shield als ungültig erklärt.
Privacy Shield	Ehemaliges Datenschutzabkommen zwischen EU und USA.
Standardvertragsklauseln	Vertraglich vereinbarte Datenschutzstandards für internationale Datenübertragungen
Betroffenenrechte	Rechte von Personen bezüglich ihrer Daten
TLS 1.3	Transport Layer Security, aktueller Standard für verschlüsselte Datenübertragung
AES-256	Advanced Encryption Standard mit 256-Bit Schlüssellänge für Datenverschlüsselung
Salt	Zufällige Daten, die beim Passwort-Hashing zur Erhöhung der Sicherheit verwendet werden
Multi-Faktor-Authentifizierung (MFA)	Sicherheitsverfahren mit mehreren Authentifizierungsfaktoren
Incident Response	Strukturierter Prozess zur Behandlung von Sicherheitsvorfällen
Penetration Testing	Sicherheitstest durch simulierte Angriffe auf IT-Systeme
CaaS Provider	Container-as-a-Service Anbieter für Cloud-Infrastruktur
AGB	Allgemeine Geschäftsbedingungen - rechtliche Regeln zwischen Anbieter und Nutzer
DSGVO	Datenschutz-Grundverordnung - EU-weites Gesetz zum Schutz personenbezogener Daten
nDSG	Neues Datenschutzgesetz der Schweiz - schweizer Pendant zur DSGVO
Salvatorische Klausel	Bestimmung, die sicherstellt, dass bei Unwirksamkeit einzelner AGB-Punkte der Rest gültig bleibt
Höhere Gewalt	Außergewöhnliche, unvorhersehbare Ereignisse (Naturkatastrophen, Pandemie, etc.)
Außerordentliche Kündigung	Sofortige Kündigung bei schwerwiegenden Vertragsverletzungen

Abkürzung / Fachwort	Erläuterung
Verzugszinsen	Zusätzliche Kosten bei verspäteter Zahlung (hier: 5% p.a.)
Basiszinssatz	Referenzzinssatz der Europäischen Zentralbank
Cross-Platform	Software, die auf verschiedenen Betriebssystemen läuft (iOS, Android, Web)
Reverse Engineering	Rückentwicklung von Software zur Analyse des Codes
Scraping/Crawling	Automatisiertes Auslesen von Daten durch Programme
Synchronisation	Abgleich von Daten zwischen verschiedenen Systemen
Vertragspartner	Die Personen/Unternehmen, die den Vertrag abschließen
Nacherfüllung	Recht auf Reparatur/Ersatz bei Mängeln
Gerichtsstand	Ort, an dem bei Rechtsstreitigkeiten geklagt werden muss
Setlisten-Management	Funktion zur Erstellung und Verwaltung von Songlisten für Auftritte
Ressourcenplanung	Verwaltung von Equipment und Proberäumen
Rollenzuweisungen	Festlegung unterschiedlicher Berechtigungen für Bandmitglieder
Betroffenenrechte	Ihre Rechte bezüglich Ihrer gespeicherten Daten
Einwilligung	Ihre ausdrückliche Zustimmung zur Datenverarbeitung
Vertragserfüllung	Rechtsgrund für notwendige Datenverarbeitung zur Service-Bereitstellung
Minderjährige	Personen unter 18 Jahren
Textform	Schriftliche Erklärung ohne Unterschrift (E-Mail reicht)
Mittelbare Schäden	Indirekte Folgeschäden
Wesentliche Vertragspflichten	Kernleistungen, die den Vertragszweck prägen
CLOUD Act	US-amerikanisches Gesetz, das Behörden den Zugriff auf Daten von US-Unternehmen ermöglicht, auch wenn diese außerhalb der USA gespeichert sind
Schrems II	EuGH-Urteil von 2020, das den Privacy Shield für unwirksam erklärte

Abkürzung / Fachwort	Erläuterung
Privacy Shield	Ehemaliges Datenschutzabkommen zwischen EU und USA (2020 für unwirksam erklärt)
Standardvertragsklauseln (SCC)	Vertraglich vereinbarte Datenschutzstandards für internationale Datenübertragungen
Betroffenenrechte	Rechte von Personen bezüglich ihrer personenbezogenen Daten (Auskunft, Berichtigung, Löschung etc.)
TLS 1.3	Transport Layer Security, aktueller Standard für verschlüsselte Datenübertragung
AES-256	Advanced Encryption Standard mit 256-Bit Schlüssellänge für Datenverschlüsselung
Salt	Zufällige Daten, die beim Passwort-Hashing zur Erhöhung der Sicherheit verwendet werden
Multi-Faktor-Authentifizierung (MFA)	Sicherheitsverfahren mit mehreren Authentifizierungsfaktoren
Incident Response	Strukturierter Prozess zur Behandlung von Sicherheitsvorfällen
Penetration Testing	Sicherheitstest durch simulierte Angriffe auf IT-Systeme
CaaS Provider	Container-as-a-Service Anbieter für Cloud-Infrastruktur
Audit-Logging	Protokollierung sicherheitsrelevanter Ereignisse für Compliance-Zwecke
Cookie-Consent	Einverständniserklärung für die Verwendung von Cookies
CLOUD Act	US-amerikanisches Gesetz, das Behörden den Zugriff auf Daten von US-Unternehmen ermöglicht, auch wenn diese außerhalb der USA gespeichert sind

Tabelle 2: Glossar 2

Einige Textzeilen von Claude