

# Fuhrparkverwaltung für die Qualitätssicherung

Entwicklung einer Software zur Reservierung und Verwaltung von Testfahrzeugen sowie zur digitalen Mängeldokumentation

Fiktives Studienprojekt

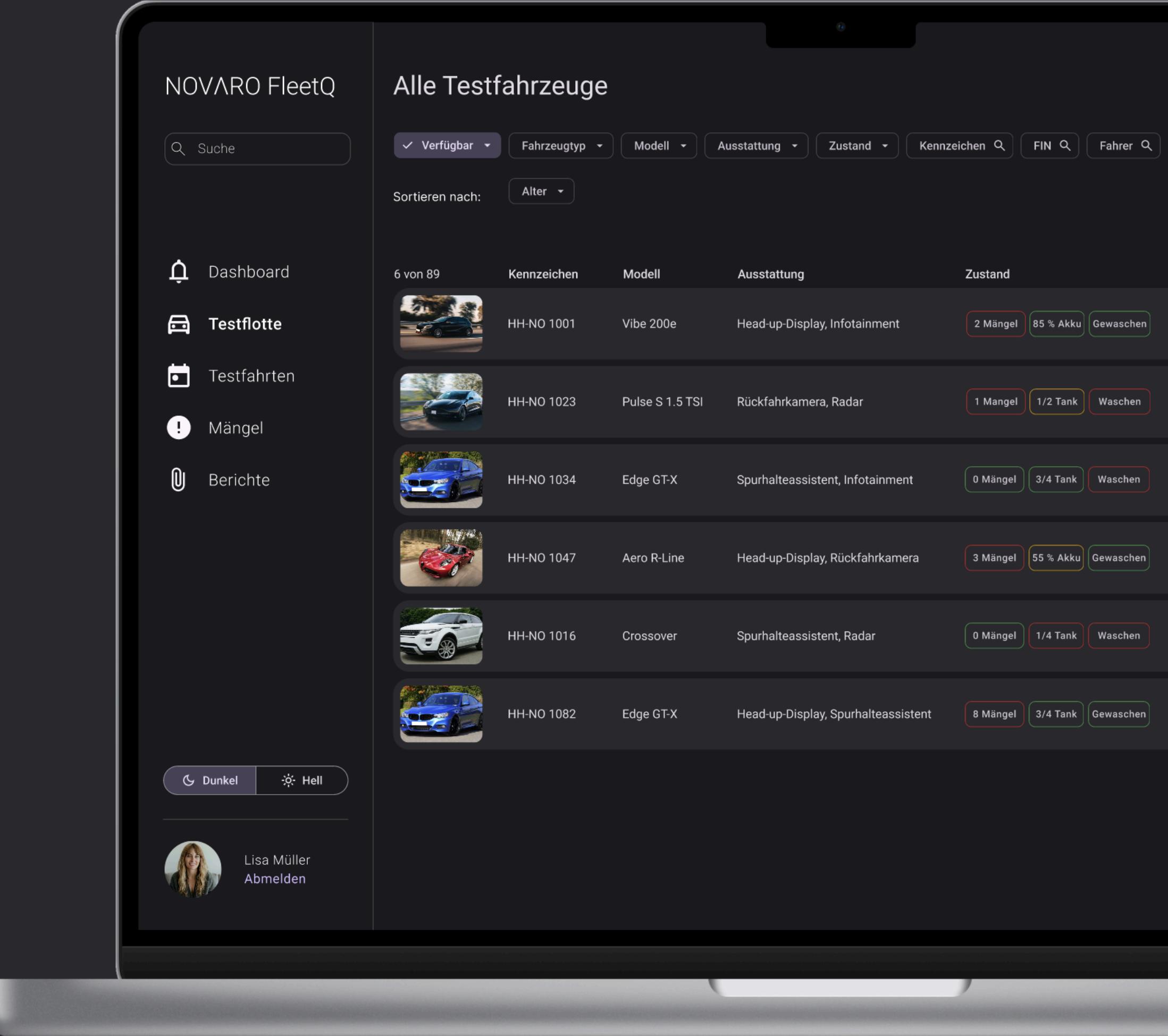
Zeitaufwand: 150 h (5 ECTS)

Dauer: 3 Monate

Device: Desktop

 Figma / Figjam

B2B



The screenshot displays the NOVARO FleetQ application interface. On the left, a sidebar menu includes links for Dashboard, Testflotte, Testfahrten, Mängel, Berichte, Dunkel (dark mode), and Hell (light mode). The main content area is titled "Alle Testfahrzeuge" and shows a list of 6 vehicles from a total of 89. Each entry includes the license plate number, model, features, and status indicators for defects, battery level, and wash status.

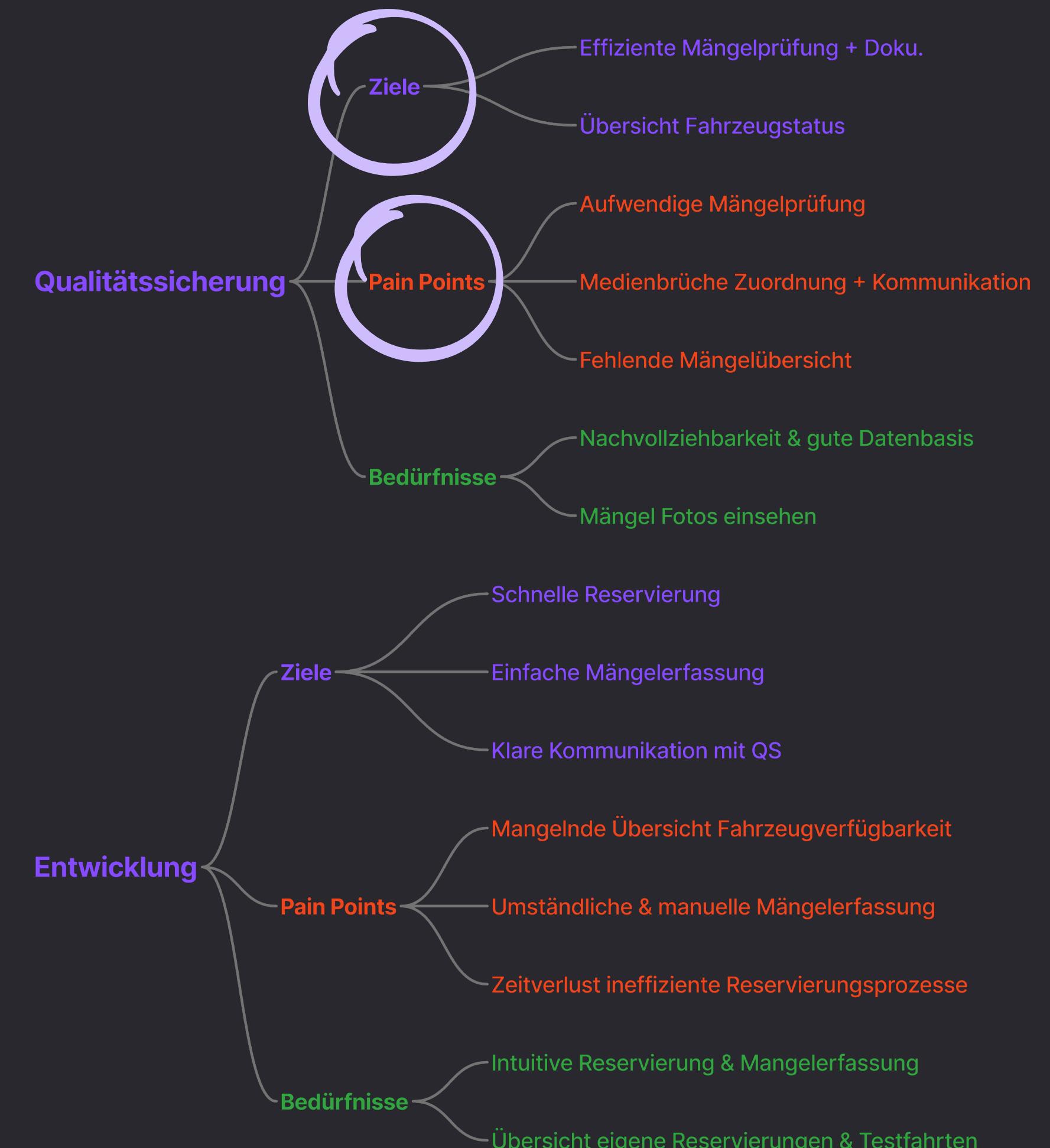
6 von 89	Kennzeichen	Modell	Ausstattung	Zustand
	HH-NO 1001	Vibe 200e	Head-up-Display, Infotainment	2 Mängel   85 % Akku   Gewaschen
	HH-NO 1023	Pulse S 1.5 TSI	Rückfahrkamera, Radar	1 Mangel   1/2 Tank   Waschen
	HH-NO 1034	Edge GT-X	Spurhalteassistent, Infotainment	0 Mängel   3/4 Tank   Waschen
	HH-NO 1047	Aero R-Line	Head-up-Display, Rückfahrkamera	3 Mängel   55 % Akku   Gewaschen
	HH-NO 1016	Crossover	Spurhalteassistent, Radar	0 Mängel   1/2 Tank   Waschen
	HH-NO 1082	Edge GT-X	Head-up-Display, Spurhalteassistent	8 Mängel   3/4 Tank   Gewaschen

## INTRO

# Problemstellung & Ziele

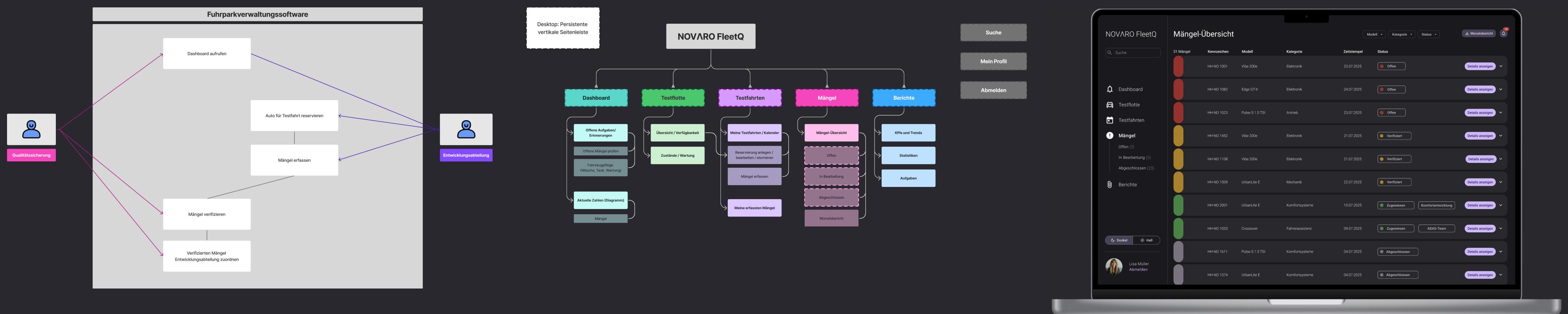
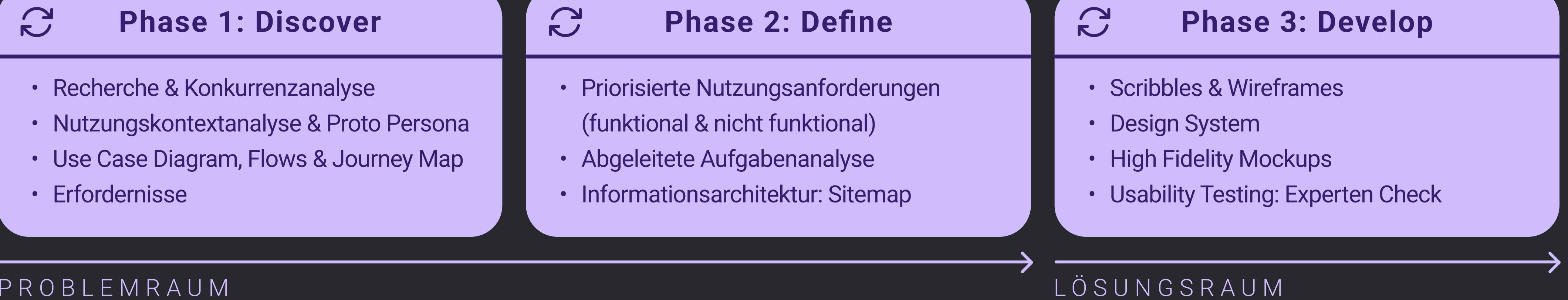
NOVARO ist ein großer deutscher Automobilhersteller der erfolgreich Fahrzeuge an eine junge Zielgruppe verkauft. Im Bereich der Qualitätssicherung und Entwicklung erfolgt die Verwaltung von Testfahrzeugen, Mängeldokumentation und Fahrzeugpflege bislang fragmentiert und mit großem manuellen Aufwand.

- ✖ Testfahrten werden **uneinheitlich** gebucht und koordiniert
- ✖ Daten sind **unstrukturiert** und Informationen gehen verloren
- ✖ Mängelmeldungen sind **aufwendig** und schwer nachvollziehbar
- ✓ Prozesse **digitalisieren**, **zentralisieren** und **beschleunigen**
- ✓ User Interface **effizient**, **effektiv** und **intuitiv** bedienbar gestalten
- ✓ Qualitätssicherung und **Produktqualität verbessern**



## DIE METHODEN

# Iterative Vorgehensweise: Human-Centered-Design



## PHASE 1: DISCOVER

# Problemraum: Dealing with constraints

## Begrenzter User Research

Aufgrund des begrenzten Zeitrahmens konnte nur wenig direkter User Research durchgeführt werden. Um dennoch **nutzerzentrierte Entscheidungen** zu ermöglichen, wurden alternative Methoden eingesetzt: eine **Konkurrenzanalyse** zur Identifikation von Best practices, **Proto-Personas** zur Annäherung an die Zielgruppen sowie eine **Nutzungskontextanalyse**, um das Umfeld besser zu verstehen.



**Lisa Müller**

*„Ich brauche ein System, das mir schnell einen Überblick über das Wichtigste gibt.“*

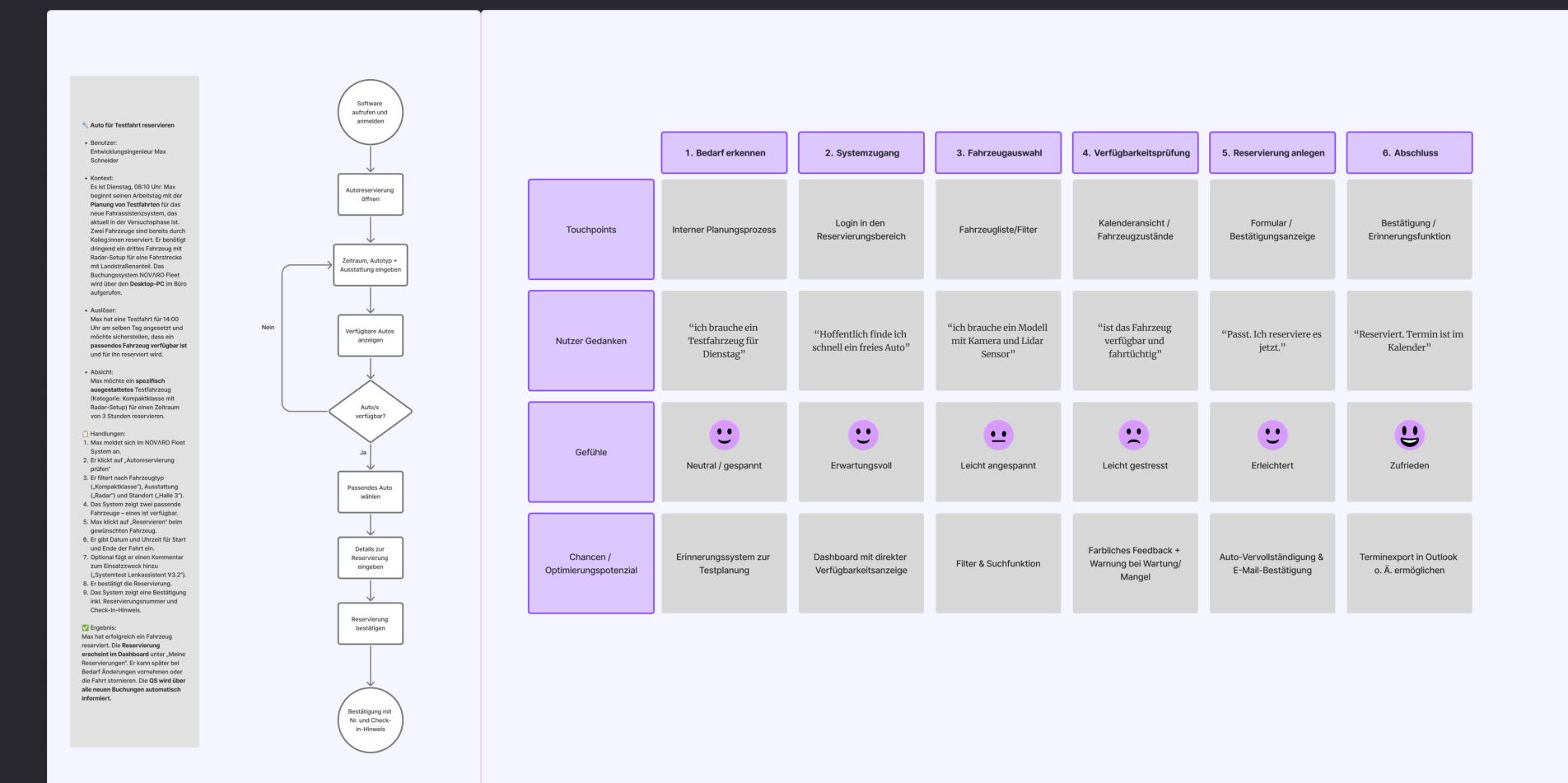
- 35 Jahre, wohnt in Hamburg
- Qualitätsingenieurin im Bereich Fahrzeugentwicklung
- 10 Jahre im Unternehmen, vertraut mit internen Prozessen
- Versiert im Umgang mit Unternehmenssoftware, nutzt regelmäßig Intranet, Testdatenbanken und Excel
- Arbeitet täglich am Desktop-PC, gelegentlich am Tablet vor Ort im Fahrzeugbereich

Ziele & Aufgaben	Pain Points	Bedürfnisse
<ul style="list-style-type: none"> <li>• Übersicht über verfügbare Testfahrzeuge behalten</li> <li>• Testfahrten koordinieren und dokumentieren</li> <li>• Mängelberichte sichten, verifizieren und weiterleiten</li> <li>• Monatsberichte für Management vorbereiten</li> <li>• Wartungszustand der Fahrzeuge prüfen und nachverfolgen</li> </ul>	<ul style="list-style-type: none"> <li>• Umständliche Excel-Listen oder manuelle Abstimmungen</li> <li>• Fehlende Übersicht bei mehreren gleichzeitig genutzten Fahrzeugen</li> <li>• Nicht nachvollziehbare Änderungsverläufe bei Mängeln</li> <li>• Zeitverlust durch doppelte oder fehlende Informationen</li> </ul>	<ul style="list-style-type: none"> <li>• Intuitives, klares Dashboard mit Filterfunktionen</li> <li>• Einfache Verifizierung von Mängeln mit Kommentarfunktion</li> <li>• Automatische Erinnerung an Wartungsintervalle</li> <li>• Exportierbare Monatsberichte ohne manuelle Nachbearbeitung</li> <li>• Verlässliche Rechtevergabe – nur sie darf Mängel verifizieren</li> </ul>

Zuverlässig & Genaue
Pragmatisch
Strukturiert

## Mehrere Nutzergruppen berücksichtigen

Eine Herausforderung bestand darin, die unterschiedlichen Anforderungen und Nutzungskontexte der beiden Zielgruppen – **Qualitätssicherung** und **Entwicklungsabteilung** – zu berücksichtigen. Deswegen wurden **User Flows** und **User Journey Maps** für beide Gruppen entwickelt, um Aufgaben, Bedürfnisse und Pain Points systematisch zu erfassen und im UI-Design zu berücksichtigen.



## PHASE 2: DEFINE

# Problemraum: Dokumentation der Anforderungen

### Erfordernisse & priorisierte Nutzungsanforderungen

- Erfordernisse ergeben sich aus Nutzungskontext, User Flows & Journeys
- Aus den Erfordernissen wurden Nutzungsanforderungen abgeleitet
- Priorisierung nach Dringlichkeit, Kritikalität und Nutzungs frequenz
- Grundlage für spätere UI-Entscheidungen

### Abgeleitete Aufgabenanalyse

- Analyse typischer Aufgaben: Ziel, Inhalt, Häufigkeit und Kritikalität
- Differenzierung nach Nutzergruppen (Qualitätssicherung & Entwicklung)
- Identifikation repetitiver und besonders wichtiger Tätigkeiten
- Unterstützung bei der Definition von UI-Schwerpunkten und Optimierungspotenzialen

E1	<p>Die Entwicklungsabteilung muss ein Fahrzeug für einen bestimmten Zeitraum reservieren können, um eine geplante Testfahrt durchzuführen.</p> <p>Die Entwicklungsabteilung muss wissen, welche Fahrzeuge zu einem bestimmten Zeitpunkt verfügbar sind, um passende Testfahrzeuge reservieren zu können.</p> <p>Die Entwicklungsabteilung muss Mängel an Fahrzeugen erfassen können, um Qualitätsprobleme zu dokumentieren.</p> <p>Die Qualitätssicherung muss erfasste Mängel verifizieren und verifizierte Mängel gezielt weiterleiten können, um die zuständige Abteilung informieren zu können, damit Mängel behoben werden.</p> <p>Die Qualitätssicherung muss den Bearbeitungsstatus eines Mangels kennen, um über den Fortschritt der Bearbeitung informiert zu sein.</p> <p>Die Entwicklungsabteilung muss wissen, welche Eigenschaften (z. B. Ausstattung, Modell) ein Fahrzeug hat, um ein geeignetes Fahrzeug für eine Testfahrt auswählen zu können.</p> <p>Die Entwicklungsabteilung muss wissen, dass Bilder die Mängeldokumentation unterstützen, um Mängel nachvollziehbar dokumentieren zu können.</p>	<p>NA1</p> <p>NA2</p> <p>NA3</p> <p>NA4</p> <p>NA5</p> <p>NA6</p> <p>NA7</p>	<p>Die Entwicklungsabteilung muss am System eine <b>Fahrzeugreservierung</b> eingeben und speichern können.</p> <p>Die Entwicklungsabteilung muss am System die <b>Verfügbarkeit von Fahrzeugen</b> erkennen können.</p> <p>Die Entwicklungsabteilung muss am System <b>Mängel eingeben, mit einem Fahrzeug verknüpfen</b> und <b>kategorisieren</b> können.</p> <p>Die Qualitätssicherung muss am System erfasste Mängel <b>verifizieren</b> und <b>verifizierte Mängel weitergeben</b> können.</p> <p>Die Qualitätssicherung muss am System den <b>Bearbeitungsstatus eines Mangels</b> erkennen können.</p> <p>Die Entwicklungsabteilung muss am System <b>Fahrzeugdetails erkennen</b> und <b>filtern</b> können.</p> <p>Die Entwicklungsabteilung muss am System <b>Bildmaterial zu einem Mangel hochladen</b> können.</p>	
	A2: Mangel erfassen	A3: Mangel verifizieren und weiterleiten	A4: Systemüberblick erhalten und Zustände einsehen	
Zeit buchen	Mangel korrekt dokumentieren	Einschätzung, ob und wohin Mangel gemeldet wird	Einen ganzheitlichen Überblick über Fahrzeugstatus, Mängel und Reservierungen erhalten	
Fahrzeug verfügbar ist	Grundlage für spätere Bewertung durch QS und Entwicklung	Filterung relevanter Probleme für Entwickler	Rechtzeitig Maßnahmen einleiten und einen reibungslosen Ablauf gewährleisten	
Fahrzeugmodell auswählen	Kategorie, Beschreibung, Dringlichkeit, Zeit/Ort	Mangelbeschreibung, Historie, evtl. Bilder	Dashboard mit Informationen zu Fahrzeugverfügbarkeit, bevorstehenden Testfahrten, offenen Mängeln, Fahrzeugzustand, Erinnerungen, Prüfverläufen	
Fahrzeuge anzeigen,	Formular öffnen, Felder ausfüllen, ggf. Bild anhängen	Bewertung vornehmen, Dringlichkeit einschätzen, Abteilung auswählen	Überblick über Testfahrten/Fahrzeuge gewinnen; offene Mängel und deren Status einsehen; Erinnerungen empfangen; Fahrzeugzustände prüfen	
Überschneidungen	Fahrzeug muss ausgewählt sein, Berechtigung notwendig	Mangel muss bereits erfasst sein, klar erkennbar	Echtzeitdaten, rollenbasierte Zugriffe, übersichtliche Darstellung, Erinnerungssystem	
Frequenz	Regelmäßig, je nach Testbedarf	Mittel – abhängig von Testverlauf	Mittel	Hoch - täglich, mehrmals
Durchführungszeit	Kurz (1-3 Minuten)	Mittel (2-5 Minuten)	Mittel (3-6 Minuten)	Kurz bis mittel (je nach Detailtiefe der Prüfung)
Repetitivität	Hoch – Aufgabe wiederholt sich häufig	Mittel – jedes Mal leicht unterschiedlich	Mittel	Hoch
Wichtigkeit	Hoch – ohne Reservierung keine Testfahrten	Hoch – Grundlage für Sicherheits- und Qualitätsmaßnahmen	Hoch – beeinflusst Arbeitslast der Entwicklung	Hoch
Dringlichkeit	Mittel bis hoch – abhängig von Projektzeitplan	Hoch – besonders bei sicherheitskritischen Mängeln	Variabel – abhängig von Mangel	Mittel bis hoch
Kritikalität	Mittel – Fehler können zu Ressourcen-Engpässen führen	Hoch – fehlerhafte Erfassung kann zu unentdeckten Problemen führen	Hoch bei sicherheitsrelevanten Mängeln	Hoch – betrifft Fahrzeugzustand und Mängelerfassung

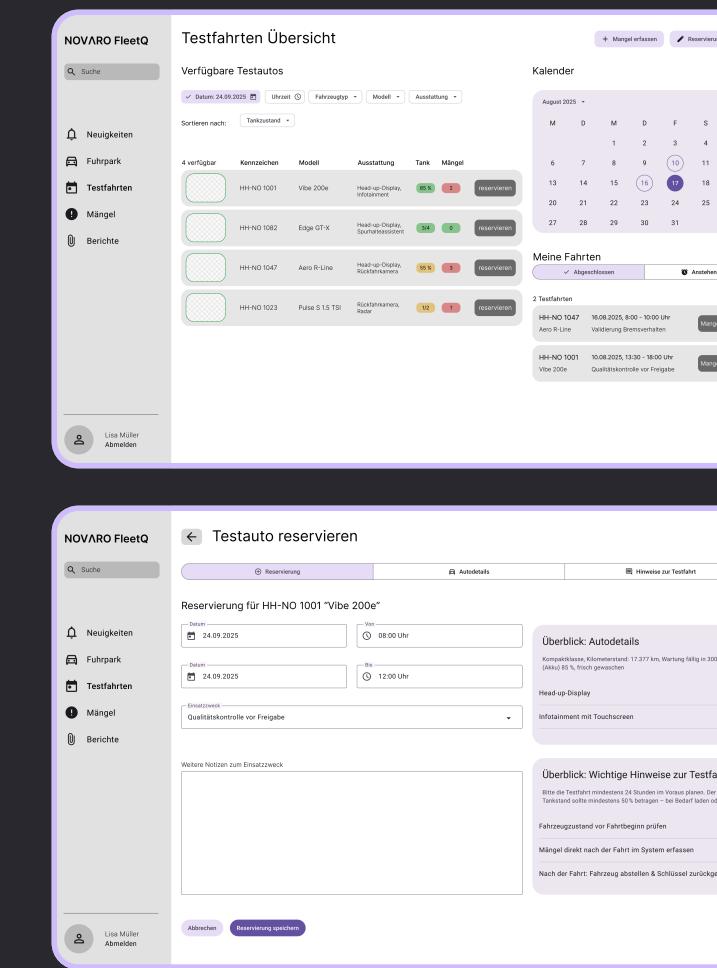
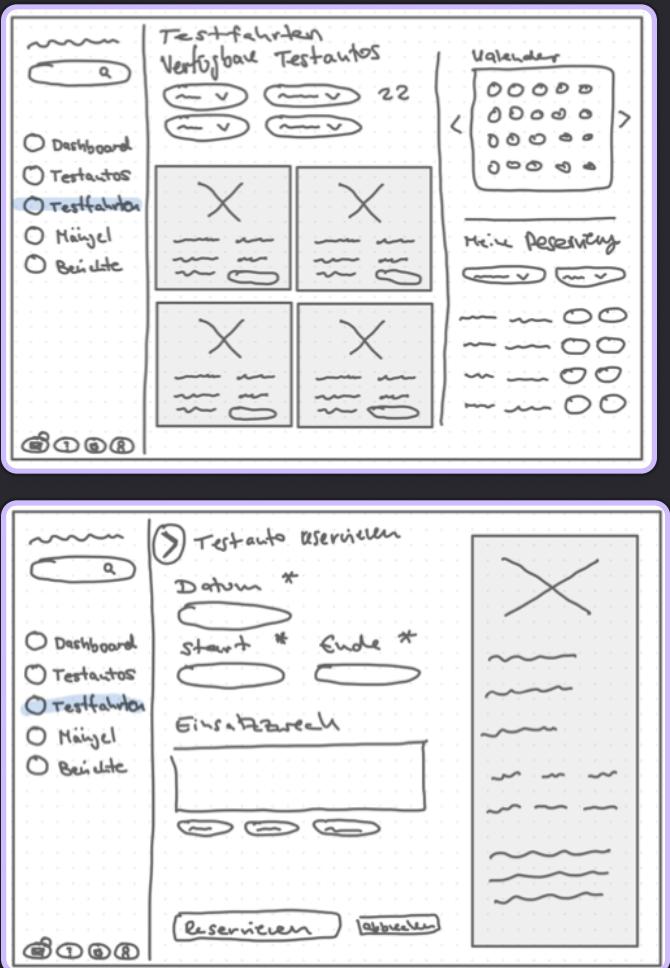
## PHASE 3: DEVELOP

# Lösungsraum: Entwicklung der Gestaltlösungen



## Scribbles & Wireframes

Erste **Ideen** wurden als **einfache** Low-Fidelity-Scribbles skizziert, um **schnell** verschiedene Ideen zu generieren. Darauf aufbauend entstanden in Figma High-Fidelity-Wireframes zur Ausarbeitung der **Elemente** und des **Layouts**.



## Designsystem & Mockups

Ich habe **Material 3** verwendet, da es **erprobt** und **flexibel** ist und umfangreiche **Guidelines** bietet. Das System lässt sich einfach an meinen Kontext anpassen und das integrierte **Dark Theme** unterstützt eine **ergonomische** und moderne Nutzung.

## PHASE 3: DEVELOP

# Das Designsystem: Material 3

**Typografie**

**DisplayLarge**  
Display / Large / Primary · Roboto 57/64 Regular 100 -25

**DisplayMedium**  
Display / Medium / Primary · Roboto 45/52 Regular 100 0

**Display Small**  
Display / Small / Primary · Roboto 36/44 Regular 100 0

**Headline Large**  
Headline / Large / Primary · Roboto 32/48 Regular 100 0

**Headline Medium**  
Headline / Medium / Primary · Roboto 28/36 Regular 100 0

**Headline Small**  
Headline / Small / Primary · Roboto 24/32 Regular 100 0

**Title Large**  
Title / Large / Primary · Roboto 22/28 Regular 100 0

**Title Medium**  
Title / Medium / Primary · Roboto 16/24 Medium 100 +0.15

**Title Small**  
Title / Small / Primary · Roboto 14/20 Medium 100 +0.1

**Label Large**  
Label / Large / Primary · Roboto 14/20 Medium 100 +0.1

**Label Medium**  
Label / Medium / Primary · Roboto 12/16 Medium 100 +0.5

**Label Small**  
Label / Small / Primary · Roboto 11/16 Medium 100 +0.5

**Body Large**  
Body / Large / Primary · Roboto 16/24 Regular 100 +0.5

**Body Medium**  
Body / Medium / Primary · Roboto 14/20 Regular 100 +0.25

**Body Small**  
Body / Small / Primary · Roboto 12/16 Regular 100 +0.4

**Farben**

Primary	P-80	Secondary	S-80	Tertiary	T-80	Error	E-80
On Primary	P-20	On Secondary	S-20	On Tertiary	T-20	On Error	E-20
Primary Container	P-30	Secondary Container	S-30	Tertiary Container	T-30	Error Container	E-30
On Primary Container	P-90	On Secondary Container	S-90	On Tertiary Container	T-90	On Error Container	E-90

Primary Fixed	Primary Fixed Dim	Secondary Fixed	Secondary Fixed Dim	Tertiary Fixed	Tertiary Fixed Dim	Error	E-80
P-90	P-80	S-90	S-80	T-90	T-80	E-90	E-80
On Primary Fixed	P-10	On Secondary Fixed	S-10	On Tertiary Fixed	T-10		
On Primary Fixed Variant	S-30	On Secondary Fixed Variant	S-30	On Tertiary Fixed Variant	T-30		

Surface Dim	Surface	Surface Bright	Inverse Surface	N-90	
N-6	N-6	N-24	Inverse Surface	N-90	
Surf. Container Lowest N-4	Surf. Container Low N-10	Surf. Container N-12	Inverse On Surface N-20	N-20	
Surf. Container High N-17	Surf. Container Highest N-22		Inverse Primary P-40	P-40	
On Surface N-90	On Surface Var. NV-80	Outline NV-60	Outline Variant NV-30	Scrim N-0	Shadow N-0

**Eingabefelder & Dropdowns**

Datum: 24.09.2025 Datum: 24.09.2025

Kategorie: Elektronik

Kategorie: Elektronik

✓ Elektronik  
Antrieb  
Mechanik  
Fahrerassistenz

Kategorie: Elektronik

Cancel OK

**Cards & Status Chips**

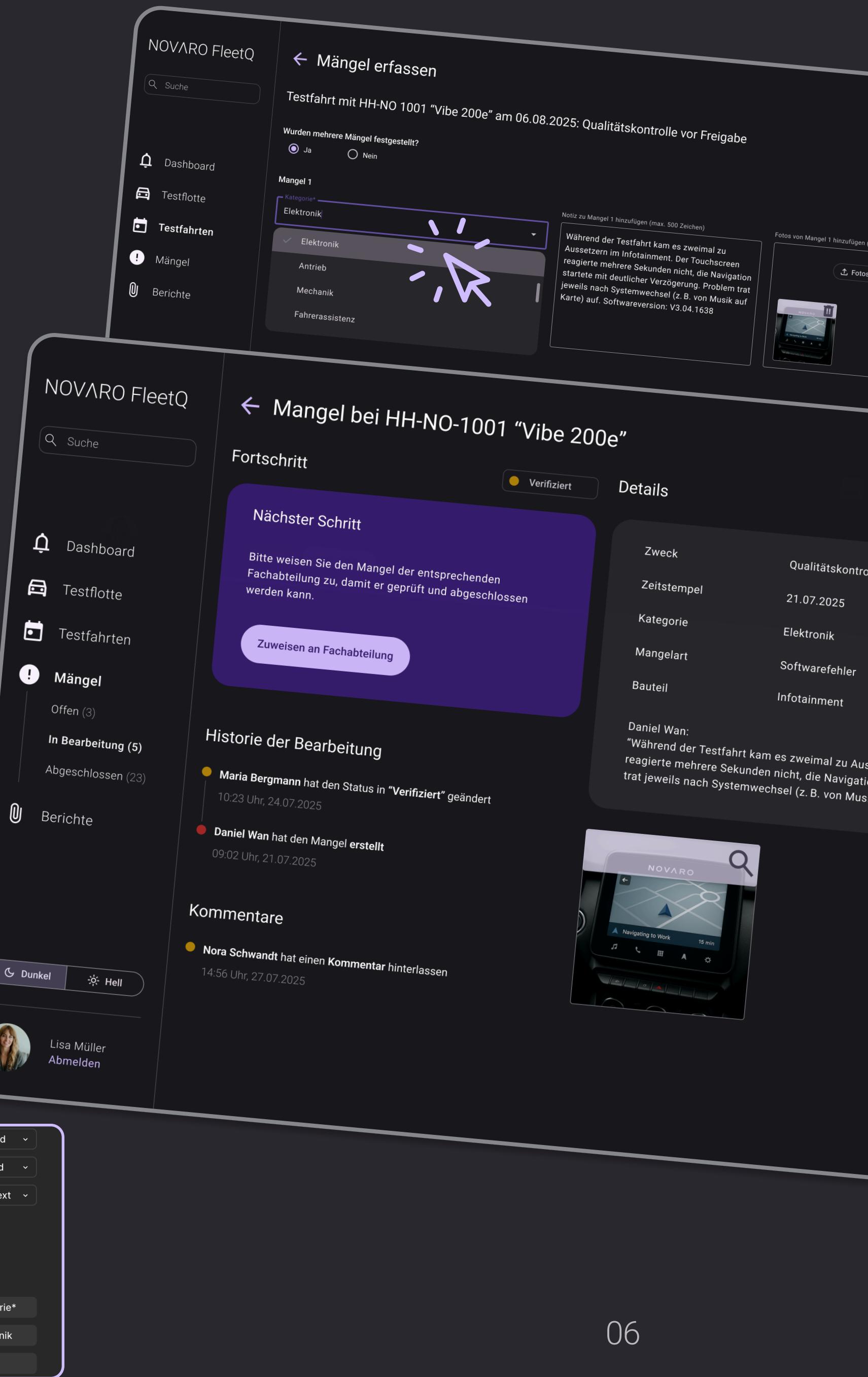
Cards: default & hover

HH-N0 1001 Vibe 200e Head-up-Display, Infotainment 2 Mängel 85 % Akku Reservieren

HH-N0 1001 Vibe 200e Head-up-Display, Infotainment 2 Mängel 85 % Akku Reservieren

Status Chips

Offen Verifiziert Zugewiesen Komfortentwicklung Abgeschlossen



← Mängel erfassen

Testfahrt mit HH-N0 1001 "Vibe 200e" am 06.08.2025: Qualitätskontrolle vor Freigabe

Wurden mehrere Mängel festgestellt?  Ja  Nein

Mangel 1

Kategorie\* Elektronik

✓ Elektronik  
Antrieb  
Mechanik  
Fahrerassistenz

Notiz zu Mangel 1 hinzufügen (max. 500 Zeichen)  
Während der Testfahrt kam es zweimal zu Aussetzern im Infotainment. Der Touchscreen reagierte mehrere Sekunden nicht, die Navigation startete mit deutlicher Verzögerung. Problem trat jeweils nach Systemwechsel (z. B. von Musik auf Karte) auf. Softwareversion: V3.04.1638

Fotos von Mangel 1 hinzufügen

← Mängel bei HH-N0-1001 "Vibe 200e"

Fortschritt

Verifiziert

Nächster Schritt

Bitte weisen Sie den Mangel der entsprechenden Fachabteilung zu, damit er geprüft und abgeschlossen werden kann.

Zuweisen an Fachabteilung

Zweck: Qualitätsskontrolle

Zeitstempel: 21.07.2025

Kategorie: Elektronik

Mängelart: Softwarefehler

Bauteil: Infotainment

Daniel Wan:  
"Während der Testfahrt kam es zweimal zu Aussetzern im Infotainment. Der Touchscreen reagierte mehrere Sekunden nicht, die Navigation startete mit deutlicher Verzögerung. Problem trat jeweils nach Systemwechsel (z. B. von Musik auf Karte) auf. Softwareversion: V3.04.1638"

Historte der Bearbeitung

- Maria Bergmann hat den Status in "Verifiziert" geändert 10:23 Uhr, 24.07.2025
- Daniel Wan hat den Mangel erstellt 09:02 Uhr, 21.07.2025

Kommentare

- Nora Schwandt hat einen Kommentar hinterlassen 14:56 Uhr, 27.07.2025

Dunkel Hell

Lisa Müller Abmelden

Style: Outlined State: Enabled Text configuration: Input text

Leading icon: Off  
Trailing icon: On  
Show support...: On

Label text: Kategorie\*

Input text: Elektronik

Supporting text:

Komponenten mit verschiedenen Varianten und Eigenschaften (Text, Boolean, Instanzen)

## FAZIT

# Lessons learned



## User Research

Aufgrund des **Zeitrahmens** konnte kein konkreter User Research betrieben werden. Zukünftig sollte dies jedoch immer stattfinden, um **Bedürfnisse** systematisch zu erheben und **Entscheidungen** auf Datenbasis zu treffen.



## Anforderungsmanagement

Die Anforderungen wurden übersichtlich in einer **Tabelle** dokumentiert. Für kommende Projekte könnte ein **interaktives Tool** wie z.B. **Notion** helfen, Anforderungen noch flexibler und nachvollziehbarer zu verwalten.



## Scribbles & Variantenbildung

Die frühe Phase der **Ideenentwicklung** wurde bewusst **schlank** gehalten. Eine erweiterte Variantenbildung kann künftig dazu beitragen, noch mehr gestalterische **Möglichkeiten** auszuloten.

