



kalm

ENDDOKUMENTATION

Interactive Media Design
Wintersemester 2020/2021
Hochschule Darmstadt
Mediencampus Dieburg

Maximilian Weber [765026]
Phuong Quach [762221]
Lisa Jochem [765004]
Ann-Kathrin Tissen [765048]

betreuende Dozenten:

Prof. Tsune Tanaka
Prof. Claudius Coenen
Katharina Bähr
Sarah Antes



Danksagung

Wir danken allen genannten Professoren und Dozent:innen für die vielfältigen Inputs und die großartige Hilfe, die wir über das gesamte Semester hinweg erhalten haben!

Vielen Dank an:

Prof. Tsune Tanaka
Prof. Claudius Coenen
Katharina Bähr
Sarah Antes
Sebastian Haase
Tomas Zebis
Martin Haas



Gendererklärung

In der folgenden Dokumentation verwenden wir, wo es möglich ist, die Partizipialform und somit eine genderneutrale Form. Wo es nicht möglich ist, verwenden wir den Gender-Gap in Form eines Doppelpunkts. Alle weiteren Geschlechter sind ausdrücklich miteinbezogen.

Beispiel:

80% der Autofahrenden bilden zu spät eine Rettungsgasse.

Der/die Rettungssanitäter:in ist unterwegs.

Zitierweise

Diese Arbeit zitiert nach den Richtlinien der American Psychological Association (APA 7. Auflage).

Abbildungen

Abbildungen ohne Quellenanmerkungen stammen von den Autor:innen.



INHALT

06 Das Team

07 Einführung

08 Aufgabenstellung

09 Problemstellung

10 Entitäten und ihr Konflikt

11 Idee

12 Zielscheibe

14 Konzept & Umsetzung

15 Charakter der Anwendung

16 Handlungsablauf

17 5W-Fragen

18 Value Proposition

19 Styleguide

20 Moodboard

21 Farbgebung

22 Typografie

23 Sounddesgin & Timing

24 Screendesign & Animation

25 Screens

29 Modell im Einsatz

32 Technik

33 Hardware

34 Hardware Komponente

35 Komponente Zusammensetzung

37 Software

39 Datentransfer und Datensicherheit

40 Entity-Relationship Model

41 Systemdiagramm

42 Swimlane-Diagramm

43 Ablaufdiagramm

44 Proof of Concept

45 Herausforderungen in der Technik

49 Ausblick

51 Management

52 Team Model Canvas

54 Kommunikationskanäle

55 Arbeitsverteilung

56 Induktion und Deduktion

57 Agiles Projektmanagement

60 Risiken

61 Arbeitspakete

63 Learnings

64 Design Thinking

66 Projekt-Learnings

68 Quellen

Das Team



Maximilian Weber

Neben 3D Modelling arbeite ich am liebsten mit Microcontrollern, weil man da am Ende des Tages was in der Hand halten und sagen kann: „Das habe ich gebaut“. In meiner Freizeit fahre ich gerne Motorrad oder spiele Gitarre.



Lisa Jochem

Im Studium habe ich gelernt, dass mir Coding nicht so leicht von der Hand geht, weswegen ich mich persönlich dort am meisten verbessern möchte. Deswegen sehe ich mich mehr im Bereich Design und Management, da mir diese Bereiche viel Spaß machen.



Phuong „Kitty“ Quach

Moderne-Medien-Enthusiastin, kritische Denkerin und Optimistin. Wird leicht inspiriert und erforscht gerne unkonventionelle Perspektiven. Versucht bei Projekten nicht nur an User-Centered Design zu denken, sondern vor allem an Society-Centered Design. Legt einen großen Wert auf die visuelle Harmonie und den kontextuellen Hintergedanken ihrer Gestaltungsideen und Farbgebung.



Ann-Kathrin Tissen

Als Mediengestalterin ins Studium gestartet habe ich neben Design auch Management und insbesondere Coding lieben gelernt. Mir macht es Spaß, neue Perspektiven kennenzulernen, unkonventionelle Ideen auszuprobieren und mich mit mir unbekannten Dingen zu beschäftigen.

Einführung

Aufgabenstellung

In diesem Semester haben wir uns mit der Thematik „Interaktion unterwegs“ beschäftigt. Wir sind ständig mobil, fahren Zug, Auto, Bus, Fahrrad oder gehen wandern oder spazieren. Um dieses große Themenfeld besser verstehen zu können, war es in diesem Semester unsere Aufgabe, Anwendungen zu entwickeln, die die Interaktion zwischen Verkehrsteilnehmern verbessern oder komplett neue Interaktionsmöglichkeiten schaffen können.

Um uns diesem Themenfeld annähern zu können, haben wir uns mit einer deduktiven Recherche zunächst dem „großen Ganzen“ angenähert. Möchte man verstehen, welche wirklich relevanten Probleme es in einem Bereich gibt, muss man zunächst den gesamten Bereich durchdringen. Nachdem wir diese Recherche vorgenommen hatten, haben wir uns auf einen thematisch abgesteckten Bereich, die Interaktion zwischen Rettungskräften und Autofahrenden, beschränkt, welche das Thema unseres Semesterprojektes darstellt.

Problemstellung

Rettungsgassen sind für Sanitäter:innen, die Feuerwehr, die Polizei usw. ein wichtiges Mittel, um den Einsatzort schnellstmöglich und ohne Behinderungen zu erreichen. In der Realität ist es allerdings der Fall, dass Sanitäter:innen des Deutschen Roten Kreuzes in 80% der Fälle beobachten, dass Einsätze der Rettungskräfte erschwert werden, weil Rettungsgassen blockiert sind. Problematisch ist dies vor allem, weil der Zeitverlust auf Grund blockierter Rettungsgassen durchschnittlich fünf Minuten beträgt, bei einem akuten Notfall ist allerdings jede Sekunde wertvoll. Besonders dann, wenn ein Patient beispielsweise reanimiert werden muss (Deutsches Rotes Kreuz, 2018). Laut ausgewerteten Einsätzen wurde nur bei 15,6% aller Einsätze spontan eine Rettungsgasse gebildet. Weiterhin problematisch ist, dass die Rettungsgasse bei 35,4% der Fahrer:innen erst durch ein Sondersignal wie dem Martinshorn gebildet wird, obwohl diese bereits entstehen sollte, wenn Fahrer:innen auf der Straße nur noch mit Schrittgeschwindigkeit unterwegs sein können (Runter Vom Gas, 2020). So kann es passieren, dass in nur einem Stau bis zu 100 Verstöße gegen Rettungsgassen dokumentiert werden können (Badische Neueste Nachrichten, 2020).

Entitäten und ihr Konflikt

Um den fehlerhaften Prozess verbessern zu können, haben wir uns die Beziehung der Entitäten genauer anschaut. Beide Entitäten geraten in Konflikt, sobald ihr Handeln und ihre Motivation sich gegenseitig an etwas hindern, das heißt in anderen Worten, wenn sie sich gegenseitig ein Hindernis in den Weg stellen. In diesem Fall wird der Einsatzwagenfahrende von dem/der Autofahrer:in aufgehalten, da sie nicht rechtzeitig reagiert hat.

Bei dieser Interaktion hat der Einsatzwagenfahrende die Motivation, schnellstmöglich ungehindert zum Einsatzort zu gelangen. Die Motivation des Autofahrenden ist es, schnellstmöglich von A nach B zu gelangen. Der Konflikt entsteht also, sobald sich ihre Wege kreuzen.

Wie bereits in der Problemstellung geschildert wurde, stellen Autofahrende das größte Problem in Zusammenhang mit blockierten Rettungsgassen dar. Einsatzwagenfahrende verlieren durch das Missachten der Rettungsgassenbildung essenzielle Zeit und es kann zu schweren Folgen kommen.

Jedoch wollen wir nicht einfach auf Kosten der Autofahrenden den Einsatzwagenfahrenden durchschleusen. Unser Ziel ist es, beiden Interaktionspartnern Zeit zu verschaffen. Damit ist nicht nur die Zeit gemeint, die der Einsatzwagenfahrende bekommt, um schneller zum Einsatzort zu kommen, oder die Zeit, die die Autofahrenden bekommen, um rechtzeitig beiseite zu fahren. Zeit verschaffen für den Autofahrenden bedeutet in diesem Fall, dass er aus seiner mentalen Stresssituation austritt, um danach klar und strukturiert handeln zu können.

Der/die Autofahrende gerät nämlich in Stress und Aggressivität, weil er in eine Situation geraten ist, mit der er nicht gerechnet hat, denn ein Unfall ist ja ein unvorhergesehenes Ereignis. Er kann die für ihn aktuell unbekannte Situation nicht einschätzen und wird verunsichert.

Menschen können es schlecht ertragen, wenn wir einer Situation ausgesetzt sind, über die wir keine Klarheit oder Kontrolle haben und Kontrollverlust führt bei uns immer zu Angst und Aggressivität (Stasch, 2020). Hinzu kommt, dass es einigen Autofahrenden schwerfällt, hinterm Steuer menschlich zu bleiben und sich wie kooperative, freundliche Wesen zu verhalten. Grund dafür sind vor allem die mangelhafte menschliche Kommunikation und Interaktion. Die Aspekte des fehlenden Blickkontakts, der eingeschränkten Kommunikation und der Anonymität haben wesentliche Auswirkungen auf das Verhalten der Autofahrenden. Das Gefühl der Anonymität fördert Aggression und rücksichtsloses Verhalten. Die Sprache und der Blickkontakt fallen ebenso weg. So werden bei den Fahrzeugen nicht die Menschen, die darin sitzen, erkannt, sondern nur noch das Fahrzeug selbst gesehen und es führt zu einer Entmenschlichung im Verkehr (Braun, 2019).

So haben wir uns vorgenommen, dem entgegenzuwirken und den Autofahrenden eine persönliche Verbindung zum anderen Verkehrsteilnehmenden, in diesem Fall dem Einsatzwagenfahrenden, zu ermöglichen. Persönliche Verbindung bedeutet hier Empathie und Verständnis für die Stresssituation des Interaktionspartners.

Idee

Dieser Problematik wollen wir uns annehmen und eine Anwendung schaffen, die die ungenaue Interaktion zwischen Rettungskräften und Autofahrer:innen verbessert und durch eine klare, zielgerichtete und empathische Interaktion austauscht. Unser Ziel ist es, Rettungskräfte dabei zu unterstützen, schnellstmöglich zu ihrem Ziel zu kommen und Autofahrenden ein stressfreies, entspanntes Bilden einer Rettungsgasse zu ermöglichen. Insgesamt möchten wir also den fehlerhaften Prozess der Interaktion berichtigen, indem wir Stress, Hektik und Missverständnisse bei und zwischen den Entitäten vermeiden wollen. Dazu müssen wir vor allem Verständnis für die Situation der Autofahrenden zeigen, da sich diese durch die vorher beschriebenen Probleme in einer Situation wiederfinden, die als sehr stressig empfunden wird.

Wir möchten uns für den Verlauf dieses Projektes darauf beschränken, unsere Anwendung für Autofahrende zu entwickeln. Da verschiedene Verkehrsteilnehmende wie Fahrradfahrende, Fußgänger:innen, Autofahrende usw. bedingt durch die unterschiedlichen Fortbewegungsmittel andere Voraussetzungen und Bedürfnisse haben, sind auch die Anforderungen an ein Produkt unterschiedlich. Aus zeitlichen Gründen und weil wir lieber ein Produkt entwickeln, welches für eine spezielle Gruppe ganz genau passt als eines, welches die Bedürfnisse unterschiedlicher Entitäten nur unzureichend abdeckt, haben wir den Entschluss gefasst, uns auf eine Personengruppe zu beschränken.

Im Rahmen unseres Semesters werden wir uns auf die Situationen auf Bundesstraßen beschränken. Der Grund für den Bezug auf die Bundesstraße ist, dass diese deutlich übersichtlichere Situationen bieten als Szenarien in der Stadt. Dort gibt es sehr viele Unbekannte, wie z.B. viele Seitenstraßen, unterschiedliche Richtungen, in welche Autos fahren können, etc.. Gleichzeitig bieten Bundesstraßen allerdings auch mehr Optionen als Autobahnen, da es mehr Abfahrten gibt und, je nach Bundesstraße, außerdem die Möglichkeit existiert, dass ein Rettungswagen auch auf die Gegenfahrbahn ausweichen muss. Um auch für komplexere Situationen in der Stadt ein gut durchdachtes Produkt anbieten zu können, wäre mehr Zeit von Nöten, weswegen wir uns auf die oben genannten Punkte beziehen möchten. Natürlich bietet unser Konzept allerdings eine Grundlage, mit welcher man mit mehr Zeit auch ein Produkt für städtische Situationen entwickeln kann.

Zielscheibe

Um für unsere Idee konkrete Ziele entwickeln zu können, haben wir die in Management vorgestellte Zielscheibe genutzt.

Den Zweck bzw. Sinn, den wir für unser Vorhaben erkannt haben, ist es, eine bessere Interaktion zwischen Rettungskräften und Autofahrenden zur Vermeidung von Problemen bei der Bildung der Rettungsgasse zu erzielen. Der momentane Konflikt, der auf Grund der mangelhaften Interaktion der Entitäten entsteht, soll durch unsere Anwendung aufgelöst werden, um stattdessen eine klare, nachsichtige und empathische Interaktion zu schaffen.

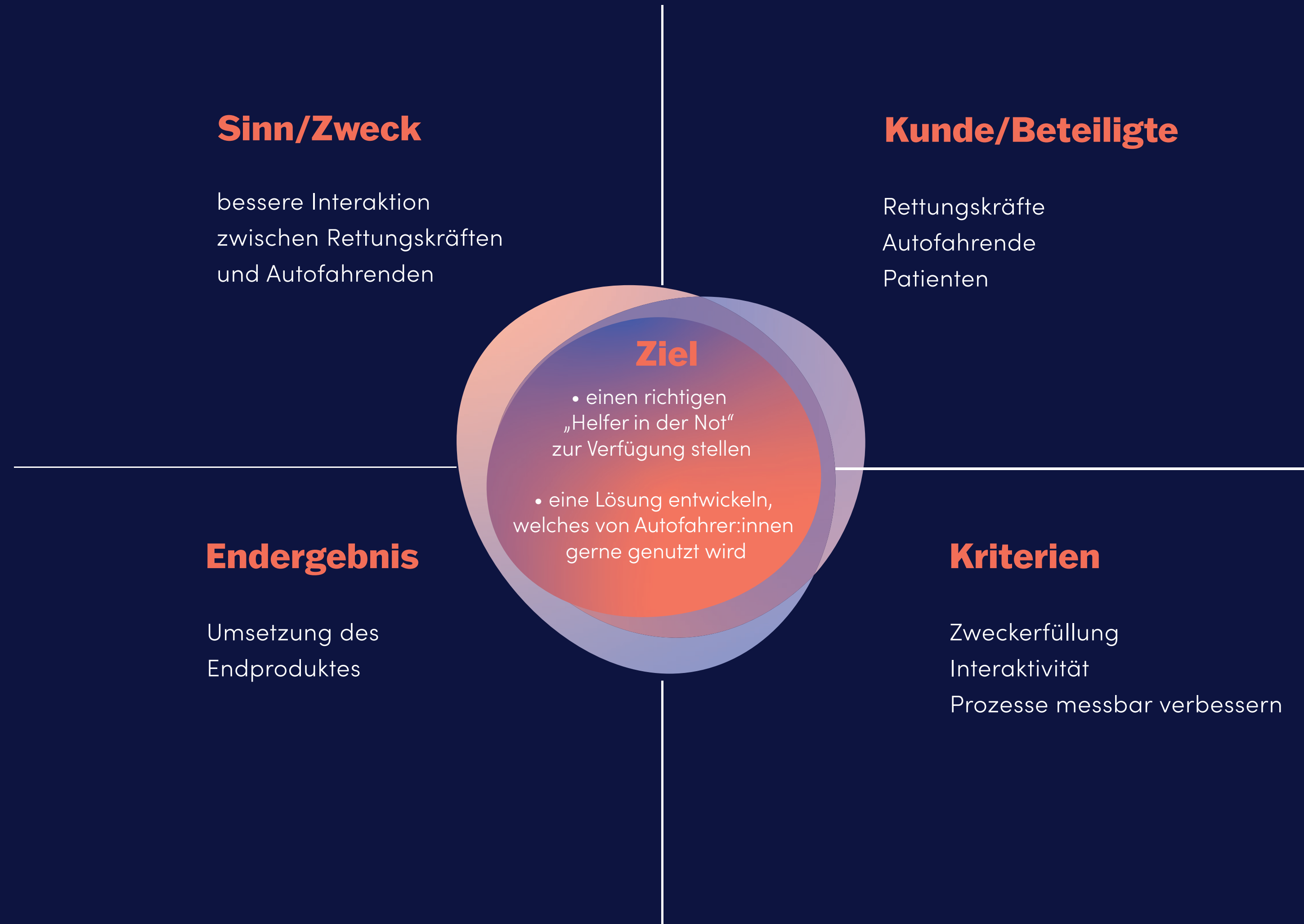
Beteiligte sind neben den Rettungskräften und Autofahrer:innen auch indirekt die Patienten der Einsatzwagenfahrenden, da für sie der zeitliche Vorteil, der durch unser Projekt geschaffen wird, im Ernstfall den Unterschied zwischen Leben und Tod ausmachen kann.

Kriterien, welches das Projekt erfüllen muss, sind, dass unser Zweck, den wir vorher beschrieben haben, erfüllt wird, und außerdem soll das entwickelte Produkt interaktiv sein. Zusätzlich ist es für uns ein wichtiges Kriterium, dass wir den fehlerhaften Prozess messbar verbessert haben. Dies könnte man beispielsweise durch eine qualitative Umfrage unter Rettungskräften überprüfen oder indem man sich die Entwicklung von Verstößen im Zusammenhang mit Rettungsgassen im Straßenverkehr genauer betrachtet.

Welches Ergebnis wollen wir bis wann erreichen? Generell wurde unser Projekt in vier Hauptphasen unterteilt. Zu Beginn eine Ideenphase, dann folgte eine Recherchephase, wobei diese beiden Phasen teilweise auch ineinander griffen. Nachdem wir eine Idee erarbeitet und eine wissenschaftliche Recherche durchgeführt hatten, folgte auf dieser Basis unsere Konzeptphase. Nachdem das Konzept erarbeitet wurde, folgte im letzten Schritt die Umsetzung des Endprodukts.

Aus diesen Teilbereichen haben wir um Endeffekt unser übergeordnetes Ziel für dieses Projekt erarbeitet. Wir möchten ein Produkt entwickeln, welches von Autofahrer:innen gerne genutzt wird, denn sie sollen das Gefühl bekommen, einen richtigen „Helfer in der Not“ an ihrer Seite zu haben. Somit soll unser Produkt die momentane Situation der Entitäten deutlich verbessern. Autofahrer:innen sollen mehr Zeit bekommen, auf die Situation zu reagieren und somit entspannter bleiben können, Rettungskräfte sollen möglichst ohne Verzögerung an ihr Ziel gelangen, um Menschenleben retten zu können.

Zielscheibe





Konzept & Umsetzung

Charakter der Anwendung

Wichtig zu klären ist nun, wie der Charakter der Anwendung aussehen muss, um zur gewünschten Art der Interaktion zu passen. KALM soll ein neutraler, seriöser Mediator und Helfer in der Not sein. Die Anwendung dient als Vermittler, der eine klare Interaktion ermöglicht. KALM ist dabei nachsichtig mit allen Beteiligten, kommuniziert trotzdem klar und deutlich und zeigt Empathie für beide Seiten, damit diese sich verstanden fühlen und in der Situation gerne aufeinander zugehen (Stasch, 2021). Diese Art des Charakters sorgt dafür, dass wir nicht nur auf eine der beiden Entitäten, nämlich die Rettungskräfte, die unter Zeitnot stehen, um ein Leben retten zu wollen, sondern auf beide eingehen. Wir „kommandieren“ die Autofahrenden nicht einfach „herum“, sondern gehen auch auf ihre Situation ein. Dadurch, dass wir ihnen Informationen über den Hergang bereitstellen, haben sie durch den empathischen Umgang einerseits mehr Kapazitäten, mit der Situation umzugehen, andererseits können sie sich durch das „Endanonymisieren“ der Situation besser in diese hineinversetzen. Dadurch können sie selber mehr Empathie mit den Rettungskräften und dem Opfer aufbringen, was ein altruistisches Verhalten und somit eine Kooperation begünstigt (Bierhoff, 2019).

Handlungsablauf

Wir möchten ein Warnsystem für Autofahrer:innen entwickeln, welches persönlich und nachsichtig arbeitet. Dabei werden zwei unterschiedliche Module entwickelt: Ein Modul im Rettungswagen sendet während eines Einsatzes Signale an Module in den Autos. Dazu wird zu Beginn eines Einsatzes das Modul zusammen mit dem Blaulicht angeschaltet. Befindet sich ein Auto im näheren Umkreis des ESW, erhält der/die Fahrer:in eine Warnung, welche möglichst viele Sinne ansprechen soll, um die Aufmerksamkeit der Fahrerin / des Fahrers schnell zu erhalten. Dabei soll diese allerdings nicht überstimulierend wirken, um die Fahrenden nicht noch mehr zu stressen. Daher wird die Warnung durch ein akustisches Signal sanft eingeleitet, wobei Musik, die gerade läuft, leiser gedreht wird. Das Lenkrad beginnt, sanft zu pulsieren. Ist das Auto mit Ambient Lights ausgestattet, pulsiert dieses in blauer Farbe im gleichen Takt wie das Lenkrad. Ist die Aufmerksamkeit der Nutzenden so geweckt, erhalten diese, um das anonyme Mindset im eigenen Auto aufzubrechen, eine persönliche, voraufgezeichnete Nachricht der Rettungskraft. Danach übernimmt wieder das System die Situation, indem es den Nutzenden kurz und knapp weitere Informationen bereitstellt. Das kann beispielsweise die voraussichtliche Dauer sein, bis der RTW am Auto vorbei fährt. KALM hilft den Autofahrenden so, die Situation zu überblicken. Außerdem unterstützt es die Fahrenden dabei, ihre oder seine Pläne anpassen zu können, in dem beispielsweise Personen, welche auf die Fahrenden warten, telefonisch erreicht werden können. Das senkt das Stresslevel der Betroffenen enorm, da sie einerseits wieder

mehr Kontrolle über die Situation erlangen und andererseits durch klare Interaktion sehr viel besser über die Umstände informiert sind. Bei Bedarf können sich Autofahrende auch anzeigen lassen, wie eine Rettungsgasse gebildet wird, da sie die Regeln oftmals nicht verinnerlicht haben. Die Warnung auf dem Display wird ausgeschaltet, wenn der Einsatzwagen sich nicht mehr im Bereich des Autos befindet oder der/die Fahrer:in die Warnung per Knopf am Lenkrad ausschaltet.

5W-Fragen

WER	Autofahrende erhalten
WIE	eine persönliche Benachrichtigung sowie eine nachsichtige Handlungsanweisung
MIT WEM	von Einsatzwagenfahrenden
WOMIT	über KALM,
WARUM	sodass sie genügend Zeit haben, eine Rettungsgasse zu bilden, einen Blick über die aktuelle Situation zu bekommen und um ihre Pläne in Ruhe umstrukturieren zu können.

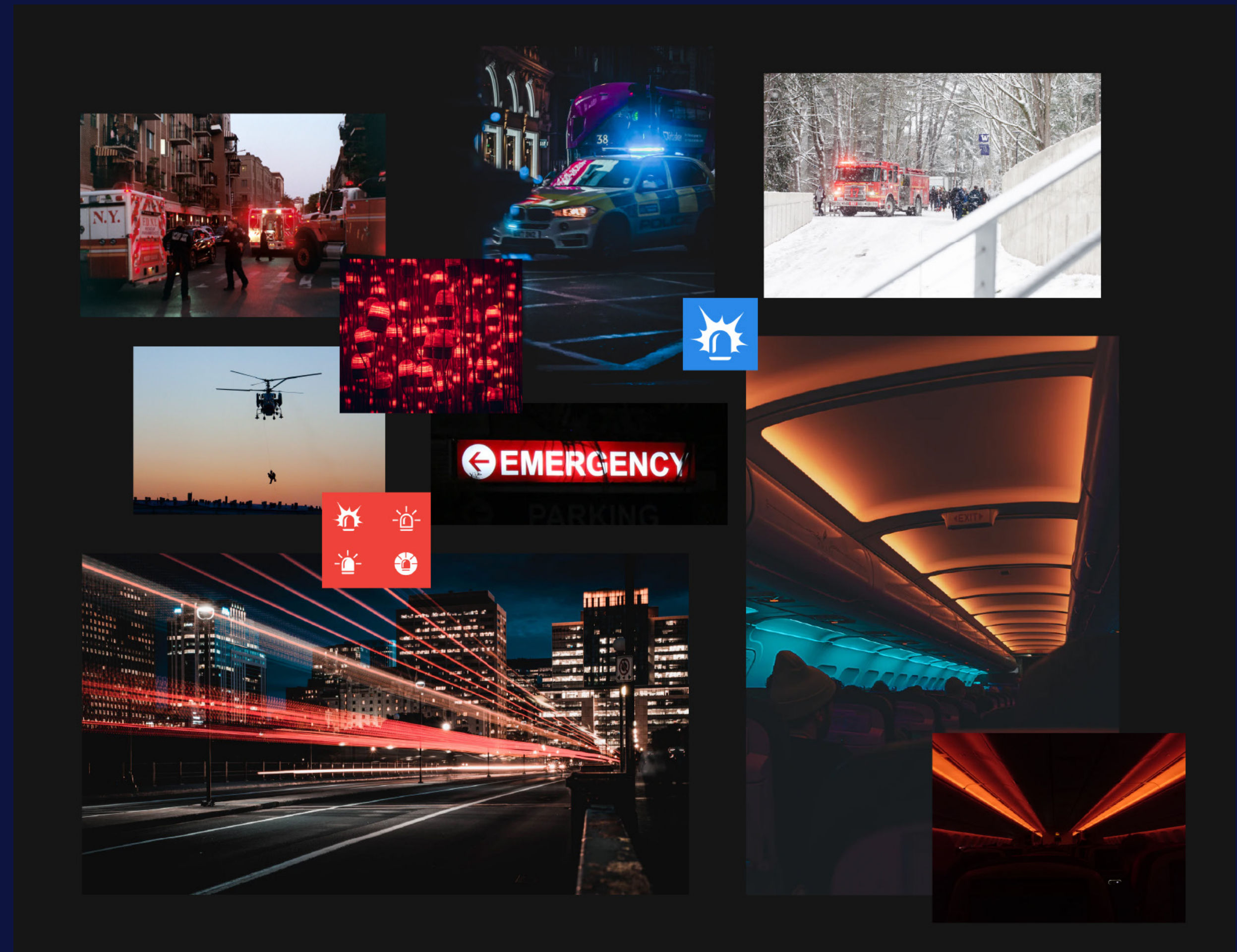
Value Proposition

Nutzergruppe Anforderung	Für Einsatzkräfte und Autofahrende, die ungehindert und stressfrei an ihrem Zielort ankommen wollen,
Produkt Kategorie, Kernnutzen	ist KALM eine Anwendung, die von Einsatzfahrenden genutzt wird, um Autofahrende frühzeitig darauf hinzuweisen, dass sich ein Einsatzwagen nähert und, dass sie eine Rettungsgasse bilden müssen.
Wettbewerb Differenzierung	Im Gegensatz zu den aktuellen Kommunikationsmitteln wie Blaulicht, Martinshorn und Riodurchsagen schaffen wir eine klare Interaktionsschnittstelle, die es Autofahrenden ermöglicht, frühzeitig auf Signale der Einsatzkräfte zu reagieren, sodass diese ungehindert an ihr Ziel kommen. Im Gegensatz zu Evam benachrichtigen wir nicht nur Autofahrende, die sich direkt vor, sondern auch die, die sich auf der weiterführenden Route des Einsatzwagens befinden.

Styleguide

Moodboard

Als Vorbereitung zur Gestaltung des Charakters der Anwendung haben wir ein Moodboard erstellt, das die Stimmung visuell zusammenträgt. Dieses Moodboard nutzen wir ebenfalls als gestalterischen Leitfaden für unsere Anwendung. Die roten und blauen Lichter des Verkehrs wirken in den gräulichen, farblich entsättigten Umgebungen der ausgesuchten Bilder nicht zu aufdringlich, jedoch stechen diese Farben durch ihre Saturation deutlich heraus. Ebenso stammen diese Farben ausschließlich von Lichtquellen. Dieses Motiv wollten wir auf das Design unserer Anwendung übertragen, da wir unter anderem auch mit Ambient Lights arbeiten. Im Ganzen soll unsere Anwendung die Aufmerksamkeit und den Fokus des Users erwecken, jedoch dabei eine beruhigende und klar strukturierte Art beibehalten.



Farbgebung

Farben können einen großen Einfluss auf die Gefühlswelt der Betrachter:innen haben, weswegen die Wahl dieser für KALM ungemein wichtig ist. Besonders für die Verwendung des Ambient lights und für die Darstellung auf dem Display wollten wir Farben nutzen, die die Intention unseres Projektes unterstützen, zugleich die Autofahrenden anspricht, ihre Aufmerksamkeit erlangt und ihnen vermittelt, dass sie in dieser Situation mit Rettungskräften zu tun haben.

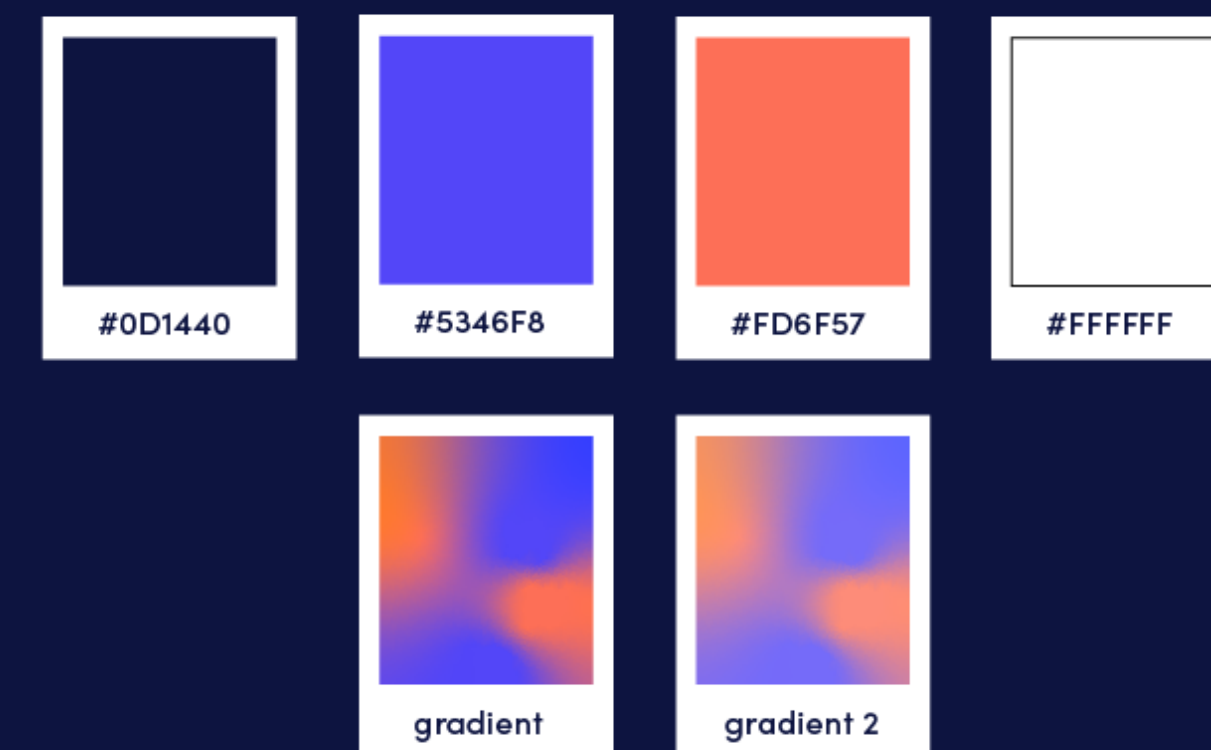
Mit diesen Beweggründen im Hinterkopf haben wir uns einerseits für ein kräftiges Blau als Hauptfarbe entschieden. Einerseits ist Blau im Straßenverkehr auch jetzt schon die Farbe der Einsatzkräfte, denn außer dem Blaulicht lässt sich dort keine blaue Beleuchtung finden. Ampeln sowie die Lichter der Autos und Baustellen scheinen dagegen Weiß, Grün, Gelb, Orange und Rot. In diesem Kontext die Farbe Blau zu nutzen ergibt also Sinn, da Autofahrende sie direkt mit Blaulicht in Verbindung bringen können und die Situation in einen ersten Kontext bringen können (Hoffmann, 2018).

Weiterhin hat Blau eine zuverlässige und seriöse Wirkung, was den Charakter von KALM als neutralem Mediator und Helfer in der Not unterstützt. Zusätzlich zu diesem Punkt schafft Blau eine räumliche Wirkung von Distanz und Entfernung, was das Gefühl der Anwender:innen unterstützt, frühzeitig gewarnt zu werden. Dies und die Tatsache, dass die Farbe Blau Betrachter:innen mit innerer Ruhe erfüllt und die Nerven

entspannt, trägt dazu bei, dass sie sich weniger hektisch und angespannt fühlen (Immos, o.D.).

In unserem Fall nutzen wir zwei unterschiedliche Blautöne, einen eher helleren und einen dunkel gehaltenen Ton, um in der Gestaltung mehr Varianz zu erreichen.

Als Kontrast und Akzentfarbe zur Grundfarbe Blau haben wir uns für einen hellen Orangeton entschieden, da Orange einen Komplementärkontrast zur Farbe Blau bietet. Dies sorgt für eine gegenseitige Steigerung beider Farben, d.h., dass sie besser zur Geltung kommen und somit durch die Nutzenden besser wahrgenommen werden können (Lehrer:innenfortbildung Baden Württemberg, o.D.). Da es bei unserer Anwendung wichtig ist, die Aufmerksamkeit der Autofahrenden möglichst schnell zu erreichen, haben wir uns dazu entschieden, diesen Effekt zu nutzen.



Typografie

Neben der Farbgestaltung haben wir uns auch Gedanken darüber gemacht, wie wir die Anwendung typografisch gestalten möchten. Dazu lässt sich im Vorfeld sagen, dass die Schriften, die wir auswählen, sich zwar in unserem Prototypen und der Dokumentation wiederfinden, im fertigen Produkt in den Autos der verschiedenen Hersteller allerdings keine Anwendung finden würden, da diese hauseigene Schriften verwenden, durch welche unsere ausgesuchten Schriften dann ersetzt würden.

Nichtsdestotrotz haben wir für unseren Prototypen Schriften ausgewählt, welche unseren Anforderungen entsprechen: Die gewählten Schriften müssen allem voran für (kleine) Bildschirme geeignet sein, da sie auch auf den Mittelkonsolendisplays der Autos noch gut lesbar und schnell erfassbar sein müssen. Hier gelten serifenlose Fonts für kleine Displays als besser lesbar als Serifenschriften (Beinert, 2019). Außerdem sollen die Fonts ein modernes Bild verkörpern.

Als Font für Fließtexte in der Dokumentation bzw. kurze Sätze auf den Displays haben wir uns für die Sofia Pro im Schnitt extra light entschieden. Sie wirkt durch ihre runden, serifenlosen und sehr geometrischen Buchstaben modern und ist durch die, im Gegensatz zu anderen Fonts, höhere x-Höhe auch in sehr kleinen Schriftgrößen gut lesbar.

Für Überschriften haben wir uns für die Franklin Gothic ATF im Schnitt heavy entschieden. Durch den gewählten Schriftschnitt wirkt die Franklin Gothic deutlich raumeinnehmender als die Sofia Pro und schafft es so, die Aufmerksamkeit der Betrachtenden auf sich zu ziehen. Sie bildet also einen Kontrast zu den anderen Texten und lässt die gesamte Gestaltung interessanter und harmonischer wirken. Die Franklin Gothic erfüllt auch unsere anderen Anforderungen, da sie jederzeit klar lesbar ist und ihr häufiger Einsatz über die Zeiten hinweg zeigt, dass sie ein zeitloses Bild verkörpert (Schriftgestaltung.com, o.D.).

MAKE YOUR PRESENCE HEARD

Franklin Gothic

Make your presence heard

Sofia Pro

Sounddesign

Unsere Anwendung warnt den User mit verschiedenen Sinneseindrücken, darunter auch Sound, da dies beim Autofahren am schnellsten aufzugreifen ist und am wenigsten störend ist. So ist Folgendes für das Sounddesign geplant:

- Ein kurzer, ruhiger und auffälliger, aber nicht penetranter Benachrichtigungston, der die erste Aufmerksamkeit auf sich rufen soll
- Ein ähnlicher Ton als Audio-Feedback nach der Bestätigung der Nachricht

Timing

Um die Autofahrenden in der stressigen Situation nicht völlig zu überlasten und um keine Reizüberflutung zu provozieren, achten wir bei der Umsetzung besonders auf das Timing, wie und wann Nutzende gewarnt werden. Die Nutzenden sollen nicht mit Text und überflüssigen Informationen überhäuft werden, deshalb versuchen uns bei den Benachrichtigungen kurz zu fassen. Dadurch schaffen wir nicht nur eine vertraute und ruhige Atmosphäre, sondern deeskalieren die Situation und beruhigen die Nutzenden, sodass sie sich kurz entspannen und im Nachhinein ihre Pläne umstrukturieren können.

Screenesign

Um Autofahrende nicht abzulenken, sind unsere Screendesigns schlicht und auf die wichtigsten Elemente reduziert. Daher befindet sich auf dem ersten Screen ausschließlich unser Logo auf dunklem Hintergrund. Sobald sich ein Einsatzwagen nähert ändert sich die Animation des Logos, um die Aufmerksamkeit des Nutzenden auf sich zu ziehen, daraufhin wird sofort die Nachricht der relevanten Informationen angezeigt.

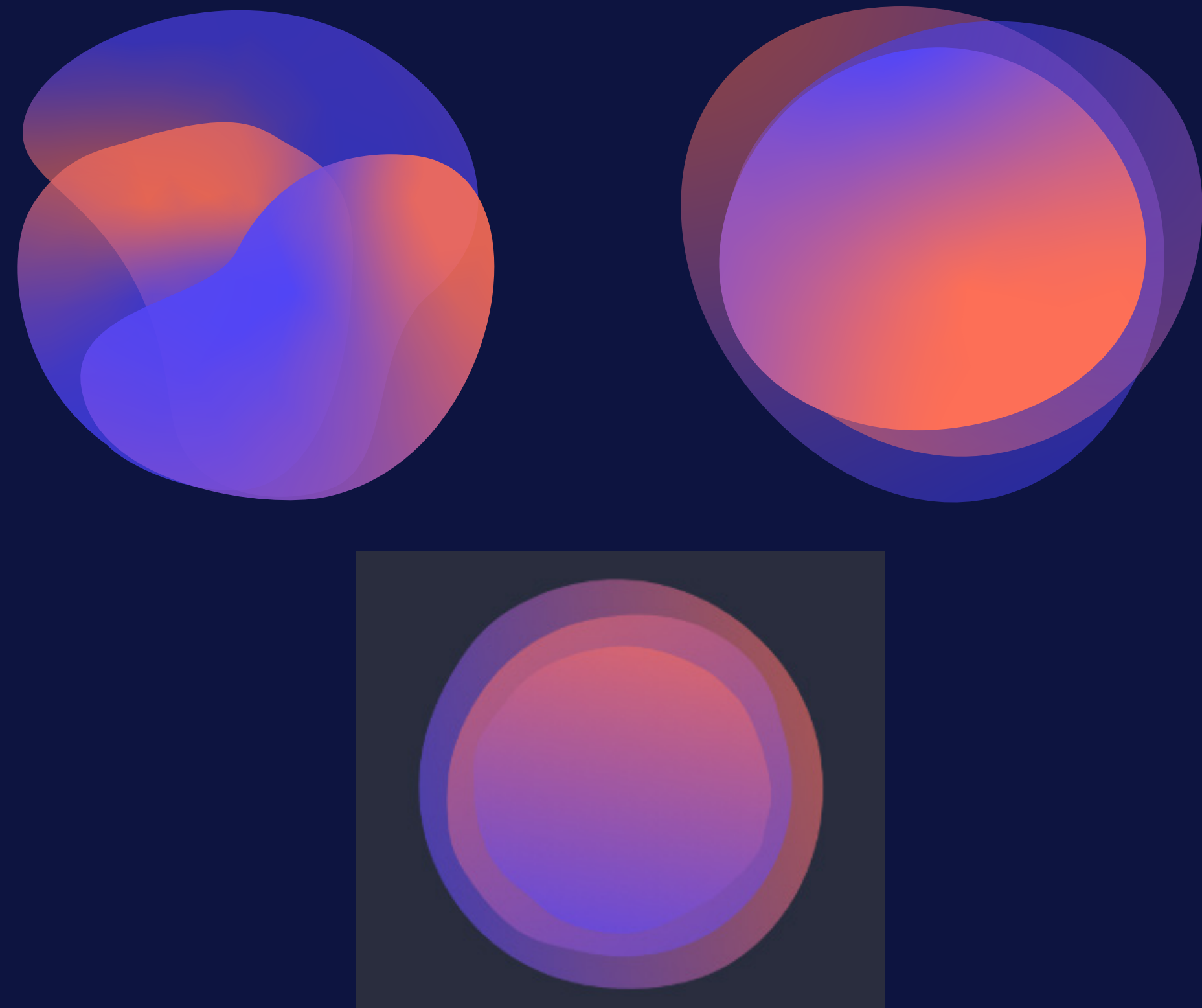
Um das Lesen auch bei verschiedenen Lichtverhältnissen zu optimieren, haben wir einen Light und Dark Mode entwickelt. In beiden Versionen beschränken wir uns auf einfache Grundformen.

Das helle Design ist für die Fahrt am Tag gedacht. Durch den in weiß gehaltenen Hintergrund soll der Bildschirm auch bei hoher Sonneneinstrahlung gut lesbar sein. Die farblichen Akzente, sowie die bunte Umrandung helfen den Text einzurahmen und so den Blick auf das Wesentliche zu lenken.

Um Autofahrende bei Nacht nicht zu blenden, bieten wir ebenfalls ein dunkles Design. Der Hintergrund bleibt wie beim ersten Screen weiterhin dunkelblau und die Schrift weiß. Wichtige Elemente werden weiterhin farblich markiert und durch die bunte Umrandung des ganzen Bildschirms in den Fokus gerückt.

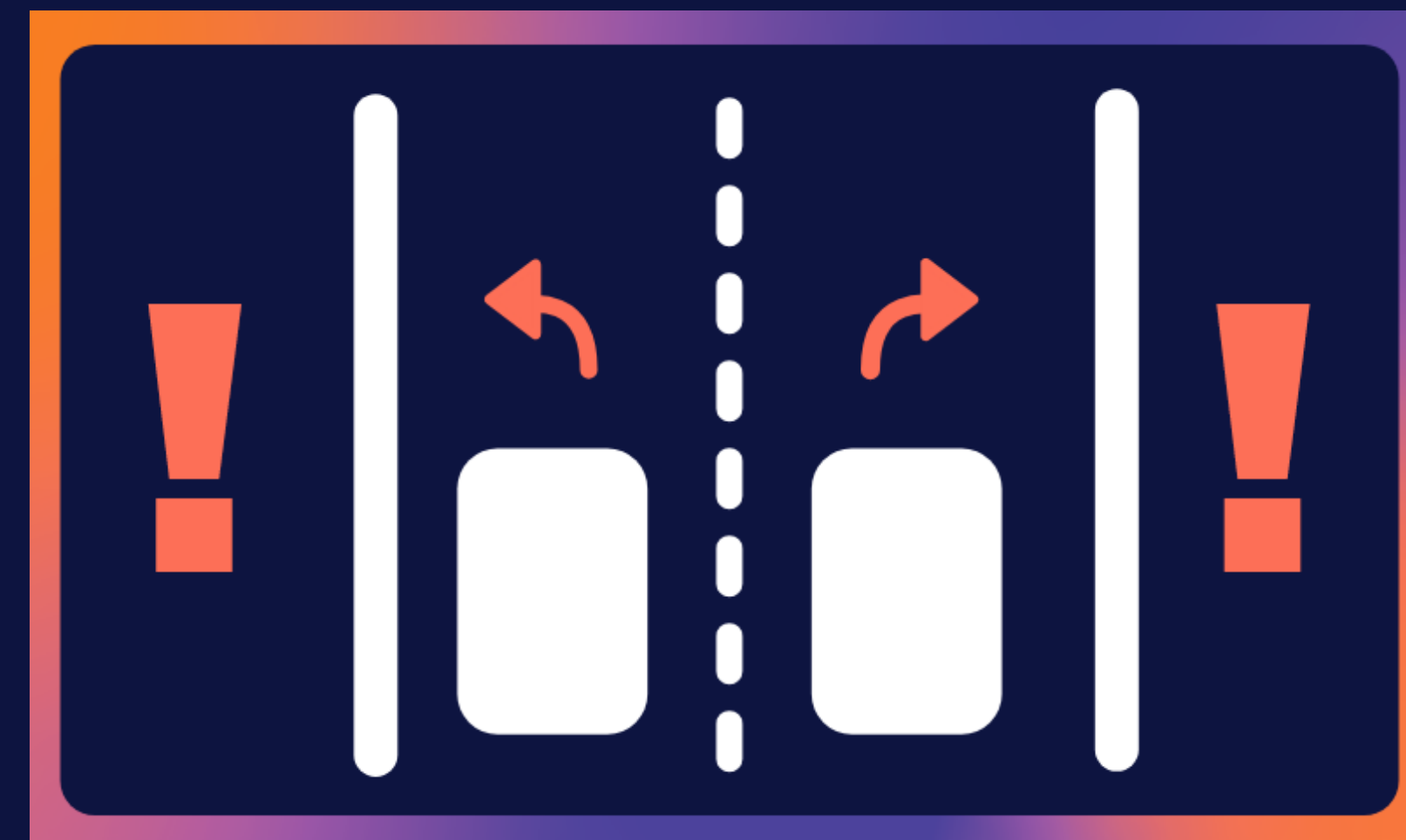
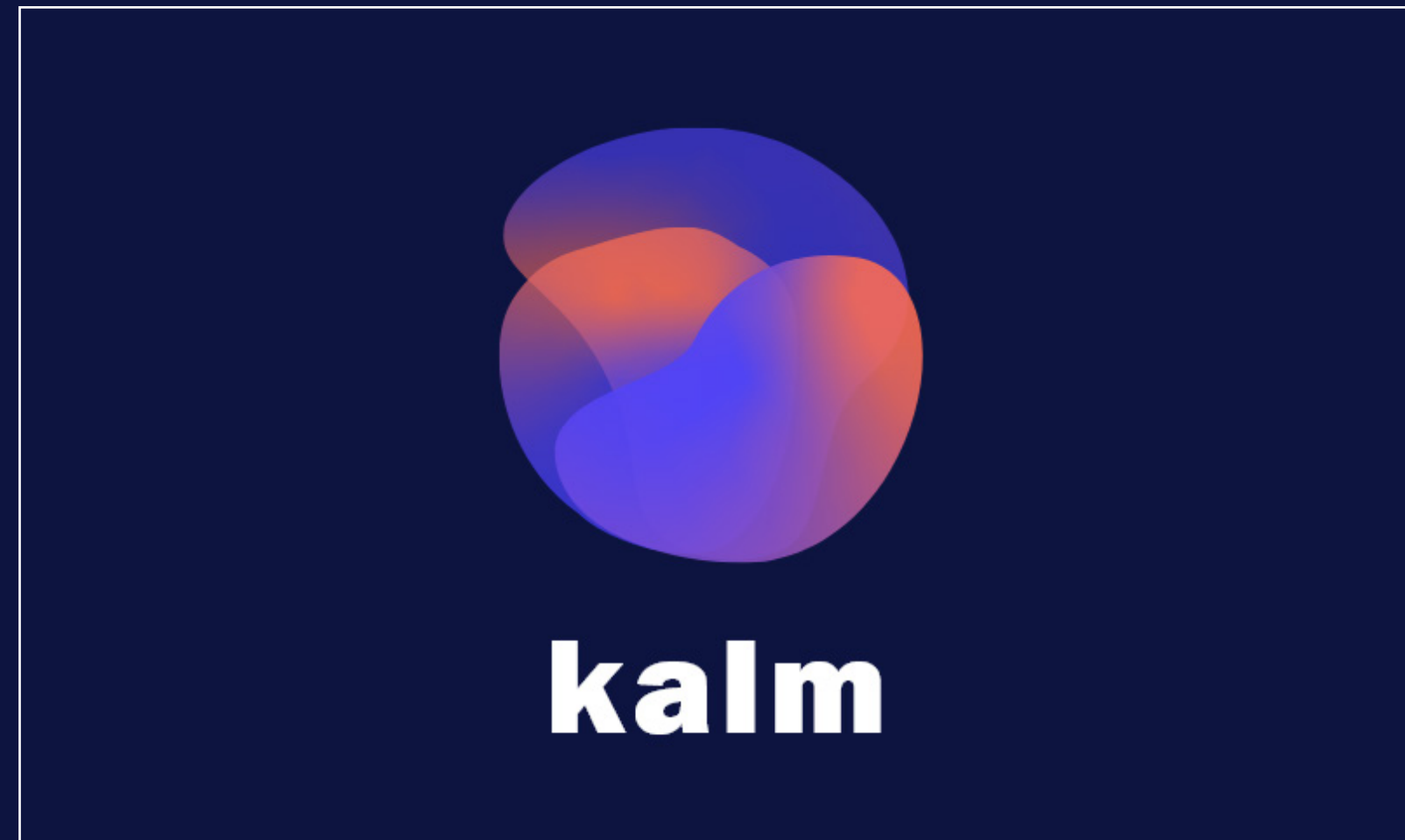
Animation

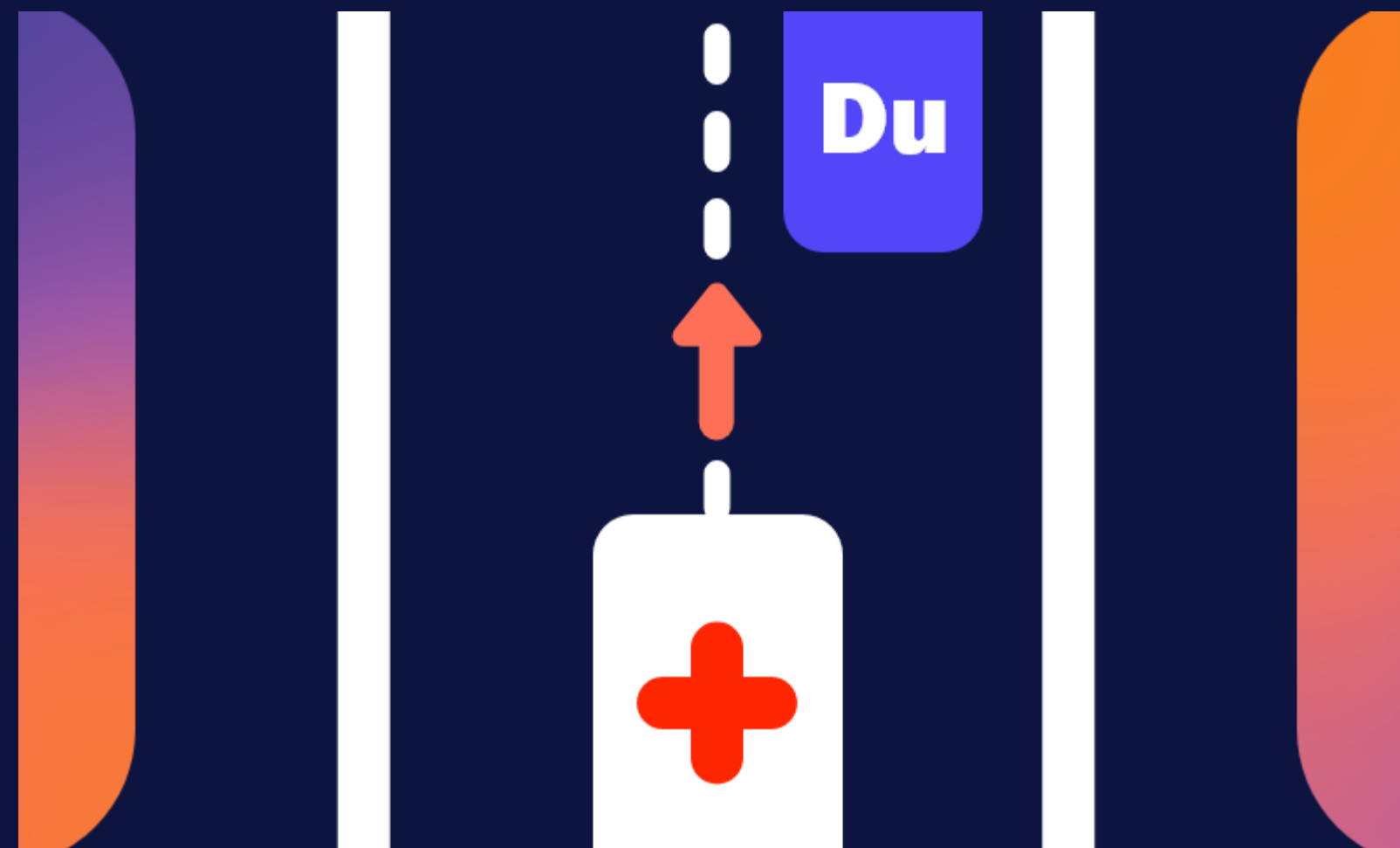
Die Animation unseres Logos soll Aufmerksamkeit schaffen aber gleichzeitig beruhigend wirken. Kreisformen wirken weich und von Natur aus beruhigend. Langsame Bewegungen wirken entspannend auf Körper und Geist.



Screens

Dark Mode





Light Mode



kalm



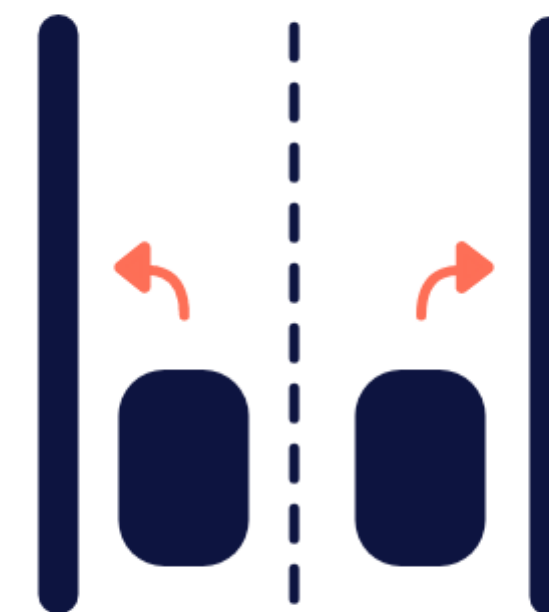
**Bitte hilf Peter,
indem du eine
Rettungsgasse
bildest.**



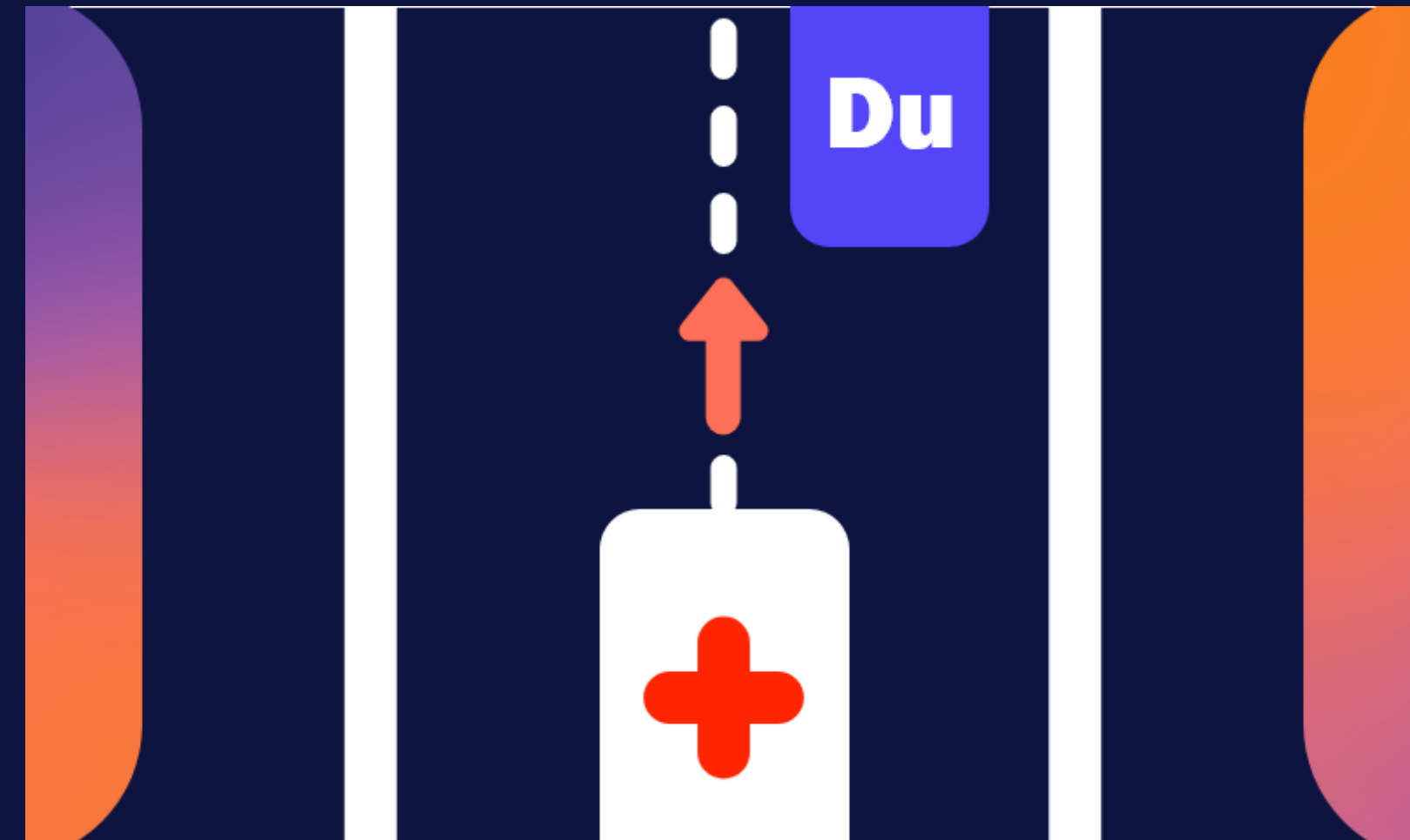
Peter, Sanitäter



**Wir müssen gleich an
dir vorbei fahren!**



**Bist du beiseite
gefahren?**



**Danke für deine Kooperation!
Damit rettest du Leben!**



Weitere Hilfe?



Nein, danke.

Modell im Einsatz



Autointerieur Modell von <https://hum3d.com/3d-models/mercedes-benz-gle-class-with-hq-interior-2019/>



kalm

SET VOL PASSANGER AIR BAG A/C



The image features a dark blue, curved shape on the left side, resembling a stylized wave or a partial circle. To its right, there are three overlapping circles of varying sizes. The circles are colored in a gradient: the largest circle on the right is a vibrant orange-red, the middle circle is a deep purple, and the smallest circle on the left is a light lavender. The word "Technik" is written in a bold, white, sans-serif font, centered within the dark blue shape.

Technik

Hardware

Nachdem die Leitstelle einen Notfall erhalten hat, informiert sie Einsatzwagenfahrende, die sich in der Nähe befinden. Parallel dazu werden Daten an Rescuetrack gesendet, die dem Einsatzwagen eine entsprechende Route zukommen lassen.

Rescuetrack ist eine Firma, welche Systemlösungen für Einsatzkräfte anbietet, darunter auch Navigation. Nachdem wir Rescuetrack kontaktierten, hat sie uns netterweise eine Schnittstelle in Form einer REST API zur Verfügung gestellt.

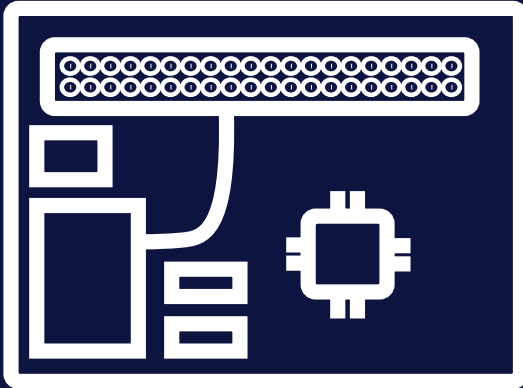
Sobald die Einsatzwagenfahrenden die Route erhalten, aktiviert sich, zusammen mit dem Blaulicht, unser System, sodass sie keine Zeit verlieren.

Der Sender im Einsatzwagen besteht aus einem RPi, der mit einem GPS Modul + Keramikantenne und einem 2.4GHz Funkmodul ausgestattet ist. Außerdem ist der Pi noch mit einer Status LED sowie einem Knopf verkabelt. Sobald das System aktiviert ist, werden Teile der Route und eine Einsatzid als JSON per Funk gesendet. Autos im Umkreis der Funkverbindung empfangen das JSON.

Die Daten werden ebenfalls von einem RPi über ein GPS Modul + Keramikantenne mit dem eigenen Standort abgeglichen. Wenn die Autos sich zwischen zwei Routenpunkten befinden, werden sie über verschiedene Mittel gewarnt, die je nach Ausstattung des Fahrzeugs variieren.

Für den Prototypen verwenden wir ein 7" LCD Touch Display als Mittelkonsole, ein Funkmodul sowie einen Vibrationsmotor, eine LED, welche die Ambient Lights der Autos simulieren soll und einen Schalter für das Lenkrad.

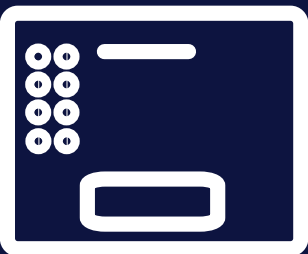
Hardware Komponente



Raspberry Pi 4
Model B



GPS Modul +
Keramikantenne



NRF24 2.4 GHz
Funkmodul



LED



RGB LED



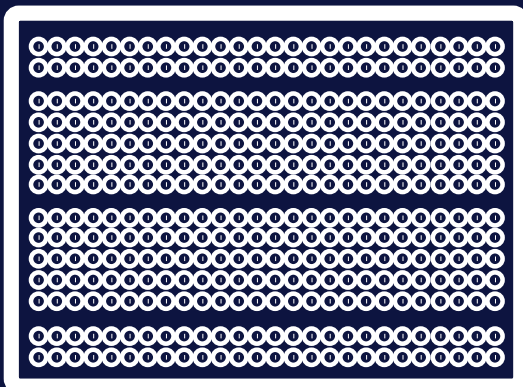
Knopf



Vibrationsmotor



Raspberry Pi 7" LCD
Touch Display



Breadboard



Widerstände



Jumperkabel

- GPIO 21 (PCM_DOUT)

GPIO 20 (PCM_DIN)

GPIO 16

Ground

GPIO 12 (PWM0)

Ground

GPIO 1 (ID_SC)

GPIO 7 (CE1)

GPIO 8 (CE0)

GPIO 25

Ground

GPIO 24

GPIO 23

Ground

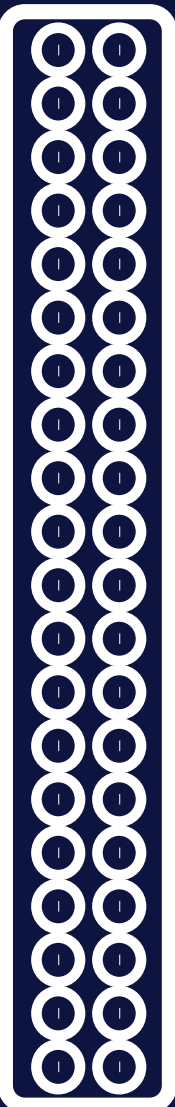
GPIO 18 (PCM_CLK)

GPIO 15 (RXD)

GPIO 14 (TXD)

Ground

5V power

5V power
- 
- Ground

GPIO 26

GPIO 19 (PCM_FS)

GPIO 13 (PWM1)

GPIO 6

GPIO 5

GPIO 9 (ID_SD)

Ground

GPIO 11 (SCLK)

GPIO 9 (MISO)

GPIO 10 (MOSI)

3V3 power

GPIO 22

GPIO 27

GPIO 17

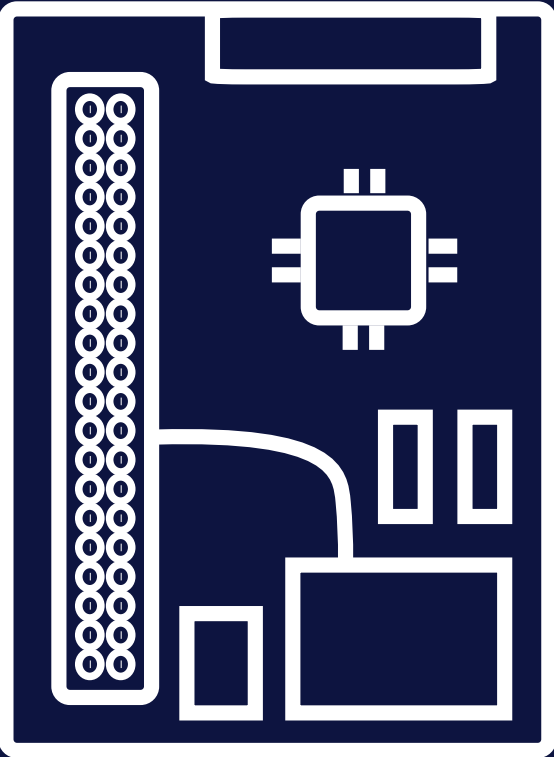
Ground

GPIO 4 (GPCLK0)

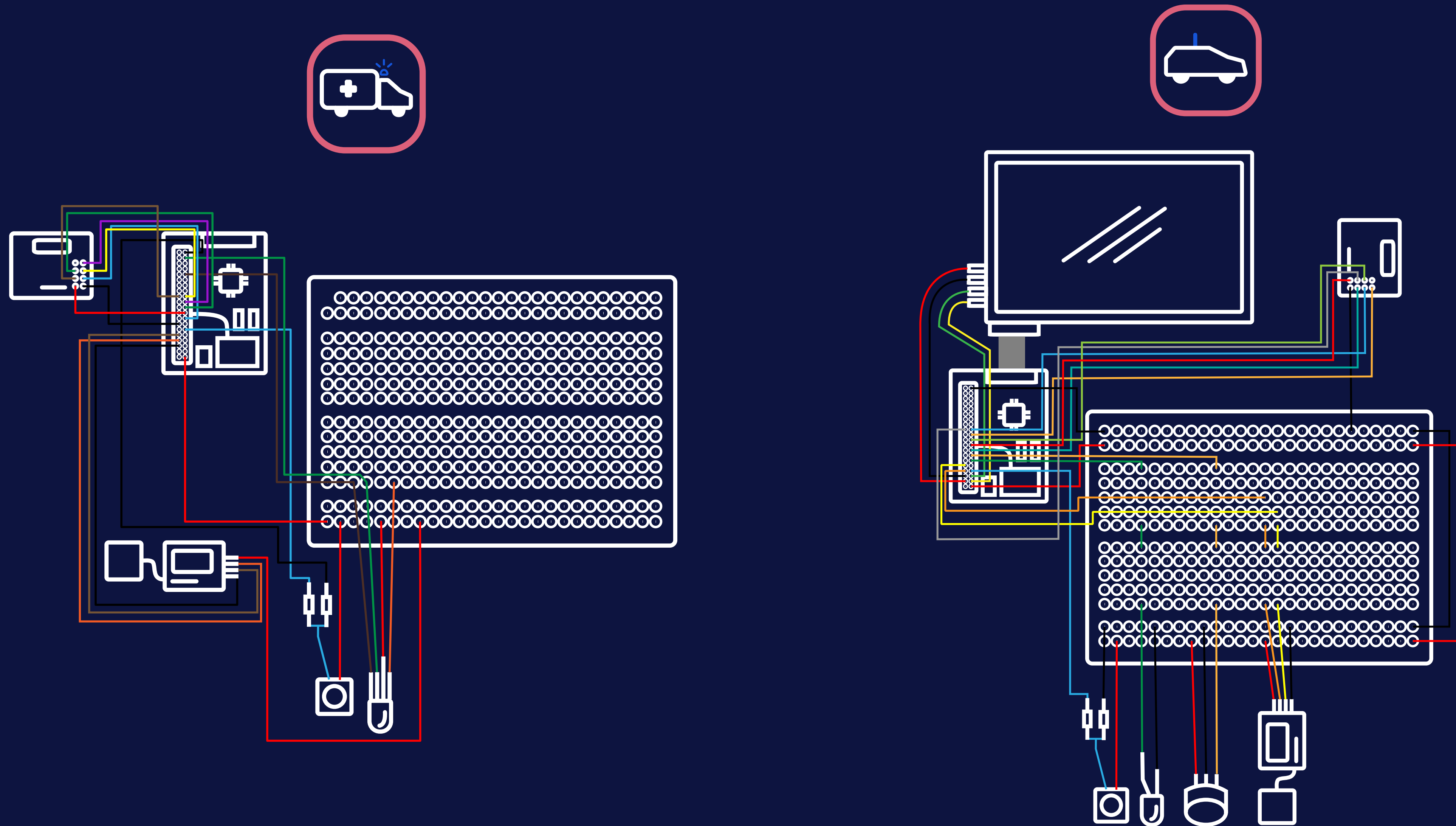
GPIO 3 (SCL)

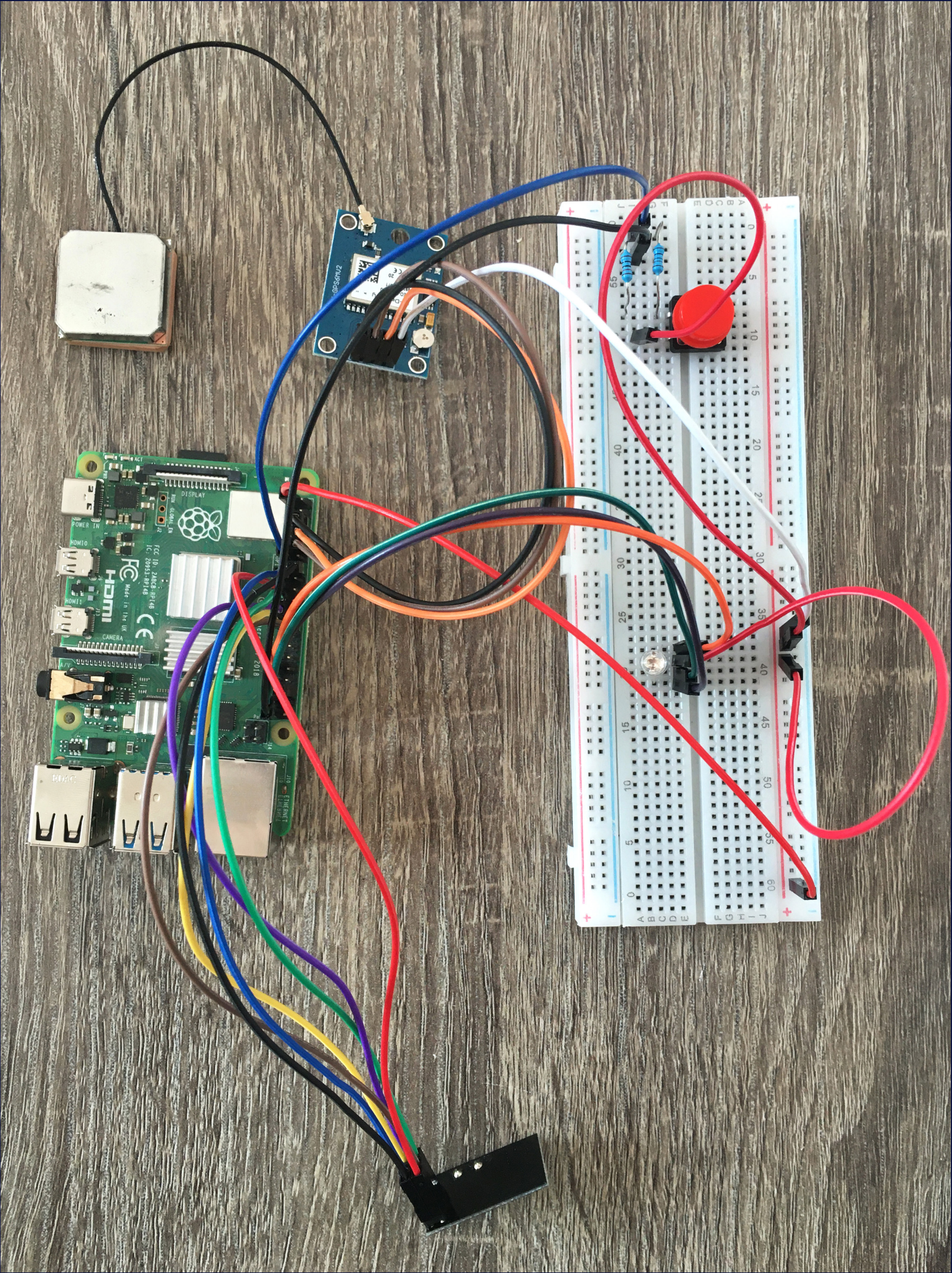
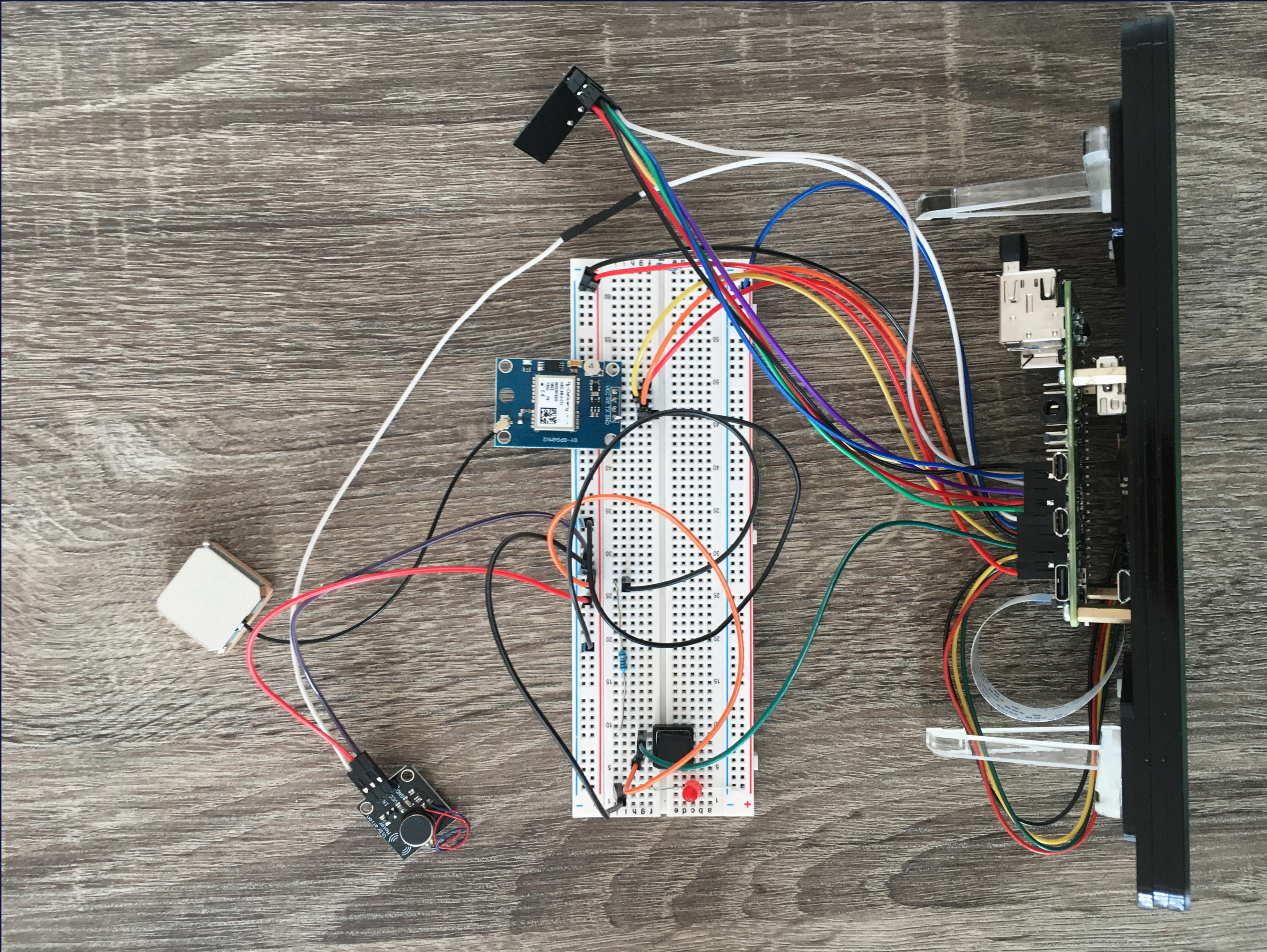
GPIO 2 (SDA)

3V3 power



Hardware Komponente Zusammensetzung



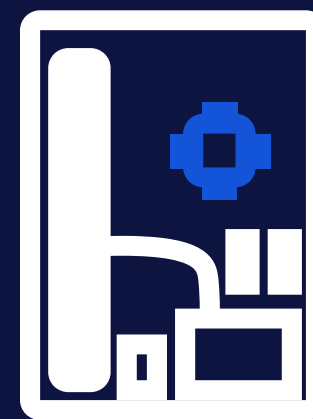
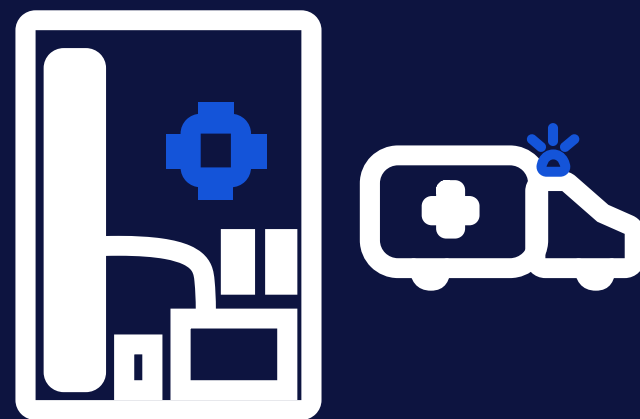


Software

Für das Backend benutzen wir node.js und für das Frontend verwenden wir HTML+CSS+JS. Das machen wir, da wir diese Programme aus diesem und aus den letzten Semestern bereits kennen.

Wir entwickeln hierbei drei verschiedene Software-Komponenten:

Erstens den RTW Node Server, der auf dem RPi im Rettungswagen läuft, zweitens den Auto Node Server, der auf dem RPi im Auto läuft und drittens die Website im Auto, die auf dem RPi Screen im Auto (auf dem Bildschirm der Mittelkonsole des Autos) angezeigt wird.



Radiodaten des RTW Node

```
{
  // random id, die dem RTW beim Start des RPi zugewiesen wird
  "evId": "ee1b4f55",

  // Die personalisierte Nachricht von dem RTW
  "message": "Hi ich bin Peter Sanitäter. Bitte lass mich vorbei",

  // Der momentane Standpunkt des RTWs
  "currentLocation": {
    "lat": 49.8721511,
    "lng": 8.6431074
  },

  // Liste von den nächsten Routenpunkten in der Reihenfolge wie sie abgefahren werden
  "route": [
    {
      "lat": 49.8722625,
      "lng": 8.6446182
    }
  ]
}
```

Link zu dem Github Repository:

<https://github.com/kitxsou/KALM>

Softwarearchitektur

ev_node

- Ev steht für „emergency vehicle“ = Einsatzwagen
- Aktualisiert alle zehn Sekunden die momentane Route des Einsatzwagens mithilfe der Rescuetrack API
- Versendet sekundlich die momentane Route, sowie die evID und die customMessage übers Radio an die umliegenden Autos

car_node

- Wartet bis es Daten eines umliegenden Einsatzwagens empfängt
- Sobald es die ersten Daten eines Wagens empfängt, aktiviert car_node die Lichter-, Sound- und Vibrationseffekte im Auto und stellt die empfangenen Daten per HTTP API zur Verfügung.
- Außerdem stellt car_node einen Webserver mit car_web bereit.
- Desweiteren hat car_node eine Route, mit der der/die Autofahrende bestätigen kann, dass er/sie beiseite gefahren ist.

car_web

- Zeigt bei default die Logoanimation an
- Alle zehn Sekunden holt es von car_node den aktuellen Status ab.
- Ist ein Einsatzwagen in der Nähe wird die Benutzerbenachrichtigung initiiert und nacheinander die verschiedenen Screens durchlaufen.
- Bei dem Screen mit dem Bestätigungsbutton wird gewartet, bis der/die Nutzende das Beiseitefahren bestätigt. Daraufhin wird die Bestätigung an die car_node API weitergeleitet.
- Nachdem alle Screens durchlaufen sind, kehrt man wieder zur Logoanimation zurück bis ein weiterer Einsatzwagen kommt.

Datentransfer und Datensicherheit

Datentransfer

Über ein Funkmodul werden Daten an alle im Umkreis liegenden Autos gesendet. Die Reichweite ist groß genug, um möglichst viele relevante Personen zu erreichen und klein genug, um die Personen, die nicht betroffen sind, nicht auch anzusprechen (2.4GHz einige 100 Meter). Damit kein Datenmissbrauch stattfindet, werden Daten verschlüsselt.

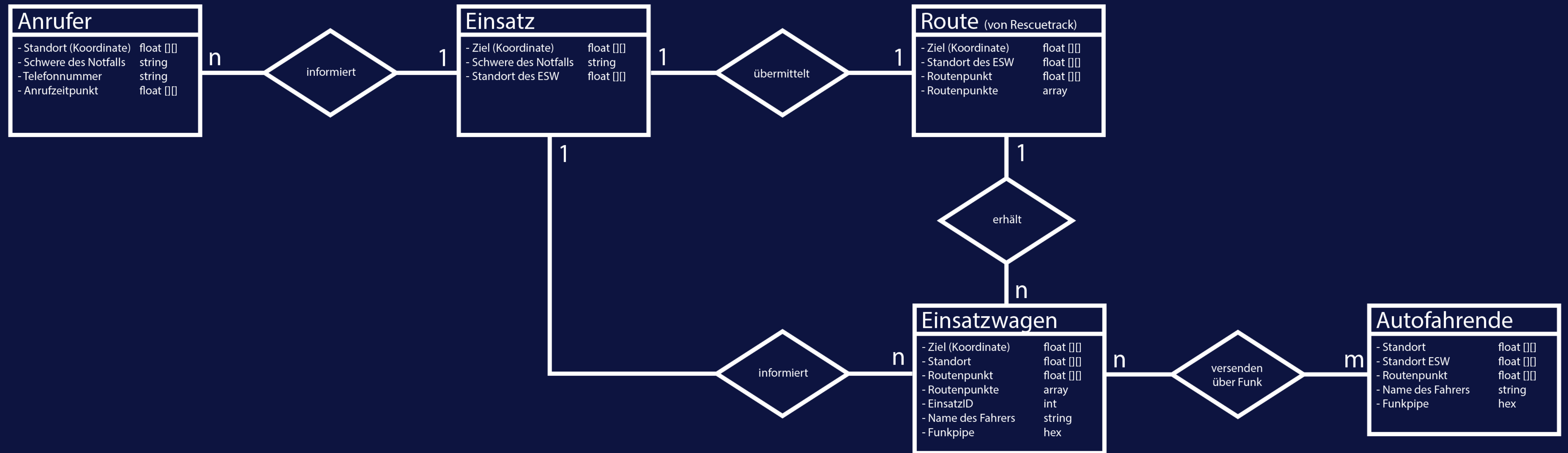
Die Route wird in kleinen Abschnitten aufgeteilt und nur an die Autofahrenden in diesem Abschnitt übermittelt. Außerdem können Einsatzkräfte das Senden der Routendaten über einen Knopf beenden.

Datensicherheit

Damit man unterscheiden kann, ob man schon Informationen von einem Einsatzfahrzeug erhalten hat bzw. man zwischen verschiedenen Einsatzfahrzeugen unterscheiden kann, braucht man eine Einsatz-ID. Diese ist ein zufällig generierter Code, welcher für jeden Einsatz neu generiert wird. Das ist auch nützlich, wenn noch Fahrzeuge hinterherkommen. Die ID gibt keine persönlichen Informationen über die Einsatzkräfte weiter.

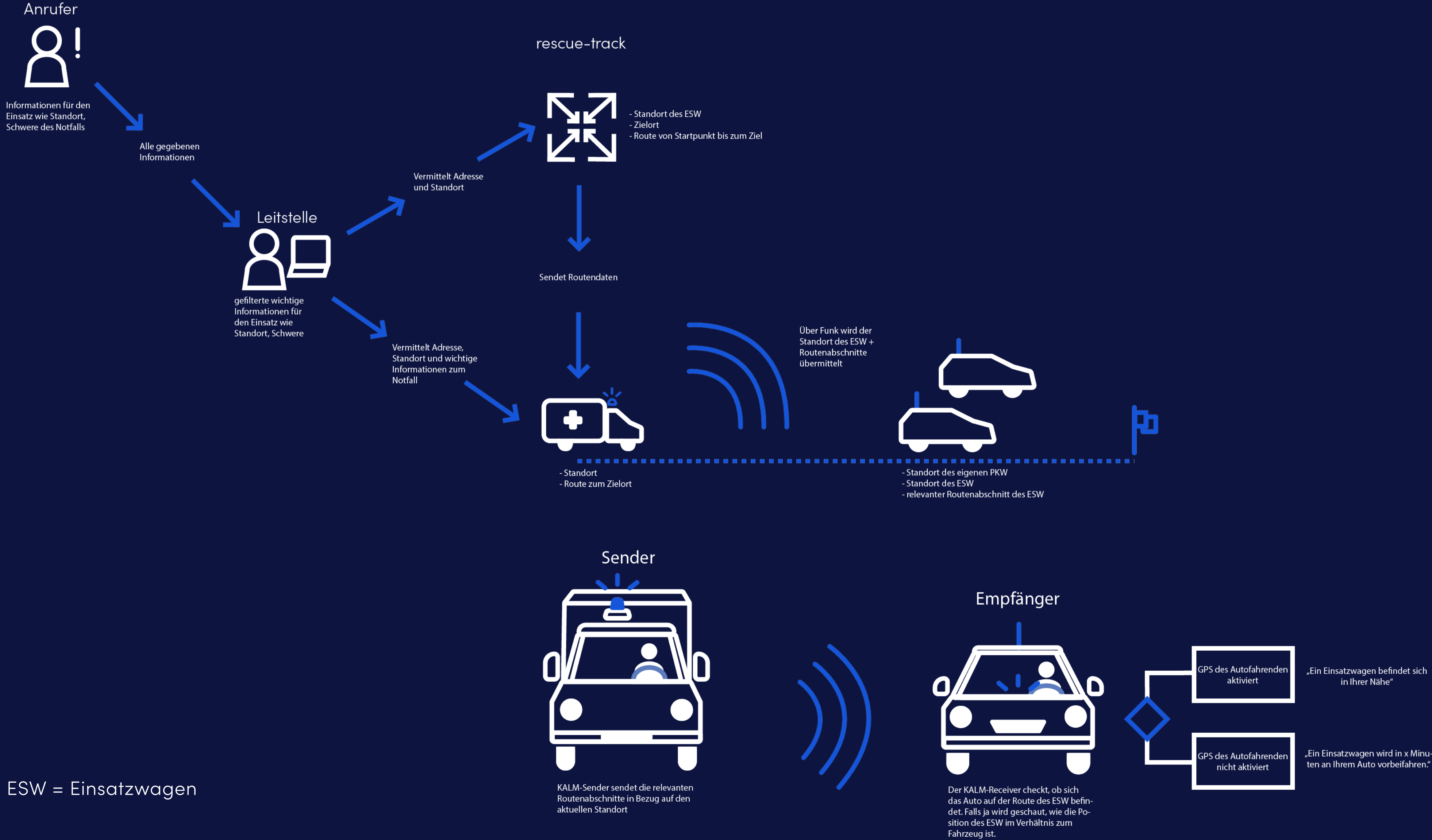
Der Standort der Autofahrenden wird nicht weitergegeben und sie können das GPS Modul jederzeit deaktivieren. Das führt allerdings dazu, dass sie eine angepasste Nachricht bekommen, weil man nicht mehr berechnen kann, ob sie sich auf der tatsächlichen Route befinden.

Entity-Relationship Model



ESW = Einsatzwagen

Systemdiagramm

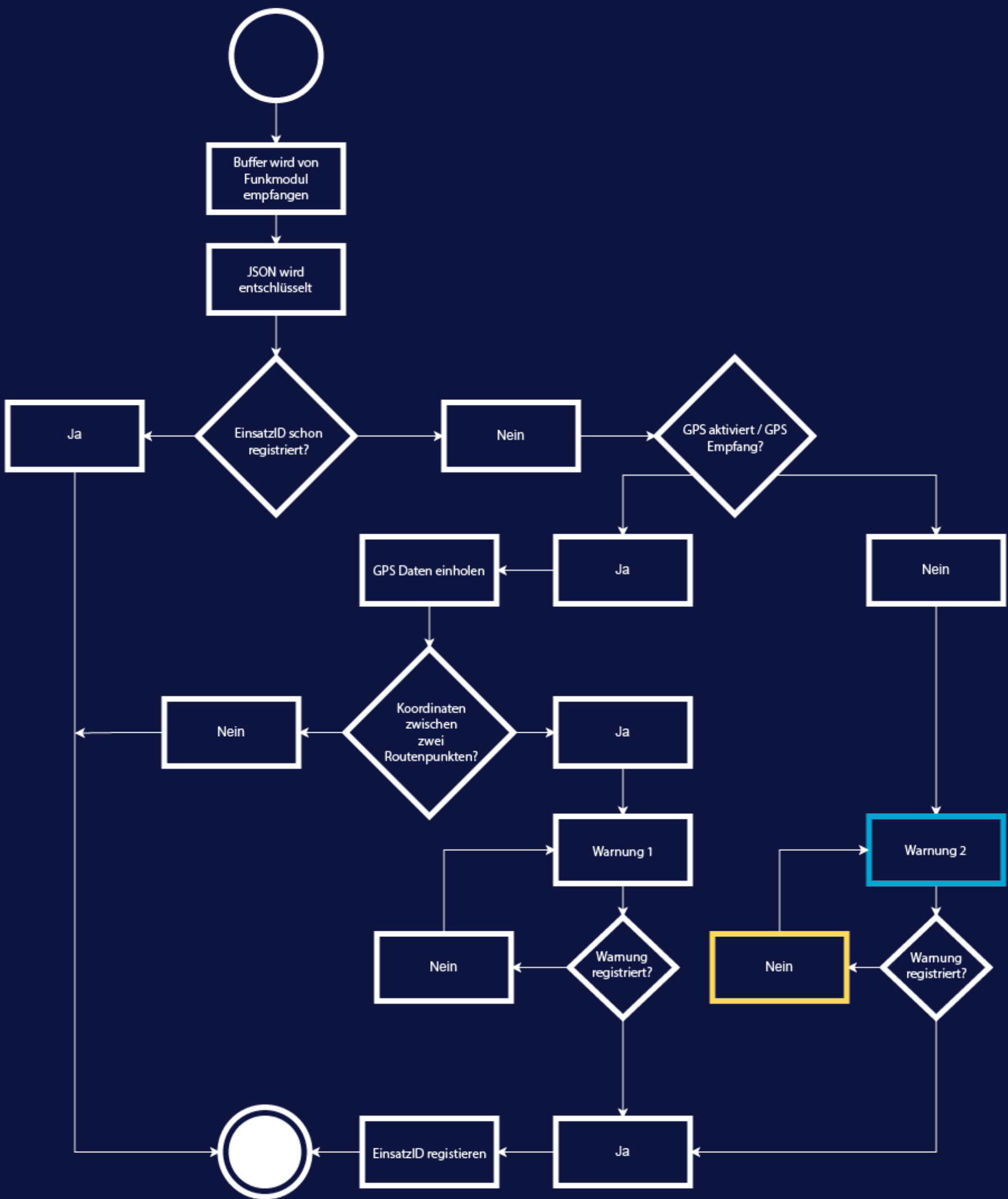
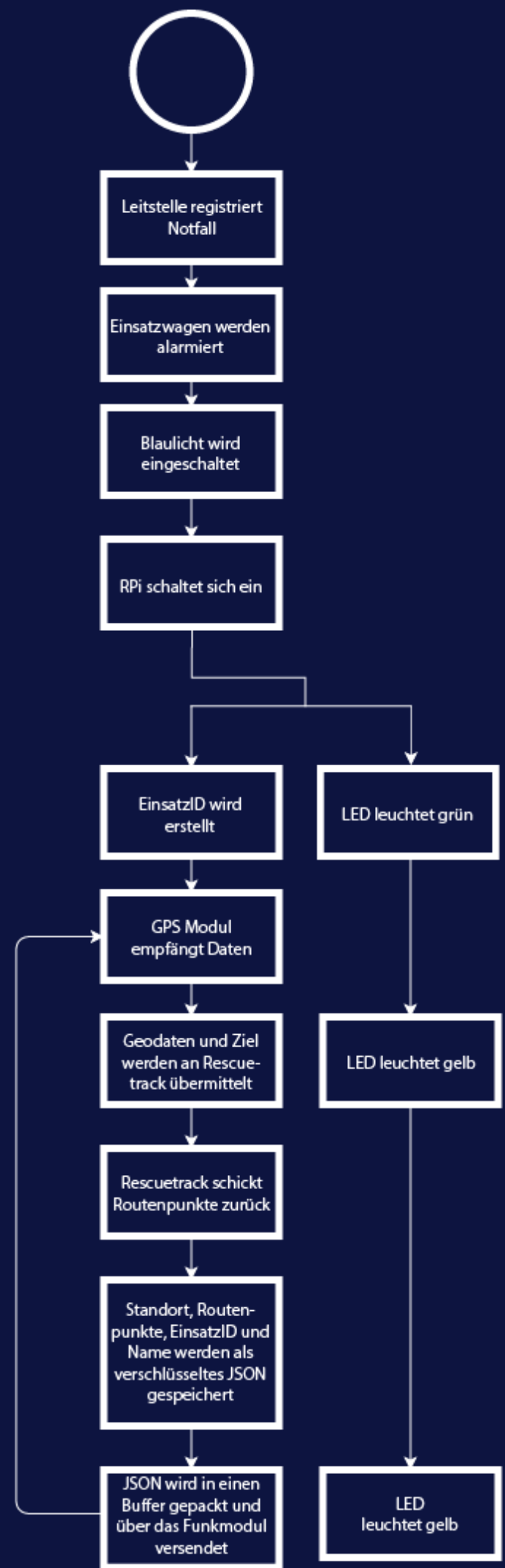


Swimlane-Diagramm



ESW = Einsatzwagen

Ablaufdiagramm



ESW = Einsatzwagen

Warnung 1

ESW kommt sicher
ESW kommt aus Richtung xy
ESW kommt in xy Minuten

Warnung 2

ESW kommt eventuell und ist in der Nähe

Proof of Concept

Unser Proof of Concept besteht aus den folgenden Knackpunkten:

- Routenabgleich und Routenplanung mit Rescue-Track
- Radio funkt nur in einem bestimmten Umkreis des Einsatzwagens
- Module besorgt und getestet und mit unserer Software ansprechbar

Wir besitzen das Nötige, um die GPS-Daten mit den Routen abzugleichen. Außerdem haben wir die Routenplanungs-Funktion durch die Zusammenarbeit mit Rescue-Track abgedeckt. Ebenso können wir beweisen, dass das Radio nur in einem bestimmten Umkreis des RTWs funkt. Alle weiteren erforderlichen Module haben wir besorgt und getestet und diese sind auch mit unserer Software ansprechbar.

Zunächst haben wir uns den 433mHz Funksensor angeschafft, jedoch fanden wir heraus, dass die Reichweite damit zu begrenzt und die Datenübertragungsmenge zu gering war. Daraufhin haben wir uns das 2.4GHz Funkmodul besorgt und verwenden es dementsprechend für den Prototypen. Nähere Details zum Funkmodul finden Sie in der Sektion „Herausforderungen in der Technik“ (S.43).

Herausforderungen in der Technik

Herausforderungen bei der Hardware

Während der Umsetzung unseres Prototypen sind wir auf diverse Herausforderungen gestoßen, die wir hier zusammen mit den Learnings, die sich daraus ergeben haben, gerne näher erläutern möchten.

Die erste Herausforderung, die wir überwinden konnten, war die Wahl des Funkmoduls. Zunächst haben wir 433 MHz Module getestet, wobei schnell deutlich wurde, dass diese die benötigten Informationen nicht übertragen konnten, da die mögliche Datenmenge sowie Reichweite nicht zur Übertragung ausgereicht haben. Daraufhin haben wir uns dazu entschieden, mit 2.4 GHz Modulen zu arbeiten, mit deren Hilfe wir theoretisch unser gesamtes JSON Objekt übertragen könnten. Praktisch stellte uns dieses Modul allerdings vor sehr große Herausforderungen, die bis zum Ende des Semesters trotz vielfältiger Ansätze nicht gelöst werden konnten. Die Möglichkeiten, die wir versucht haben zu nutzen, werden wir nun näher erläutern.

Zunächst war geplant, wie bei den anderen Modulen auch, eine Node Library zu nutzen, da wir diese mit Javascript verwenden können und diese Sprache aus den letzten Semestern und den neuen Inputs dieses Semester gut kennen. Leider trafen wir hier auf die ersten Probleme. Wir haben es zunächst mit Hilfe folgender Erklärung: <https://www.npmjs.com/package/nrf24> geschafft, uns das benötigte JSON Objekt zuschicken zu können. Bei unserem nächsten Treffen, bei welchem

wir weitere Funktionen zu unserem Prototypen hinzufügen wollten, funktionierte der vorher tadellos funktionierende Code nicht mehr. Weder beim Code noch beim Aufbau des Prototypen hatte sich etwas geändert. Beim Senden trat nun mehr ein dauerhafter Error („failed to send“) auf, da der Success-Callback nicht mehr erfolgreich war. Beim Empfänger wurden uns zeitweise auch Hardware Failures durch den Code ausgegeben, allerdings nur dann, wenn wir den Wert „Autoack“ auf true setzen. Wir haben dann die komplette Verkabelung sowie die Module an sich ausgetauscht, was leider keinen Erfolg brachte. Insgesamt haben wir sechs unterschiedliche Funkmodule ausprobiert, was uns zu der Vermutung führt, dass die Hardware entgegen des geworfenen Errors nicht das Problem ist. Ein weiteres Problem des Empfängers ist, dass dieser nicht von uns interpretierbare Daten erhält, und auch diese lassen sich nicht sicher reproduzieren. Mal erhalten wir sehr viele Einsen, mal leere Buffer, die sehr viele Nullen enthalten und mal leere Arrays. Unsere Vermutung, die wir leider nicht überprüfen konnten, da uns hierzu die technischen Möglichkeiten in der begrenzten Zeit gefehlt haben, ist, dass ein anderes Gerät in der Nähe auf der selben Frequenz senden könnte. Das könnte die Verbindung zwischen unseren Geräten stören und die seltsamen Daten schicken. Wir haben dann nochmal ausprobiert, die Channels, auf welcher die Module funken sollen, zu ändern, aber auch das hat keinen Unterschied gemacht.

Weiterhin ist in der Dokumentation der library aufgefallen, dass möglicherweise die Pins, die im Code mit CS und CE benannt werden, erst durch eine Veränderung der Einstellungen im Pi angesteuert werden können. Durch weitere Recherche haben wir herausgefunden, dass man dies besser über pigpio machen sollte. Dies haben wir dann direkt probiert, allerdings ohne Erfolg.

Bei den genannten Punkten hat es auch keinen Unterschied gemacht, welcher PI der Sender und welcher der Empfänger ist. Im nächsten Schritt haben wir dann noch versucht, die vorgegebenen Beispiele aus der library auszuprobieren und auch die Module nochmal neu zu installieren. Auch hier trat allerdings keine Änderung ein. Auch haben wir außer dem Funkmodul die anderen Module probeweise abgekabelt, um zu schauen, ob sich hier etwas gegenseitig stört. Aber auch dies hat keinen Unterschied gemacht. Wir haben das Setup auch zwischen einem Raspberry Pi und einem Arduino ausprobiert, was leider auch nicht funktioniert hat.

Auf Grund dieser gescheiterten Versuche sind wir dann dazu übergegangen, andere Möglichkeiten auszuprobieren. Zunächst haben wir versucht mittels dieses Tutorials <https://tutorials-raspberrypi.de/funkkommunikation-zwischen-raspberry-pis-und-arduinis-2-4-ghz/> einfach mit „sudo ./gettingstarted“ im Terminal (nach Installation der library) zu schauen, ob dieser einfache Test klappt. Auch dies führte nicht zum gewünschten Erfolg. Es kam die weitere Vermutung auf, dass das Problem evtl. die Programmiersprache sein könnte und so hatten wir uns genutzt wir dieses Tutorial: https://www.youtube.com/watch?v=tB_GebqaGas&ab_channel=AlexanderBaran-Harper bei welchem schnell

negativ auffiel, dass die verwendete library sehr alt und schlecht dokumentiert ist. Daher haben wir diese Idee wieder verworfen und uns im nächsten Schritt die library <https://circuitpython-nrf24l01.readthedocs.io/en/latest/> angeschaut. Wir versuchten hier zunächst, die example Datei „simple example“ zu nutzen, hier stellte sich allerdings schnell heraus, dass die Board Pins nicht erkannt werden, obwohl das zugehörige Modul installiert ist. Der Code bricht in Zeile 11 ab, weil der Parameter „board.D4“, welcher aber vom Beispiel so vorgegeben wird, nicht gefunden wird. Nach einer kurzen Recherche haben wir herausgefunden, dass die Funktion „DigitalInOut“ als Argument einen Eintrag der Kategorie „microcontroller.pin“ erwartet (<https://circuitpython.readthedocs.io/en/6.0.x/shared-bindings/digitalio/#digitalio.DigitalInOut>) Die Dokumentation dazu (<https://circuitpython.readthedocs.io/en/6.0.x/shared-bindings/microcontroller/index.html#microcontroller.Pin>) haben wir uns dann ebenfalls angeschaut. In dieser stand, dass man statt „board“ auch „microcontroller.pin“ verwenden kann. Also versuchten wir, statt board.D4 microcontroller.GPIO4 sowie microcontroller.pin.GPIO4 zu verwenden. Diese beiden Werte werden zwar von der Autocompletion vorgeschlagen, werfen aber ebenfalls die Fehlermeldung „microcontroller has no attribute GPIO4“.

Leider hat uns an dieser Stelle die Zeit gefehlt, noch stärker in die Fehlersuche einzutauchen. Um die restlichen Funktionen des Prototypen trotzdem zeigen zu können, haben wir uns dazu entschieden, einen Workaround mit Hilfe von Web-Sockets aufzubauen. Natürlich wäre dies für das fertige Produkt nicht sinnvoll, da die zwei Teile unseres Prototypen für Web-Sockets im selben Netzwerk sein müssen. Für unseren Fall ist es aber eine Möglichkeit, die allgemeine Funktionalität trotzdem zeigen zu können.

Für den Fall, dass wir mehr Zeit hätten, hätten wir gerne noch einige Möglichkeiten ausprobiert. Zunächst würden wir gerne testen, ob es in der Umgebung andere, störende Frequenzen gibt, die dafür sorgen könnten, dass unsere Module nicht miteinander kommunizieren können. Außerdem würden wir im nächsten Schritt gerne eine Möglichkeit finden, unsere komplette Hardware auf Fehler überprüfen zu können. Mit einem Multimeter könnte man beispielsweise überprüfen, ob überall genug Spannung ankommt. Sollte hier festgestellt werden, dass Komponenten beschädigt sind, würden wir diese austauschen und außerdem die Verkabelung von einem Experten anschauen lassen. Ist hier alles in Ordnung, ließe sich ausschließen, dass die Hardware das Problem ist und man könnte dazu übergehen, das Problem in der Software zu testen. Zunächst könnte man hier die Raspberry Pis komplett neu aufsetzen, um auszuschließen, dass durch mögliche Einstellungen, die wir getätigt haben, etwas kaputt gegangen ist. Wir könnten außerdem nochmal schauen, ob bei den Pinbelegungen ein Fehler aufgetreten ist und sich daher Fehler eingeschlichen haben. Dann sollten wir Schritt für Schritt den Code neu aufbauen und schauen, ob sich vorher Fehler eingeschlichen haben.

Außerdem haben wir in unseren Prototypen aus zeitlichen Gründen noch keine Unterscheidung einarbeiten können, ob ein:e Autofahrer:in GPS aktiviert hat oder nicht. Momentan gibt es nur die Option, das GPS aktiviert zu haben und dann die direkten Koordinaten zu vergleichen. Dann wird den Autofahrenden eine konkrete Warnung gegeben, wenn ein ESW sicher an ihnen vorbei fährt. Allerdings möchten wir den Nutzenden gerne die Möglichkeit geben, selber zu entscheiden, ob sie mit ihren Standortdaten arbeiten möchten oder nicht. Falls nicht, würden sie

einfach benachrichtigt werden, dass sich im Umkreis ein ESW befindet und dieser nur evtl. an dem/der jeweiligen Autofahrenden vorbei fährt.

Trotz oder gerade auf Grund dieser Hindernisse haben wir sehr viel lernen können. Generell haben wir den grundlegenden Umgang mit dem Raspberry Pi und dem entsprechenden Betriebssystem Linux kennengelernt. Dies wird uns auch in den kommenden Semestern einen großen Vorteil verschaffen, wenn wir wieder damit in Berührung kommen werden. Zusätzlich konnten wir Kenntnisse im Umgang mit Verkabelung, Widerständen und Stromflüssen sammeln. Unser Umgang mit dem Terminal und Node hat sich gefestigt und generell konnten wir erste Erfahrungen in der Kombination von Hard- und Software sammeln. Wir haben gelernt, wie wir einzelne Hardwareteile mit dem dazu passenden Code ansprechen können. Außerdem haben wir eine neue Programmiersprache, Python, kennengelernt. Auch das kann uns in Zukunft helfen, weil es uns mehr Variabilität in der Umsetzung geben wird.

Im Bezug auf die Fehlersuche beim Funkmodul konnten wir unser Durchhaltevermögen und unsere Disziplin festigen. Wir haben bis zum Schluss nicht aufgeben wollen und haben gelernt, auch dann weiter zu machen, wenn man schon sehr lange an einer Sache sitzt.

Anpassung der Software

Durch die Probleme mit der Datenübertragung über Funk haben wir uns entschieden, für unseren Prototypen mit Websockets die Funkkommunikation zu simulieren.

In `ev_node` wird ein WebSocket-Server gestartet, der anstelle des Funkmoduls alle zehn Sekunden seine Position, die Route und die personalisierte Nachricht verschickt. `Car_node` verbindet sich beim Start mit dem WebSocket-Server und nimmt die Daten entgegen.

Da es für den Prototypen aufgrund der geringen Reichweite von WLAN (< 50m) und der Ungenauigkeit unserer GPS-Module keinen Sinn mehr macht die Überprüfung, ob sich das Auto auf der Route des Einsatzwagens befindet einzubauen, haben wir diese nicht implementiert. Aus dem gleichen Grund haben wir auch die Berechnung der Richtung des Einsatzwagens in Relation zum Auto weggelassen.

Außerdem haben wir bei dem „weitere Hilfe“-Screen drei Buttons mit einer Anruf-Funktion, einer Message-Funktion und einer „mehr Info“-Funktion. Diese wurden lediglich zu Mock-Up Zwecken eingebaut, jedoch nicht implementiert.

Ausblick

Während der begrenzten Zeit, welche wir in diesem Semester zur Verfügung hatten, konnten wir nicht alle Features und Situationen, welche wir gerne eingebracht hätten, in unserem Prototypen umsetzen. Zunächst einmal wäre ein weiterer Schritt, KALM auch auf andere Situationen im Straßenverkehr, wie z.B. den Stadtverkehr, auszuweiten. Dazu sind weitere, tiefgreifendere Recherchen nötig, um alle momentan noch unbekannten Situationen im Stadtverkehr einschätzen und in unserer Anwendung beachten zu können. Weiterhin könnten nicht nur Fahrzeuge, sondern auch Ampeln mit einer Warnung ausgestattet werden. So würden auch Verkehrsteilnehmende gewarnt werden, die KALM nicht besitzen.

Ein weiterer Punkt, den wir gerne noch mit einarbeiten möchten, ist, KALM auch für andere Verkehrsteilnehmende bereit zu stellen. Man kann allerdings nicht ein Produkt für alle erstellen, da jede Gruppe im Verkehr andere Ansprüche an unser Produkt stellt. Fahrradfahrende haben beispielsweise keine Mittelkonsole am Fahrrad und somit auch kein Display, über welches sie die Warnung erhalten könnten. Außerdem besitzen sie kein Radio, mit welchem sie den Funk aus dem Rettungswagen erhalten könnten. Somit wird deutlich, dass hier ein anderer Prototyp mit einem abgeänderten Konzept von Nöten wäre, um die Bedürfnisse dieser Entität erfüllen zu können. Aufgrund des zeitlich begrenzten Rahmens und da wir ein Produkt erschaffen möchten, welches die Bedürfnisse unserer Entität zu 100% erfüllt, haben wir uns auf Autofahrende begrenzt. Unser Prototyp bezieht sich dabei auf Grund der

ausgereifteren technischen Möglichkeiten auf Autos, welche technisch moderner (Display, Ambient Lights, Steuerelemente am Lenkrad) sind. Für ältere Modelle wäre es eine Möglichkeit, eine Ausgabe rein über das Radio zu erstellen.

Im weiteren Verlauf möchten wir in den Prototypen eine Sprachsteuerung mit einbauen. So könnten Teile unseres Konzeptes, wie z.B. die Möglichkeit, KALM zu sagen, dass man jemanden anrufen möchte, im Produkt umgesetzt werden können. Dies sorgt dafür, dass die Autofahrenden im Gegensatz zum Tippen auf dem Display möglichst wenig abgelenkt werden. Allerdings möchten wir gleichzeitig darauf achten, dass die Barrierefreiheit auch für z.B. taube Menschen gegeben bleibt. Daher muss die Möglichkeit, das Display zu benutzen, bestehen bleiben. Durch diese Möglichkeit und die Vibration im Lenkrad können auch Gehörlose KALM verwenden.

In Zukunft könnte, je nach Automodell, die Warnung nicht mehr auf dem Display, sondern auf einem Head Up Display angezeigt werden. Dies würde dafür sorgen, dass Nutzende den Blick nicht mehr von der Straße abwenden müssen. Nützlich ist dies allerdings nur, wenn gleichzeitig die Sprachsteuerung ermöglicht würde.

Darüber hinaus würden wir eine asymmetrische Verschlüsselung anwenden wollen, um die Rettungskräfte zu schützen. So würden wir sicherstellen, dass sich Unberechtigte nicht als Einsatzkräfte ausgeben und unseren Dienst manipulieren. Zudem würde die Route immer vor dem Zielort abgebrochen werden, um die Privatsphäre aller Beteiligten zu schützen. Da die technische Umsetzung dafür jedoch den zeitlichen Rahmen des Projekts sprengen würde, können wir dies leider nicht für unseren Prototyp realisieren.

Bezüglich der Ambient Lights und des Displays haben wir uns überlegt, dass es zu Komplikationen kommen könnte, wenn bestimmte Lichtverhältnisse aufkommen. Idealerweise würden die Ambients Lights sich den Lichtverhältnissen anpassen, sodass sie zu jeder Zeit gut zu erkennen sind und es auch eine Option geben würde, die für farbenblinde Menschen vorteilhaft wäre. Dazu müsste man sich näher damit beschäftigen, wie dies technologisch umzusetzen ist, jedoch würde dies ebenfalls den zeitlichen Rahmen sprengen.

Management

Team Model Canvas

Damit jedes Teammitglied an einem Strang zieht und das Team als Ganzes nach einem gemeinsamen Ziel strebt, haben wir uns während der Kennenlernphase ein Team Model Canvas angelegt und die daraus resultierenden Ergebnisse gemeinsam besprochen. Wir haben unsere persönlichen Projektziele ausgetauscht und mit den Teamzielen abgeglichen. Dabei kam heraus, dass jeder an seinen eigenen Skills feilen möchte, und manche den Fokus auf spezifische Skills setzen, an denen sie dieses Semester mehr arbeiten möchten. Dies wird in der Umsetzungsphase vom Team natürlich berücksichtigt.

Mit diesem Modell werden auch die Grundwerte, sowie Erwartungen des Teams festgesetzt, um die gegenseitige Unterstützung und Ergänzung zu fördern. Der zwischenmenschliche Umgang wird somit reguliert.

Zum Zeitpunkt der Erstellung hatten wir uns noch nicht notiert, was die Stärken und Schwächen des Teams sind, da sie noch nicht bekannt waren. Allerdings haben wir uns spontan auf den Teamnamen „KALM“ geeinigt (ein Akronym für Kitty, Ann-Kathrin, Lisa, Max). Der Teamname ist ziemlich passend, da wir eher ruhige und entspannte Persönlichkeiten sind.

Team Model Canvas



Kommunikationskanäle

Signal

Signal nutzen wir, um neue Termine auszumachen, uns auf dem neusten Stand zu halten, kurze Infos auszutauschen und Organisatorisches zu klären. Da es auf dem Handy läuft, können hier jederzeit und schnell Fragen beantwortet werden.

Conzeptboard

Unser Konzeptboard „KALM“ ist der Ort, an dem wir alle Ideen und Fortschritte festhalten. Hier erstellen wir Mindmaps, sammeln Stichpunkte, entwickeln und verwerfen neue Ideen. Dieses digitale Whiteboard begleitet uns durch das ganze Projekt.

Trello

In Trello befindet sich ein auf unser Team angepasstes Kanban-Board. Näheres dazu wird im Kapitel „Agiles Projektmanagement“ unter „Kanban-Board“ erklärt.

Discord

Discord ist unser Hauptkommunikations-channel. Hier finden mehrmals die Woche Teammeetings statt. Wir bringen uns auf den neusten Stand, besprechen unseren Arbeitsstand und wie wir weiter vorgehen werden. Wichtige Quellen sowie alle nützlichen Infos werden in dem Discordchat gesammelt.

Arbeitsverteilung

Zu Beginn der gemeinsamen Projektarbeit haben wir einen Arbeitsplan aufgesetzt, in welchem wir unsere Treffen notierten. Die Treffen im ganzen Team finden regelmäßig dienstags und freitags ab 08:30 Uhr statt.

Bei jedem gemeinsamen Treffen stellen wir uns die erledigte Arbeit vor und schauen, was als nächstes zu tun ist, ob ein Zwischentreffen nötig ist, wie lange die nächsten Gruppentreffen voraussichtlich dauern werden und legen die Arbeitspakete für den nächsten Wochensprint fest. Außerdem werden alle Treffen dokumentiert, sodass keine Informationen verloren gehen.

Damit jeder etwas zu tun hat, halten wir die Arbeitsverteilung stets ausgeglichen. Zwischen Treffen arbeiten wir alleine oder in Untergruppen an verschiedenen Arbeitspaketen, sodass wir unsere Zeit effektiv nutzen.

Wir haben für jeden Bereich (Design, Coding) Verantwortliche, die dafür sorgen, dass bis zum Teamtreffen alle nötigen Aufgaben erarbeitet werden. Um den Spaß an der Arbeit am Projekt nicht zu verlieren, haben alle die Chance, an den Dingen zu arbeiten, an denen sie arbeiten wollen. Nach jedem Sprint legen wir gemeinsam fest, wer mit wem welche Aufgabenpakete erledigt.

Induktion und Deduktion

Bei der Definition von Induktion und Deduktion und der Vorgehensweise haben wir uns auf die „Induktiv und deduktiv vorgehen in 4 Schritten“ Methode (Pfeiffer, 2018) gestützt.

Um ein geeignetes Thema für unser Projekt zu entwickeln, arbeiteten wir in unserer Ideenphase zunächst induktiv, die eigene Forschung wurde zuerst getrieben und daraufhin eine Theorie aufgestellt. Dafür haben wir unsere eigenen Beobachtungen und Erfahrungen gesammelt. Zuerst haben wir alle Probleme der Interaktionen unterwegs zusammengetragen, um diese in verschiedenen Bereiche einzuteilen und nach Wichtigkeit zu bewerten. Daraus folgte eine engere Auswahl mithilfe einer Entscheidungsmatrix und wir konnten uns daraufhin auf eine konkrete Problemstellung einigen. Für unser weiteres Vorgehen stellten wir eine genau definierte Theorie zu unserem Problem auf.

1. Beginne mit einer Beobachtung.

Ein Mensch hat Probleme bei der frühzeitigen Bildung einer Rettungsgasse und ist gestresst.

2. Sammle Daten und schau nach, ob du ein Muster erkennen kannst.

Wir untersuchen durch Umfragen und Interviews mit Experten, ob Menschen Probleme/gestresst sind bei der Bildung einer Rettungsgasse haben.

3. Stelle eine mögliche Hypothese auf.

Wenn wir 80 Leute befragen und alle ein Problem mit der Bildung einer Rettungsgasse haben/gestresst sind, dann haben alle dieses Problem.

4. Stelle eine Theorie (Generalisierung) auf.

Die Meisten fangen an eine Rettungsgasse zu bilden sobald der Verkehr ins Stocken kommt. Die Meisten reagieren verunsichert oder gestresst.

Um unsere abgeleitete These zu überprüfen, sind wir in unserer Forschungsphase deduktiv vorgegangen. Durch genaue Recherche und Durchführungen von Interviews mit Experten, untersuchten wir, ob unsere These bestätigt oder widerlegt wird.

Es wurde belegt, dass ein Problem bei der Bildung von Rettungsgassen besteht. Ebenfalls wurde belegt das Autofahrende gestresst reagieren.

Nach näheren Untersuchungen und Coachings mit unseren Projektbetreuern sind wir zu dem Schluss gekommen, dass es sinnvoller gewesen wäre, zuerst deduktiv vorzugehen, um das große Ganze zu betrachten, bevor wir uns auf ein Thema einigen. Dies ist mit der Redewendung „den Wald vor lauter Bäumen nicht sehen“ gut veranschaulicht.

Agiles Projektmanagement

Arbeitsweise

Unser Team möchte flexibel, agil und autonom arbeiten. Mit anderen Worten: unser Team führt und reguliert sich selbst. Wir möchten unsere Arbeitsprozesse selbst gestalten, unsere eigenen Prioritäten setzen und behalten unseren Fortschritt selbst im Blick.

Scrum

In Kombination mit den Kanban-Boards arbeiten wir in der Umsetzungsphase mit wöchentlichen Sprints. Hierbei spielt die Selbstorganisation der Teammitglieder eine grundlegende Rolle. Jeder nimmt sich Aufgaben vom Produkt-Backlog, die er voraussichtlich am Ende der Woche oder innerhalb des selbst bestimmten Zeitraums fertigstellt. So arbeiten wir Schritt für Schritt die Aufgaben von der Liste ab und haben durchgehend funktionsfähige Zwischenprodukte. Natürlich kommunizieren wir während jedem Sprint miteinander, wie der aktuelle Zustand ist. Die Sprints bauen zudem aufeinander auf.

Nach jedem Sprint folgt ein Sprint Review, um im Team zu überprüfen, ob die Zwischenprodukte in Ordnung sind oder noch etwas daran verändert werden muss und holen uns anschließend von unseren Projektbetruern, in unserem Fall unseren Professoren, regelmäßig Feedback.

Kanban-Board

Wir haben uns dazu entschieden, mit dem evolutionären Change Management Tool, dem Kanban-Board, zu arbeiten. Indem wir Schritt für Schritt kleine Änderungen machen, reduzieren wir das Risiko jeder einzelnen Veränderung. Unser Team notiert sich alle zu erledigenden Aufgaben in der Backlog-Sektion unseres Kanban-Boards, damit wir eine klare Übersicht darüber behalten und bei der Projektentwicklung agil bleiben können.

Wir haben verschiedene Spalten für einzelne Arbeitspakete angelegt, so behalten wir einen besseren Überblick. In der Liste „Info“ befindet sich eine Legende mit wichtigen Terminen, Links, Projektphasen und einer Einteilung der Disziplinen nach Farben.

Jeder nimmt sich Aufgaben aus dem Backlog heraus und gibt an, in welchem Bearbeitungszustand (Entwickeln, Testen und Abgeschlossen) sich diese befinden. Zusätzlich gibt es noch die Spalte „Wochenaufgaben“, in der alle anstehende Aufgaben für die Woche aufgeführt werden. Da dies präsenzfrei funktionieren soll, haben wir uns dazu entschieden, unser Kanban-Board, wie bereits erwähnt, auf Trello zu erstellen.

Alle Aufgaben werden benannt und wenn nötig genauer beschrieben. Sie werden den Teammitgliedern sowie den Disziplinen zugeteilt und mit einer Frist belegt. Dadurch sind alle auf dem gleichen Stand und können jederzeit nachvollziehen, woran das andere Teammitglied arbeitet.

Kanban-Board

The screenshot displays a Kanban board for 'Team 5' with a top navigation bar and several columns representing different stages of the workflow.

Top Bar: Board ▾ Team 5 ☆ Privat-Team Privat PQ M AT L Einladen Kalender Butler (2 Tips) ... Menü anzeigen

Columns and Cards:

- Info:** Allgemein, Design, Coding, Management, Doku, Präsentation, Präsi, Links, Boards, Termine, Ideenphase (20. Nov. 2020).
- Backlog Doku:** Doku Coding: Datentransfer und Sicherheit (2. Jan., L, M); Management Doku: Kommunikation innerhalb der Gruppe (2. Jan., AT); Management Doku: Methoden Management (2. Jan., L, PQ); Management Doku: Arbeitsplanung (2. Jan., M); Management Doku: Arbeitsverteilung (2. Jan., M); Management Doku: Herausforderungen im Team (2. Jan., L, PQ); Management Doku: Ausblick (2. Jan., AT); Allgemein Doku: DoR + NVS.
- Backlog:** Design: Screendesign / UI (28. Jan., L, PQ); HTML Interface (+ Eine weitere Karte hinzufügen).
- Wochenaufgabe 11.01 - 18.01.:** Design: Formgebung (28. Dez. 2020, M); Allgemein Doku: Stichpunkte Konzeptdoku (4, 6. Jan., AT, L, M, PQ); Diagramme für Doku anpassen; Konzeptdoku (4, 13. Jan., AT, L, M, PQ); Handlungsablauf (2, 5. Jan., AT, L, M, PQ); Allgemein Coding: Prototyp (+ Eine weitere Karte hinzufügen).
- Entwickeln:** Peer review; Doku alle Texte durchgehen; Preview Board; Allgemein Doku: Styleguide der Doku (6. Jan., L); Präsentation Allgemein: Styleguide der Präsi (6. Jan., L); Präsentation Allgemein: Konzeptpräsentation (2, AT, L, M, PQ) (+ Eine weitere Karte hinzufügen).
- Testen:** + Eine Karte hinzufügen.
- Release:** Allgemein: Beziehung (21. Dez. 2020); Design: Wie formulieren (+ Eine weitere Karte hinzufügen).

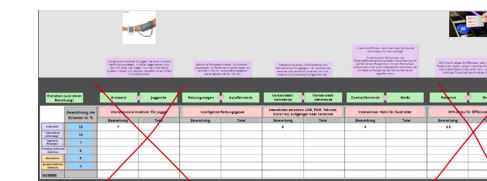
Fortschrittsbericht

Wir haben uns im Team dazu entschieden, nicht mit traditionellen Fortschrittsberichten zu arbeiten. Um jedes Team-Meeting und Team-Coaching zu protokollieren, haben wir unsere eigene Art von Fortschrittsbericht entwickelt, mit der wir am besten arbeiten können. Die Aufgabe des Protokollierens übernimmt immer dieselbe Person bei Teammeetings, bei Coachings können andere Teammitglieder ebenfalls Notizen machen, diese werden von dem/der Protokollant:in zusammengefasst. Die Fortschrittsberichte werden mit dem Programm „Notion“ erstellt und direkt nach den Treffen auf die Nextcloud in den gemeinsamen internen Teamordner mit dem zugehörigem Datum hochgeladen. Bei den Berichten sind immer Start- und Endzeit bzw. Dauer notiert sowie Entscheidungen, Überlegungen, Problemstellungen, Ergebnisse, Zwischenstände. Ein Beispielfortschrittsbericht ist rechts abgebildet.

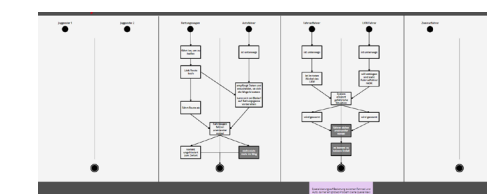
24.11.20

Dauer: 8.30-11Uhr, 12-14Uhr

- Entscheidungsmatrix für Ideen erstellt und die dazugehörigen Muss- und Kannkriterien ausgewählt (angefangen auszufüllen, dabei diskutiert ob unsere Ideen überhaupt interaktiv sind, manche Ideen gestrichen)



- (NFC-Chip mit Check-in und Check-out statt Papiertickets: Ist das zu stationär?)
- Problem: Wie unterscheidet sich Interaktion von Kommunikation? (Bei Rettungsgassenidee und Zweiradhelmidée) Was braucht eine Kommunikation um eine Interaktion zu sein?
 - Mit Team 6 (Diandra, Isabel, Christina, Tom) und Sascha besprochen
 - Wir sollten bei unseren Ideen an Handlungspartner und Beziehungen denken, nicht an einzelne Gadgets
 - Bei ihnen haben Aktivitätsdiagramme geholfen, um zu verstehen, wo sich die Schnittstellen der Handlungspartner befinden
- Aktivitätsdiagramme erstellt für die finalen Ideen



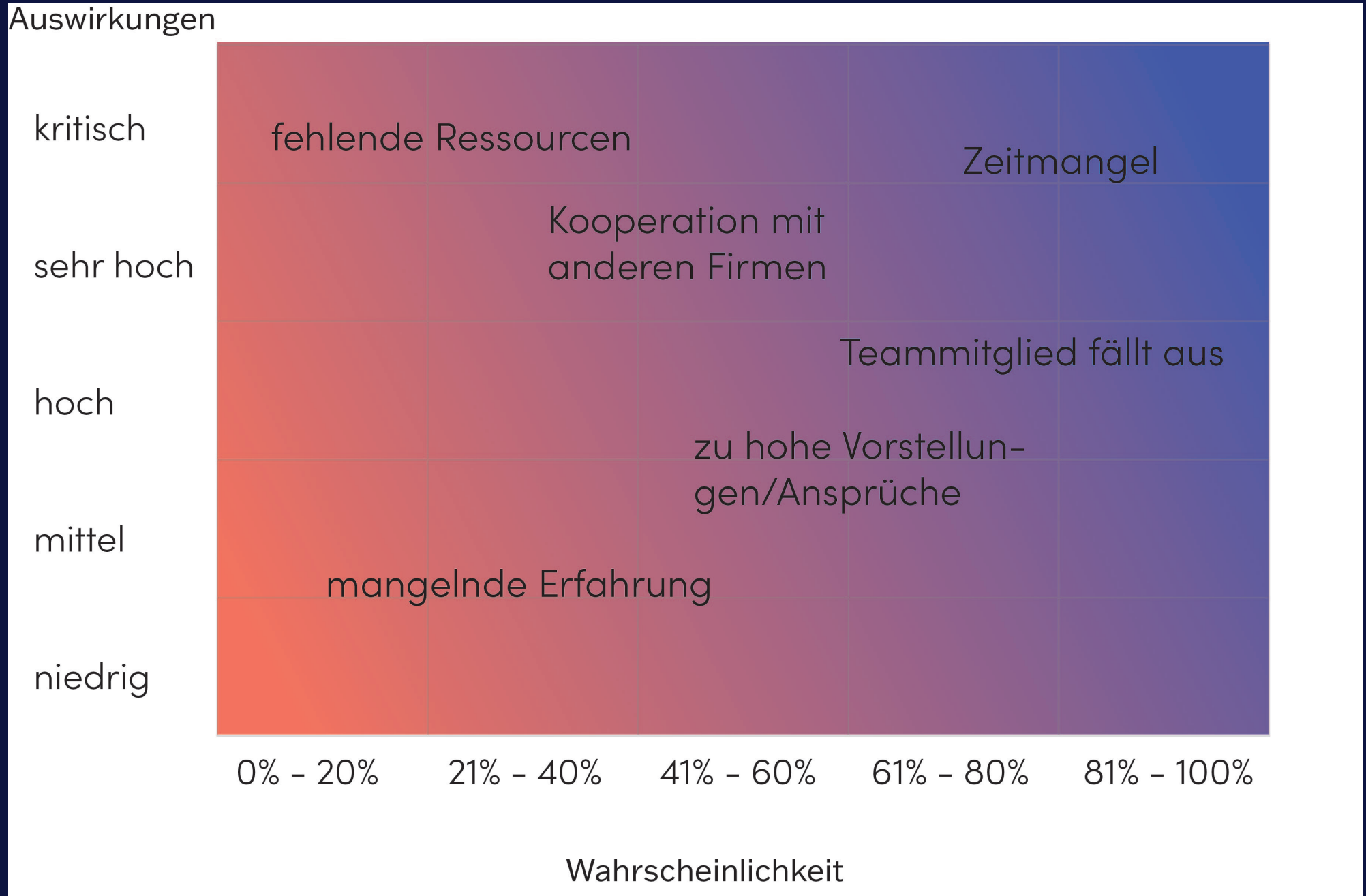
Aufgaben fürs nächste Meeting

- Wie ist die Beziehung der Interaktion?
- Idee mit Zweirädern: konkrete Beispiele?
- Entscheidungsmatrix für die letzten beiden Ideen

Risiken

Beispiel aus unserem Projekt:

Ereignis	Mangelnde Erfahrung	Kooperation mit anderen „Firmen“
Wahrscheinlichkeit	30%	ca. 50%
Präventive Maßnahme	- Jemanden ins Team holen der mehr Erfahrung in dem Bereich hat - mehr Arbeitsstunden einplanen	- mehr Pufferzeit einplanen - einen Vertrag aufsetzen - vorher Deadline setzen
Ursache	- wenig bis keine Erfahrung in dem Bereich gesammelt	Keiner Rückmeldung der Partner Firma
Risiko	Es kostet Zeit sich in neue Sachen/Programme einzuarbeiten	Komponenten fehlen z.B.: Software, Hardware, usw.
Auswirkung	- Zeit geht verloren - es müssen mehr Arbeitsstunden investiert werden	- Projekt kann nicht weitergeführt werden - Projekt gerät ins Stocken
Korrektive Maßnahme	/	nach Alternativen suchen



Arbeitspakete

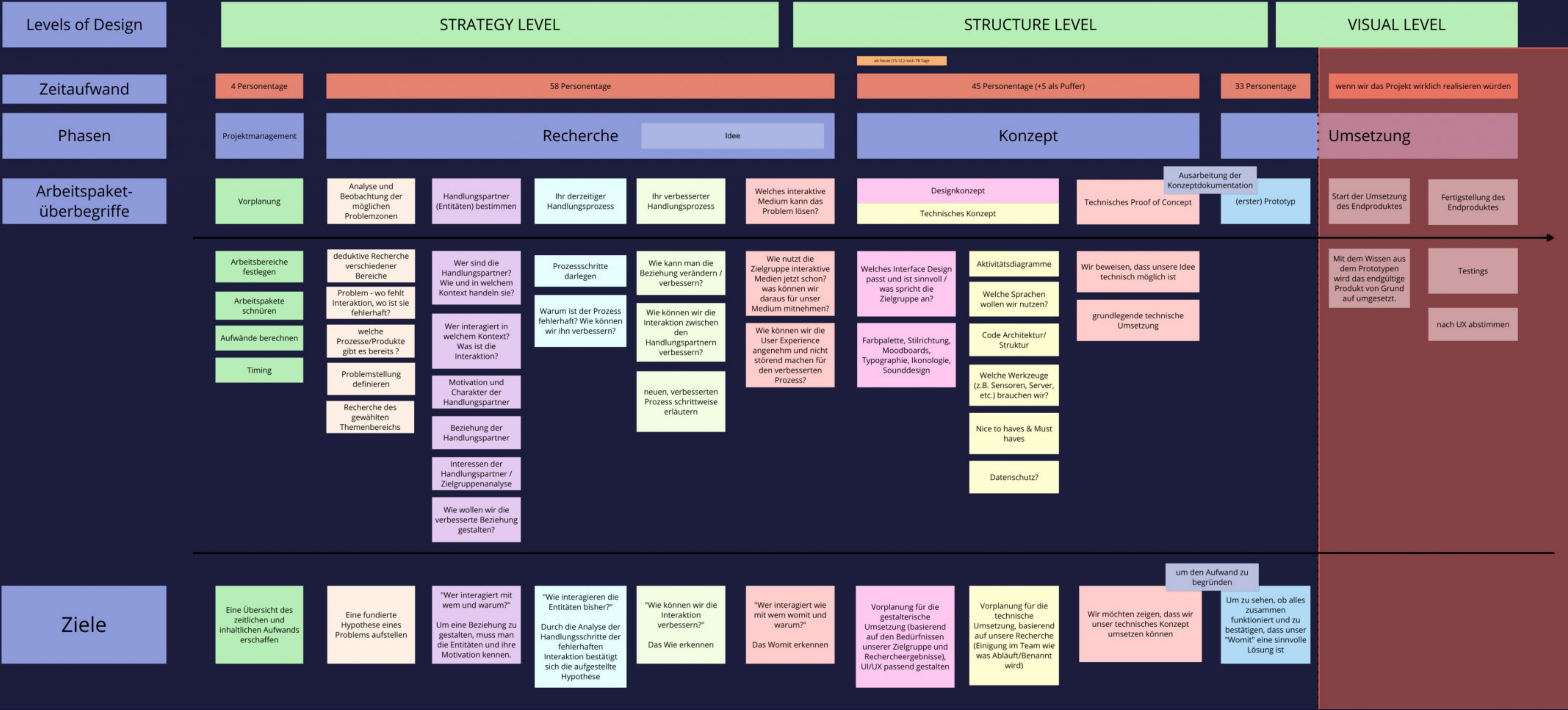
Arbeitsplanung

Um unser Projekt genauer zu strukturieren, haben wir eine Arbeitsplanung erstellt. Dafür haben wir zuerst alle Rahmenbedingungen aufgestellt, zum Beispiel, welche Tools wir benutzen, worüber wir kommunizieren, welche wichtige Termine nötig sind und welche Arbeitsmethoden wir verwenden. Danach haben wir unseren Gesamtprozess in Arbeitspakete heruntergebrochen und in einen Ablaufplan eingetragen. Dadurch kann am besten nachvollzogen werden, welche Schritte abhängig voneinander sind oder parallel laufen können.

Arbeitspakete

Zuerst haben wir unser Projekt in grobe Phasen geteilt. Danach haben wir uns alle Aufgaben notiert und diese in Arbeitspakete geschnürt. Die Arbeitspakete bestehen aus einem Überbegriff und mehreren kleinen, konkreten Aufgaben (siehe Abbildung auf der nächsten Seite).

Arbeitspakete



Learnings

The background features a dark blue wave-like shape on the left side, which contains the word 'Learnings'. To the right of this wave are three large, overlapping circles. The top circle is a light pinkish-red, the middle circle is a medium purple, and the bottom circle is a darker purple. These circles overlap each other and the dark blue wave, creating a layered, abstract design.

Design Thinking

Definition von Interaktion

Eine Interaktion ist das aufeinander bezogene Handeln zweier oder mehrerer Entitäten oder auch die Wechselbeziehung zwischen Handlungspartnern. Ein aufeinander bezogenes Handeln setzt eine Motivation voraus. Die Gestaltung von Interaktionen bedeutet also Beziehungen zu gestalten. (Tanaka, 2020)

„Technologien sind keine Lösungen, sondern ein Medium, mit der sich eine Lösung umsetzen lässt.“

– Herr Tanaka (am 06.01.21 im Coaching)

Beziehungen und Interaktionen gestalten statt Produkte

Dieses Semester wurde uns besonders klar, dass wir nicht in Produkten denken sollen, sondern an erster Stelle Beziehungen, Handlungen und Interaktionen in den Fokus stellen sollen. Wir als Designer:innen sollen Menschen und deren Beziehungen zu anderen Menschen oder zu digitalen Tools in den Vordergrund stellen und nicht nur innovative Technologien und abstrakte Konzepte ohne Kontext ausarbeiten. Die Beziehungsgestaltung der Interaktionspartner bildet einen essenziellen Aspekt des Design Thinkings, und wir als Team haben versucht, durchgehend dieses Learning im Laufe des Projekts nicht aus den Augen zu verlieren und zu festigen. Dieses Learning wird vor allem durch „Learning by Doing“ und „Project based Learning“ verankert.

Prozess zur Problemdefinition

Um zu verstehen und herauszufinden, was das eigentliche Problem einer Situation ist, haben wir die Methode aus dem Design Thinking gelernt wie Kleinkinder mehrmals „Warum?“ zu fragen. Sie ist unter anderem auch bekannt als die „5 Warum-Methode“ (5 Warum-Methode | Design Thinking Methoden Katalog, o. D.), nicht zu verwechseln mit der „6 W-Methode“. Diese Methode soll uns dabei helfen, ein grundlegendes Verständnis für die Situation aufzubauen und unsere Antworten dabei mehrmals zu hinterfragen, um am Ende bei einem gewissen Schmerzpunkt zu landen. Dort würden wir dann ansetzen wollen, um das Problem an der Wurzel zu packen und die Ursache des Problems anzugehen, statt zu versuchen, ein Symptom zu behandeln. Durch diese Methode soll mit anderen Worten Empathie entwickelt werden.

User Forschung

Für uns war es dieses Semester eine große Herausforderung, einzugestehen, dass wir anfangs unseren Nutzenden bzw. unsere Entitäten nicht genug kannten, wie wir es eigentlich tun sollen. Da wir zudem am Anfang induktiv gearbeitet haben und von unseren eigenen Erfahrungen daraufhin Annahmen gemacht haben über den Nutzenden und der Situation, sind wir immer wieder auf das Problem gestoßen, dass wir „Reverse Engineering“ betrieben haben. Dies war auf vielen Aspekten kontraproduktiv. Beispielsweise haben wir anfangs dem/der Autofahrenden keine Empathie gezeigt und haben ihre Charaktereigenschaften angenommen (Sie seien beispielsweise grundlos genervt und von Natur aus unkooperativ etc.). Als wir darauf hingewiesen wurden, wurde uns bewusst, dass es tiefere Gründe für ihre Stresssituation gibt und haben diese dementsprechend recherchiert. Zusammenfassend kann man sagen, dass das wichtigste Learning dieses Semesters bezüglich Design Thinkings das Verbot von Annahmen und Vermutungen ist, und wir alles auf wissenschaftlicher Basis fundieren sollen, um diese zu vermeiden.

Projekt-Learnings

Unsere Herausforderungen

Unsere größte Herausforderung war die Zeit. Während der Ideenphase fiel es uns schwer, uns auf ein Thema zu einigen und dies kostete uns ein paar Tage. Verstärkt wurde dies in der Recherchephase und der Konzeptphase, da wir zusätzlich neue Design-Thinking Methoden lernen sollten und dies für uns Neuland war. Ebenso erlernten wir in Coding neue, wichtige technische Inhalte, die wir auf unser Projekt anwenden sollten und dies benötigte ebenfalls Zeit. Diese Hürden waren zeitintensiv, da wir häufig nach den Coachings einige Schritte zurückgehen mussten, um die Kritikpunkte der Professoren zu bearbeiten.

Außerdem wurden einige Teamdiskussionen durch den fehlenden direkten Menschenkontakt aufgrund des präsenzfrierten Unterrichts zusätzlich erschwert. Es gab einige Missverständnisse, die eventuell durch Körpersprache und Mimik sogar hätten verhindert werden können. Ebenso lernt man seine Teammitglieder nicht so gut kennen, wie bei eigentlicher Präsenz-Zusammenarbeit.

Des Weiteren hatte unser Team an manchen Zeitpunkten keine klare Richtung und war sich des weiteren Vorgehens nicht ganz bewusst. Letztendlich haben wir unsere Spur allmählich wiedergefunden, doch da ist uns öfter Zeit verloren gegangen.

Unsere Erkenntnisse bis jetzt

Eine der wichtigsten Erkenntnisse des Projekts war die Wichtigkeit des Notierens der Entscheidungen und Besprechungsinhalte. Damit man jedes Teammitglied ins Boot holt und das ganze Team „on the same page“ ist, müssen klare Entscheidungen im Team getroffen und hinuntergeschrieben werden, damit dies nicht in Vergessenheit gerät oder Unsicherheiten bezüglich des Projekts auftauchen. Eine weitere grundlegende Erkenntnis ist, dass die Hindernisse, die uns von den Professoren aufgestellt werden, uns bei der Weiterentwicklung des Projekts als auch unserer Person weiterhelfen sollen und die Kritikpunkte nicht als persönliche Kritik aufgefasst werden sollen. Außerdem wurde uns nun klar, wie wesentlich die Einplanung von Pufferzeiten wirklich ist.

Weitere Learnings

Besonders in der Vorplanung des Projektes haben wir viel dazugelernt. Im Nachhinein haben wir eingesehen, dass es wichtig ist, konkrete Arbeitspakete zu schnüren und dabei realistisch zu bleiben. Mit einer besseren Vorplanung hätten wir uns viel Stress ersparen können. Ebenfalls haben wir gelernt, mit welchen Arbeitsmethoden und Tools unser Team am besten klarkommt und diese speziell auf unsere Bedürfnisse angepasst.

Besonders in Zeiten des Home-Office ist uns klar geworden, wie wichtig feste Termine und offene und ehrliche Gespräche sind, um Missverständnisse zu klären. Wichtig ist auch, den eigenen Alltag so zu planen, dass man nicht den ganzen Tag lang vor dem eigenen PC sitzt, damit man einen gesunden mentalen Zustand beibehält.

Ein Einblick ins Risikomanagement war sehr interessant und hilfreich, jedoch fehlte uns hier jegliche Erfahrung diese realistisch einzuschätzen. Um Risiken frühzeitig zu erkennen und zu vermeiden, fehlen uns eventuell praktische Erfahrungen.

Als Team sind wir sehr dankbar dafür, dass dieses Semester keine gravierenden Probleme im Bereich Krisenmanagement aufgetreten sind. Wir freuen uns darauf, das angeeignete Wissen auf die weiteren Projekte übertragen zu können und neue Erfahrungen zu sammeln.

Quellen

- In 80 Prozent aller Fälle funktioniert die Rettungsgasse nicht richtig. (2018, 19. November). [www.drk.de](https://www.drk.de/presse/pressemitteilungen/meldung/in-80-prozent-aller-faelle-funktioniert-die-rettungsgasse-nicht-richtig/).
<https://www.drk.de/presse/pressemitteilungen/meldung/in-80-prozent-aller-faelle-funktioniert-die-rettungsgasse-nicht-richtig/>
- Neuer Spot zur Rettungsgasse: Deshalb zählt jede Sekunde. (2020, 20. Februar). [runter vom Gas](https://www.runtervomgas.de/news/artikel/neuer-spot-zur-rettungsgasse-deshalb-zaehlt-jede-sekunde.html).
<https://www.runtervomgas.de/news/artikel/neuer-spot-zur-rettungsgasse-deshalb-zaehlt-jede-sekunde.html>
- Badische Neueste Nachrichten. (2020, 24. Juli). Keine Rettungsgasse: So kämpfen sich die Helfer auf der A5 zu den Verletzten vor.
<https://bnn.de/kraichgau/bruchsal/keine-rettungsgasse-so-kaempfen-sich-die-helfer-zu-den-verletzten-vor>
- Braun, J. (2019, 3. Juli). Warum wir uns beim Autofahren in Unmenschen verwandeln – Wie im Straßenverkehr die Menschlichkeit verloren geht. Psychophilie.
<https://www.psychophilie.com/blog/2019/5/20/warum-wir-uns-beim-autofahren-in-unmenschen-verwandeln-wie-im-strassenverkehr-die-menschlichkeit-verloren-geht>
- Müller, C. (2019, 21. November). „Wir nehmen Konflikte im Straßenverkehr zu persönlich“ – _erklärt | TÜV NORD. TÜV NORD GROUP. <https://www.tuev-nord.de/explore/de/erklaert/wir-nehmen-konflikte-im-strassenverkehr-zu-persoendlich/>
- Bierhoff, H. (2019). Empathie-Altruismus-Hypothese – Dorsch – Lexikon der Psychologie. Dorsch Lexikon der Psychologie.
<https://dorsch.hogrefe.com/stichwort/empathie-altruismus-hypothese>
- Hoffmann, S. (2018, 19. März). Darum wählte man für Blaulicht die Farbe Blau – und nicht Rot. [geo](https://www.geo.de/wissen/18549-rtkl-endlich-verstehen-darum-waehlte-man-fuer-blaulicht-die-farbe-blau-und-nicht-rot).
<https://www.geo.de/wissen/18549-rtkl-endlich-verstehen-darum-waehlte-man-fuer-blaulicht-die-farbe-blau-und-nicht-rot>
- Immos, F. (o.D.). Farben Wahrnehmung Assoziation Psychoenergetik.
<http://gestaltung.wilhelm-ostwald-schule.de/wp-content/uploads/2010/04/bedeutung-der-farben.pdf>
- Stauss, T. (o. D.). Komplementär-Kontrast. Lehrer:innenfortbildung Baden Württemberg.
https://lehrerfortbildung-bw.de/st_digital/medienkompetenz/gestaltung-farbe/kontrast/komp-kon/

- Pfeiffer, F. (2020, 4. März). Deduktiv in deiner wissenschaftlichen Arbeit vorgehen. Scribbr.
<https://www.scribbr.de/methodik/deduktiv/>
- Beinert, W. (2019, 24. Januar). Lesbarkeit von Schriften und Schriftsätzen in der Typografie. Typolexikon.
<https://www.typolexikon.de/lesbarkeit/>
- Schriftportrait Franklin Gothic – Franklin Gothic kaufen. (o. D.). Schriftgestaltung.com. Abgerufen am 19. Januar 2021, von
<https://schriftgestaltung.com/schriftlexikon/schriftportrait/franklin-gothic.html>
- 5 Warum-Methode | Design Thinking Methoden Katalog. (o. D.). msg systems ag. Abgerufen am 13. Februar 2021, von
<https://www.designthinking-methods.com/1Empathie/5-warum.html>

Delegation of Rights and Non Violation Statement

Kalm is a project, developed in the framework of the study course Interactive Media Design of Hochschule Darmstadt – Faculty of Media, during the Winter Term 2020. It is a group work of: Phuong Quach, Ann-Kathrin Tissen, Lisa Jochem and Maximilian Weber. It was mentored by: Prof. Tsune Tanaka, Prof. Claudius Coenen and Sarah Antes. We herewith delegate the non exclusive and timewise non restricted rights to publish and present the results of the project Kalm to the Professors of Hochschule Darmstadt and to the coaches directly connected to the academic supervision of this project, named above. In the same time the student project team declares that with the project no intellectual properties rights of third parties have been harmed.



Ort, Datum, Unterschrift

(Phuong Quach)



Ort, Datum, Unterschrift

(Ann-Kathrin Tissen)



Ort, Datum, Unterschrift

(Lisa Jochem)



Ort, Datum, Unterschrift

(Maximilian Weber)

Plagiatserklärung

Hiermit erklären wir, Phuong Quach, Ann-Kathrin Tissen, Lisa Jochem and Maximilian Weber, dass wir die vorliegende Arbeit selbständig verfasst, keine anderen als die angegebenen Quellen und Hilfsmittel benutzt und alle Stellen der Arbeit, die anderen Werken dem Wortlaut oder dem Sinn nach entnommen sind, in jedem einzelnen Fall kenntlich gemacht haben. Darüber hinaus haben wir keine Arbeit mit ähnlichem Inhalt an einer anderen Stelle eingereicht.



Ort, Datum, Unterschrift

(Phuong Quach)



Ort, Datum, Unterschrift

(Ann-Kathrin Tissen)



Ort, Datum, Unterschrift

(Lisa Jochem)



Ort, Datum, Unterschrift

(Maximilian Weber)



kalm