

# Anhang

## I. Einrichtung & Installation

Dieses Kapitel enthält eine Schritt-für-Schritt Anleitung zur Nutzung des im Rahmen des Projekts entwickelten Demonstratorsystems. Je nach Art der Datenübertragung sind unter Umständen umfassende Erweiterungen des Programmcodes für die Inbetriebnahme notwendig.

### What's in the Package?

Die ZIP-Datei enthält neben dem Code des NuR.E-Systems auch verschiedene Konfigurationsdateien. Diese müssen nicht zwingend genutzt werden, können aber als Orientierung dienen. Zusätzlich liegt eine .sql-Datei bei, welche das für das DemonstratorSystem genutzte Datenbankschema widerspiegelt. Das Datenbankschema ist sehr konkret auf das DemonstratorSystem und die Art und Weise der Datenübertragung zugeschnitten und muss angepasst werden.

Das ZIP-Archiv enthält die folgenden Ordner:

client-app

Enthält den Source Code der Client-App. Um mit diesem arbeiten zu können, müssen zunächst mit dem Befehl *npm install* alle Abhängigkeiten installiert werden. Anschließend kann mit dem Befehl *npm run build* ein dist-Verzeichnis erzeugt werden, welches dann auf den Webserver kopiert werden kann.

## server

Enthält sowohl den Source Code für die GraphQL-API als auch für die Datenaufbereitung:

- cockpit-server
- lean-data-processing-service

Auch hier müssen jeweils mit *npm install* zunächst die notwendigen Abhängigkeiten installiert und der Code mit *npm run build* kompiliert werden.

## configs

Enthält die folgenden Konfigurationsdateien und Vorlagen:

- nginxConfig  
Enthält eine beispielhafte NGINX-Konfiguration. Um diese zu nutzen, müssen der *server\_name* geändert und ein SSL-Zertifikat hinzugefügt werden.
- exampleKioskSettings.xml  
Konfigurationsdatei für die Kiosk Pro App. Dient nur als Vorlage, eine genaue Anleitung zur Nutzung ist dem Kapitel *Tablet-Konfiguration* zu entnehmen.
- datenbank\_schema.sql  
SQL-Datei die automatisch alle relevanten Tabellen erzeugt. Je nach vorhandenen Daten-Kanälen müssen die Tabellen noch angepasst werden.

## Hardwareanforderungen

In erster Linie wird ein iPad Mini benötigt. Das entwickelte Interface ist für dieses Gerät optimiert und funktioniert auf anderen Geräten unter Umständen nur eingeschränkt. Sollte also ein anderes Gerät genutzt werden, sind gegebenenfalls noch Änderungen am Code der Client-App

notwendig. Für die Installation im Bus ist zudem eine entsprechende Halterung notwendig, mit welcher das Tablet so im Bus positioniert werden kann, sodass es die Sicht des Fahrpersonals nicht behindert, aber dennoch einfach zu betrachten und zu bedienen ist. Bestenfalls ist die Halterung auch diebstahlsicher. Das iPad benötigt natürlich auch eine Internetverbindung. Sollte dies nicht über einen bereits im Bus vorhandenen WLAN-Hotspot möglich sein, muss das iPad zusätzlich mit einer SIM-Karte mit ausreichend Datenvolumen ausgestattet werden. Neben einer Internetverbindung ist auch eine durchgehende Stromversorgung notwendig.

Zusätzlich zu dem iPad ist auch ein Server für die Datenbank und den Anwendungsserver notwendig. Je nach Anzahl der Busse sind die Hardwareanforderungen an diesen unterschiedlich hoch. Für den Testbetrieb mit einem einzelnen Bus wurde ein v-Server mit 1GB Ram sowie einer Intel Xeon E5420-CPU mit einer Taktung von 2,5 GHz genutzt. Dieser Server war im Schnitt zu etwa 2% ausgelastet.

### **Einrichtung des Servers & Softwareinstallation**

Für den Server empfiehlt sich die Nutzung eines Debian-Derivates, wie z. B. Ubuntu, als Betriebssystem. Das System wurde ausschließlich auf einem Ubuntu-Server getestet. Auf anderen Betriebssystemen sind gegebenenfalls Änderungen notwendig.

Auf einem frischen Ubuntu-Server müssen zunächst die folgenden Pakete installiert werden:

- MySQL Server
- Node.js
- Redis
- PM2 (optional)
- Webserver (z. B. NGINX)

Für alle diese Pakete findet sich eine Vielzahl an Installationsanleitungen im Internet. Das Paket PM2 ist optional und sorgt dafür, dass das NuR.E-System nach jedem Server-Neustart automatisch wieder gestar-

tet wird. Dies kann jedoch auch anders gelöst werden. Auch muss nicht zwingend NGINX verwendet werden, es eignen sich durchaus auch Alternativen wie beispielsweise Apache. Für NGINX liegt jedoch eine Beispielkonfiguration bei.

Nach der grundlegenden Einrichtung müssen sowohl Server- als auch Client-Code auf den Server kopiert werden. Anschließend im Server-Verzeichnis mit dem Befehl *npm install* alle notwendigen Abhängigkeiten installieren.

### Datenbank-Konfiguration

Die Konfiguration der Datenbank hängt vor allem davon ab, wie die Fahrdaten der Busse in das System gelangen. Das ist vorrangig vom Hersteller des Busses anhängig. Im Falle des entwickelten Demonstratorsystems werden die Fahrdaten zunächst an der Hersteller gesendet und dort in einer MySQL-Datenbank abgelegt. Das Übertragen der Daten von dort ans das NuR.E-System gestaltete sich aufgrund der IT-Infrastruktur des Busherstellers erstmal schwierig. Als einzige mögliche Lösung mussten die Daten mit Hilfe der MySQL Federated Storage Engine<sup>14</sup> übertragen werden. Der große Vorteil dieser Methode besteht darin, dass keine Eingriffe in die IT-Infrastruktur des Herstellers erforderlich sind und nur minimale Änderungen der Datenbank notwendig sind. Es müssen lediglich neue Tabellen angelegt werden, welche die Federated Storage Engine nutzen. Sieht das Datenbankschema für jeden Bus eine Tabelle vor, muss entsprechend für jeden Bus eine weitere Tabelle angelegt werden. Dies war hier der Fall. Über MySQL-Trigger werden die Daten dann, sobald sie vom Bus gesendet und in der jeweiligen Tabelle abgelegt wurden, automatisch in die entsprechende Federated-Tabelle kopiert.

Für jede Federated-Tabelle in der Datenbank des Herstellers, muss dann eine Zieltabelle mit dem gleichen Schema in der NuR.E-Datenbank existieren. Über die Federated Storage Engine werden die Daten vollautomatisch in diese übertragen. Die Tabellen in der NuR.E-Datenbank dür-

---

14 Für weitere Informationen siehe: <https://dev.mysql.com/doc/refman/8.0/en/federated-storage-engine.html> (abgerufen am 19.12.2019)

fen dabei nicht die Federated Storage Engine nutzen, es können dementsprechend ganz reguläre InnoDB- oder MyISAM-Tabellen dafür erstellt werden. Werden die Rohdaten anschließend nicht mehr benötigt, kann auch die Blackhole-Engine genutzt werden. Dann werden die Daten nicht gespeichert, sondern nur ins Binary Log geschrieben. Dies spart Platz und benötigt weniger CPU-Leistung. Damit der Node-Server neue Datenpunkte möglichst verzögerungsfrei bekommt, gibt es im wesentlichen zwei Möglichkeiten. Zum einen kann über ein Polling der entsprechenden Tabellen herausgefunden werden, ob ein neuer Datensatz hinzugekommen ist. Diese Methode ist jedoch, was die benötigte Rechenleistung angeht, relativ aufwendig. Zudem entsteht dadurch eine zusätzliche Latenz. Eine bessere Variante ist das *Binary Log*. Das Binary Log schreibt jede auftretende Änderung in der Datenbank oder in spezifizierten Tabellen in eine Textdatei. Der Node-Server kann auf dieses Binary Log dann einen *Listener* setzen, der über alle Änderungen automatisch und ohne große Verzögerung informiert wird. Das Binary Log muss zunächst in der MySQL-Konfigurationsdatei aktiviert werden. Es empfiehlt sich zudem, nur die relevanten Tabellen auch zu loggen, um unnötige Last auf den MySQL-Server zu vermeiden.

Diese Lösung funktioniert grundlegend sehr gut. Für den Betrieb von Systemen mit Echtzeit-Daten sollte und muss die Datenübertragung aufgrund der so entstehenden Latenz jedoch anders gelöst werden. Wie bereits angesprochen kann das beste Ergebnis erzielt werden, wenn die Daten vom Bus direkt an einen Endpoint des NuR.E-System gesendet werden, von dem aus die Daten dann weiterverarbeitet werden können.

Neben der speziell für die Datenübertragung notwendigen Konfiguration, sind bei einer erstmaligen Einrichtung eines MySQL-Servers die Einstellungen auf die Hardware anzupassen. Außerdem muss für das NuR.E-System ein Benutzerkonto mit Lese- und Schreibrechten für sämtliche genutzten Datenbanken eingerichtet werden.

## Anbindung anderer Datenquellen

Das NuR.E-System ist darauf ausgelegt, auch mit anderen Datenquellen arbeiten zu können. Bei der Anbindung neuer Datenquellen müssen je-

doch einige Dinge beachtet werden. Zunächst müssen die Daten in das System gelangen. Dazu kann beispielsweise ein Service entwickelt werden, der nur die Daten annimmt und aufbereitet und anschließend über Redis dem NuR.E-System zur Verfügung stellt. Durch die Nutzung von Redis spielt es keine Rolle, mit welchen Programmiersprachen dieser Service entwickelt wird, solange ein Redis-Client<sup>15</sup> für die gewählte Programmiersprache verfügbar ist.

Nachdem der Service die Daten angenommen hat, müssen die Datenkanäle gemappt werden. Für eine korrekte Funktionsweise des Systems werden die folgenden Datenkanäle benötigt:

- QRemain\_Ah (Batterieladestand in Ah)
- UInvL\_V (Spannung linke Batterien in V)
- UInvR\_V (Spannung rechte Batterien in V)
- PMot\_kw (Motorleistung in kW)
- STravel\_km (Zurückgelegte Strecke in km)

Die Fahrdaten müssen dann in Form eines JSON-Strings über Redis im Topic „vehicleData“ gepublished werden. Das JSON-Objekt hat zwei Felder. Das Feld „tablename“ enthält die ID des Busses, das Feld „data“ die eigentlichen Fahrdaten.

### **Tablet-Konfiguration**

Bei der Konfiguration des Tablets müssen einige Dinge beachtet werden. So muss sichergestellt werden, dass auch nach einem Neustart des Tablets automatisch der Browser mit der entsprechenden URL geöffnet wird. Ein Neustart kann zum Beispiel dann auftreten, wenn der Akku über Nacht leerläuft. Weiterhin sollten alle Hardware-Tasten deaktiviert und die Nutzung auf den Browser mit der entsprechenden Webseite limitiert werden. Um dies zu realisieren gibt es viele verschiedene Möglichkeiten. Die besten Ergebnisse liefert eine Kombination aus dem *Single App Mode* und der *Kiosk Pro Plus*-App. Daher werden an dieser Stelle die für eine

---

15 Liste verfügbarer Redis-Clients: <https://redis.io/clients> (abgerufen am 19.12.2019)

korrekte Konfiguration des Single App Mode und der Kiosk Pro Plus App notwendigen Schritte beschrieben.

Für die Konfiguration des Single App Mode ist ein MAC sowie die kostenlose Software *Apple Configurator 2* notwendig. Für die korrekte Konfiguration des Single App Mode müssen folgende Schritte befolgt werden:

1. iPad über USB mit dem MAC verbinden
2. Im Apple Configurator mit der rechten Maustaste auf das iPad klicken und „Prepare“ auswählen
3. „Manual Configuration“ und „Supervise Devices“ auswählen und den Anweisungen folgen
4. In den iPad-Einstellungen den Sperrbildschirm deaktivieren
5. Im Apple Configurator mit der rechten Maustaste auf das iPad klicken und „Advanced -> Start Single App Mode“ auswählen
6. Kiosk Pro Plus auswählen und anschließend auf „Options“ klicken
7. Motion, Volume Buttons, Side Switch, Sleep/Wake Button und Auto Lock deaktivieren
8. Kiosk Pro Plus konfigurieren

Für die Basiskonfiguration kann die beiliegende Konfigurationsdatei genutzt werden. Diese muss auf einem Webserver abgelegt werden. Anschließend muss in den *Kiosk Pro Settings* die Option „Remote Settings Control“ aktiviert und konfiguriert werden<sup>16</sup>. Nachdem die Kiosk Pro Settings mit dem Server synchronisiert wurden, muss die Option vorerst wieder deaktiviert werden. Nun können die Kiosk Pro Settings so angepasst werden, wie sie benötigt werden.

Im folgenden sind alle relevanten Einstellungen, welche modifiziert werden müssen, aufgelistet:

- UniquePadID  
Wird zur Identifizierung des Busses genutzt

16 Für weitere Informationen siehe: <https://docs.kioskproapp.com/article/747-remote-settings-control> (abgerufen am 19.12.2019)

- settingsPassCode
- homePage  
Hat die Form `https://example.com?id=username&token=password`
- externalSettingsFile  
Hat die Form `https://example.com/kiosk_settings/$UniquePadID$.xml?token=password`

Ist die Konfiguration abgeschlossen, können die Einstellungen exportiert werden. Die so entstehende XML-Datei kann die auf dem Webserver liegende Datei ersetzen. Zum Schluss kann die Option „Remote Settings Control“ wieder aktiviert werden. Dadurch können Änderungen an der Konfiguration auch aus der Ferne aufgespielt werden. Wichtig ist, dass Änderungen an der Konfiguration nun nur noch über die auf einem Webserver liegende XML-Datei möglich sind. In der App vorgenommenen Änderungen werden vom Server wieder überschrieben, sofern die Option Remote Settings Control aktiviert ist. Der Name der XML-Datei sollte der UniquePadID entsprechen.

### **Verknüpfung aller Komponenten**

Um die einzelnen Komponenten miteinander zu verknüpfen, müssen entsprechend alle IPs/URLs und Zugangsdaten angepasst werden. In der folgenden Liste sind alle notwendigen Anpassungen aufgelistet.

- Sowohl im cockpit-server als auch im data-processing-service befinden sich config-Verzeichnisse mit Konfigurationsdateien. Hier müssen die Zugangsdaten zur Datenbank eingetragen werden.
- Im lean-cockpit-Verzeichnis muss in den .env-Dateien die IP/ URL der GraphQL-API eingetragen werden. Sollte die Client App nicht im root-Verzeichnis liegen, muss gegebenenfalls noch die PUBLIC\_PATH-Variable angepasst werden. Ist die App unter `www.example.com/nure` erreichbar, muss PUBLIC\_PATH den Wert „/nure“ haben.
- Die Authentifizierung des Clients läuft über einen Query-String in der URL. Wird die App mit der URL `www.example.com?id=`

user&token=password aufgerufen, werden die Werte „user“ und „password“ zur Authentifizierung genutzt. Dementsprechend muss dafür auch ein entsprechender Eintrag in der „Users“-Tabelle vorhanden sein. Das Passwort muss dabei als md5-Hash gespeichert werden. Auch die ID des Busses muss im Query-String mit Parameter „busId“ angegeben werden. Die URL mit allen Query-Parametern kann in der Kiosk Pro App als „Homepage“ eingetragen werden.

### **Live-Betrieb**

Für den Live-Betrieb müssen zunächst alle relevanten Services gestartet werden. Dies beinhaltet den MySQL-Server, Redis, den data-processing-Service sowie den NuR.E-Anwendungsserver. Zusätzlich muss sichergestellt sein, dass sowohl Server als auch Client von außen über den Webserver erreichbar sind. Nach dem erfolgreichen Starten aller Services kann auf dem Tablet das NuR.E-System aufgerufen werden. In einem ersten Schritt muss die zu fahrende Umlaufstrecke in Kilometern angegeben werden. Anschließend ist das Assistenzsystem einsatzbereit. Das Fahrpersonal wird automatisch zu Beginn der Fahrt aufgefordert, die Umlaufstrecke erneut einzugeben.

## **II. Weiterführende Informationen**

### **Verbrauchswertberechnung**

Grundsätzlich lässt sich der Verbrauch eines Fahrzeugs auf zwei Arten darstellen: in Form eines Momentanverbrauchs oder eines Durchschnittsverbrauchs über einer Referenzperiode. Der Momentanverbrauch gibt eine direkte Rückmeldung zum Verbrauch in diesem Moment, also der vom Motor gelieferten elektrische Leistung zum Zeitpunkt der Messung. Der Wert zeigt maximal transparent, was in diesem Moment „verbraucht“ wird. Es ist aber auch stark volatil (bei der Beschleunigung ist der Wert sehr hoch, beim Rekuperieren sogar negativ, da Energie generiert wird) und es lassen sich nur indirekt Rückschlüsse auf den durchschnittlichen Verbrauch des Fahrzeugs ziehen.

Aus diesem Grund wird in den meisten Fahrzeugen ein Durchschnittsverbrauch über einer Referenzperiode angegeben. Statt einer Momentbetrachtung wird eine Referenzperiode herangezogen – die letzten 5 Kilometer, die letzten 10 Minuten, die letzte Fahrt auf dieser Strecke – und die dort verbrauchte Energie als Durchschnittsverbrauch angezeigt.

Neben diesen beiden Methoden kann auch eine dritte Möglichkeit genutzt werden: eine Verbrauchsprognose basierend auf dem geplanten Streckenprofil und äußereren Einflüssen. Im Gegensatz zu dem Momentanverbrauch (gegenwärtige Daten) und dem Durchschnittsverbrauch über eine Referenzperiode (vergangene Daten) wird hierbei versucht, zu erwartende Einflüsse einzubeziehen und daraus einen Verbrauch abzuleiten. Im Gegensatz zum Individualverkehr bietet der ÖPNV hier klare Vorteile, da die geplante Strecke genau bekannt ist und auch Umwelt-einflüsse (Verkehr, Fahrgastaufkommen, etc.) durch Erfahrungswerte besser eingeschätzt werden können.

Für Verbrauchswerte, sowohl ein Momentanverbrauch zu einem Zeitpunkt X als auch ein Durchschnittsverbrauch über eine Referenzperiode, gibt es verschiedene Berechnungswege. Um konsistente Ergebnisse und Vergleichswerte zu erhalten, sollte frühzeitig eine Variante definiert für die Realisierung von Assistenzsystemen definiert werden. Im Folgenden werden drei Varianten vorgestellt.

#### Verbrauch über Akkuladung

Eine Möglichkeit der Verbrauchsberechnung ist, die Differenz der Akkuladung zwischen zwei Zeitpunkten zu bestimmen und in das Verhältnis zur zurückgelegten Strecke zu setzen. Beispiel: bei Zeitpunkt A liegt eine Restakkukapazität von 100 kWh vor. Eine Stunde später (Zeitpunkt B) verbleiben noch 80 kWh und es wurden 50 km zurückgelegt. Der Verbrauch beträgt somit 20 kWh auf 50 km, also 0,4 kWh/km. Ein Momentanverbrauch könnte über eine kürzere Referenzperiode extrapoliert werden, z. B. eine Minute.

## Verbrauch über Leistung & Distanz

Eine zweite Möglichkeit bietet die am Motor anliegende Leistung. Sie kann über einen Zeitraum hinweg summiert werden und ergibt so einen Verbrauch über diesen Zeitraum.

## Verbrauch über Leistung & Geschwindigkeit

Ein direkterer Ansatz ergibt sich durch die Hochrechnung der Leistung. Hierbei wird nicht die Vergangenheit betrachtet (zum Beispiel der letzte Kilometer), sondern die aktuell anliegende Leistung hochgerechnet. Die Leistung an den beiden letzten Datenpunkten wird wieder gemittelt und mit der Dauer des Datenpunktes multipliziert. Die resultierenden kWh werden durch die zurückgelegte Distanz (in Kilometern) dividiert und somit auf einen Kilometer normiert (kWh/km).

Diese Zahl ist stark schwankend, da jeder Beschleunigungs- oder Rekuperationsvorgang direkt abgebildet wird.

## Beispiele für Nutzerstudien

### Anforderungsanalyse: Qualitative Interviewstudie + Fragebogen

Zur Analyse wie die Reichweiteninteraktion im Elektrobus stattfindet, also wie die Reichweite bewertet und gesteigert wird und welche Einflüsse dies erschweren bzw. unterstützen können, wurde eine Interviewstudie mit 10 Busfahrer:innen (2 weiblich; Alter:  $M = 48.4$ ;  $SD = 10.8$ ) mit Elektrobus Erfahrung (Fahrstunden:  $M = 14.6$ ;  $SD = 8.8$ ) durchgeführt. Die Interviewstudie dauerte insgesamt ca. 20 Minuten und beinhaltete sowohl Interviewfragen als auch quantitative Fragen auf einem Fragebogen. Die Fragen des Interviews zielten auf die Schwierigkeiten in der Abschätzung der Reichweite, die Kommunikation mit der Leitstelle, die gewünschten Informationen und die Strategien/Barrieren der Reichweitensteigerung ab. Im Fragebogen wurde neben demographischen Daten auch abgefragt zu welchen Dienstzeitpunkten (z. B. Dienstbeginn, bei Wendepausen) am meisten Zeit und am meisten Bedarf für Reichweiteninformationen besteht. Außerdem wurde eine Skala zur Eco-Driving

Motivation abgefragt. Die Teilnehmer:innen erhielten zu einem späteren Zeitpunkt einen weiteren Fragebogen (ca. 10 Minuten), in dem die *subjective range competence*-Skala<sup>17</sup> und die komfortable Reichweite<sup>18</sup> abgefragt wurde. Im Anschluss an die Nutzungskontextanalyse wurde ein erster Prototyp durch ein Video präsentiert und mithilfe von vier Fragen evaluiert (ca. 10 Minuten).

Die quantitativen Daten wurden mit SPSS bereinigt und ausgewertet, die Interviewantworten wurden transkribiert und mit der thematischen Analyse nach Braun und Clarke<sup>19</sup> ausgewertet. Durch die thematische Analyse werden angesprochene Themen kodiert, zusammengefasst und als Struktur angeordnet. Die finale Struktur ist in Abbildung 16 zu sehen.

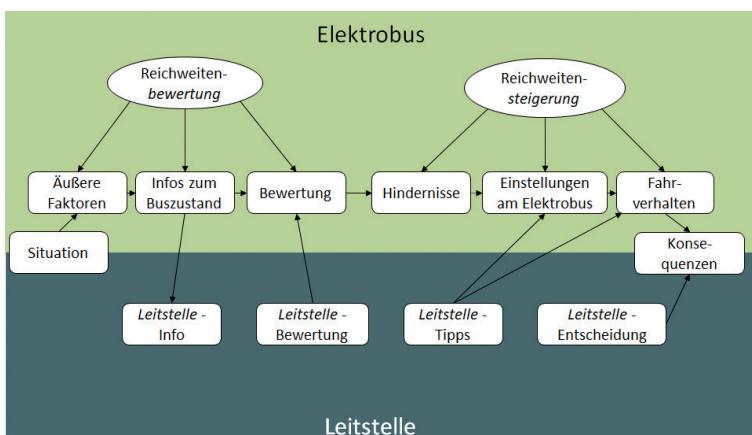


Abbildung 16. Themen der Reichweiteninteraktion im Elektrobus im ÖPNV

- 17 Franke, T., & Krems, J. F. (2013). Interacting with limited mobility resources: Psychological range levels in electric vehicle use. *Transportation Research Part A*, 48, 109–122.
- 18 Franke, T., Günther, M., Trantow, M., Rauh, N., & Krems, J. F. (2015). Range comfort zone of electric vehicle users – concept and assessment. *IET Intelligent Transport Systems*, 9(7), 740–745.
- 19 Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative Research in Psychology*, 3(2), 77–101.

Die Bewertung der Reichweite wird durch äußere Faktoren wie Fahrgastaufkommen oder dem Wetter erschwert. Andererseits werden Informationen zum Buszustand (z. B. Ladestand) als hilfreich empfunden und für die Bewertung herangezogen. Eine Reichweitensteigerung wird aus Sicht eines Teils des Fahrpersonals durch Hindernisse wie der Verkehrssituation erschwert. Die Reichweite kann sowohl durch Einstellungen am Elektrobus (Nebenverbraucher) als auch durch das Ändern des Fahrverhaltens gesteigert werden. Der gesamte Interaktionsprozess wird durch die Kommunikation mit der Leitstelle begleitet. Daneben gibt es noch einige Besonderheiten, die die Reichweiteninteraktion im ÖPNV charakterisieren. Zum einen wird der Elektrobus an einem Tag von drei Fahrer:innen gefahren, die sich die Batterieladung, durch die abwechselnde, gemeinsame Nutzung, teilen und somit die nachfolgenden Fahrer von den vorherigen abhängig sind. Außerdem gaben manche Fahrer:innen an, dass im ÖPNV kaum Steigerungspotential vorhanden ist, da bereits sparsam gefahren wird, und die Fahrer nichts weiter an dem Fahrverhalten ändern können.

Das Fahrpersonal gab an, dass sie die meiste Zeit für Informationen zur Reichweite zum Dienstbeginn ( $M = 5:30$  min;  $SD = 1:06$  min) und während der Wendepausen ( $M = 2:12$  min;  $SD = 1:06$  min) haben. Der meiste Bedarf (Skala: 1 = „gar kein Bedarf“ bis 4 = „hoher Bedarf“) bestehe aber während der Ablösung eines Kollegen ( $M = 3,3$ ;  $SD = 1,1$ ), zum Dienstbeginn ( $M = 3,2$ ;  $SD = 1,1$ ) und während der Wendepausen ( $M = 2,9$ ;  $SD = 0,9$ ). Wenn die aktuelle Reichweitenanzeige 100 km Restreichweite anzeigen würden, würde das Fahrpersonal mit gutem Gefühl noch  $M = 70,5$  km ( $SD = 9,6$ ) fahren (ComfDist100). Bei einer angezeigten Restreichweite von 50 km noch  $M = 27,0$  km ( $SD = 5,4$ ; ComfDist50).

Bei der subjective-range-competence Skala (SRC) wurde durchschnittlich ein Skalenwert von  $M = 3,6$  ( $SD = 0,6$ ) erreicht. In den Zusammenhangsmaßen der Variablen erreichte nur der Zusammenhang zwischen ComfDist50 und der bisherigen Linienbuserfahrung das Signifikanzniveau ( $r = -.70$ ;  $p < .05$ ) und der Zusammenhang von Motivation und ComfDist100 ( $r = .76$ ;  $p < .05$ ). Der Zusammenhang zwischen ComfDist100 und der SRC verfehlt das Signifikanzniveau knapp ( $r = .56$ ;  $p = .09$ ) und könnte möglicherweise bei einer größeren Stichprobe signifikant werden.

Die Ergebnisse deuten auf ein großes Unterstützungspotential durch nutzerzentrierte Systeme hin. Die komfortable Reichweite von ca. 70 km (von 100 km) zeigt wie groß der Sicherheitspuffer sein müsste, bevor das Fahrpersonal ein zusätzlichen Belastungsfaktor im Alltag erfahren würden. Die Themen der Reichweiteninteraktion (Abbildung 16) sollten bei der Einführung eines Systems unbedingt mitbetrachtet werden.

#### Evaluation von Systemkomponenten – Online-Experiment

Primäres Ziel dieser Evaluation ist die Beantwortung der Forschungsfrage: „Inwieweit kann das Vertrauen in die Reichweitenangabe durch Transparenz und Anpassbarkeit der Berechnung der Reichweitendarstellung erhöht werden?“ Für die Evaluation lassen sich zwei Hypothesen ableiten:

1. Eine transparente Darstellung der Reichweitenberechnung hat im Vergleich zu einer Reichweitenangabe, deren Herleitung nicht erkennlich ist, einen signifikanten positiven Effekt auf das subjektive Vertrauen in die Reichweitenangabe und die komfortable Reichweite.
2. Die Möglichkeit, die Reichweitenberechnung zu anzupassen, hat im Vergleich zu einer Reichweitenangabe, die nicht angepasst werden kann, einen signifikanten positiven Effekt auf das subjektive Vertrauen in die Reichweitenangabe und die komfortable Reichweite.

Die Studie wurde als Teil einer umfassenderen Online-Evaluation im Rahmen des Forschungsprojekts NuR.E in Form eines Fragebogens konzipiert. Vier kurze Videos ergänzen die textuellen Fragen und stellen Konzepte sowie den Kontext der Studie animiert dar. Die Zielgruppe der Befragung sind Elektroautofahrer (BEV); auf eine Evaluation mit Elektrobusfahrer:innen musste zeitlich und organisatorisch bedingt verzichtet werden.

Zur Akquise wurden fünf Facebookgruppen und fünf Foren mit Bezug zur Elektromobilität herangezogen, in denen ein Einladungstext und der Link zur Online-Evaluation gepostet wurden. Die Facebook-Beiträge

wurden um ein Logo ergänzt. Dadurch sollte die Sichtbarkeit des Beitrags und damit die Zahl der Leser erhöht werden.

Der A/B-Test bestand aus drei Varianten, die im Within-Subject-Design (Messwiederholungsdesign) getestet wurden. Um Reihenfolgeeffekte zu minimieren, wurden die Varianten randomisiert präsentiert. Anzeige A war eine numerische Angabe der Reichweite ohne weitere Informationen. Anzeige B enthielt Zusatzinformationen zur Reichweitenberechnung, welche die Reichweitenangabe transparenter gestaltete. Anzeige C bot schließlich die Möglichkeit, Einfluss auf die Berechnung zu nehmen und den Referenzverbrauch selbst zu wählen. Alle drei Anzeigenkonzepte wurden zunächst in jeweils einem Video erläutert. Anschließend wurden die abhängigen Variablen *Vertrauen* und *komfortable Reichweite* mittels 5 bzw. 2 Items erfasst, die bereits in einer anderen Studie in diesem Kontext eingesetzt wurden.

Die Ergebnisse des A/B-Tests wurden mittels einer einfaktoriellen Varianzanalyse mit Messwiederholung (repeated-measures ANOVA) in R analysiert. Die Anzeige in ihren drei Varianten bildete dabei den Faktor, der in jeder Messung verändert wurde. Die Signifikanz des Effekts zwischen den Anzeigevarianten wurde mittels Kontrasten bestimmt, um Rückschlüsse auf beide unabhängigen Variablen (UVs) zu erhalten. Eine Aussage über die Effektstärke wurde schließlich durch Pearson's *r* getroffen.

Ein erster Blick auf die deskriptiven Werte der Vertrauen-Skala zeigte das Bild, welches auch in den Hypothesen formuliert wurde: Das angegebene Systemvertrauen in die Reichweitenangabe war bei einer transparenteren (Anzeige B,  $M = 4.05$ ,  $SD = 1.10$ ) und anpassbaren Reichweitenberechnung (Anzeige C,  $M = 4.63$ ,  $SD = 1.04$ ) höher als bei einer reinen Reichweitenangabe als Zahl (Anzeige A,  $M = 3.68$ ,  $SD = 1.13$ ).

Mauchly's Test gab an, dass die Voraussetzung der Sphärizität verletzt wurde ( $\chi^2(94) = .71$ ,  $p < .001$ ), Greenhouse-Geisser relativierte dies jedoch ( $W = .71$ ,  $p < .001$ ). Die ANOVA zeigte, dass der Effekt signifikant war ( $p < .001$ ), und zwar laut der betrachteten Kontraste sowohl zwischen Anzeige A und B ( $p < .001$ ), Anzeige B und C ( $p < .001$ ) und An-

zeige A und C ( $p < .001$ ). Der Effekt war in beiden Fällen mittel ausgeprägt.

Bei der komfortablen Reichweite schien der Effekt ähnlich stark ausgeprägt wie bezüglich des Systemvertrauens. Die Mittelwerte der Anzeigen A und B wiesen einen geringfügig kleineren Unterschied auf als zwischen Anzeige B und C.

Auch hier zeigte Mauchly's Test eine Verletzung der Sphärizität an ( $\chi^2(94) = .55, p < .001$ ) und Greenhouse-Geisser berichtete ( $W = .69, p < .001$ ). Die ANOVA wies ebenfalls einen signifikanten Effekt aus ( $p < .001$ ), der auch in den beiden Kontrasten zu finden war ( $p_{AvsB} < .001; p_{BvsC} < .001$ ). Pearson's  $r$  zufolge war der Effekt stärker ausgeprägt als bei der Vertrauen-Skala ( $r = .40$  bzw.  $r = .45$ ), aber immer noch mittel.

### III. Tabellen

Tabelle 1: Auflistung der Datenkanäle

#	Kanal	Einheit (ideal)	Relevanz
1	Datum	DD.MM.YYYY	must-have
2	Zeit	HH:MM:SS,ms	must-have
3	Geografische Breite	Dezimalgrad	must-have
4	Geografische Länge	Dezimalgrad	must-have
5	Verbleibende Ladung	Ah	must-have
6	Leistung (am Motor anliegend)	kW	must-have
7	Spannung der Akkus	V	must-have
8	Drehzahl des Motors	rpm	must-have
9	Status der Zündung	codiert	should-have
10	Gaspedalstellung	Prozent	should-have
11	Bremspedalstellung	Prozent	should-have
12	Geschwindigkeit nach GPS-Daten	km/h	good-to-have

13	Fahrtrichtung nach GPS-Daten	Dezimalgrad	good-to-have
14	State of Charge / Akkuladung	Prozent	good-to-have
15	Genutzte Ladung (gesamt)	Ah	good-to-have
16	Generierte Ladung (gesamt)	Ah	good-to-have
17	Geschwindigkeit laut Fahrtenschreiber	km/h	good-to-have
18	Geschwindigkeit laut Tacho	km/h	good-to-have
19	Beschleunigung	m/s	good-to-have
20	Gefahrene Kilometer	Km	good-to-have
21	Verbleibende Reichweite	Km	good-to-have
22	Verbrauch pro Kilometer	kWh/km	good-to-have

**Tabelle 2. Indikatoren und Metriken für energieeffizientes Fahren**

#	Indikator	Berechnungslogik der Metrik	Erklärung
1	Gesamteffizienz	$\text{zurückgelegte Strecke} \div (\text{Ladestand}_{\text{Start}} - \text{Ladestand}_{\text{Ende}})$	Zeigt das Verhältnis von zurückgelegten Kilometer und der verbrauchten Energie an. Je besser die Energieeffizienz desto größer der Wert.
2	Energieverbrauch	$\Sigma \text{Energieverbrauch pro Datenpunkt} \div \Sigma \text{zurückgelegte Strecke pro Datenpunkt}$	Berechnet den Energieverbrauch. Je höher der Wert desto schlechter die Energieeffizienz.
3	Beschleunigungseffizienz	$\begin{aligned} &\text{Über alle Beschleunigungen hinweg:} \\ &\Sigma \text{Geschwindigkeitszuwachs} \div \Sigma \text{aufgewendete Leistung} \end{aligned}$	Zeigt das Verhältnis von dem Geschwindigkeitszuwachs und der dafür aufgewendeten Energie an. Je besser die Energieeffizienz, desto höher der Wert.
4	Schwungausnutzung	$\text{Streckenabschnitt ohne Gas- oder Bremspedal} \div \text{Gesamte Strecke}$	Zeigt das Verhältnis der Gesamtstrecke und der Streckenabschnitte, die ohne Benutzung des Gas- oder Bremspedals zurückgelegt wurden, an. Je besser die Energieeffizienz, desto höher der Wert.
5	Bremsen im Rekuperationsbereich	$\text{Streckenabschnitt im Rekuperationsbereich} \div \text{Gesamte Strecke}$	Zeigt das Verhältnis der Gesamtstrecke und der Streckenabschnitte, die im Rekuperationsbereich zurückgelegt wurden, an. Je besser die Energieeffizienz, desto höher der Wert.

#	Indikator	Berechnungslogik der Metrik	Erklärung
6	Beschleunigungsintensität	$(\sum \text{Beschleunigung in m/s}^2) \div \text{Distanz}$	= Seltenes und sanftes Beschleunigen. Zeigt die „Gesamtheit“ der Beschleunigungen an. Dieser Wert ist ein reiner „Verhaltens“-Wert und schließt den Energieverbrauch aus der Berechnung aus. Je besser die Energieeffizienz desto geringer der Wert.
7	Entschleunigungsintensität	$(\sum \text{neg. Beschleunigung in m/s}^2) \div \text{Distanz}$	= Seltenes und sanftes Verzögern. Zeigt die „Gesamtheit“ der Bremsvorgänge an. Dieser Wert ist ein reiner Verhaltens-Wert und schließt den Energieverbrauch aus der Berechnung aus. Je besser die Energieeffizienz desto geringer der Wert.
8	Energieflussintensität	$(\sum  \text{Leistung}  \times \text{Dauer}) \div \text{Distanz}$	= Vorausschauendes Fahren. Berechnet die Fläche unter der Kurve der am E-Motor anliegenden Leistung (sowohl positiv als auch negativ). Dadurch wird eine ruhige Fahrweise mit wenigen und sanften Be- und Entschleunigungen angezeigt. Je besser die Energieeffizienz desto geringer der Wert.