

Schriftliche Dokumentation der App

Fachhochschule Aachen
Fachbereich 5 – Elektrotechnik und Informationstechnik



Dokumentation der Scheibner-App

Mobile Informationssysteme
- Dr. ir. Kris Cardinaels -

03.07.2019

Björn Kops
David Neuroth
Felix Bengsch
Florentin Bürvenich

Inhaltsverzeichnis

1	Anforderungen (David)	1
1.1	Optionale Anforderungen	1
2	Funktionalität (Kps)	2
2.1	Profilverwaltung	2
2.2	Simulationsdaten laden	2
2.2.1	Laden vom Server	2
2.2.2	Laden aus QR-Code	2
2.3	Daten bearbeiten	2
2.4	Simulation	2
2.4.1	Diagramme	2
2.4.2	Tabelle	2
2.5	Kommentar	2
3	Navigation (David)	3
3.1	Ablauf einer Simulation	3
3.2	Wechseln des Profils	3
4	Einstellungen (brv)	4
4.1	Sprache	5
4.2	Eingabemethode	5
5	Benutzertests- und Interviews (Felix)	7
5.1	Benutzertest- und Interview 1	7
5.2	Benutzertest- und Interview 2	9
5.3	Benutzertest- und Interview 3	11
6	Technische Umsetzung (brv)	13
6.1	Speicherung der Profildaten	13
6.2	Speicherung der Einstellungen	15

6.3	Verwaltung des App-States	15
6.4	Daten hinzufügen	15
6.5	Diagramme	15

1 Anforderungen (David)

- Mehrere Profile - Eine Simulation pro Profil (wurde auch nur eine umgesetzt) -

1.1 Optionale Anforderungen

2 Funktionalität (Kps)

2.1 Profilverwaltung

2.2 Simulationsdaten laden

2.2.1 Laden vom Server

2.2.2 Laden aus QR-Code

2.3 Daten bearbeiten

2.4 Simulation

2.4.1 Diagramme

2.4.2 Tabelle

2.5 Kommentar

3 Navigation (David)

3.1 Ablauf einer Simulation

3.2 Wechseln des Profils

4 Einstellungen (brv)

Über das Einstellungs-menü lässt sich die App in zwei Punkten konfigurieren:

1. Einstellung der Sprache (s. ??)
2. Einstellung der Eingabemethode (s. ??)

Die Einstellungsseite wird je nach Ansicht, in der sich der Nutzer aktuell befindet, auf verschiedene Arten geöffnet:

1. **Profilverwaltung:** Die AppBar der Profilverwaltung enthält einen Button, der aus einem Zahnrad-Icon besteht und per direktem Klick die Einstellungsansicht öffnet (s. Abb. 4.1 links).
2. **Sonstige Ansichten:** Alle weiteren Ansichten (außer der Profilverwaltung) enthalten in der AppBar ein Dropdown (s. Abb. 4.1 Mitte), mithilfe dessen über den zweiten Menüpunkt die Einstellungsseite geöffnet werden kann (s. Abb. 4.1 rechts).

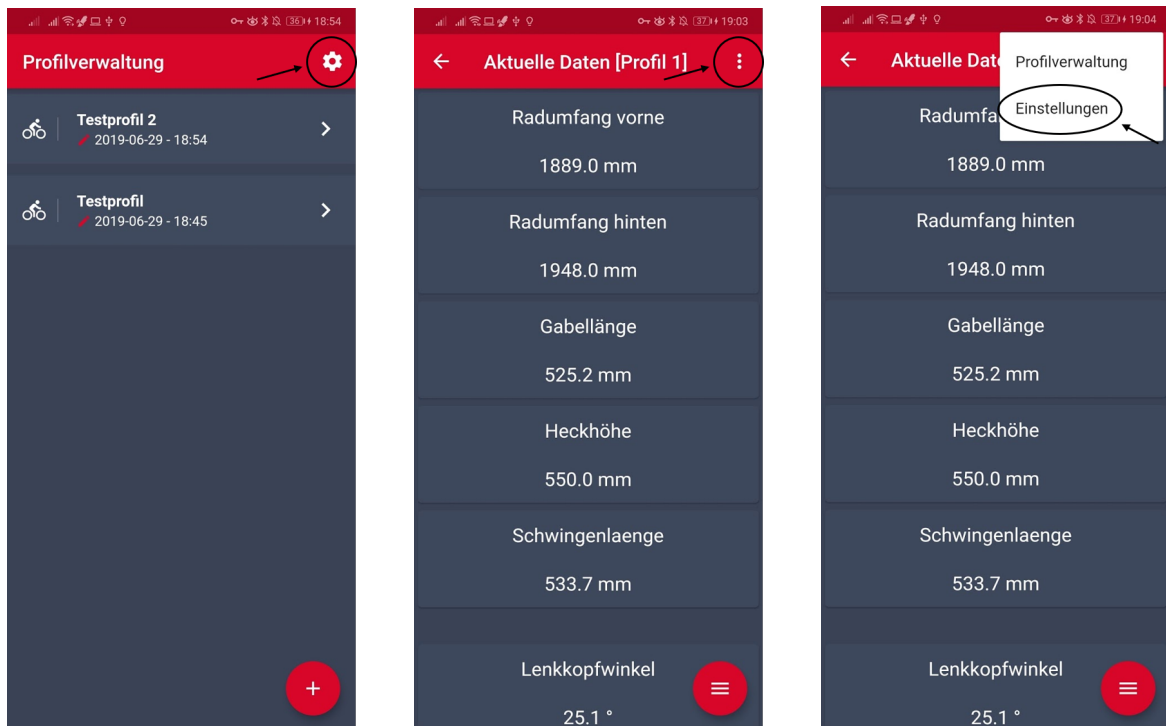


Abbildung 4.1: Links: Profilverwaltung, Mitte: Button für Dropdown-Menü, Rechts: Dropdown-Menü

4.1 Sprache

Die App ist in den Sprachen Deutsch und Englisch verfügbar. Die Einstellung wird standardmäßig auf die im Gerät eingestellte Sprache gesetzt und kann manuell verändert werden (s. Abb. 4.2 rechts)

4.2 Eingabemethode

Neue Werte, die von der App simuliert werden sollen, werden über die Ansicht SSimulation eingegeben. Hierfür kann in den Einstellungen angepasst werden, ob die Werte mit Textfeldern oder Slidern eingegeben werden sollen (s. Abb. 4.2 links).



Abbildung 4.2: Links: Einstellung der Eingabemethode, Rechts: Einstellung der Sprache

5 Benutzertests- und Interviews (Felix)

Es wurden drei Interviews mit Personen durchgeführt, welche die App bewerten sollen. Den Probanden wurde dabei jeweils nur erklärt, was der Nutzen einer Messung und Simulation von *MEGA m.a.x.* und wofür die App dienen soll, alles weitere mussten sich die Personen selbst erarbeiten. Damit sollte getestet werden, ob die App intuitiv benutzbar und gut zu bedienen sind. Die Probanden sollten die App jeweils an Hand dieser vier Punkte bewerten:

- Wie ist der erste Eindruck?
- Was ist positiv aufgefallen?
- Was ist negativ aufgefallen?
- Ist die App intuitiv bedienbar?

Nach der Beantwortung dieser Fragen wird jeweils noch beschrieben, welche Anpassungen auf Grund des Feedbacks des jeweiligen Interviews durchgeführt wurden.

5.1 Benutzertest- und Interview 1

Wie ist der erste Eindruck? Mein erster Eindruck der App ist sehr positiv. Die Corporate Identity ist konsequent umgesetzt worden und die Farben sind untereinander sehr stimmig. Des Weiteren ist die Geschwindigkeit der App sehr hoch, sodass man nie lang warten muss. Sogar die Simulation von Eingabewerten geht sehr schnell, obwohl ich mir vorstellen kann, dass sehr komplexe Berechnungen dahinterstehen.

Was ist positiv aufgefallen? Wie bereits erwähnt, ist mir die hohe Geschwindigkeit der App aufgefallen. Auch die übersichtliche Darstellung, sowohl der veränderbaren Werte, als auch der simulierten Werte, sorgt meiner Meinung nach dafür, dass die App in sich sehr aufgeräumt wirkt.

Was ist negativ aufgefallen? Ein Punkt, der mir nicht so gut gefallen hat ist, dass das Textfeld für den Kommentar nur eine Zeile hat. Drückt man innerhalb dieses Textfeldes auf “Enter”, so wird dieses um eine Zeile länger. Ich finde dies sehr unintuitiv, weil sonst die Gefahr besteht, dass der Benutzer denkt, dass dort nur einzeilige Kommentare möglich sind.

Ist die App intuitiv bedienbar? Nach der Einführung in *MEGA m.a.x.* und eine Erläuterung, wofür diese App dienen soll, konnte ich direkt loslegen und Profile anlegen sowie Simulationen durchführen. Es war keine weitere Anleitung notwendig, was sehr für die Intuitivität spricht.

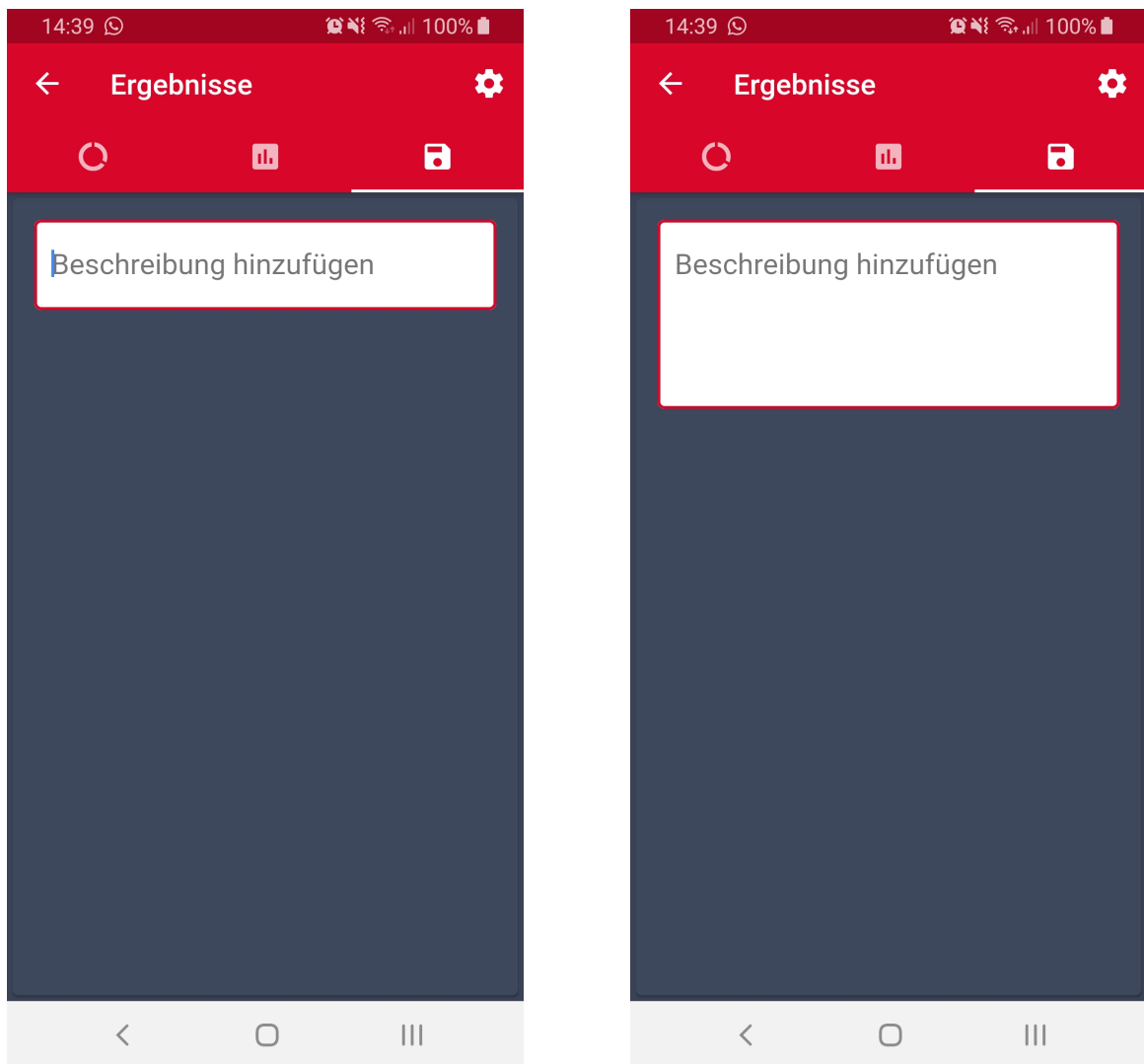


Abbildung 5.1: Links: Kommentarfeld vor dem Interview, rechts nach dem Interview

Nach Durchführung des Interviews wurde das Kommentarfeld so angepasst, dass es etwas größer ist, um Verwirrung zu vermeiden. Dies ist in ?? zu sehen. Um diesen Effekt zu erreichen, wurde die Eigenschaft `minLines` des `TextField`s auf den Wert 4 gesetzt, vorher war dieser nicht gesetzt.

5.2 Benutzertest- und Interview 2

Wie ist der erste Eindruck? Die App wirkt auf den ersten Blick sehr strukturiert und leicht zu bedienen. Wäre ich ein potentieller Kunde von Scheibner, würde es mir auf jeden Fall Spaß machen, mit dieser App zu arbeiten. Nach einiger Zeit habe ich die bildliche Darstellung der Simulationsergebnisse in Form von Diagrammen entdeckt, welche die Ergebnisse sehr anschaulich darstellen.

Was ist positiv aufgefallen? Mir haben die Einstellungsmöglichkeiten sehr gut gefallen. Es ist beispielsweise für einen Benutzer immer angenehmer, wenn die App automatisch in seiner Landessprache gestaltet ist und nicht in einer "Standardsprache" wie englisch. Weiterhin bietet die Einstellung, die Eingabe der veränderbaren Werte wahlweise durch Textfelder oder per Slider vorzunehmen, sehr viel Komfort. Will man einen sehr genauen Wert eingeben, so ist die Eingabe per Textfeld von Vorteil, soll nur eine ungefähre Simulation durchgeführt werden, so eignen sich die Slider besser, weil die Werte so schnell verändert werden können.

Ein weiterer Punkt, der mir sehr positiv aufgefallen ist, dass mehrere Fahrzeuge, sprich Profile in der App angelegt und verwaltet werden können. Dies bietet insbesondere Besitzern von mehreren Motorrädern die Möglichkeit, all ihre Fahrzeuge in einer App zu simulieren.

Was ist negativ aufgefallen? Was mir nicht gut gefallen hat, ist die Tatsache, dass die Einstellungen auf der Einstellungsseite nicht sehr gut zu erkennen waren. Das lag daran, dass die Texte schwarz und der Hintergrund ein sehr dunkles grau sind.

Ist die App intuitiv bedienbar? Da ich keinerlei weitere Anleitung benötigt habe, ist die App aus meiner Sicht intuitiv bedienbar. Für die Anwender, die vorher immer in eine Werkstatt vor Ort gehen mussten, um eine neue Simulation auszuführen, wird sie sicherlich einen hohen Mehrwert haben. Für die Werkstätten hat dies auch den Vorteil, dass sich die Mitarbeiter nicht mehr mit der stupiden Arbeit der Simulation befassen müssen:

Es reicht dann, die Messung durchzuführen, dem Kunden die ID der Messung mitzuteilen und dieser kann die Simulation bequem selbst zu Hause durchführen. Schließlich kann er mit seinen angepassten und fertig simulierten Werten zurück zur Werkstatt kommen, um sein Motorrad entsprechende umzurüsten.

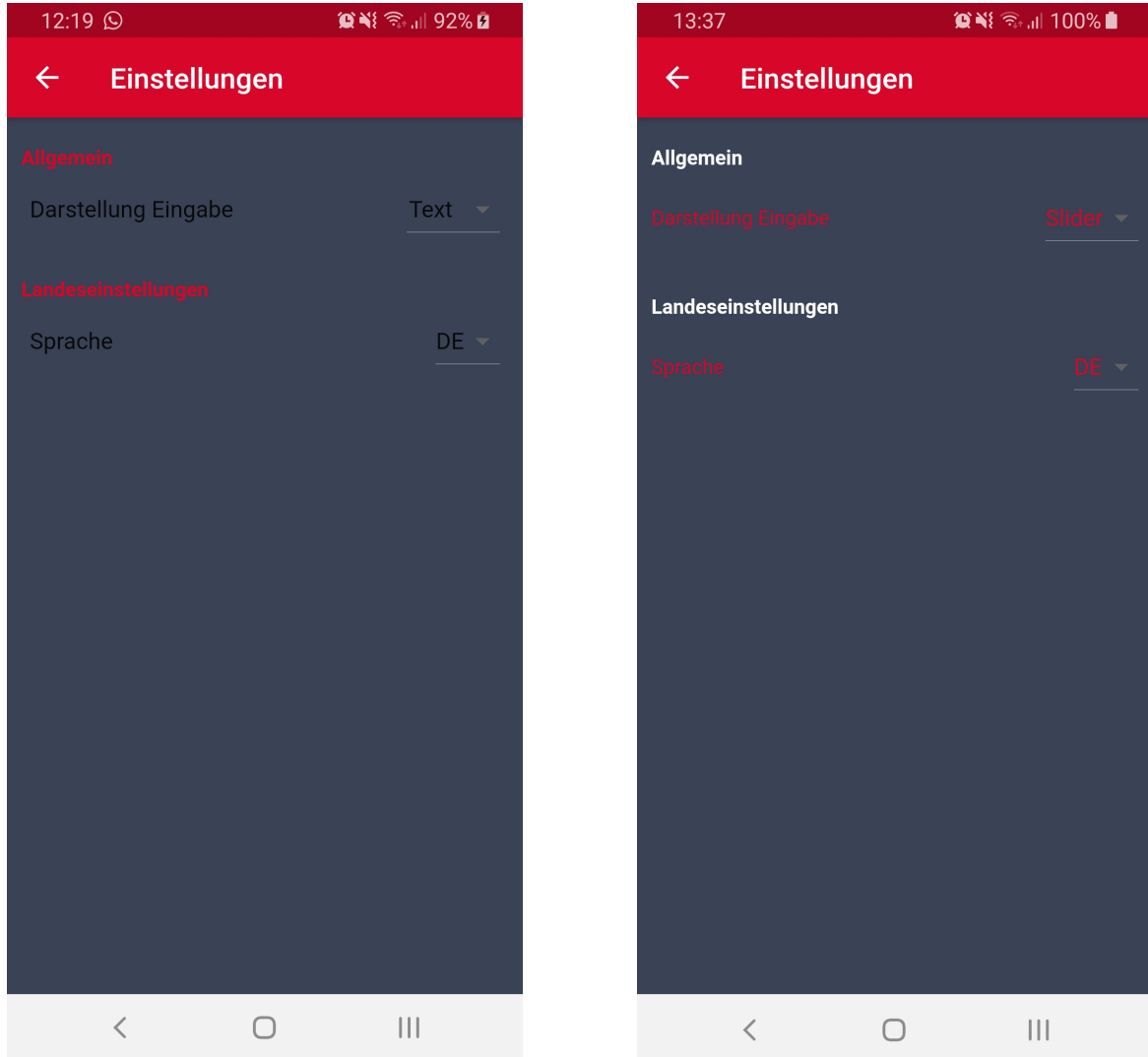


Abbildung 5.2: Links: Einstellungsseite vor dem Interview, rechts nach dem Interview

Nach dem Feedback wurde die Einstellungsseite so angepasst, dass der Text der einzelnen Kategorien weiß und der Text der einzelnen Einstellungen dunkelrot ist. Diese Anpassung zog einige Anpassungen nach sich, weil zuvor das Flutter-Paket *preferences* verwendet wurde. Dies lässt aber leider keine Konfiguration der Farben zu, sodass die komplette Einstellungsseite selbst programmiert werden musste. Der zu Grunde liegende *PrefService*, welcher auf die *SharedPreferences* zugreift, wird weiterhin verwendet. Das dyna-

mische Setzen der Bedienungselemente wie die Auswahlliste und das automatische Speichern der neuen Werte musste nachprogrammiert werden. Abb. 5.2 zeigt die Einstellungsseite vor und nach den Anpassungen

5.3 Benutzertest- und Interview 3

Wie ist der erste Eindruck? Die App macht auf mich einen sehr guten Eindruck, es gibt keinen Punkt, der mich extrem gestört hat. Die Darstellung der einzelnen Daten ist sehr strukturiert, außerdem beschränkt sich die App auf das Wesentliche und ist nicht überladen. Dadurch war es sehr angenehm, sie zu benutzen. Insbesondere gefällt mir die Farbgebung sehr gut, das Rot wirkt angenehm und auch die weiße Schrift in Kombination mit dem dunkelgrauen Hintergrund wirkt sehr stimmig.

Was ist positiv aufgefallen? Insgesamt hat mir die App sehr gut gefallen, sodass es keinen einzelnen Punkt gibt, den ich besonders hervorheben möchte. Die App als Ganzes wirkt in sich abgeschlossen und es gibt kein Element, welches in ihr “fremd” wirkt. Sie bietet einige Komfortfunktionen, beispielsweise das Einlesen der Simulationsdaten über einen zuvor generierten QR-Code, aber auch das Einlesen über die ID der Simulation ist sehr gut gelungen. Hat man die Daten als QR-Code zur Verfügung, wo ist die App komplett ohne Internet-Zugriff verwendbar, da alle Daten lokal auf dem Gerät gespeichert werden, auch dies ist ein positiver Aspekt.

Was ist negativ aufgefallen? Trotz aller zuvor erwähnten positiven Aspekte gibt es einen Punkt, den ich als störend empfunden habe: Befindet man sich auf der Seite, auf der die Simulationsergebnisse angezeigt werden, so ist es nicht möglich, das Profil zu wechseln, ohne den ganzen Weg “zurückzugehen”. Dies ist insbesondere dann sehr nervig, wenn man mehrere Profile hat, zwischen denen man oft wechseln muss. Ich würde mir deshalb eine Funktion wünschen, durch die ich das Profil auch dann schnell wechseln kann, wenn ich mich auf der Seite mit den Simulationsergebnissen befinde.

Ist die App intuitiv bedienbar? Durch die Verwendung des *Materialdesigns* passt sich die App nahtlos an bereits bestehende Apps an und wirkt nicht wie ein Fremdkörper. Dadurch fiel mir die Bedienung sehr leicht und ich habe keine weitere Hilfe benötigt. Außerdem werden bereits durch andere Apps etablierte Gesten wie das Wischen über ein

Listenelement zum löschen verwendet, sodass sich die App auch in diesem Punkt gut ein-
gliedert.

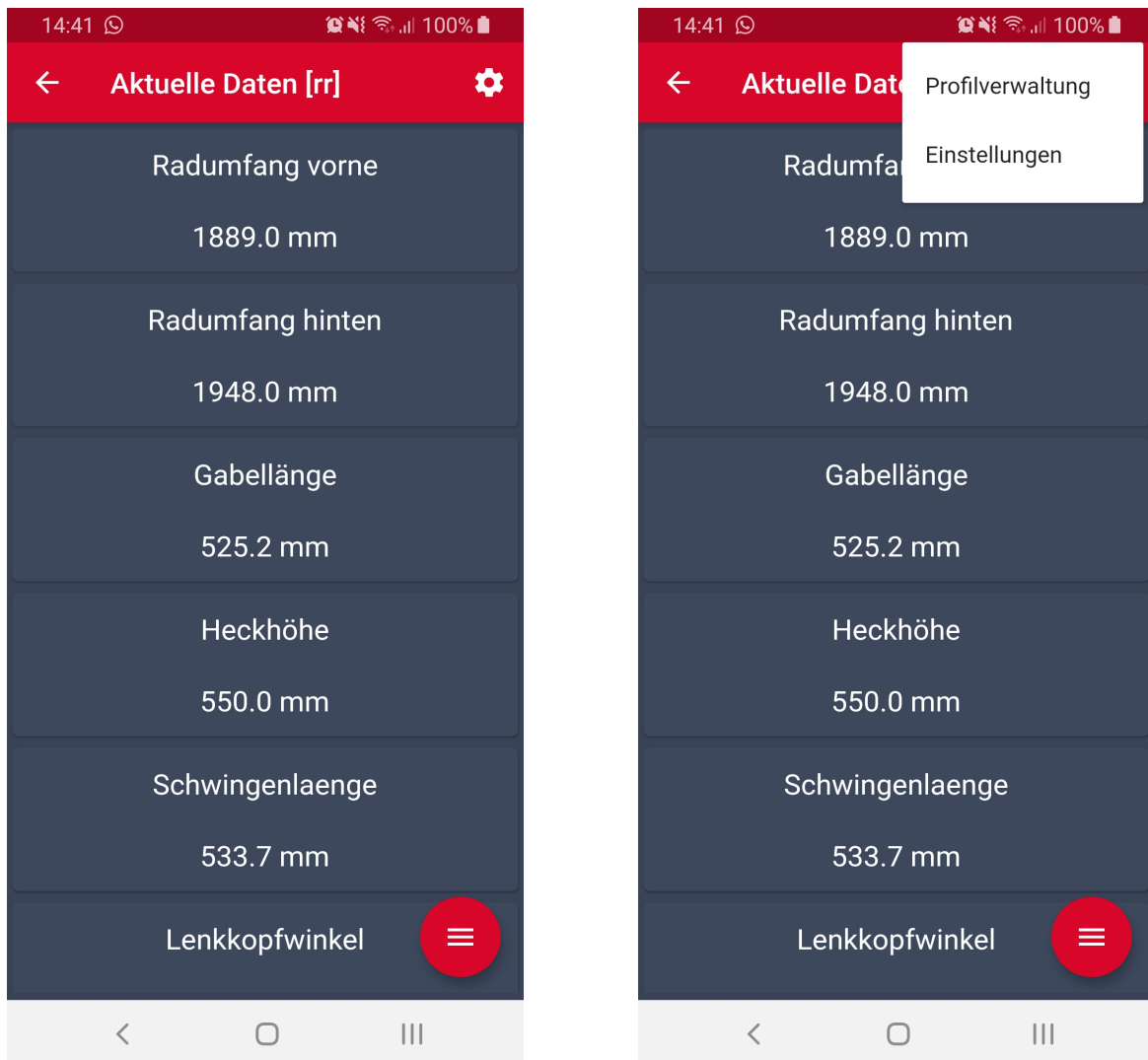


Abbildung 5.3: Links: Übersichtsseite vor dem Interview, rechts nach dem Interview. Zu sehen ist das Drei-Punkt-Menü, was eine verbesserte Navigation ermöglicht.

Die Kritik des Probanden wurde zum Anlass genommen, die Navigation der App zu verändern: Befand man sich vorher auf der Seite, auf der die Ergebnisse der Simulation dargestellt werden, so musste man drei mal zurück gehen, um das Profil zu ändern. Nach der Anpassung ist dies nun über das Drei-Punkt-Menü möglich, die ist in ... zu sehen.

6 Technische Umsetzung (brv)

6.1 Speicherung der Profildaten

Profildaten, die der Nutzer in der App eingibt, werden persistent gespeichert. Die Daten werden mit dem Flutter Plugin SQLite (<https://pub.dev/packages/sqlite>) gespeichert. Das Plugin speichert Daten in einer SQLite Datenbank.

Profildaten werden bei jeder Änderung automatisch gespeichert. Hierzu gehört nicht nur das Laden neuer Daten, sondern auch das Verändern von Eingabevariablen und dem Kommentarfeld.

Die Datenverwaltung basiert im wesentlichen auf zwei Komponenten:

- **Profilklasse (*profile.dart*):** Die Profilklasse repräsentiert Profile, die in der App gespeichert werden können. Jedes Profil besteht dabei aus einer ID, einem Namen, dem letzten Änderungsdatum, der ID, mit der die Messung vom Server geladen wurde (falls die Messung nicht per QR-Code gescannt wurde), einem Satz von Messwerten, einem Satz von Simulationswerten und dem optionalen Kommentar. Die Klasse verfügt außerdem über die Methode *toMap()*, die ein Profil in eine Map konvertiert, die dann in der Datenbank gespeichert werden kann.
- **Datenbank-Hilfsklasse (*database_helpers.dart*):** Die Hilfsklasse stellt sämtliche Funktionalitäten zur Verfügung, die genutzt werden, um mit der SQLite Datenbank zu interagieren. Die Klasse verwaltet die Erstellung der Datenbank-Tabelle und kann Daten zur Datenbank hinzufügen, löschen und bestehende Daten aktualisieren.

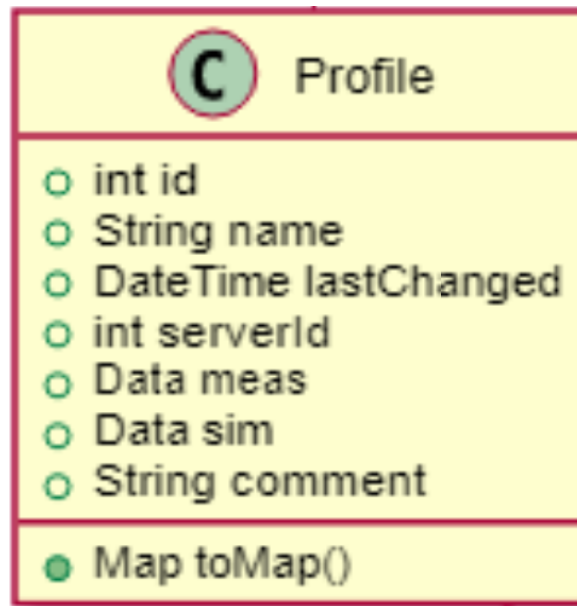


Abbildung 6.1: Aufbau der Profilklasse (*profile.dart*)

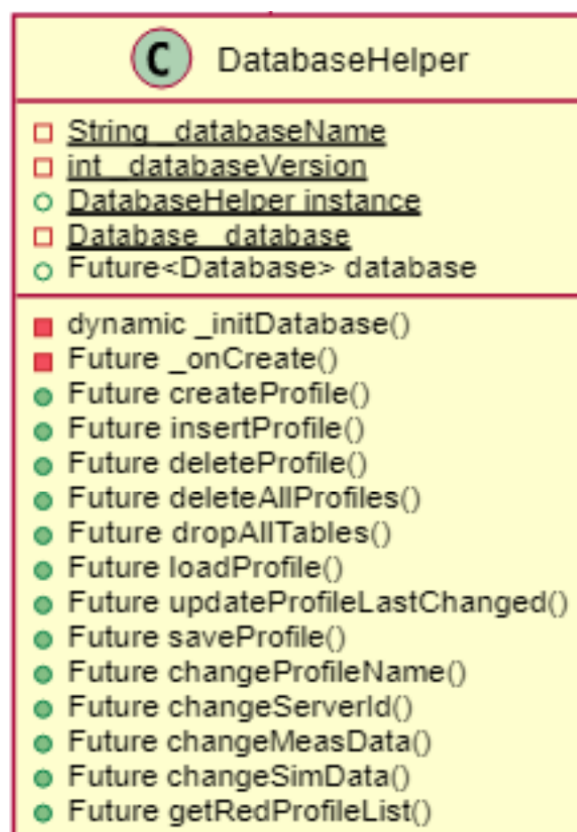


Abbildung 6.2: Aufbau der Hilfsklasse (*database_helpers.dart*)

6.2 Speicherung der Einstellungen

Die Daten zur Einstellung der App bestehen aus Präferenzen zur Sprache und zur Eingabemethode. Diese werden mithilfe des *Shared preferences* Plugins (<https://pub.dev/packages/sqlite>) persistent gespeichert. Da diese Daten nur sehr wenig Speicherplatz benötigen und eine Speicherung mithilfe von Shared preferences sehr einfach ist, wurde hierfür diese Technik gewählt. <https://pub.dev/packages/sqlite>

6.3 Verwaltung des App-States

- ScopedModel

6.4 Daten hinzufügen

- QR: Library - Vom Server: HTTP-Request (Domain, an die der Request geht) -

6.5 Diagramme

- Library - KPS: Erklären, was da wichtig ist

6.6

6.7

6.8

6.9

6.10